

Beim vorliegenden Dokument handelt es sich um einen Entwurf der gematik in Vorbereitung auf zukünftige normative Festlegungen als Grundlage entsprechender Zulassungs- und Bestätigungsverfahren. Die gematik veröffentlicht diesen Entwurf mit dem Ziel, dass sich Interessierte bereits frühzeitig einen Überblick über die mögliche Weiterentwicklung der Telematikinfrastuktur verschaffen können. Die gematik übernimmt keine Gewähr für die Aktualität, Richtigkeit und Vollständigkeit dieses Entwurfes und behält sich das Recht vor, ohne vorherige Ankündigung Änderungen oder Ergänzungen vorzunehmen oder von den Regelungen insgesamt bzw. teilweise Abstand zu nehmen.

## Elektronische Gesundheitskarte und Telematikinfrastuktur

# Spezifikation Konnektor

Version: 5.~~11~~12.0 CC  
Revision: 295363306115  
Stand: 09.12.~~11~~2020  
Status: zur Abstimmung freigegeben  
Klassifizierung: öffentlich\_Entwurf  
Referenzierung: gemSpec\_Kon

22

## Dokumentinformationen

### Änderungen zur Vorversion

Anpassungen des vorliegenden Dokumentes im Vergleich zur Vorversion können Sie der nachfolgenden Tabelle entnehmen.

26

### Dokumentenhistorie

| Version | Stand    | Kap./<br>Seite | Grund der Änderung, besondere<br>Hinweise             | Bearbeitung |
|---------|----------|----------------|---|-------------|
| 5.1.0   | 05.10.17 |                | Initialversion Online-Produktivbetrieb<br>(Stufe 2.1) | gematik     |
| 5.2.0   | 18.12.17 |                | Einarbeitung Erratas 1.6.4-1 bis 1.6.4-3,<br>P15.1    | gematik     |
| 5.3.0   | 14.05.18 |                | Einarbeitung P15.2, P15.4 und P15.5                   | gematik     |
| 5.4.0   | 26.10.18 |                | Einarbeitung P15.8 und P15.9                          | gematik     |
| 5.5.0   | 18.12.19 |                | Einarbeitung P17.1                                    | gematik     |
| 5.6.0   | 15.05.19 |                | Einarbeitung P18.1                                    | gematik     |
| 5.7.0   | 28.06.19 |                | Einarbeitung P19.1                                    | gematik     |
| 5.8.0   | 02.10.19 |                | Einarbeitung P20.1/2                                  | gematik     |
| 5.9.0   | 02.03.20 |                | Einarbeitung P21.1                                    | gematik     |
| 5.9.1   | 26.06.20 |                | Einarbeitung P21.3                                    | gematik     |
| 5.9.2   | 27.08.20 |                | Einarbeitung P21.4                                    | gematik     |
| 5.9.3   | 21.09.20 |                | Einarbeitung P21.5                                    | gematik     |
| 5.9.4   | 05.11.20 |                | Einarbeitung P21.6                                    | gematik     |
| 5.10.0  | 30.06.20 |                | Einarbeitung P22.1                                    | gematik     |

|        |          |  |                                     |         |
|--------|----------|--|-------------------------------------|---------|
| 5.11.0 | 12.11.20 |  | Einarbeitung Scope-Themen zu R4.0.1 | gematik |
| 5.12.0 | 09.12.20 |  | Einarbeitung P22.5                  | gematik |

ENTWURF

## Inhaltsverzeichnis

|    |  |           |
|----|--|-----------|
| 30 | <b>1 Einordnung des Dokumentes .....</b>                                     | <b>24</b> |
| 31 | <b>1.1 Zielsetzung .....</b>   | <b>24</b> |
| 32 | <b>1.2 Zielgruppe .....</b>  | <b>24</b> |
| 33 | <b>1.3 Geltungsbereich .....</b>   | <b>24</b> |
| 34 | <b>1.4 Abgrenzung des Dokuments .....</b>                                    | <b>25</b> |
| 35 | <b>1.5 Methodik .....</b>  | <b>25</b> |
| 36 | 1.5.1 Anforderungen .....  | 25        |
| 37 | 1.5.2 Offene Punkte .....  | 25        |
| 38 | 1.5.3 Erläuterungen zur Spezifikation des Außenverhaltens .....              | 25        |
| 39 | 1.5.4 Erläuterungen zur Dokumentenstruktur und „Dokumentenmechanismen“ ..... | 26        |
| 40 | 1.5.4.1 Modulare Spezifikation über Funktionsmerkmale .....                  | 26        |
| 41 | 1.5.4.2 Technische Use Cases – TUCs .....                                    | 27        |
| 42 | 1.5.4.3 Event Mechanismus .....  | 29        |
| 43 | 1.5.4.4 Konfigurationsparameter und Zustandswerte .....                      | 29        |
| 44 | <b>2 Systemüberblick .....</b>   | <b>30</b> |
| 45 | <b>2.1 Logische Struktur .....</b>   | <b>32</b> |
| 46 | <b>2.2 Sicherer Datenspeicher .....</b>                                      | <b>34</b> |
| 47 | <b>2.3 Überblick Konnektoridentität .....</b>                                | <b>34</b> |
| 48 | <b>2.4 Mandantenfähigkeit .....</b>  | <b>35</b> |
| 49 | <b>2.5 Versionierung .....</b>   | <b>35</b> |
| 50 | <b>2.6 Fachanwendungen .....</b>   | <b>35</b> |
| 51 | <b>2.7 Netzseitige Einsatzszenarien .....</b>                                | <b>36</b> |
| 52 | 2.7.1 Parameter ANLW_ANBINDUNGS_MODUS .....                                  | 36        |
| 53 | 2.7.2 Parameter ANLW_INTERNET_MODUS .....                                    | 36        |
| 54 | <b>2.8 Lokale und entfernte Kartenterminals .....</b>                        | <b>37</b> |
| 55 | <b>2.9 Standalone-Szenario .....</b>   | <b>37</b> |
| 56 | <b>3 Übergreifende Festlegungen .....</b>                                    | <b>38</b> |
| 57 | <b>3.1 Konnektoridentität und gSMC-K .....</b>                               | <b>41</b> |
| 58 | 3.1.1 Organisatorische Anforderungen und Sperrprozesse .....                 | 42        |
| 59 | <b>3.2 Bootup-Phase .....</b>  | <b>44</b> |
| 60 | <b>3.3 Betriebszustand .....</b>   | <b>45</b> |
| 61 | 3.3.1 Betriebsaspekte .....  | 58        |
| 62 | <b>3.4 Fachliche Anbindung der Clientsysteme .....</b>                       | <b>59</b> |
| 63 | 3.4.1 Betriebsaspekte .....  | 63        |
| 64 | <b>3.5 Clientsystemschnittstelle .....</b>                                   | <b>65</b> |
| 65 | 3.5.1 SOAP-Schnittstelle .....   | 65        |
| 66 | 3.5.2 Statusrückmeldung und Fehlerbehandlung .....                           | 66        |
| 67 | 3.5.3 Transport großer Dokumente .....                                       | 68        |



|     |   |           |
|-----|---|-----------|
| 68  | <b>3.6 Verwendung manuell importierter CA-Zertifikate</b>                             | <b>69</b> |
| 69  | <b>3.7 Testunterstützung</b>  | <b>70</b> |
| 70  | <b>4 Funktionsmerkmale</b>  | <b>72</b> |
| 71  | <b>4.1 Anwendungskonnektor</b>  | <b>72</b> |
| 72  | 4.1.1 Zugriffsberechtigungsdienst   | 72        |
| 73  | 4.1.1.1 Funktionsmerkmalweite Aspekte   | 72        |
| 74  | 4.1.1.2 Durch Ereignisse ausgelöste Reaktionen  | 83        |
| 75  | 4.1.1.3 Interne TUCs, nicht durch Fachmodule nutzbar                                  | 83        |
| 76  | 4.1.1.4 Interne TUCs, auch durch Fachmodule nutzbar                                   | 83        |
| 77  | 4.1.1.4.1 TUC_KON_000 „Prüfe Zugriffsberechtigung“                                    | 83        |
| 78  | 4.1.1.5 Operationen an der Außenschnittstelle   | 95        |
| 79  | 4.1.1.6 Betriebsaspekte   | 95        |
| 80  | 4.1.2 Dokumentvalidierungsdienst  | 95        |
| 81  | 4.1.2.1 Funktionsmerkmalweite Aspekte   | 96        |
| 82  | 4.1.2.2 Durch Ereignisse ausgelöste Reaktionen  | 96        |
| 83  | 4.1.2.3 Interne TUCs, nicht durch Fachmodule nutzbar                                  | 96        |
| 84  | 4.1.2.4 Interne TUCs, auch durch Fachmodule nutzbar                                   | 96        |
| 85  | 4.1.2.4.1 TUC_KON_080 „Dokument validieren“   | 96        |
| 86  | 4.1.2.5 Operationen an der Außenschnittstelle   | 99        |
| 87  | 4.1.2.6 Betriebsaspekte   | 99        |
| 88  | 4.1.3 Dienstverzeichnisdienst   | 99        |
| 89  | 4.1.3.1 Funktionsmerkmalweite Aspekte   | 99        |
| 90  | 4.1.3.2 Durch Ereignisse ausgelöste Reaktionen  | 103       |
| 91  | 4.1.3.3 Interne TUCs, nicht durch Fachmodule nutzbar                                  | 103       |
| 92  | 4.1.3.4 Interne TUCs, auch durch Fachmodule nutzbar                                   | 103       |
| 93  | 4.1.3.4.1 TUC_KON_041 „Einbringen der Endpunktinformationen während der Bootup-Phase“ | 103       |
| 94  | 4.1.3.5 Operationen an der Außenschnittstelle   | 104       |
| 95  | 4.1.3.6 Betriebsaspekte   | 105       |
| 96  | 4.1.4 Kartenterminaldienst  | 106       |
| 97  | 4.1.4.1 Funktionsmerkmalweite Aspekte   | 110       |
| 98  | 4.1.4.2 Durch Ereignisse ausgelöste Reaktionen  | 113       |
| 99  | 4.1.4.3 Interne TUCs, nicht durch Fachmodule nutzbar                                  | 114       |
| 100 | 4.1.4.3.1 TUC_KON_050 „Beginne Kartenterminalsitzung“                                 | 114       |
| 101 | 4.1.4.3.2 TUC_KON_054 „Kartenterminal hinzufügen“                                     | 120       |
| 102 | 4.1.4.3.3 TUC_KON_053 „Paire Kartenterminal“  | 122       |
| 103 | 4.1.4.3.4 TUC_KON_055 „Befülle CT-Object“   | 127       |
| 104 | 4.1.4.4 Interne TUCs, auch durch Fachmodule nutzbar                                   | 128       |
| 105 | 4.1.4.4.1 TUC_KON_051 „Mit Anwender über Kartenterminal interagieren“                 | 128       |
| 106 | 4.1.4.4.2 TUC_KON_056 „Karte anfordern“   | 130       |
| 107 | 4.1.4.4.3 TUC_KON_057 „Karte auswerfen“   | 133       |
| 108 | 4.1.4.4.4 TUC_KON_058 „Displaygröße ermitteln“  | 135       |
| 109 | 4.1.4.5 Operationen an der Außenschnittstelle   | 137       |
| 110 | 4.1.4.5.1 RequestCard   | 137       |
| 111 | 4.1.4.5.2 EjectCard   | 140       |
| 112 | 4.1.4.6 Betriebsaspekte   | 142       |
| 113 |   |           |

|     |  |     |
|-----|--|-----|
| 114 | 4.1.4.6.1 Allgemeine Betriebsaspekte .....                     | 142 |
| 115 | 4.1.4.6.2 Kartenterminals pflegen .....                        | 144 |
| 116 | 4.1.4.6.3 Import der Kartenterminal-Informationen .....        | 148 |
| 117 | 4.1.5 Kartendienst.....  | 149 |
| 118 | 4.1.5.1 Funktionsmerkmalweite Aspekte.....                     | 151 |
| 119 | 4.1.5.2 Durch Ereignisse ausgelöste Reaktionen .....           | 156 |
| 120 | 4.1.5.3 Interne TUCs, nicht durch Fachmodule nutzbar .....     | 157 |
| 121 | 4.1.5.3.1 TUC_KON_001 „Karte öffnen“.....                      | 157 |
| 122 | 4.1.5.4 Interne TUCs, auch durch Fachmodule nutzbar .....      | 159 |
| 123 | 4.1.5.4.1 TUC_KON_026 „Liefere CardSession“.....               | 159 |
| 124 | 4.1.5.4.2 TUC_KON_012 „PIN verifizieren“.....                  | 161 |
| 125 | 4.1.5.4.3 TUC_KON_019 „PIN ändern“.....                        | 166 |
| 126 | 4.1.5.4.4 TUC_KON_021 „PIN entsperren“.....                    | 170 |
| 127 | 4.1.5.4.5 TUC_KON_022 „Liefere PIN Status“.....                | 174 |
| 128 | 4.1.5.4.6 TUC_KON_027 „PIN Schutz ein /ausschalten“.....       | 176 |
| 129 | 4.1.5.4.7 TUC_KON_023 „Karte reservieren“.....                 | 180 |
| 130 | 4.1.5.4.8 TUC_KON_005 „Card to Card authentisieren“.....       | 181 |
| 131 | 4.1.5.4.9 TUC_KON_202 „LeseDatei“.....                         | 186 |
| 132 | 4.1.5.4.10 TUC_KON_203 „SchreibeDatei“.....                    | 188 |
| 133 | 4.1.5.4.11 TUC_KON_204 „LöscheDateiInhalt“.....                | 190 |
| 134 | 4.1.5.4.12 TUC_KON_209 „LeseRecord“.....                       | 192 |
| 135 | 4.1.5.4.13 TUC_KON_210 „SchreibeRecord“.....                   | 194 |
| 136 | 4.1.5.4.14 TUC_KON_211 „LöscheRecordInhalt“.....               | 196 |
| 137 | 4.1.5.4.15 TUC_KON_214 „FügeHinzuRecord“.....                  | 198 |
| 138 | 4.1.5.4.16 TUC_KON_215 „SucheRecord“.....                      | 200 |
| 139 | 4.1.5.4.17 TUC_KON_018 „eGK-Sperrung prüfen“.....              | 202 |
| 140 | 4.1.5.4.18 TUC_KON_006 „Datenzugriffsaudit eGK schreiben“..... | 203 |
| 141 | 4.1.5.4.19 TUC_KON_218 „Signiere“.....                         | 205 |
| 142 | 4.1.5.4.20 TUC_KON_219 „Entschlüssele“.....                    | 207 |
| 143 | 4.1.5.4.21 TUC_KON_200 „SendeAPDU“.....                        | 209 |
| 144 | 4.1.5.4.22 TUC_KON_024 „Karte zurücksetzen“.....               | 210 |
| 145 | 4.1.5.4.23 TUC_KON_216 „LeseZertifikat“.....                   | 212 |
| 146 | 4.1.5.4.24 TUC_KON_036 „LiefereFachlicheRolle“.....            | 214 |
| 147 | 4.1.5.5 Operationen an der Außenschnittstelle.....             | 215 |
| 148 | 4.1.5.5.1 VerifyPin .....                                      | 216 |
| 149 | 4.1.5.5.2 ChangePin .....                                      | 219 |
| 150 | 4.1.5.5.3 GetPinStatus.....                                    | 222 |
| 151 | 4.1.5.5.4 UnblockPin .....                                     | 225 |

|     |   |     |
|-----|---|-----|
| 152 | 4.1.5.5.5 EnablePin .....                                     | 228 |
| 153 | 4.1.5.5.6 DisablePin .....                                    | 231 |
| 154 | 4.1.5.6 Betriebsaspekte .....                                 | 234 |
| 155 | 4.1.5.6.1 TUC_KON_025 "Initialisierung Kartendienst" .....    | 234 |
| 156 | 4.1.5.6.2 Kartenübersicht und PIN-Management .....            | 235 |
| 157 | 4.1.6 Systeminformationsdienst .....                          | 236 |
| 158 | 4.1.6.1 Funktionsmerkmalweite Aspekte .....                   | 237 |
| 159 | 4.1.6.2 Durch Ereignisse ausgelöste Reaktionen .....          | 240 |
| 160 | 4.1.6.3 Interne TUCs, nicht durch Fachmodule nutzbar .....    | 240 |
| 161 | 4.1.6.4 Interne TUCs, auch durch Fachmodule nutzbar .....     | 240 |
| 162 | 4.1.6.4.1 TUC_KON_256 „Systemereignis absetzen“ .....         | 240 |
| 163 | 4.1.6.4.2 TUC_KON_252 „Liefere KT_Liste“ .....                | 245 |
| 164 | 4.1.6.4.3 TUC_KON_253 „Liefere Karten_Liste“ .....            | 246 |
| 165 | 4.1.6.4.4 TUC_KON_254 „Liefere Ressourcendetails“ .....       | 248 |
| 166 | 4.1.6.5 Operationen an der Außenschnittstelle .....           | 250 |
| 167 | 4.1.6.5.1 GetCardTerminals .....                              | 251 |
| 168 | 4.1.6.5.2 GetCards .....                                      | 254 |
| 169 | 4.1.6.5.3 GetResourceInformation .....                        | 259 |
| 170 | 4.1.6.5.4 Subscribe .....                                     | 263 |
| 171 | 4.1.6.5.5 Unsubscribe .....                                   | 266 |
| 172 | 4.1.6.5.6 RenewSubscriptions .....                            | 268 |
| 173 | 4.1.6.5.7 GetSubscription .....                               | 270 |
| 174 | 4.1.6.5.8 GetLeCards .....                                    | 272 |
| 175 | 4.1.6.6 Betriebsaspekte .....                                 | 272 |
| 176 | 4.1.7 Verschlüsselungsdienst .....                            | 273 |
| 177 | 4.1.7.1 Funktionsmerkmalweite Aspekte .....                   | 273 |
| 178 | 4.1.7.2 Durch Ereignisse ausgelöste Reaktionen .....          | 275 |
| 179 | 4.1.7.3 Interne TUCs, nicht durch Fachmodule nutzbar .....    | 275 |
| 180 | 4.1.7.4 Interne TUCs, auch durch Fachmodule nutzbar .....     | 275 |
| 181 | 4.1.7.4.1 TUC_KON_070 „Daten hybrid verschlüsseln“ .....      | 275 |
| 182 | 4.1.7.4.2 TUC_KON_071 „Daten hybrid entschlüsseln“ .....      | 284 |
| 183 | 4.1.7.4.3 TUC_KON_072 „Daten symmetrisch verschlüsseln“ ..... | 289 |
| 184 | 4.1.7.4.4 TUC_KON_073 „Daten symmetrisch entschlüsseln“ ..... | 290 |
| 185 | 4.1.7.4.5 TUC_KON_075 „Symmetrisch verschlüsseln“ .....       | 291 |
| 186 | 4.1.7.4.6 TUC_KON_076 „Symmetrisch entschlüsseln“ .....       | 293 |
| 187 | 4.1.7.5 Operationen an der Außenschnittstelle .....           | 294 |
| 188 | 4.1.7.5.1 EncryptDocument .....                               | 294 |
| 189 | 4.1.7.5.2 DecryptDocument .....                               | 300 |
| 190 | 4.1.7.6 Betriebsaspekte .....                                 | 303 |
| 191 | 4.1.8 Signaturdienst .....                                    | 303 |
| 192 | 4.1.8.1 Funktionsmerkmalweite Aspekte .....                   | 303 |
| 193 | 4.1.8.1.1 Dokumentensignatur .....                            | 303 |

|     |   |     |
|-----|---|-----|
| 194 | 4.1.8.1.2 Signaturrichtlinien .....                                     | 310 |
| 195 | 4.1.8.1.3 Signaturzeitpunkt .....                                       | 310 |
| 196 | 4.1.8.1.4 Jobnummer .....   | 310 |
| 197 | 4.1.8.1.5 Komfortsignatur .....   | 312 |
| 198 | 4.1.8.2 Durch Ereignisse ausgelöste Reaktionen .....                    | 314 |
| 199 | 4.1.8.3 Interne TUCs, nicht durch Fachmodule nutzbar .....              | 314 |
| 200 | 4.1.8.3.1 TUC_KON_155 „Dokumente zur Signatur vorbereiten“ .....        | 315 |
| 201 | 4.1.8.3.2 TUC_KON_165 „Signaturvoraussetzungen für nonQES prüfen“ ..... | 319 |
| 202 | 4.1.8.3.3 TUC_KON_166 „nonQES Signaturen erstellen“ .....               | 320 |
| 203 | 4.1.8.3.4 TUC_KON_152 „Signaturvoraussetzungen für QES prüfen“ .....    | 322 |
| 204 | 4.1.8.3.5 TUC_KON_154 „QES Signaturen erstellen“ .....                  | 323 |
| 205 | 4.1.8.3.6 TUC_KON_168 „Einzelsignatur QES erstellen“ .....              | 327 |
| 206 | 4.1.8.3.7 TUC_KON_158 „Komfortsignaturen erstellen“ .....               | 328 |
| 207 | 4.1.8.4 Interne TUCs, auch durch Fachmodule nutzbar .....               | 331 |
| 208 | 4.1.8.4.1 TUC_KON_160 „Dokumente nonQES signieren“ .....                | 331 |
| 209 | 4.1.8.4.2 TUC_KON_161 „nonQES Dokumentsignatur prüfen“ .....            | 337 |
| 210 | 4.1.8.4.3 TUC_KON_162 „Kryptographische Prüfung der XML-                |     |
| 211 | Dokumentensignatur“ .....   | 344 |
| 212 | 4.1.8.4.4 TUC_KON_150 „Dokumente QES signieren“ .....                   | 345 |
| 213 | 4.1.8.4.5 Anforderungen an die Stapelsignatur .....                     | 350 |
| 214 | 4.1.8.4.6 TUC_KON_151 „QES Dokumentensignatur prüfen“ .....             | 352 |
| 215 | 4.1.8.4.7 TUC_KON_170 „Dokumente mit Komfort signieren“ .....           | 358 |
| 216 | 4.1.8.4.8 TUC_KON_171 „Komfortsignatur einschalten“ .....               | 361 |
| 217 | 4.1.8.4.9 TUC_KON_172 „Komfortsignatur ausschalten“ .....               | 363 |
| 218 | 4.1.8.4.10 TUC_KON_173 „Liefere Signaturmodus“ .....                    | 365 |
| 219 | 4.1.8.5 Operationen an der Außenschnittstelle .....                     | 367 |
| 220 | 4.1.8.5.1 SignDocument (nonQES und QES) .....                           | 367 |
| 221 | 4.1.8.5.2 VerifyDocument (nonQES und QES) .....                         | 381 |
| 222 | 4.1.8.5.3 StopSignature .....   | 387 |
| 223 | 4.1.8.5.4 GetJobNumber .....  | 388 |
| 224 | 4.1.8.5.5 ActivateComfortSignature .....                                | 389 |
| 225 | 4.1.8.5.6 DeactivateComfortSignature .....                              | 391 |
| 226 | 4.1.8.5.7 GetSignatureMode .....  | 392 |
| 227 | 4.1.8.6 Betriebsaspekte .....   | 394 |
| 228 | 4.1.9 Zertifikatsdienst .....   | 396 |
| 229 | 4.1.9.1 Funktionsmerkmalweite Aspekte .....                             | 396 |
| 230 | 4.1.9.2 Durch Ereignisse ausgelöste Reaktionen .....                    | 402 |
| 231 | 4.1.9.3 Interne TUCs, nicht durch Fachmodule nutzbar .....              | 402 |
| 232 | 4.1.9.3.1 TUC_KON_032 „TSL aktualisieren“ .....                         | 402 |
| 233 | 4.1.9.3.2 TUC_KON_031 „BNetzA-VL aktualisieren“ .....                   | 407 |

|     |   |     |
|-----|---|-----|
| 234 | 4.1.9.3.3 TUC_KON_040 „CRL aktualisieren“ .....                       | 408 |
| 235 | 4.1.9.3.4 TUC_KON_033 „Zertifikatsablauf prüfen“ .....                | 410 |
| 236 | 4.1.9.4 Interne TUCs, auch durch Fachmodule nutzbar .....             | 413 |
| 237 | 4.1.9.4.1 TUC_KON_037 „Zertifikat prüfen“ .....                       | 413 |
| 238 | 4.1.9.4.2 TUC_KON_042 „CV-Zertifikat prüfen“ .....                    | 418 |
| 239 | 4.1.9.4.3 TUC_KON_034 „Zertifikatsinformationen extrahieren“ .....    | 420 |
| 240 | 4.1.9.5 Operationen an der Außenschnittstelle .....                   | 423 |
| 241 | 4.1.9.5.1 CheckCertificateExpiration .....                            | 424 |
| 242 | 4.1.9.5.2 ReadCardCertificate .....                                   | 427 |
| 243 | 4.1.9.5.3 VerifyCertificate .....                                     | 431 |
| 244 | 4.1.9.6 Betriebsaspekte .....   | 433 |
| 245 | 4.1.9.6.1 TUC_KON_035 „Zertifikatsdienst initialisieren“ .....        | 433 |
| 246 | 4.1.10 Protokollierungsdienst .....                                   | 441 |
| 247 | 4.1.10.1 Funktionsmerkmalweite Aspekte .....                          | 442 |
| 248 | 4.1.10.2 Durch Ereignisse ausgelöste Reaktionen .....                 | 444 |
| 249 | 4.1.10.3 Interne TUCs, nicht durch Fachmodule nutzbar .....           | 444 |
| 250 | 4.1.10.4 Interne TUCs, auch durch Fachmodule nutzbar .....            | 444 |
| 251 | 4.1.10.4.1 TUC_KON_271 „Schreibe Protokolleintrag“ .....              | 444 |
| 252 | 4.1.10.5 Operationen an der Außenschnittstelle .....                  | 448 |
| 253 | 4.1.10.6 Betriebsaspekte .....  | 448 |
| 254 | 4.1.10.6.1 TUC_KON_272 „Initialisierung Protokollierungsdienst“ ..... | 450 |
| 255 | 4.1.11 TLS Dienst .....   | 452 |
| 256 | 4.1.11.1 Funktionsmerkmalweite Aspekte .....                          | 452 |
| 257 | 4.1.11.2 Durch Ereignisse ausgelöste Reaktionen .....                 | 452 |
| 258 | 4.1.11.3 Interne TUCs, nicht durch Fachmodule nutzbar .....           | 452 |
| 259 | 4.1.11.4 Interne TUCs, auch durch Fachmodule nutzbar .....            | 452 |
| 260 | 4.1.11.4.1 TUC_KON_110 „Kartenbasierte TLS-Verbindung aufbauen“ ..... | 452 |
| 261 | 4.1.11.4.2 TUC_KON_111 „Kartenbasierte TLS-Verbindung abbauen“ .....  | 454 |
| 262 | 4.1.11.5 Operationen an der Außenschnittstelle .....                  | 455 |
| 263 | 4.1.11.6 Betriebsaspekte .....  | 455 |
| 264 | 4.1.12 LDAP-Proxy .....   | 455 |
| 265 | 4.1.12.1 Funktionsmerkmalweite Aspekte .....                          | 455 |
| 266 | 4.1.12.2 Durch Ereignisse ausgelöste Reaktionen .....                 | 455 |
| 267 | 4.1.12.3 Interne TUCs, nicht durch Fachmodule nutzbar .....           | 455 |
| 268 | 4.1.12.4 Interne TUCs, auch durch Fachmodule nutzbar .....            | 456 |
| 269 | 4.1.12.4.1 TUC_KON_290 „LDAP-Verbindung aufbauen“ .....               | 456 |
| 270 | 4.1.12.4.2 TUC_KON_291 „Verzeichnis abfragen“ .....                   | 457 |
| 271 | 4.1.12.4.3 TUC_KON_292 „LDAP-Verbindung trennen“ .....                | 457 |
| 272 | 4.1.12.4.4 TUC_KON_293 „Verzeichnisabfrage abbrechen“ .....           | 459 |
| 273 | 4.1.12.5 Operationen an der Außenschnittstelle .....                  | 459 |
| 274 | 4.1.12.5.1 Unterstützte LDAPv3-Operationen .....                      | 459 |
| 275 | 4.1.12.6 Betriebsaspekte .....  | 460 |
| 276 | 4.1.13 Authentifizierungsdienst .....                                 | 460 |
| 277 | 4.1.13.1 Funktionsmerkmalweite Aspekte .....                          | 460 |
| 278 | 4.1.13.1.1 Externe Authentisierung .....                              | 460 |

|     |   |            |
|-----|---|------------|
| 279 | 4.1.13.2 Durch Ereignisse ausgelöste Reaktionen.....              | 461        |
| 280 | 4.1.13.3 Interne TUCs.....  | 461        |
| 281 | 4.1.13.4 Operationen an der Außenschnittstelle.....               | 461        |
| 282 | 4.1.13.4.1 ExternalAuthenticate.....                              | 461        |
| 283 | 4.1.13.5 Betriebsaspekte.....                                     | 465        |
| 284 | <b>4.2 Netzkonnektor.....</b>                                     | <b>466</b> |
| 285 | 4.2.1 Anbindung LAN/WAN.....                                      | 466        |
| 286 | 4.2.1.1 Funktionsmerkmalweite Aspekte.....                        | 467        |
| 287 | 4.2.1.1.1 Netzwerksegmentierung.....                              | 468        |
| 288 | 4.2.1.1.2 Routing und Firewall.....                               | 470        |
| 289 | 4.2.1.2 Durch Ereignisse ausgelöste Reaktionen.....               | 479        |
| 290 | 4.2.1.3 Interne TUCs, nicht durch Fachmodule nutzbar.....         | 480        |
| 291 | 4.2.1.3.1 TUC_KON_305 „LAN Adapter initialisieren“.....           | 480        |
| 292 | 4.2.1.3.2 TUC_KON_306 „WAN Adapter initialisieren“.....           | 481        |
| 293 | 4.2.1.3.3 TUC_KON_304 „Netzwerk Routen einrichten“.....           | 483        |
| 294 | 4.2.1.4 Interne TUCs, auch durch Fachmodule nutzbar.....          | 485        |
| 295 | 4.2.1.5 Operationen an der Außenschnittstelle.....                | 485        |
| 296 | 4.2.1.6 Betriebsaspekte.....                                      | 485        |
| 297 | 4.2.2 DHCP Server.....  | 492        |
| 298 | 4.2.2.1 Funktionsmerkmalweite Aspekte.....                        | 493        |
| 299 | 4.2.2.2 Durch Ereignisse ausgelöste Reaktionen.....               | 493        |
| 300 | 4.2.2.3 Interne TUCs, nicht durch Fachmodule nutzbar.....         | 493        |
| 301 | 4.2.2.4 Interne TUCs, auch durch Fachmodule nutzbar.....          | 493        |
| 302 | 4.2.2.5 Operationen an der Außenschnittstelle.....                | 493        |
| 303 | 4.2.2.5.1 Liefere Netzwerkinformationen über DHCP.....            | 493        |
| 304 | 4.2.2.6 Betriebsaspekte.....                                      | 494        |
| 305 | 4.2.2.6.1 TUC_KON_343 „Initialisierung DHCP Server“.....          | 498        |
| 306 | 4.2.3 DHCP Client.....  | 499        |
| 307 | 4.2.3.1 Funktionsmerkmalweite Aspekte.....                        | 499        |
| 308 | 4.2.3.2 Durch Ereignisse ausgelöste Reaktionen.....               | 499        |
| 309 | 4.2.3.3 Interne TUCs, nicht durch Fachmodule nutzbar.....         | 499        |
| 310 | 4.2.3.3.1 TUC_KON_341 „DHCP Informationen beziehen“.....          | 499        |
| 311 | 4.2.3.4 Interne TUCs, auch durch Fachmodule nutzbar.....          | 501        |
| 312 | 4.2.3.5 Operationen an der Außenschnittstelle.....                | 501        |
| 313 | 4.2.3.6 Betriebsaspekte.....                                      | 501        |
| 314 | 4.2.4 VPN Client.....   | 502        |
| 315 | 4.2.4.1 Funktionsmerkmalweite Aspekte.....                        | 502        |
| 316 | 4.2.4.2 Durch Ereignisse ausgelöste Reaktionen.....               | 502        |
| 317 | 4.2.4.3 Interne TUCs, nicht durch Fachmodule nutzbar.....         | 503        |
| 318 | 4.2.4.3.1 TUC_KON_321 „Verbindung zu dem VPN-Konzentrator der TI  |            |
| 319 | aufbauen“.....  | 503        |
| 320 | 4.2.4.3.2 TUC_KON_322 „Verbindung zu dem VPN-Konzentrator des SIS |            |
| 321 | aufbauen“.....  | 506        |
| 322 | 4.2.4.4 Interne TUCs, auch durch Fachmodule nutzbar.....          | 508        |
| 323 | 4.2.4.5 Operationen an der Außenschnittstelle.....                | 508        |
| 324 | 4.2.4.6 Betriebsaspekte.....                                      | 508        |
| 325 | 4.2.5 Zeitdienst.....   | 510        |
| 326 | 4.2.5.1 Funktionsmerkmalweite Aspekte.....                        | 510        |



|     |  |            |
|-----|--|------------|
| 327 | 4.2.5.2 Durch Ereignisse ausgelöste Reaktionen .....                     | 511        |
| 328 | 4.2.5.3 Interne TUCs, nicht durch Fachmodule nutzbar .....               | 511        |
| 329 | 4.2.5.4 Interne TUCs, auch durch Fachmodule nutzbar .....                | 511        |
| 330 | 4.2.5.4.1 TUC_KON_351 „Liefere Systemzeit“ .....                         | 511        |
| 331 | 4.2.5.5 Operationen an der Außenschnittstelle .....                      | 512        |
| 332 | 4.2.5.5.1 Sync_Time .....  | 512        |
| 333 | 4.2.5.6 Betriebsaspekte .....  | 513        |
| 334 | 4.2.5.6.1 TUC_KON_352 Initialisierung Zeitdienst .....                   | 514        |
| 335 | 4.2.6 Namensdienst und Dienstlokalisierung .....                         | 515        |
| 336 | 4.2.6.1 Funktionsmerkmalweite Aspekte .....                              | 515        |
| 337 | 4.2.6.2 Durch Ereignisse ausgelöste Reaktionen .....                     | 517        |
| 338 | 4.2.6.3 Interne TUCs, nicht durch Fachmodule nutzbar .....               | 517        |
| 339 | 4.2.6.4 Interne TUCs, auch durch Fachmodule nutzbar .....                | 517        |
| 340 | 4.2.6.4.1 TUC_KON_361 „DNS-Namen auflösen“ .....                         | 517        |
| 341 | 4.2.6.4.2 TUC_KON_362 „Liste der Dienste abrufen“ .....                  | 520        |
| 342 | 4.2.6.4.3 TUC_KON_363 „Dienstdetails abrufen“ .....                      | 521        |
| 343 | 4.2.6.5 Operationen an der Außenschnittstelle .....                      | 522        |
| 344 | 4.2.6.5.1 GetIPAddress .....   | 522        |
| 345 | 4.2.6.6 Betriebsaspekte .....  | 523        |
| 346 | 4.2.7 Optionale Verwendung von IPv6 .....                                | 525        |
| 347 | <b>4.3 Konnektormanagement .....</b>                                     | <b>525</b> |
| 348 | 4.3.1 Zugang und Benutzerverwaltung des Konnektormanagements .....       | 527        |
| 349 | 4.3.2 Konnektornamen und Versionsinformationen .....                     | 530        |
| 350 | 4.3.3 Konfigurationsdaten: Persistieren sowie Export-Import .....        | 531        |
| 351 | 4.3.4 Administration von Fachmodulen .....                               | 533        |
| 352 | 4.3.5 Neustart und Werksreset .....                                      | 534        |
| 353 | 4.3.6 Leistungsumfänge und Standalone-Szenarios .....                    | 534        |
| 354 | 4.3.7 Online-Anbindung verwalten .....                                   | 536        |
| 355 | 4.3.8 Remote Management (Optional) .....                                 | 539        |
| 356 | 4.3.9 Software- und Konfigurationsaktualisierung (KSR-Client) .....      | 544        |
| 357 | 4.3.9.1 Funktionsmerkmalweite Aspekte .....                              | 544        |
| 358 | 4.3.9.2 Durch Ereignisse ausgelöste Reaktionen .....                     | 545        |
| 359 | 4.3.9.3 Interne TUCs, nicht durch Fachmodule nutzbar .....               | 545        |
| 360 | 4.3.9.3.1 TUC_KON_280 „Konnektoraktualisierung durchführen“ .....        | 545        |
| 361 | 4.3.9.3.2 TUC_KON_281 „Kartenterminalaktualisierung anstoßen“ .....      | 549        |
| 362 | 4.3.9.3.3 TUC_KON_282 „UpdateInformationen beziehen“ .....               | 551        |
| 363 | 4.3.9.3.4 TUC_KON_283 „Infrastruktur-Konfiguration aktualisieren“ .....  | 553        |
| 364 | 4.3.9.4 Interne TUCs, auch durch Fachmodule nutzbar .....                | 557        |
| 365 | 4.3.9.4.1 TUC_KON_285 „UpdateInformationen für Fachmodul beziehen“ ..... | 557        |
| 366 | 4.3.9.4.2 TUC_KON_286 „Paket für Fachmodul laden“ .....                  | 559        |
| 367 | 4.3.9.5 Operationen an der Außenschnittstelle .....                      | 561        |
| 368 | 4.3.9.6 Betriebsaspekte .....  | 561        |
| 369 | 4.3.9.6.1 TUC_KON_284 KSR-Client initialisieren .....                    | 561        |
| 370 | 4.3.10 Konnektorstatus .....   | 569        |
| 371 | <b>4.4 Hardware-Merkmale des Konnektors .....</b>                        | <b>569</b> |

|   |            |
|---|------------|
| <b>5 Anhang A – Verzeichnisse</b>   | <b>573</b> |
| 5.1 Abkürzungen   | 573        |
| 5.2 Glossar   | 575        |
| 5.3 Abbildungsverzeichnis   | 575        |
| 5.4 Tabellenverzeichnis   | 577        |
| 5.5 Referenzierte Dokumente   | 602        |
| 5.5.1 Dokumente der gematik   | 602        |
| 5.5.2 Weitere Dokumente   | 603        |
| <b>6 Anhang B – Profilierung der Signatur und Verschlüsselungsformate (normativ)</b>                  | <b>611</b> |
| 6.1 Profilierung der Verschlüsselungsformate  | 611        |
| 6.2 Profilierung der Signaturformate  | 611        |
| 6.3 Profilierung VerificationReport   | 613        |
| 6.4 Profilierung der Dokumentenformate und Nachrichten  | 618        |
| <b>7 Anhang D – Übersicht über die verwendeten Versionen</b>  | <b>620</b> |
| <b>8 Anhang F – Übersicht Events</b>  | <b>630</b> |
| <b>9 Anhang H – Mapping von „Architektur der TI-Plattform“ auf Konnektorspezifikation</b>             | <b>648</b> |
| <b>10 Anhang I – Umsetzungshinweise (informativ)</b>  | <b>657</b> |
| 10.1 Systemüberblick  | 657        |
| 10.1.1 Hinweise zur Sicherheitsevaluierung nach Common Criteria                                       | 657        |
| 10.1.1.1 Separationsmechanismen des Konnektors  | 657        |
| 10.1.1.2 Granularität der TSF   | 658        |
| 10.2 Übergreifende Festlegungen   | 659        |
| 10.2.1 Interne Mechanismen  | 659        |
| 10.2.1.1 Zufallszahlen und Schlüssel  | 659        |
| 10.3 Funktionsmerkmale  | 659        |
| 10.3.1 Anwendungskonnektor  | 659        |
| 10.3.1.1 Administration des Informationsmodells   | 659        |
| 10.3.1.2 Vorgehensvariante für das Handling von CardSessions  | 660        |
| 10.3.1.3 Darstellung von Terminal-Anzeigen auf einem Kartenterminal                                   | 661        |
| <b>11 Anhang K – Szenarien im dezentralen Umfeld</b>  | <b>664</b> |
| 11.1 Szenario 1: Einfache Installation ohne spezielle Anforderungen und ohne bestehende Infrastruktur | 664        |
| 11.1.1 Beschreibung des Szenarios   | 664        |
| 11.1.2 Voraussetzungen  | 665        |
| 11.1.3 Auswirkungen   | 665        |
| 11.2 Szenario 2: Installation mit mehreren Behandlungsräumen  | 666        |
| 11.2.1 Beschreibung des Szenarios   | 666        |
| 11.2.2 Voraussetzungen  | 666        |



|     |   |            |
|-----|---|------------|
| 412 | 11.2.3 Auswirkungen.....  | 667        |
| 413 | <b>11.3 Szenario 3: Integration in bestehende Infrastruktur ohne</b>        |            |
| 414 | <b>Netzsegmentierung.....</b>   | <b>667</b> |
| 415 | 11.3.1 Beschreibung des Szenarios.....                                      | 667        |
| 416 | 11.3.2 Voraussetzungen.....   | 668        |
| 417 | 11.3.3 Auswirkungen.....  | 668        |
| 418 | <b>11.4 Szenario 4: Integration in bestehende Infrastruktur mit</b>         |            |
| 419 | <b>Netzsegmentierung.....</b>   | <b>669</b> |
| 420 | 11.4.1 Beschreibung des Szenarios.....                                      | 669        |
| 421 | 11.4.2 Voraussetzungen.....   | 669        |
| 422 | 11.4.3 Auswirkungen.....  | 670        |
| 423 | <b>11.5 Szenario 5: Zentral gesteckter HBA.....</b>                         | <b>670</b> |
| 424 | 11.5.1 Beschreibung des Szenarios.....                                      | 670        |
| 425 | 11.5.2 Voraussetzungen.....   | 671        |
| 426 | 11.5.3 Auswirkung.....  | 671        |
| 427 | <b>11.6 Szenario 6: Installation mit zentralem PS.....</b>                  | <b>672</b> |
| 428 | 11.6.1 Beschreibung des Szenarios.....                                      | 672        |
| 429 | 11.6.2 Voraussetzungen.....   | 673        |
| 430 | 11.6.3 Auswirkungen.....  | 673        |
| 431 | <b>11.7 Szenario 7: Mehrere Mandanten.....</b>                              | <b>674</b> |
| 432 | 11.7.1 Beschreibung des Szenarios.....                                      | 674        |
| 433 | 11.7.2 Voraussetzungen.....   | 674        |
| 434 | 11.7.3 Auswirkungen.....  | 675        |
| 435 | <b>11.8 Szenario 9: Standalone Konnektor – Physische Trennung.....</b>      | <b>675</b> |
| 436 | 11.8.1 Beschreibung des Szenarios.....                                      | 676        |
| 437 | 11.8.2 Voraussetzungen.....   | 676        |
| 438 | 11.8.3 Auswirkung.....  | 677        |
| 439 | <b>12 Anhang L – Datentypen von Eingangs- und Ausgangsdaten..</b>           | <b>678</b> |
| 440 | <b>1 Einordnung des Dokumentes .....</b>                                    | <b>24</b>  |
| 441 | 1.1 Zielsetzung.....  | 24         |
| 442 | 1.2 Zielgruppe.....   | 24         |
| 443 | 1.3 Geltungsbereich.....  | 24         |
| 444 | 1.4 Abgrenzung des Dokuments.....   | 25         |
| 445 | 1.5 Methodik.....   | 25         |
| 446 | 1.5.1 Anforderungen.....  | 25         |
| 447 | 1.5.2 Offene Punkte.....  | 25         |
| 448 | 1.5.3 Erläuterungen zur Spezifikation des Außenverhaltens.....              | 25         |
| 449 | 1.5.4 Erläuterungen zur Dokumentenstruktur und „Dokumentenmechanismen“..... | 26         |
| 450 | 1.5.4.1 Modulare Spezifikation über Funktionsmerkmale.....                  | 26         |
| 451 | 1.5.4.2 Technische Use Cases - TUCs.....                                    | 27         |
| 452 | 1.5.4.3 Event-Mechanismus.....  | 29         |
| 453 | 1.5.4.4 Konfigurationsparameter und Zustandswerte.....                      | 29         |
| 454 | <b>2 Systemüberblick .....</b>  | <b>30</b>  |
| 455 | 2.1 Logische Struktur.....  | 32         |

|     |   |           |
|-----|---|-----------|
| 456 | <b>2.2 Sicherer Datenspeicher .....</b>                         | <b>34</b> |
| 457 | <b>2.3 Überblick Konnektoridentität.....</b>                    | <b>34</b> |
| 458 | <b>2.4 Mandantenfähigkeit.....</b>                              | <b>35</b> |
| 459 | <b>2.5 Versionierung .....</b>                                  | <b>35</b> |
| 460 | <b>2.6 Fachanwendungen.....</b>                                 | <b>35</b> |
| 461 | <b>2.7 Netzseitige Einsatzszenarien .....</b>                   | <b>36</b> |
| 462 | 2.7.1 Parameter ANLW_ANBINDUNGS_MODUS.....                      | 36        |
| 463 | 2.7.2 Parameter ANLW_INTERNET_MODUS.....                        | 36        |
| 464 | <b>2.8 Lokale und entfernte Kartenterminals .....</b>           | <b>37</b> |
| 465 | <b>2.9 Standalone-Szenario .....</b>                            | <b>37</b> |
| 466 | <b>3 Übergreifende Festlegungen .....</b>                       | <b>38</b> |
| 467 | <b>3.1 Konnektoridentität und gSMC-K .....</b>                  | <b>41</b> |
| 468 | 3.1.1 Organisatorische Anforderungen und Sperrprozesse.....     | 42        |
| 469 | <b>3.2 Bootup-Phase .....</b>                                   | <b>44</b> |
| 470 | <b>3.3 Betriebszustand.....</b>                                 | <b>45</b> |
| 471 | 3.3.1 Betriebsaspekte .....                                     | 58        |
| 472 | <b>3.4 Fachliche Anbindung der Clientsysteme .....</b>          | <b>59</b> |
| 473 | 3.4.1 Betriebsaspekte .....                                     | 63        |
| 474 | <b>3.5 Clientsystemschnittstelle .....</b>                      | <b>65</b> |
| 475 | 3.5.1 SOAP-Schnittstelle .....                                  | 65        |
| 476 | 3.5.2 Statusrückmeldung und Fehlerbehandlung.....               | 66        |
| 477 | 3.5.3 Transport großer Dokumente .....                          | 68        |
| 478 | <b>3.6 Verwendung manuell importierter CA-Zertifikate .....</b> | <b>69</b> |
| 479 | <b>3.7 Testunterstützung .....</b>                              | <b>70</b> |
| 480 | <b>4 Funktionsmerkmale .....</b>                                | <b>72</b> |
| 481 | <b>4.1 Anwendungskonnektor.....</b>                             | <b>72</b> |
| 482 | 4.1.1 Zugriffsberechtigungsdienst.....                          | 72        |
| 483 | 4.1.1.1 Funktionsmerkmalweite Aspekte.....                      | 72        |
| 484 | 4.1.1.2 Durch Ereignisse ausgelöste Reaktionen .....            | 83        |
| 485 | 4.1.1.3 Interne TUCs, nicht durch Fachmodule nutzbar .....      | 83        |
| 486 | 4.1.1.4 Interne TUCs, auch durch Fachmodule nutzbar .....       | 83        |
| 487 | 4.1.1.4.1 TUC_KON_000 „Prüfe Zugriffsberechtigung“ .....        | 83        |
| 488 | 4.1.1.5 Operationen an der Außenschnittstelle.....              | 95        |
| 489 | 4.1.1.6 Betriebsaspekte .....                                   | 95        |
| 490 | 4.1.2 Dokumentvalidierungsdienst .....                          | 95        |
| 491 | 4.1.2.1 Funktionsmerkmalweite Aspekte.....                      | 96        |
| 492 | 4.1.2.2 Durch Ereignisse ausgelöste Reaktionen .....            | 96        |
| 493 | 4.1.2.3 Interne TUCs, nicht durch Fachmodule nutzbar .....      | 96        |
| 494 | 4.1.2.4 Interne TUCs, auch durch Fachmodule nutzbar .....       | 96        |
| 495 | 4.1.2.4.1 TUC_KON_080 „Dokument validieren“ .....               | 96        |
| 496 | 4.1.2.5 Operationen an der Außenschnittstelle.....              | 99        |
| 497 | 4.1.2.6 Betriebsaspekte .....                                   | 99        |
| 498 | 4.1.3 Dienstverzeichnisdienst.....                              | 99        |

|     |   |     |
|-----|---|-----|
| 499 | 4.1.3.1 Funktionsmerkmalweite Aspekte.....                                | 99  |
| 500 | 4.1.3.2 Durch Ereignisse ausgelöste Reaktionen .....                      | 103 |
| 501 | 4.1.3.3 Interne TUCs, nicht durch Fachmodule nutzbar .....                | 103 |
| 502 | 4.1.3.4 Interne TUCs, auch durch Fachmodule nutzbar .....                 | 103 |
| 503 | 4.1.3.4.1 TUC_KON_041 „Einbringen der Endpunktinformationen während der   |     |
| 504 | Bootup-Phase“ .....   | 103 |
| 505 | 4.1.3.5 Operationen an der Außenschnittstelle.....                        | 104 |
| 506 | 4.1.3.6 Betriebsaspekte .....   | 105 |
| 507 | 4.1.4 Kartenterminaldienst .....  | 106 |
| 508 | 4.1.4.1 Funktionsmerkmalweite Aspekte.....                                | 110 |
| 509 | 4.1.4.2 Durch Ereignisse ausgelöste Reaktionen .....                      | 113 |
| 510 | 4.1.4.3 Interne TUCs, nicht durch Fachmodule nutzbar .....                | 114 |
| 511 | 4.1.4.3.1 TUC_KON_050 „Beginne Kartenterminalsitzung“ .....               | 114 |
| 512 | 4.1.4.3.2 TUC_KON_054 „Kartenterminal hinzufügen“ .....                   | 120 |
| 513 | 4.1.4.3.3 TUC_KON_053 „Paire Kartenterminal“ .....                        | 122 |
| 514 | 4.1.4.3.4 TUC_KON_055 „Befülle CT-Object“ .....                           | 127 |
| 515 | 4.1.4.4 Interne TUCs, auch durch Fachmodule nutzbar .....                 | 128 |
| 516 | 4.1.4.4.1 TUC_KON_051 „Mit Anwender über Kartenterminal interagieren“ ... | 128 |
| 517 | 4.1.4.4.2 TUC_KON_056 „Karte anfordern“ .....                             | 130 |
| 518 | 4.1.4.4.3 TUC_KON_057 „Karte auswerfen“ .....                             | 133 |
| 519 | 4.1.4.4.4 TUC_KON_058 „Displaygröße ermitteln“ .....                      | 135 |
| 520 | 4.1.4.5 Operationen an der Außenschnittstelle.....                        | 137 |
| 521 | 4.1.4.5.1 RequestCard .....   | 137 |
| 522 | 4.1.4.5.2 EjectCard .....   | 140 |
| 523 | 4.1.4.6 Betriebsaspekte .....   | 142 |
| 524 | 4.1.4.6.1 Allgemeine Betriebsaspekte .....                                | 142 |
| 525 | 4.1.4.6.2 Kartenterminals pflegen .....                                   | 144 |
| 526 | 4.1.4.6.3 Import der Kartenterminal-Informationen .....                   | 148 |
| 527 | 4.1.5 Kartendienst.....   | 149 |
| 528 | 4.1.5.1 Funktionsmerkmalweite Aspekte.....                                | 151 |
| 529 | 4.1.5.2 Durch Ereignisse ausgelöste Reaktionen .....                      | 156 |
| 530 | 4.1.5.3 Interne TUCs, nicht durch Fachmodule nutzbar .....                | 157 |
| 531 | 4.1.5.3.1 TUC_KON_001 „Karte öffnen“ .....                                | 157 |
| 532 | 4.1.5.4 Interne TUCs, auch durch Fachmodule nutzbar .....                 | 159 |
| 533 | 4.1.5.4.1 TUC_KON_026 „Liefere CardSession“ .....                         | 159 |
| 534 | 4.1.5.4.2 TUC_KON_012 „PIN verifizieren“ .....                            | 161 |
| 535 | 4.1.5.4.3 TUC_KON_019 „PIN ändern“ .....                                  | 166 |
| 536 | 4.1.5.4.4 TUC_KON_021 „PIN entsperren“ .....                              | 170 |
| 537 | 4.1.5.4.5 TUC_KON_022 „Liefere PIN-Status“ .....                          | 174 |
| 538 | 4.1.5.4.6 TUC_KON_027 „PIN-Schutz ein-/ausschalten“ .....                 | 176 |
| 539 | 4.1.5.4.7 TUC_KON_023 „Karte reservieren“ .....                           | 180 |
| 540 | 4.1.5.4.8 TUC_KON_005 „Card-to-Card authentisieren“ .....                 | 181 |
| 541 | 4.1.5.4.9 TUC_KON_202 „LeseDatei“ .....                                   | 186 |

|     |   |     |
|-----|---|-----|
| 542 | 4.1.5.4.10 TUC_KON_203 „SchreibeDatei“ .....                    | 188 |
| 543 | 4.1.5.4.11 TUC_KON_204 „LöscheDateiInhalt“ .....                | 190 |
| 544 | 4.1.5.4.12 TUC_KON_209 „LeseRecord“ .....                       | 192 |
| 545 | 4.1.5.4.13 TUC_KON_210 „SchreibeRecord“ .....                   | 194 |
| 546 | 4.1.5.4.14 TUC_KON_211 „LöscheRecordInhalt“ .....               | 196 |
| 547 | 4.1.5.4.15 TUC_KON_214 „FügeHinzuRecord“ .....                  | 198 |
| 548 | 4.1.5.4.16 TUC_KON_215 „SucheRecord“ .....                      | 200 |
| 549 | 4.1.5.4.17 TUC_KON_018 „eGK-Sperrung prüfen“ .....              | 202 |
| 550 | 4.1.5.4.18 TUC_KON_006 „Datenzugriffsaudit eGK schreiben“ ..... | 203 |
| 551 | 4.1.5.4.19 TUC_KON_218 „Signiere“ .....                         | 205 |
| 552 | 4.1.5.4.20 TUC_KON_219 „Entschlüssele“ .....                    | 207 |
| 553 | 4.1.5.4.21 TUC_KON_200 „SendeAPDU“ .....                        | 209 |
| 554 | 4.1.5.4.22 TUC_KON_024 „Karte zurücksetzen“ .....               | 210 |
| 555 | 4.1.5.4.23 TUC_KON_216 „LeseZertifikat“ .....                   | 212 |
| 556 | 4.1.5.4.24 TUC_KON_036 „LiefereFachlicheRolle“ .....            | 214 |
| 557 | 4.1.5.5 Operationen an der Außenschnittstelle.....              | 215 |
| 558 | 4.1.5.5.1 VerifyPin .....                                       | 216 |
| 559 | 4.1.5.5.2 ChangePin .....                                       | 219 |
| 560 | 4.1.5.5.3 GetPinStatus.....                                     | 222 |
| 561 | 4.1.5.5.4 UnblockPin .....                                      | 225 |
| 562 | 4.1.5.5.5 EnablePin .....                                       | 228 |
| 563 | 4.1.5.5.6 DisablePin .....                                      | 231 |
| 564 | 4.1.5.6 Betriebsaspekte .....                                   | 234 |
| 565 | 4.1.5.6.1 TUC_KON_025 "Initialisierung Kartendienst".....       | 234 |
| 566 | 4.1.5.6.2 Kartenübersicht und PIN-Management.....               | 235 |
| 567 | 4.1.6 Systeminformationsdienst .....                            | 236 |
| 568 | 4.1.6.1 Funktionsmerkmalweite Aspekte.....                      | 237 |
| 569 | 4.1.6.2 Durch Ereignisse ausgelöste Reaktionen .....            | 240 |
| 570 | 4.1.6.3 Interne TUCs, nicht durch Fachmodule nutzbar .....      | 240 |
| 571 | 4.1.6.4 Interne TUCs, auch durch Fachmodule nutzbar .....       | 240 |
| 572 | 4.1.6.4.1 TUC_KON_256 „Systemereignis absetzen“ .....           | 240 |
| 573 | 4.1.6.4.2 TUC_KON_252 „Liefere KT_Liste“ .....                  | 245 |
| 574 | 4.1.6.4.3 TUC_KON_253 „Liefere Karten_Liste“ .....              | 246 |
| 575 | 4.1.6.4.4 TUC_KON_254 „Liefere Ressourcendetails“ .....         | 248 |
| 576 | 4.1.6.5 Operationen an der Außenschnittstelle.....              | 250 |
| 577 | 4.1.6.5.1 GetCardTerminals.....                                 | 251 |
| 578 | 4.1.6.5.2 GetCards.....   | 254 |
| 579 | 4.1.6.5.3 GetResourceInformation .....                          | 259 |
| 580 | 4.1.6.5.4 Subscribe .....                                       | 263 |

|     |   |     |
|-----|---|-----|
| 581 | 4.1.6.5.5 Unsubscribe .....   | 266 |
| 582 | 4.1.6.5.6 RenewSubscriptions .....                                      | 268 |
| 583 | 4.1.6.5.7 GetSubscription .....   | 270 |
| 584 | 4.1.6.6 Betriebsaspekte .....   | 272 |
| 585 | 4.1.7 Verschlüsselungsdienst .....                                      | 273 |
| 586 | 4.1.7.1 Funktionsmerkmalweite Aspekte .....                             | 273 |
| 587 | 4.1.7.2 Durch Ereignisse ausgelöste Reaktionen .....                    | 275 |
| 588 | 4.1.7.3 Interne TUCs, nicht durch Fachmodule nutzbar .....              | 275 |
| 589 | 4.1.7.4 Interne TUCs, auch durch Fachmodule nutzbar .....               | 275 |
| 590 | 4.1.7.4.1 TUC_KON_070 „Daten hybrid verschlüsseln“ .....                | 275 |
| 591 | 4.1.7.4.2 TUC_KON_071 „Daten hybrid entschlüsseln“ .....                | 284 |
| 592 | 4.1.7.4.3 TUC_KON_072 „Daten symmetrisch verschlüsseln“ .....           | 289 |
| 593 | 4.1.7.4.4 TUC_KON_073 „Daten symmetrisch entschlüsseln“ .....           | 290 |
| 594 | 4.1.7.4.5 TUC_KON_075 „Symmetrisch verschlüsseln“ .....                 | 291 |
| 595 | 4.1.7.4.6 TUC_KON_076 „Symmetrisch entschlüsseln“ .....                 | 293 |
| 596 | 4.1.7.5 Operationen an der Außenschnittstelle .....                     | 294 |
| 597 | 4.1.7.5.1 EncryptDocument .....   | 294 |
| 598 | 4.1.7.5.2 DecryptDocument .....   | 300 |
| 599 | 4.1.7.6 Betriebsaspekte .....   | 303 |
| 600 | 4.1.8 Signaturdienst .....  | 303 |
| 601 | 4.1.8.1 Funktionsmerkmalweite Aspekte .....                             | 303 |
| 602 | 4.1.8.1.1 Dokumentensignatur .....                                      | 303 |
| 603 | 4.1.8.1.2 Signaturrichtlinien .....                                     | 310 |
| 604 | 4.1.8.1.3 Signaturzeitpunkt .....                                       | 310 |
| 605 | 4.1.8.1.4 Jobnummer .....   | 310 |
| 606 | 4.1.8.1.5 Komfortsignatur .....   | 312 |
| 607 | 4.1.8.2 Durch Ereignisse ausgelöste Reaktionen .....                    | 314 |
| 608 | 4.1.8.3 Interne TUCs, nicht durch Fachmodule nutzbar .....              | 314 |
| 609 | 4.1.8.3.1 TUC_KON_155 „Dokumente zur Signatur vorbereiten“ .....        | 315 |
| 610 | 4.1.8.3.2 TUC_KON_165 „Signaturvoraussetzungen für nonQES prüfen“ ..... | 319 |
| 611 | 4.1.8.3.3 TUC_KON_166 „nonQES Signaturen erstellen“ .....               | 320 |
| 612 | 4.1.8.3.4 TUC_KON_152 "Signaturvoraussetzungen für QES prüfen" .....    | 322 |
| 613 | 4.1.8.3.5 TUC_KON_154 "QES Signaturen erstellen" .....                  | 323 |
| 614 | 4.1.8.3.6 TUC_KON_168 „Einzelsignatur QES erstellen“ .....              | 327 |
| 615 | 4.1.8.3.7 TUC_KON_158 "Komfortsignaturen erstellen" .....               | 328 |
| 616 | 4.1.8.4 Interne TUCs, auch durch Fachmodule nutzbar .....               | 331 |
| 617 | 4.1.8.4.1 TUC_KON_160 „Dokumente nonQES signieren“ .....                | 331 |
| 618 | 4.1.8.4.2 TUC_KON_161 „nonQES Dokumentsignatur prüfen“ .....            | 337 |
| 619 | 4.1.8.4.3 TUC_KON_162 „Kryptographische Prüfung der XML-                |     |
| 620 | Dokumentensignatur“ .....   | 344 |
| 621 | 4.1.8.4.4 TUC_KON_150 „Dokumente QES signieren“ .....                   | 345 |

|     |  |     |
|-----|--|-----|
| 622 | 4.1.8.4.5 Anforderungen an die Stapelsignatur .....                  | 350 |
| 623 | 4.1.8.4.6 TUC_KON_151 „QES Dokumentensignatur prüfen“ .....          | 352 |
| 624 | 4.1.8.4.7 TUC_KON_170 „Dokumente mit Komfort signieren“ .....        | 358 |
| 625 | 4.1.8.4.8 TUC_KON_171 „Komfortsignatur einschalten“ .....            | 361 |
| 626 | 4.1.8.4.9 TUC_KON_172 „Komfortsignatur ausschalten“ .....            | 363 |
| 627 | 4.1.8.4.10 TUC_KON_173 „Liefere Signaturmodus“ .....                 | 365 |
| 628 | 4.1.8.5 Operationen an der Außenschnittstelle.....                   | 367 |
| 629 | 4.1.8.5.1 SignDocument (nonQES und QES) .....                        | 367 |
| 630 | 4.1.8.5.2 VerifyDocument (nonQES und QES) .....                      | 381 |
| 631 | 4.1.8.5.3 StopSignature .....  | 387 |
| 632 | 4.1.8.5.4 GetJobNumber .....   | 388 |
| 633 | 4.1.8.5.5 ActivateComfortSignature .....                             | 389 |
| 634 | 4.1.8.5.6 DeactivateComfortSignature.....                            | 391 |
| 635 | 4.1.8.5.7 GetSignatureMode .....                                     | 392 |
| 636 | 4.1.8.6 Betriebsaspekte .....  | 394 |
| 637 | 4.1.9 Zertifikatsdienst.....   | 396 |
| 638 | 4.1.9.1 Funktionsmerkmalweite Aspekte.....                           | 396 |
| 639 | 4.1.9.2 Durch Ereignisse ausgelöste Reaktionen .....                 | 402 |
| 640 | 4.1.9.3 Interne TUCs, nicht durch Fachmodule nutzbar .....           | 402 |
| 641 | 4.1.9.3.1 TUC_KON_032 „TSL aktualisieren“ .....                      | 402 |
| 642 | 4.1.9.3.2 TUC_KON_031 „BNetzA-VL aktualisieren“ .....                | 407 |
| 643 | 4.1.9.3.3 TUC_KON_040 „CRL aktualisieren“ .....                      | 408 |
| 644 | 4.1.9.3.4 TUC_KON_033 „Zertifikatsablauf prüfen“ .....               | 410 |
| 645 | 4.1.9.4 Interne TUCs, auch durch Fachmodule nutzbar .....            | 413 |
| 646 | 4.1.9.4.1 TUC_KON_037 „Zertifikat prüfen“ .....                      | 413 |
| 647 | 4.1.9.4.2 TUC_KON_042 „CV-Zertifikat prüfen“ .....                   | 418 |
| 648 | 4.1.9.4.3 TUC_KON_034 „Zertifikatsinformationen extrahieren“ .....   | 420 |
| 649 | 4.1.9.5 Operationen an der Außenschnittstelle.....                   | 423 |
| 650 | 4.1.9.5.1 CheckCertificateExpiration .....                           | 424 |
| 651 | 4.1.9.5.2 ReadCardCertificate .....                                  | 427 |
| 652 | 4.1.9.5.3 VerifyCertificate .....                                    | 431 |
| 653 | 4.1.9.6 Betriebsaspekte .....  | 433 |
| 654 | 4.1.9.6.1 TUC_KON_035 „Zertifikatsdienst initialisieren“ .....       | 433 |
| 655 | 4.1.10 Protokollierungsdienst.....                                   | 441 |
| 656 | 4.1.10.1 Funktionsmerkmalweite Aspekte .....                         | 442 |
| 657 | 4.1.10.2 Durch Ereignisse ausgelöste Reaktionen.....                 | 444 |
| 658 | 4.1.10.3 Interne TUCs, nicht durch Fachmodule nutzbar .....          | 444 |
| 659 | 4.1.10.4 Interne TUCs, auch durch Fachmodule nutzbar .....           | 444 |
| 660 | 4.1.10.4.1 TUC_KON_271 „Schreibe Protokolleintrag“ .....             | 444 |
| 661 | 4.1.10.5 Operationen an der Außenschnittstelle.....                  | 448 |
| 662 | 4.1.10.6 Betriebsaspekte.....  | 448 |
| 663 | 4.1.10.6.1 TUC_KON_272 „Initialisierung Protokollierungsdienst ..... | 450 |



|     |   |            |
|-----|---|------------|
| 664 | 4.1.11 TLS-Dienst .....   | 452        |
| 665 | 4.1.11.1 Funktionsmerkmalweite Aspekte .....                          | 452        |
| 666 | 4.1.11.2 Durch Ereignisse ausgelöste Reaktionen .....                 | 452        |
| 667 | 4.1.11.3 Interne TUCs, nicht durch Fachmodule nutzbar .....           | 452        |
| 668 | 4.1.11.4 Interne TUCs, auch durch Fachmodule nutzbar .....            | 452        |
| 669 | 4.1.11.4.1 TUC_KON_110 „Kartenbasierte TLS-Verbindung aufbauen“ ..... | 452        |
| 670 | 4.1.11.4.2 TUC_KON_111 „Kartenbasierte TLS-Verbindung abbauen“ .....  | 454        |
| 671 | 4.1.11.5 Operationen an der Außenschnittstelle .....                  | 455        |
| 672 | 4.1.11.6 Betriebsaspekte .....  | 455        |
| 673 | 4.1.12 LDAP-Proxy .....   | 455        |
| 674 | 4.1.12.1 Funktionsmerkmalweite Aspekte .....                          | 455        |
| 675 | 4.1.12.2 Durch Ereignisse ausgelöste Reaktionen .....                 | 455        |
| 676 | 4.1.12.3 Interne TUCs, nicht durch Fachmodule nutzbar .....           | 455        |
| 677 | 4.1.12.4 Interne TUCs, auch durch Fachmodule nutzbar .....            | 456        |
| 678 | 4.1.12.4.1 TUC_KON_290 „LDAP-Verbindung aufbauen“ .....               | 456        |
| 679 | 4.1.12.4.2 TUC_KON_291 „Verzeichnis abfragen“ .....                   | 457        |
| 680 | 4.1.12.4.3 TUC_KON_292 „LDAP-Verbindung trennen“ .....                | 457        |
| 681 | 4.1.12.4.4 TUC_KON_293 „Verzeichnisabfrage abbrechen“ .....           | 459        |
| 682 | 4.1.12.5 Operationen an der Außenschnittstelle .....                  | 459        |
| 683 | 4.1.12.5.1 Unterstützte LDAPv3 Operationen .....                      | 459        |
| 684 | 4.1.12.6 Betriebsaspekte .....  | 460        |
| 685 | 4.1.13 Authentifizierungsdienst .....                                 | 460        |
| 686 | 4.1.13.1 Funktionsmerkmalweite Aspekte .....                          | 460        |
| 687 | 4.1.13.1.1 Externe Authentisierung .....                              | 460        |
| 688 | 4.1.13.2 Durch Ereignisse ausgelöste Reaktionen .....                 | 461        |
| 689 | 4.1.13.3 Interne TUCs .....   | 461        |
| 690 | 4.1.13.4 Operationen an der Außenschnittstelle .....                  | 461        |
| 691 | 4.1.13.4.1 ExternalAuthenticate .....                                 | 461        |
| 692 | 4.1.13.5 Betriebsaspekte .....  | 465        |
| 693 | 4.1.14 Betriebsdatenmeldedienst .....                                 | 466        |
| 694 | <b>4.2 Netzkonnektor .....</b>  | <b>466</b> |
| 695 | 4.2.1 Anbindung LAN/WAN .....   | 466        |
| 696 | 4.2.1.1 Funktionsmerkmalweite Aspekte .....                           | 467        |
| 697 | 4.2.1.1.1 Netzwerksegmentierung .....                                 | 468        |
| 698 | 4.2.1.1.2 Routing und Firewall .....                                  | 470        |
| 699 | 4.2.1.2 Durch Ereignisse ausgelöste Reaktionen .....                  | 479        |
| 700 | 4.2.1.3 Interne TUCs, nicht durch Fachmodule nutzbar .....            | 480        |
| 701 | 4.2.1.3.1 TUC_KON_305 „LAN-Adapter initialisieren“ .....              | 480        |
| 702 | 4.2.1.3.2 TUC_KON_306 „WAN-Adapter initialisieren“ .....              | 481        |
| 703 | 4.2.1.3.3 TUC_KON_304 „Netzwerk-Routen einrichten“ .....              | 483        |
| 704 | 4.2.1.4 Interne TUCs, auch durch Fachmodule nutzbar .....             | 485        |
| 705 | 4.2.1.5 Operationen an der Außenschnittstelle .....                   | 485        |
| 706 | 4.2.1.6 Betriebsaspekte .....   | 485        |
| 707 | 4.2.2 DHCP-Server .....   | 492        |
| 708 | 4.2.2.1 Funktionsmerkmalweite Aspekte .....                           | 493        |
| 709 | 4.2.2.2 Durch Ereignisse ausgelöste Reaktionen .....                  | 493        |
| 710 | 4.2.2.3 Interne TUCs, nicht durch Fachmodule nutzbar .....            | 493        |

|     |   |            |
|-----|---|------------|
| 711 | 4.2.2.4 Interne TUCs, auch durch Fachmodule nutzbar .....         | 493        |
| 712 | 4.2.2.5 Operationen an der Außenschnittstelle.....                | 493        |
| 713 | 4.2.2.5.1 Liefere Netzwerkinformationen über DHCP .....           | 493        |
| 714 | 4.2.2.6 Betriebsaspekte .....                                     | 494        |
| 715 | 4.2.2.6.1 TUC_KON_343 „Initialisierung DHCP-Server“ .....         | 498        |
| 716 | 4.2.3 DHCP-Client .....   | 499        |
| 717 | 4.2.3.1 Funktionsmerkmalweite Aspekte.....                        | 499        |
| 718 | 4.2.3.2 Durch Ereignisse ausgelöste Reaktionen .....              | 499        |
| 719 | 4.2.3.3 Interne TUCs, nicht durch Fachmodule nutzbar .....        | 499        |
| 720 | 4.2.3.3.1 TUC_KON_341 „DHCP-Informationen beziehen“ .....         | 499        |
| 721 | 4.2.3.4 Interne TUCs, auch durch Fachmodule nutzbar .....         | 501        |
| 722 | 4.2.3.5 Operationen an der Außenschnittstelle.....                | 501        |
| 723 | 4.2.3.6 Betriebsaspekte .....                                     | 501        |
| 724 | 4.2.4 VPN-Client .....  | 502        |
| 725 | 4.2.4.1 Funktionsmerkmalweite Aspekte.....                        | 502        |
| 726 | 4.2.4.2 Durch Ereignisse ausgelöste Reaktionen .....              | 502        |
| 727 | 4.2.4.3 Interne TUCs, nicht durch Fachmodule nutzbar .....        | 503        |
| 728 | 4.2.4.3.1 TUC_KON_321 „Verbindung zu dem VPN-Konzentrator der TI  |            |
| 729 | aufbauen“ .....   | 503        |
| 730 | 4.2.4.3.2 TUC_KON_322 „Verbindung zu dem VPN-Konzentrator des SIS |            |
| 731 | aufbauen“ .....   | 506        |
| 732 | 4.2.4.4 Interne TUCs, auch durch Fachmodule nutzbar .....         | 508        |
| 733 | 4.2.4.5 Operationen an der Außenschnittstelle.....                | 508        |
| 734 | 4.2.4.6 Betriebsaspekte .....                                     | 508        |
| 735 | 4.2.5 Zeitdienst .....  | 510        |
| 736 | 4.2.5.1 Funktionsmerkmalweite Aspekte.....                        | 510        |
| 737 | 4.2.5.2 Durch Ereignisse ausgelöste Reaktionen .....              | 511        |
| 738 | 4.2.5.3 Interne TUCs, nicht durch Fachmodule nutzbar .....        | 511        |
| 739 | 4.2.5.4 Interne TUCs, auch durch Fachmodule nutzbar .....         | 511        |
| 740 | 4.2.5.4.1 TUC_KON_351 „Liefere Systemzeit“ .....                  | 511        |
| 741 | 4.2.5.5 Operationen an der Außenschnittstelle.....                | 512        |
| 742 | 4.2.5.5.1 Sync_Time .....   | 512        |
| 743 | 4.2.5.6 Betriebsaspekte .....                                     | 513        |
| 744 | 4.2.5.6.1 TUC_KON_352 Initialisierung Zeitdienst.....             | 514        |
| 745 | 4.2.6 Namensdienst und Dienstlokalisierung .....                  | 515        |
| 746 | 4.2.6.1 Funktionsmerkmalweite Aspekte.....                        | 515        |
| 747 | 4.2.6.2 Durch Ereignisse ausgelöste Reaktionen .....              | 517        |
| 748 | 4.2.6.3 Interne TUCs, nicht durch Fachmodule nutzbar .....        | 517        |
| 749 | 4.2.6.4 Interne TUCs, auch durch Fachmodule nutzbar .....         | 517        |
| 750 | 4.2.6.4.1 TUC_KON_361 „DNS-Namen auflösen“ .....                  | 517        |
| 751 | 4.2.6.4.2 TUC_KON_362 „Liste der Dienste abrufen“ .....           | 520        |
| 752 | 4.2.6.4.3 TUC_KON_363 „Dienstdetails abrufen“ .....               | 521        |
| 753 | 4.2.6.5 Operationen an der Außenschnittstelle.....                | 522        |
| 754 | 4.2.6.5.1 GetIPAddress .....                                      | 522        |
| 755 | 4.2.6.6 Betriebsaspekte .....                                     | 523        |
| 756 | 4.2.7 Optionale Verwendung von IPv6.....                          | 525        |
| 757 | <b>4.3 Konnektormanagement .....</b>                              | <b>525</b> |
| 758 | 4.3.1 Zugang und Benutzerverwaltung des Konnektormanagements..... | 527        |



|     |  |            |
|-----|--|------------|
| 759 | 4.3.2 Konnektorname und Versionsinformationen .....                      | 530        |
| 760 | 4.3.3 Konfigurationsdaten: Persistieren sowie Export-Import .....        | 531        |
| 761 | 4.3.4 Administration von Fachmodulen .....                               | 533        |
| 762 | 4.3.5 Neustart und Werksreset .....                                      | 534        |
| 763 | 4.3.6 Leistungsumfänge und Standalone-Szenarios .....                    | 534        |
| 764 | 4.3.7 Online-Anbindung verwalten .....                                   | 536        |
| 765 | 4.3.8 Remote Management (Optional) .....                                 | 539        |
| 766 | 4.3.9 Software- und Konfigurationsaktualisierung (KSR-Client) .....      | 544        |
| 767 | 4.3.9.1 Funktionsmerkmalweite Aspekte .....                              | 544        |
| 768 | 4.3.9.2 Durch Ereignisse ausgelöste Reaktionen .....                     | 545        |
| 769 | 4.3.9.3 Interne TUCs, nicht durch Fachmodule nutzbar .....               | 545        |
| 770 | 4.3.9.3.1 TUC_KON_280 „Konnektoraktualisierung durchführen“ .....        | 545        |
| 771 | 4.3.9.3.2 TUC_KON_281 „Kartenterminalaktualisierung anstoßen“ .....      | 549        |
| 772 | 4.3.9.3.3 TUC_KON_282 „UpdateInformationen beziehen“ .....               | 551        |
| 773 | 4.3.9.3.4 TUC_KON_283 „Infrastruktur Konfiguration aktualisieren“ .....  | 553        |
| 774 | 4.3.9.4 Interne TUCs, auch durch Fachmodule nutzbar .....                | 557        |
| 775 | 4.3.9.4.1 TUC_KON_285 „UpdateInformationen für Fachmodul beziehen“ ..... | 557        |
| 776 | 4.3.9.4.2 TUC_KON_286 „Paket für Fachmodul laden“ .....                  | 559        |
| 777 | 4.3.9.5 Operationen an der Außenschnittstelle .....                      | 561        |
| 778 | 4.3.9.6 Betriebsaspekte .....  | 561        |
| 779 | 4.3.9.6.1 TUC_KON_284 KSR-Client initialisieren .....                    | 561        |
| 780 | 4.3.10 Konnektorstatus .....   | 569        |
| 781 | <b>4.4 Hardware-Merkmale des Konnektors .....</b>                        | <b>569</b> |
| 782 | <b>5 Anhang A – Verzeichnisse .....</b>                                  | <b>573</b> |
| 783 | <b>5.1 Abkürzungen .....</b>   | <b>573</b> |
| 784 | <b>5.2 Glossar .....</b>   | <b>575</b> |
| 785 | <b>5.3 Abbildungsverzeichnis .....</b>                                   | <b>575</b> |
| 786 | <b>5.4 Tabellenverzeichnis .....</b>                                     | <b>577</b> |
| 787 | <b>5.5 Referenzierte Dokumente .....</b>                                 | <b>602</b> |
| 788 | 5.5.1 Dokumente der gematik .....  | 602        |
| 789 | 5.5.2 Weitere Dokumente .....  | 603        |
| 790 | <b>6 Anhang B – Profilierung der Signatur- und</b>                       |            |
| 791 | <b>Verschlüsselungsformate (normativ) .....</b>                          | <b>611</b> |
| 792 | <b>6.1 Profilierung der Verschlüsselungsformate .....</b>                | <b>611</b> |
| 793 | <b>6.2 Profilierung der Signaturformate .....</b>                        | <b>611</b> |
| 794 | <b>6.3 Profilierung VerificationReport .....</b>                         | <b>613</b> |
| 795 | <b>6.4 Profilierung der Dokumentenformate und Nachrichten .....</b>      | <b>618</b> |
| 796 | <b>7 Anhang F – Übersicht Events .....</b>                               | <b>630</b> |
| 797 | <b>8 Anhang H – Mapping von „Architektur der TI-Plattform“ auf</b>       |            |
| 798 | <b>Konnektorspezifikation .....</b>                                      | <b>648</b> |

|     |   |            |
|-----|---|------------|
| 799 | <b>9 Anhang I – Umsetzungshinweise (informativ).....</b>                            | <b>657</b> |
| 800 | <b>9.1 Systemüberblick .....</b>  | <b>657</b> |
| 801 | 9.1.1 – Hinweise zur Sicherheitsevaluierung nach Common Criteria .....              | 657        |
| 802 | 9.1.1.1 Separationsmechanismen des Konnektors.....                                  | 657        |
| 803 | 9.1.1.2 Granularität der TSF .....  | 658        |
| 804 | <b>9.2 Übergreifende Festlegungen.....</b>  | <b>659</b> |
| 805 | 9.2.1 Interne Mechanismen .....   | 659        |
| 806 | 9.2.1.1 Zufallszahlen und Schlüssel .....   | 659        |
| 807 | <b>9.3 Funktionsmerkmale .....</b>  | <b>659</b> |
| 808 | 9.3.1 Anwendungskonnektor.....  | 659        |
| 809 | 9.3.1.1 Administration des Informationsmodells .....                                | 659        |
| 810 | 9.3.1.2 Vorgehensvariante für das Handling von CardSessions.....                    | 660        |
| 811 | 9.3.1.3 Darstellung von Terminal-Anzeigen auf einem Kartenterminal.....             | 661        |
| 812 | <b>10 Anhang K – Szenarien im dezentralen Umfeld .....</b>                          | <b>664</b> |
| 813 | <b>10.1 Szenario 1: Einfache Installation ohne spezielle Anforderungen und ohne</b> |            |
| 814 | <b>bestehende Infrastruktur .....</b>   | <b>664</b> |
| 815 | 10.1.1 Beschreibung des Szenarios.....  | 664        |
| 816 | 10.1.2 Voraussetzungen.....   | 665        |
| 817 | 10.1.3 Auswirkungen .....   | 665        |
| 818 | <b>10.2 Szenario 2: Installation mit mehreren Behandlungsräumen.....</b>            | <b>666</b> |
| 819 | 10.2.1 Beschreibung des Szenarios.....  | 666        |
| 820 | 10.2.2 Voraussetzungen.....   | 666        |
| 821 | 10.2.3 Auswirkungen .....   | 667        |
| 822 | <b>10.3 Szenario 3: Integration in bestehende Infrastruktur ohne</b>                |            |
| 823 | <b>Netzsegmentierung .....</b>  | <b>667</b> |
| 824 | 10.3.1 Beschreibung des Szenarios.....  | 667        |
| 825 | 10.3.2 Voraussetzungen.....   | 668        |
| 826 | 10.3.3 Auswirkungen .....   | 668        |
| 827 | <b>10.4 Szenario 4: Integration in bestehende Infrastruktur mit</b>                 |            |
| 828 | <b>Netzsegmentierung .....</b>  | <b>669</b> |
| 829 | 10.4.1 Beschreibung des Szenarios.....  | 669        |
| 830 | 10.4.2 Voraussetzungen.....   | 669        |
| 831 | 10.4.3 Auswirkungen .....   | 670        |
| 832 | <b>10.5 Szenario 5: Zentral gesteckter HBA .....</b>                                | <b>670</b> |
| 833 | 10.5.1 Beschreibung des Szenarios.....  | 670        |
| 834 | 10.5.2 Voraussetzungen.....   | 671        |
| 835 | 10.5.3 Auswirkung .....   | 671        |
| 836 | <b>10.6 Szenario 6: Installation mit zentralem PS .....</b>                         | <b>672</b> |
| 837 | 10.6.1 Beschreibung des Szenarios.....  | 672        |
| 838 | 10.6.2 Voraussetzungen.....   | 673        |
| 839 | 10.6.3 Auswirkungen .....   | 673        |
| 840 | <b>10.7 Szenario 7: Mehrere Mandanten .....</b>                                     | <b>674</b> |
| 841 | 10.7.1 Beschreibung des Szenarios.....  | 674        |
| 842 | 10.7.2 Voraussetzungen.....   | 674        |
| 843 | 10.7.3 Auswirkungen .....   | 675        |
| 844 | <b>10.8 Szenario 9: Standalone Konnektor - Physische Trennung .....</b>             | <b>675</b> |
| 845 | 10.8.1 Beschreibung des Szenarios.....  | 676        |

|     |   |            |
|-----|---|------------|
| 846 | 10.8.2 Voraussetzungen.....                                       | 676        |
| 847 | 10.8.3 Auswirkung .....   | 677        |
| 848 | <b>11 Anhang L – Datentypen von Eingangs- und Ausgangsdaten..</b> | <b>678</b> |
| 849 |   |            |

ENTWURF

850

## 1 Einordnung des Dokumentes

### 1.1 Zielsetzung

852 Die vorliegende Spezifikation definiert die Anforderungen zu Herstellung, Test und  
853 Betrieb des Produkttyps Konnektor.

854 Dieses Dokument beschreibt die dezentrale Komponente zur sicheren Anbindung von  
855 Clientsystemen der Institutionen und Organisationen des Gesundheitswesens an die  
856 Telematikinfrastruktur – den Konnektor. Der Konnektor ist einerseits verantwortlich für  
857 den Zugriff auf die in der Einsatzumgebung befindlichen Kartenterminals sowie Karten  
858 und andererseits für die Kommunikation mit den zentralen Diensten der TI-Plattform und  
859 fachanwendungsspezifischen Diensten. Aus den Kommunikationsbeziehungen mit  
860 Clientsystem, Kartenterminals, Karten und zentralen Diensten der TI-Plattform und  
861 fachanwendungsspezifischen Diensten resultieren vom Konnektor anzubietende  
862 Schnittstellen, die gemeinsam in diesem Dokument sowie den  
863 fachanwendungsspezifischen Fachmodulspezifikationen normativ geregelt werden. Vom  
864 Konnektor genutzte Schnittstellen liegen zumeist in anderen Verantwortungsbereichen  
865 (zentrale TI-Plattform aber auch Schnittstellen der Kartenterminals und Karten). Diese  
866 werden in den übergreifenden Spezifikationen der TI sowie den Produkttypspezifikationen  
867 definiert.

868 Dieses Dokument regelt somit nur einen Teil des Konnektors (wenngleich auch den  
869 Wesentlichen). Für die Implementierung eines Konnektors ist entsprechend die Kenntnis  
870 aller weiteren Spezifikationen erforderlich. Die Gesamtheit aller für den Konnektor  
871 relevanten Dokumente wird im Produkttypsteckbrief des Konnektors erhoben.

### 1.2 Zielgruppe

873 Das Dokument richtet sich an Konnektorhersteller sowie Hersteller und Anbieter von  
874 Produkttypen (dies beinhaltet auch die Anbieter zur G2-Ausschreibung), die hierzu eine  
875 Schnittstelle besitzen.

### 1.3 Geltungsbereich

877 Dieses Dokument enthält normative Festlegungen zur Telematikinfrastruktur des  
878 Deutschen Gesundheitswesens. Der Gültigkeitszeitraum der vorliegenden Version und  
879 deren Anwendung in Zulassungsverfahren werden durch die gematik GmbH in  
880 gesonderten Dokumenten (z. B. Dokumentenlandkarte, Produkttypsteckbrief,  
881 Leistungsbeschreibung) festgelegt und bekannt gegeben.

### Wichtiger Schutzrechts-/Patentrechtshinweis

883 *Die nachfolgende Spezifikation ist von der gematik allein unter technischen*  
884 *Gesichtspunkten erstellt worden. Im Einzelfall kann nicht ausgeschlossen werden, dass*  
885 *die Implementierung der Spezifikation in technische Schutzrechte Dritter eingreift. Es ist*  
886 *allein Sache des Anbieters oder Herstellers, durch geeignete Maßnahmen dafür Sorge zu*  
887 *tragen, dass von ihm aufgrund der Spezifikation angebotene Produkte und/oder*  
888 *Leistungen nicht gegen Schutzrechte Dritter verstoßen und sich ggf. die erforderlichen*

889 Erlaubnisse/Lizenzen von den betroffenen Schutzrechtsinhabern einzuholen. Die gematik  
890 GmbH übernimmt insofern keinerlei Gewährleistungen.

## 891 1.4 Abgrenzung des Dokuments

892 Spezifiziert werden in dem Dokument die von dem Produkttyp bereitgestellten  
893 (angebotenen) Schnittstellen. Benutzte Schnittstellen werden hingegen in der  
894 Spezifikation desjenigen Produkttypen beschrieben, der diese Schnittstelle bereitstellt.  
895 Auf die entsprechenden Dokumente wird referenziert.

896 Die vollständige Anforderungslage für den Produkttyp ergibt sich aus weiteren Konzept-  
897 und Spezifikationsdokumenten, diese sind in dem Produkttypsteckbrief des Produkttyps  
898 Konnektor verzeichnet.

## 899 1.5 Methodik

### 900 1.5.1 Anforderungen

901 Anforderungen als Ausdruck normativer Festlegungen werden durch eine eindeutige ID  
902 sowie die dem RFC 2119 [RFC2119] entsprechenden, in Großbuchstaben geschriebenen  
903 deutschen Schlüsselworte MUSS, DARF NICHT, SOLL, SOLL NICHT, KANN  
904 gekennzeichnet.

905 Sie werden im Dokument wie folgt dargestellt:

906 **<AFO-ID> - <Titel der Afo>**

907 Text / Beschreibung

908 [**<=>**]

909

910 Dabei umfasst die Anforderung sämtliche innerhalb der Afo-ID und der Textmarke  
911 angeführten Inhalte.

### 912 1.5.2 Offene Punkte

913 Zum Zeitpunkt der Spezifikationserstellung konnten nicht alle Details abschließend  
914 geklärt werden, insbesondere, da Abstimmungsbedarf mit der umsetzenden Industrie  
915 besteht. Details, die keine produkttypübergreifenden Auswirkungen haben und die im  
916 Rahmen des Verhandlungsverfahrens mit der Industrie besprochen werden müssen,  
917 werden als „Offene Punkte“ ausgewiesen und wie folgt im Dokument kenntlich gemacht:

*Die XYZ müssen noch definiert werden.*

918

919

### 920 1.5.3 Erläuterungen zur Spezifikation des Außenverhaltens

921 Der Konnektor stellt einen vergleichsweise komplexen Produkttyp dar, dessen  
922 Beschreibung eine Herausforderung darstellt und somit in vielen verschiedenen Varianten

923 möglich wäre. An dieser Stelle folgen daher wesentliche Informationen, die das korrekte  
924 Verstehen der Spezifikation fördern:

925 Die Spezifikation des Konnektors ist eine Black-Box-Spezifikation, das heißt alle  
926 Festlegungen dienen ausschließlich der Beschreibung des von der Komponente  
927 verlangten Verhaltens an der Außenschnittstelle.

928 Normative Festlegungen, die eine Festlegung des inneren Verhalten vermuten lassen  
929 (beispielsweise die Definitionen der Technischen Use Cases - TUCs) sind nur in so weit  
930 normativ, wie ihre Festlegungen auf die Außenschnittstelle wirken. Sie legen explizit nicht  
931 die intern zu verwendende Implementierung fest. Die Notwendigkeit für diese Art der  
932 „scheinbaren internen Beschreibung“ ergibt sich aus der Komplexität der  
933 Gesamtkomponente, sowie dem Bedarf, wiederholt ähnlich Verhaltensweisen in  
934 Außenschnittstellen darstellen zu müssen. In diesem Fall werden die sich wiederholenden  
935 Verhaltensanteile in internen TUCs zur editoriiellen Wiederverwendung gekapselt. Die  
936 konkrete konnektorinterne Modularisierung bleibt dem Hersteller freigestellt.  
937 Insbesondere bleibt es dem Hersteller freigestellt, intern bereits Mechanismen für  
938 kommende Releases zu realisieren, sofern diese an der Außenschnittstelle keine  
939 Auswirkung zeigen.

940 Die einzige Abweichung von dieser Vorgehensweise ergibt sich für Sicherheitsaspekte.  
941 Hier können interne Vorgänge normativ gefordert sein, die sich an der Außenschnittstelle  
942 nicht manifestieren (Beispiel „Verpflichtung auf sicheres Löschen eines temporären  
943 Schlüssels nach Gebrauch“). In diesem Fall erfolgt die Überprüfung der Einhaltung dieser  
944 Anforderungen im Rahmen der CC-Evaluierung.

#### 945 **1.5.4 Erläuterungen zur Dokumentenstruktur und** 946 **„Dokumentenmechanismen“**

##### 947 **1.5.4.1 Modulare Spezifikation über Funktionsmerkmale**

948 Die Beschreibung des Konnektors erfolgt soweit wie möglich modular, d. h. alle Aspekte,  
949 die für einen logischen Bereich relevant sind, werden in einem Kapitel beschrieben. Diese  
950 logischen Bereiche werden als Funktionsmerkmal bezeichnet.

951 Funktionsmerkmale kennzeichnet ein eigener Verantwortungsbereich. In diesen  
952 Verantwortungsbereich greifen keine anderen Funktionsmerkmale ein. So kann ein  
953 logischer Bereich vollständig durchdrungen werden, ohne dass in anderen Kapiteln  
954 Anforderungen zu erwarten wären, die das Verhalten des Funktionsmerkmals  
955 beeinflussen. Da zwischen Funktionsmerkmalen Wechselwirkungen bestehen (Die  
956 Erkennung einer gesteckten Karte im Kartenterminaldienst löst eine Reaktion im  
957 Kartendienst aus), wurden zur „dokumententechnischen Interaktion“ zwischen  
958 Funktionsmerkmalen ein interner Event-Mechanismus sowie Konfigurationsparameter  
959 und Zustandswerte eingeführt (siehe Folgekapitel).

960 Funktionsmerkmale bestehen (bis auf wenige Ausnahmen) immer aus folgenden  
961 Unterkapiteln:

- 962 1. Funktionsmerkmalweite Aspekte
- 963 2. Durch Ereignisse ausgelöste Reaktionen
- 964 3. Interne TUCs, **nicht** durch Fachmodule nutzbar
- 965 4. Interne TUCs, **auch** durch Fachmodule nutzbar
- 966 5. Operationen an der Außenschnittstelle
- 967 6. Betriebsaspekte

968 Die Unterkapitel 1-5 dienen der funktionalen Beschreibung des Funktionsmerkmals.  
 969 Punkte, die zum Funktionieren des Funktionsmerkmals relevant sind:  
 970 Initialisierungsaspekte, durch den Administrator festzulegenden Konfigurationsparameter  
 971 etc., werden im Unterkapitel Betriebsaspekte erfasst.  
 972 In jedem Funktionsmerkmal sind immer alle Unterkapitel enthalten, auch wenn es im  
 973 konkreten Einzelfall dort keine Inhalte gibt. Diese feste Struktur innerhalb der  
 974 Funktionsmerkmale erleichtert die Orientierung und erhöht somit die Lesbarkeit.

#### 975 **1.5.4.2 Technische Use Cases - TUCs**

976 Innerhalb der Funktionsmerkmale in Kapitel 4 erfolgt eine Unterscheidung der TUCs in  
 977 solche, die nur durch die Basisdienste des Konnektors aufgerufen werden dürfen (rein  
 978 interne TUCs) und solche die neben den Basisdiensten auch durch Fachmodule genutzt  
 979 werden dürfen. Diese Unterteilung ergibt sich ausschließlich aus dem Bedarf der  
 980 editoriellen Steuerung der verschiedenen Spezifikationen (Konnektor- und  
 981 Fachmodulspezifikationen). Es besteht im Rahmen der Implementierung des Konnektors  
 982 keine Anforderung diese Trennung intern durchzusetzen.

983 Die Beschreibung der TUCs erfolgt nach folgendem Muster:

- 984 • TUC-Tabelle
- 985 • Aktivitäts- oder Sequenzdiagramm (optional)
- 986 • Fehlercodetabelle

987 Dabei wird innerhalb der TUC-Tabelle in der Zeile „Standardablauf“ ausschließlich der  
 988 Gut-Durchlauf beschrieben. Fehler, die innerhalb dieses Ablaufs auftreten können,  
 989 werden in der Zeile „Fehlerfälle“ erhoben. Dabei wird auf die jeweilige Schrittnummer  
 990 innerhalb des Ablaufs referenziert. In dieser Tabellenzeile werden nur Fehlercodes  
 991 erhoben, die im jeweiligen Fehlerfall geworfen werden müssen. Die genauen  
 992 Festlegungen zu den Fehlern, neben Fehlercode auch: ErrorType, Severity und  
 993 Fehlertext, werden in der Fehlercodetabelle festgelegt.

994 Die Spezifikation, in der ein TUC definiert wird, ist an den mittleren drei Buchstaben der  
 995 TUC-Referenz zu erkennen:

- 996 • TUC\_KON\_xxx entsprechend in dieser Konnektorspezifikation
- 997 • TUC\_PKI\_xxx in der PKI-Spezifikation [gemSpec\_PKI]
- 998 • TUC\_VPN\_ZD-xxxx in der Spezifikation des VPN-Zugangsdienstes  
 999 [gemSpec\_VPN\_ZugD]
- 1000 • TUC\_VZD\_xxx in der Spezifikation des Verzeichnisdienstes [gemSpec\_VZD]

#### 1001 **Festlegungen zur Schreibweise von Eingangs- und Ausgangsdaten von TUCs**

1002 a) Eingangs- und Ausgangsparameter werden in TUC-Tabellen wie folgt beschrieben:

1003 Name des Eingangs- bzw. Ausgangsparameters

1004 gefolgt von (falls definiert): [Datentyp]

1005 gefolgt von (falls zutreffend):

- 1006 - *optional*; *default*: <Defaultwert> bzw.
- 1007 - *optional*; <erklärender Text>



1008 Hierbei bedeuten:

1009

1010 - *optional*; kennzeichnet optionale Ein- und Ausgangsparameter

1011 *default*: <Defaultwert> definiert den Defaultwert für den Fall, dass der

1012 Eingangsparameter leer ist bzw. nicht übergeben wurde

1013 /<erklärender Text> beschreibt Bedingungen, unter denen der

1014 Eingangsparameter optional ist

1015 gefolgt von (falls vorhanden): (<erklärender Text>)

1016 b) Namen mit kleinem Anfangsbuchstaben bezeichnen Ein- und Ausgangsparameter;

1017 Namen mit großem Anfangsbuchstaben bezeichnen Datentypen.

1018 Beispiel:

|               |  |
|---------------|--|
| Eingangsdaten | <ul style="list-style-type: none"> <li>• mandantId</li> <li>• allWorkplaces [Boolean] – <i>optional</i>; <i>default</i>: <i>false</i><br/>(Dieser Schalter gibt an, ob eine mandantenweite Zugriffsberechtigung zum Tragen kommt...)</li> <li>• userId – <i>optional/verpflichtend</i>, wenn <i>cardType = HBAX</i></li> </ul> |
| Ausgangsdaten | <ul style="list-style-type: none"> <li>• pinStatus [PinStatus]</li> <li>• leftTries – <i>optional/verpflichtend</i>, wenn pinStatus = VERIFYABLE<br/>(Anzahl der verbleibenden Versuche für die Verifikation der PIN)</li> </ul>   |

1019

1020

1021 Die im Dokument verwendeten Datentypen sind definiert in [Anhang L – Datentypen von

1022 Eingangs- und Ausgangsdaten].

### 1023 Festlegungen zur Schreibweise des Aufrufs von TUCs

1024 Ein TUC-Aufruf erfolgt nach folgendem Muster:

1025 <TUC-Bezeichner> {

1026     <TUC-Eingangsparameter Name> = <TUC Eingangsparameter Wert>;

1027     ... }

1028 Beispiel:

1029 TUC\_KON\_256 {

1030     topic = „CT/DISCONNECTED“;

1031     eventType = Op;

1032     severity = Info;

1033     parameters = („CtID=\$CT.CTID, Hostname=\$CT.HOSTNAME“) }

1034 Vereinfachung:



1035 Ist <TUC-Eingangsparameter Name> des aufzurufenden TUCs gleich der Variablen, die  
1036 als < TUC Eingangsparameter Wert> gesetzt wird, so kann dieser Bezeichner ohne  
1037 Zuweisung geschrieben werden.

1038 Beispiel: (cardSession und pinRef sind Eingangsdaten des aufrufenden TUCs):

1039 TUC\_KON\_022 „Liefere PIN-Status“ {cardSession=cardSession; pinRef=pinRef}

1040 vereinfachte Schreibweise:

1041 TUC\_KON\_022 „Liefere PIN-Status“ {cardSession; pinRef}

#### 1042 **1.5.4.3 Event-Mechanismus**

1043 Der in Kapitel 4.1.6 spezifizierte Event-Mechanismus zur Unterrichtung von  
1044 Clientsystemen wird innerhalb dieser Spezifikation auch zur internen Verzahnung der  
1045 einzelnen Funktionsmerkmale eingesetzt. So wird ein Ereignis, das in der  
1046 Managementschnittstelle durch Änderung eines Konfigurationsparameters ausgelöst wird,  
1047 innerhalb des DHCP-Kapitels als Trigger für eine Lease-Erneuerung verwendet. Dies  
1048 bedeutet nicht, dass im Rahmen der Implementierung intern ein Event-Mechanismus  
1049 zwischen den Modulen verwendet werden muss. Auch hier dient die Form der Darstellung  
1050 (Events) lediglich der editoriellen Kopplung verschiedener Verhaltensbeschreibungen.

1051 Um den Ursprung eines Events erkennen zu können, verwenden alle Events ein Haupt-  
1052 Topic passend zum Funktionsmerkmal: „DHCP/LAN\_CLIENT/RENEW“ wird im  
1053 Funktionsmerkmal DHCP ausgelöst, „CARD/INSERTED“ wird im Funktionsmerkmal  
1054 Kartendienst ausgelöst usw.

#### 1055 **1.5.4.4 Konfigurationsparameter und Zustandswerte**

1056 Werte die der Administrator des Konnektors einsehen oder verändern können muss,  
1057 werden zusätzlich zu den Festlegungen in Kapitel 4.3 Konnektormanagement auch pro  
1058 Funktionsmerkmal in den jeweiligen Unterkapiteln „Betriebsaspekte“ erhoben. Diese  
1059 **Konfigurationsparameter** werden über eine ReferenzID definiert. Definierte  
1060 Konfigurationsparameter können in allen Kapiteln der Spezifikation referenziert werden.  
1061 Den Ort, an welchem ein solcher Konfigurationsparameter definiert/erhoben und somit  
1062 dessen Bedeutung beschrieben wird, lässt sich über den Präfix der ReferenzID erkennen:  
1063 CERT\_CRL\_DOWNLOAD\_ADDRESS (also „Cert“) wird im Zertifikatsdienst definiert,  
1064 MGM\_LU\_ONLINE (also „MGM“) wird im Konnektormanagement definiert usw.

1065 Die ReferenzIDs der Konfigurationsparameter besitzen in ihrer Schreibweise nur  
1066 innerhalb des Dokuments Gültigkeit. In der Umsetzung können für die  
1067 Konfigurationswerte herstellerspezifische Beschreibungen und Labels verwendet werden.

1068 Vergleichbar zu diesen Konfigurationsparametern, sind **Zustandswerte**. Auch diese  
1069 werden über ReferenzIDs definiert, nur können sie nicht durch den Administrator  
1070 verändert oder eingesehen werden. Sie finden nur konnektorintern Verwendung und sind  
1071 für die Beschreibung der Verhaltensweise notwendig, Beispiele sind CTM\_CT\_LIST für  
1072 die Liste der durch den Konnektor verwalteten Kartenterminals oder CM\_CARD\_LIST für  
1073 die Liste der aktuell erreichbaren Karten. Zustandswerte verwenden die gleichen Präfixe  
1074 wie Konfigurationsparameter.

1075

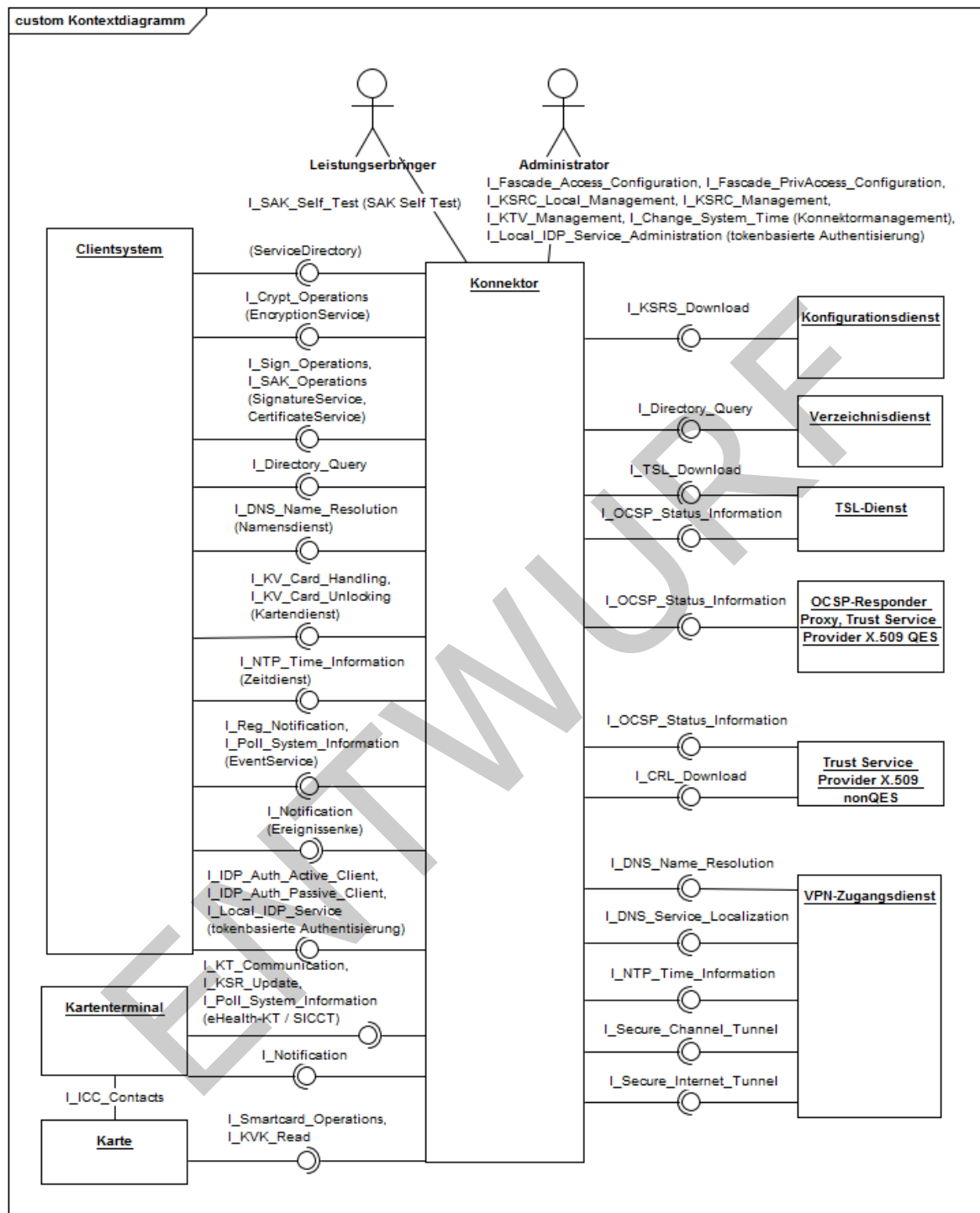
---

## 2 Systemüberblick

---

- 1076 Der Konnektor ist ein Produkttyp der TI gemäß [gemKPT\_Arch\_TIP#5.3.9].
- 1077 Er bietet seine Basisdienste sowohl intern den in ihm laufenden Fachmodulen an, als  
1078 auch externen Clientsystemen über die Konnektorauschnittstellen.
- 1079 Im lokalen Netz der Einsatzumgebung kommuniziert das Clientsystem mit dem  
1080 Konnektor über dessen LAN-seitiges Ethernet-Interface. Alleinig der Konnektor  
1081 kommuniziert mit den in lokalen Netzen angeschlossenen Kartenterminals und Karten.  
1082 Auch die Kommunikation mit den zentralen Diensten der TI-Plattform und  
1083 fachanwendungsspezifischen Diensten erfolgt ausschließlich über den Konnektor über  
1084 dessen WAN-seitiges Ethernet-Interface.
- 1085 Um die lokale Anzeige für die Signaturerstellung und Signaturprüfung zu realisieren, wird  
1086 ein Signaturproxy verwendet, der die Schnittstellen I\_Sign\_Operations und  
1087 I\_SAK\_Operations sowie ServiceDirectory kapselt. Der Signaturproxy ist aus Gründen der  
1088 Übersichtlichkeit nicht in der Abbildung PIC\_KON\_116 dargestellt, seine Spezifikation  
1089 findet sich in [gemSpec\_Kon\_SigProxy].
- 1090 Abbildung PIC\_KON\_116 stellt die Schnittstellen im Umfeld des Konnektors dar.

1091



**Abbildung 1: PIC\_KON\_116 Schnittstellen des Konnektors von und zu anderen Produkttypen**

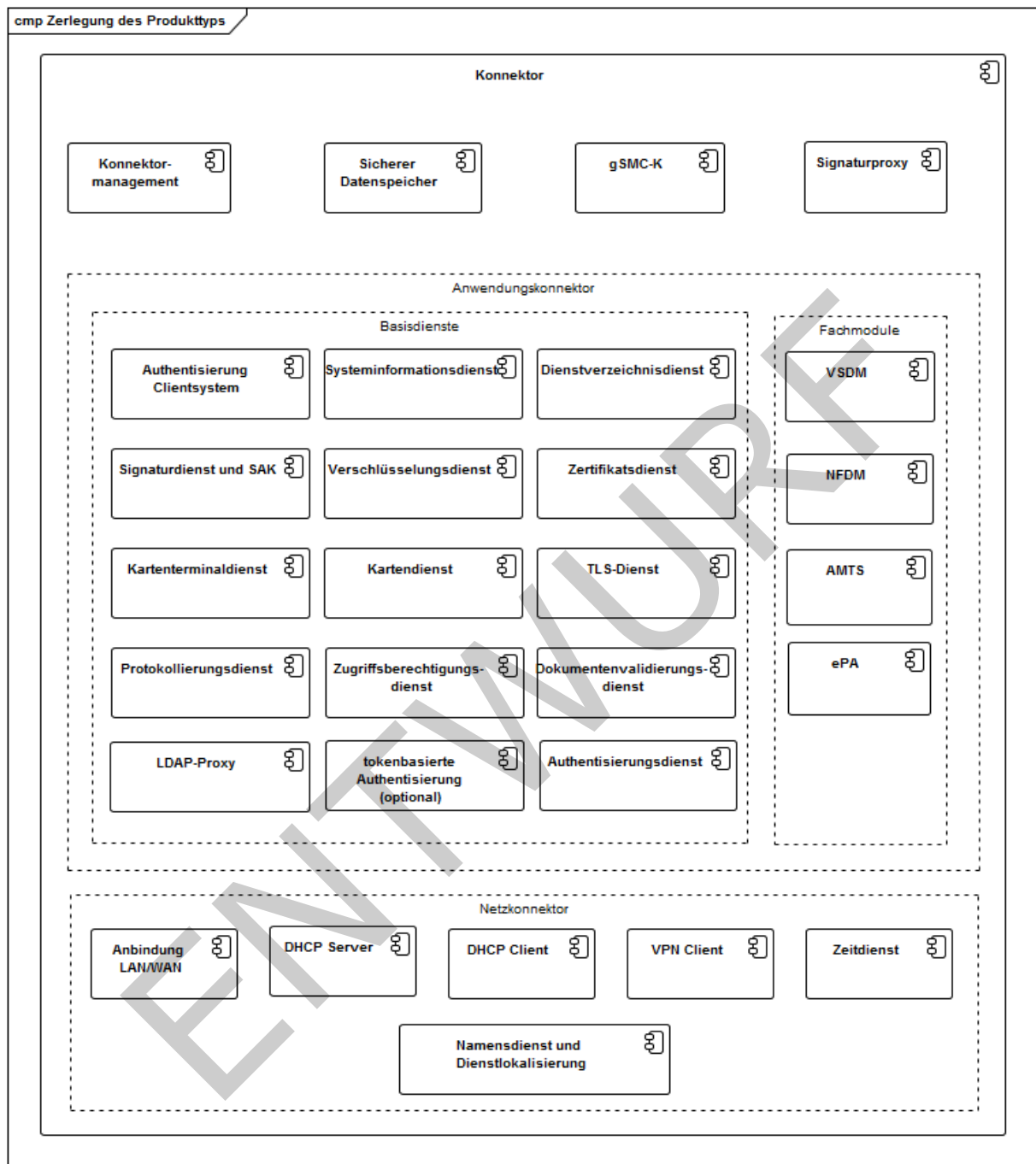
Die logischen Außenschnittstellen aus [gemKPT\_Arch\_TIP] werden im Konnektor technisch vorrangig als SOAP-Schnittstellen ausgeprägt. Von dieser Regel wird insbesondere bei Netzwerkschnittstellen abgewichen, wenn bereits etablierte Schnittstellenstandards für Basisdienste existieren (IPsec, TLS, NTP, DNS etc.). Eine

- 1100 Übersicht der Zuordnung „logische Schnittstellen → technische Schnittstellen“ findet sich  
1101 in Anhang H.
- 1102 Zum Nachweis der Sicherheit müssen Konnektoren im Rahmen der Zulassung nach  
1103 Common Criteria gegen die Schutzprofile [PP\_NK] und [PP\_KON] evaluiert und zertifiziert  
1104 werden.
- 1105 Die zu verwendenden kryptographischen Verfahren und zugehörige Parameter (z. B.  
1106 Schlüssellängen) für alle kryptographischen Operationen innerhalb der  
1107 Telematikinfrastruktur, werden durch das Dokument „Verwendung kryptographischer  
1108 Algorithmen in der Telematikinfrastruktur“ [gemSpec\_Krypt] normativ geregelt.

## 1109 2.1 Logische Struktur

- 1110 Der Produkttyp Konnektor besitzt eine Vielzahl verschiedenster Operationen und  
1111 Verhaltensweisen an seiner Außenschnittstelle. Um sein komplexes Gesamtverhalten  
1112 sinnvoll beschreiben zu können, wird der Konnektor innerhalb dieser Spezifikation logisch  
1113 unterteilt und strukturiert. Es wird primär zwischen Anwendungs- und Netzkonnektor  
1114 unterschieden, begleitet von Mechanismen, die blockübergreifend beschrieben werden.
- 1115 Der logische Aufbau des Konnektors ist in Abbildung PIC\_KON\_117 dargestellt.
- 1116 • Der Anwendungskonnektor bietet anwendungsnahe Basisdienste (inklusive  
1117 Signaturdienst) und Fachmodule zur Nutzung durch ein Clientsystem an.
  - 1118 • Der Anwendungskonnektor bietet zusätzlich zu den in Kap. 4.1 beschriebenen  
1119 Basisdiensten den optionalen Dienst „tokenbasierte Authentisierung“, der in  
1120 [gemSpec\_Kon\_TBAuth] beschrieben ist.
  - 1121 • Der Netzkonnektor bietet transportnahe Basisdienste und verbindet das lokale  
1122 Netz der Nutzer mit der zentralen TI-Plattform.
  - 1123 • Die gSMC-K ist zwar ein eigenständiger Produkttyp innerhalb der TI, wird im  
1124 Konnektor jedoch als Verbaukomponente betrachtet. Sie enthält die  
1125 kryptographischen Identitäten des Konnektors, sowie Steuerdaten  
1126 (Umgebungsinformationen TU/RU/PU, zugehörige Adressbereiche,  
1127 herstellersistem spezifische Konfigurationsdaten), die aus Sicherheitsgründen  
1128 unveränderlich in den Konnektor eingebracht werden müssen.
  - 1129 • Das Konnektormanagement dient der administrativen Verwaltung und Steuerung  
1130 des gesamten Konnektors.
  - 1131 • Der Sichere Datenspeicher dient der integren, vertraulichen und authentischen  
1132 Persistierung von veränderlichen Daten (siehe auch Kapitel 2.2).
  - 1133 • Der Signaturproxy ist eine Komponente, die zwischengeschaltet auf der  
1134 Kommunikationsstrecke zwischen Client-System und Konnektor dafür sorgt, dass  
1135 die zu signierenden oder zu prüfenden Dokumente dem Nutzer angezeigt werden.  
1136 Die Beschreibung des Signaturproxy befindet sich in [gemSpec\_Kon\_SigProxy]

1137



**Abbildung 2: PIC\_KON\_117 Logische Zerlegung des Konnektors in Anwendungs- und Netzkonnektor**

Diese logische Unterteilung schreibt in keiner Art und Weise die spätere Implementierung durch den Hersteller vor. Der Hersteller kann seine interne Modularisierung des Konnektors frei wählen. Normativ wirksam ist ausschließlich das durch die Detailfestlegungen in Summe beschriebene Verhalten an den Außenschnittstellen des Konnektors als Ganzes.

## 1147 2.2 Sicherer Datenspeicher

1148 Wie im vorherigen Kapitel dargestellt, wird für den Konnektor ein Datenspeicher  
1149 angenommen, in welchem der Konnektor alle sicherheitskritischen, veränderlichen Daten  
1150 dauerhaft speichert, die für seinen Betrieb relevant sind. Dieser Datenspeicher sichert die  
1151 Integrität, Authentizität und Vertraulichkeit der in ihm hinterlegten Daten bzw. der aus  
1152 ihm entnommenen Daten. Alleinig der Konnektor hat auf diesen Datenspeicher Zugriff.  
1153 Für folgende, im weiteren Verlauf der Spezifikation anfallende Daten wird angenommen,  
1154 dass diese im Sicheren Datenspeicher persistiert werden:

- 1155 • Der Trust Store des Zertifikatsdienstes
- 1156 • Die Konfigurationsdaten des Konnektormanagements
- 1157 • Die Konfigurationsdaten aller Funktionsmerkmale

1158 Ferner stellt der Konnektor den in ihm laufenden Fachmodulen ebenfalls eine Nutzung  
1159 dieses Datenspeichers für ihre sensiblen Daten zur Verfügung.

1160 Da es sich bei dem Sicheren Datenspeicher um ein internes Modul handelt, welches an  
1161 der Außenschnittstelle nicht testbar ist, werden an dieses Modul im Rahmen dieser  
1162 Spezifikation keine Anforderungen erhoben. Da dieses logische Modul aber essenzielle  
1163 Sicherheitsfunktionen bietet, ohne die ein Konnektor nicht sicher betrieben werden kann,  
1164 werden die Funktionen, die ein Hersteller für sein Konnektormodell real umsetzt, um die  
1165 notwendigen sicheren Speicherfunktionen zu realisieren, im Rahmen der CC-Evaluierung  
1166 geprüft werden. Näheres hierzu regeln die Schutzprofile des Konnektors.

## 1167 2.3 Überblick Konnektoridentität

1168 Die Geräteidentität des Konnektors (Konnektoridentität) teilt sich in drei Identitäten auf:

- 1169 • ID.NK.VPN für den Netzkonnektor  
1170 Die Identität des Netzkonnektors dient der Authentisierung gegenüber den  
1171 zentralen Netzwerkdiensten und wird für die Anmeldung an den VPN-Konzentrator  
1172 genutzt.
- 1173 • ID.AK.AUT für den Anwendungskonnektor  
1174 Die Identität des Anwendungskonnektors dient der Authentisierung gegenüber  
1175 den Clientsystemen im Rahmen von TLS-Verbindungen.
- 1176 • ID.SAK.AUT für die im Anwendungskonnektor enthaltene  
1177 Signaturanwendungskomponente  
1178 Die Identität des Signaturdienstes dient zur Authentisierung gegenüber den  
1179 Kartenterminals. Darüber hinaus muss sich der Signaturdienst des Konnektors  
1180 gegenüber dem Heilberufsausweis mittels eines kartenverifizierbaren Zertifikats  
1181 (C.SAK.AUTD\_CVC) mit entsprechendem Profil ausweisen, um Stapelsignaturen  
1182 durchführen zu können.

1183 In der Regel ergibt sich aus dem Kontext, welche Identität gemeint ist, sodass in diesen  
1184 Fällen nur kurz von der Konnektoridentität geschrieben wird.

1185 Die Geräteidentitäten werden durch asymmetrische Schlüssel und X.509-Zertifikate  
1186 umgesetzt. In Abhängigkeit vom gewählten kryptographischen Verfahren werden RSA-  
1187 Schlüssel bzw. ECC-Schlüssel verwendet.

## 1188 2.4 Mandantenfähigkeit

1189 Den Anforderungen aus [gemKPT\_Arch\_TIP#TIP1-A\_2200] folgend, wird die  
1190 Mandantenfähigkeit innerhalb des Konnektors nicht durch eine einzelne Funktion,  
1191 sondern durch Berücksichtigung in einer Reihe von Funktionsmerkmalen umgesetzt.

1192 Die Mandantenfähigkeit wirkt dabei auf:

- 1193 • Zugriffsberechtigungsdienst: Kapitel 4.1.1  
1194 (und über diesen auf alle Karten- und Kartenterminaloperationen)
- 1195 • Systeminformationsdienst: Kapitel 4.1.6

## 1196 2.5 Versionierung

1197 Gemäß [gemSpec\_OM] müssen Konnektor und Kartenterminals über eine Versionierung  
1198 verfügen. Die relevanten Versionsinformationen sind durch das O&M-Schema  
1199 ProductInformation.xsd definiert. Ferner definiert [gemSpec\_OM], dass Konnektor und  
1200 Kartenterminal das Konzept der Firmware-Gruppe verwenden müssen. Daher verfügen  
1201 die beiden Produkttypen auch über eine aktuelle Firmware-Gruppenversion.

1202 Versionsinformationen werden innerhalb des Konnektor an folgenden Stellen ver- und  
1203 bearbeitet:

- 1204 • Dienstverzeichnisdienst (Kapitel 4.1.3): Ausgabe der Konnektorversion über SOAP
- 1205 • Kartenterminaldienst (Kapitel 4.1.4): Anzeige der Versionsinformationen der  
1206 verwalteten Kartenterminals
- 1207 • Konnektormanagement (Kapitel 4.3):
  - 1208 • Anzeige der Versionsinformationen des Konnektors (Kapitel 4.3.2)
  - 1209 • Software-Aktualisierung (KSR-Client) für Konnektor und Kartenterminals  
1210 (Kapitel 4.3.9)

## 1211 2.6 Fachanwendungen

1212 Der Konnektor ist als Plattformkomponente der TI für die Erbringung von Basisdiensten  
1213 verantwortlich. Fachliche Funktionalitäten werden über die Fachmodule bereitgestellt.

1214 Das Fachmodul wird dabei als integraler Bestandteil des Konnektors verstanden  
1215 (Konnektor als Monolith), d. h., die Spezifikationen zu Konnektor (als  
1216 Plattformkomponente) und dem Fachmodul sind zwar getrennt, werden aber von einem  
1217 Hersteller in einer Gesamtkomponente umgesetzt. Die inneren Schnittstellen zwischen  
1218 Fachmodul und Konnektor sind von außen nicht erkennbar.

1219 In dieser Ausbaustufe unterstützt der Konnektor die Fachanwendungen VSDM,  
1220 AMTS, NFDM und ePA über jeweils ein Fachmodul.

1221 Neben Fachanwendungen, die über ihr Fachmodul mit einem gesicherten Fachdienst  
1222 kommunizieren, unterstützt der Konnektor einen Zugriff von Clientsystemen auf offene  
1223 Fachdienste.



## 2.7 Netzseitige Einsatzszenarien

Der Konnektor unterstützt unterschiedliche netzseitige Einsatzszenarien, die in Anhang K beispielhaft dargestellt sind.

Der Konnektor bietet hierzu Konfigurations-Parameter, die je nach netzseitigem Einsatzszenario konfiguriert werden müssen.

### 2.7.1 Parameter ANLW\_ANBINDUNGS\_MODUS

#### **Konfiguration 1: Konnektor als Gateway (ANLW\_ANBINDUNGS\_MODUS = InReihe):**

Diese Konfiguration ist geeignet für Szenarien, in denen der Konnektor zwischen das lokale Netz und das Internet Access Gateway (IAG) (z. B. Router mit DSL-/Kabelmodem) geschaltet wird. (vgl. Anhang K, Szenario 1)

**Konfiguration 2: Konnektor eingebettet in existierende Infrastruktur (ANLW\_ANBINDUNGS\_MODUS = Parallel):** Diese Konfiguration ist geeignet für Szenarien, in denen der Konnektor als weiteres Gerät in die bestehende Netzwerkinfrastruktur integriert wird. (vgl. Anhang K, Szenario 3)

Aus Sicherheitsgründen soll die Kommunikation der Clientsysteme mit dem Konnektor hierbei verschlüsselt erfolgen (ANCL\_TLS\_MANDATORY=Enabled). Falls diese Kommunikation unverschlüsselt erfolgt (ANCL\_TLS\_MANDATORY=Disabled), übernimmt der Nutzer die Verantwortung für die Sicherstellung der vertraulichen Übertragung.

Für den Einsatz und die Nutzung von DHCP gibt es im Zusammenhang mit diesem Konfigurationsparameter folgende Möglichkeiten:

- Die Netzwerkinfrastruktur der Einsatzumgebung verwendet den DHCP-Server des Konnektors (siehe Kap. 4.2.2).
- Ein bestehender DHCP-Server im Netz der Einsatzumgebung wird weiter verwendet und derart konfiguriert, dass als Default Gateway und DNS-Server entweder bestehende Infrastruktur oder der Konnektor verwendet wird.
- Es kommt kein DHCP-Server zum Einsatz. Bei allen Clients im Netz der Einsatzumgebung werden das Default Gateway und der DNS-Server statisch auf den Konnektor gesetzt.

Die DHCP-Konfiguration ist in Konfiguration 1 in aller Regel die folgende: Die WAN-Seite des Konnektors verwendet den DHCP-Server des bestehenden IAG. An der LAN-Seite stellt der Konnektor einen DHCP-Server für alle Clients zur Verfügung.

### 2.7.2 Parameter ANLW\_INTERNET\_MODUS

Grundsätzlich routet der Konnektor im Modus ANLW\_INTERNET\_MODUS=SIS alle für das Internet bestimmten Pakete von Clients, die ihn als Default Gateway verwenden, in den VPN-Tunnel zum SIS, während er im Modus ANLW\_INTERNET\_MODUS=Keiner diese Pakete verwirft.

Im Unterschied zu (ANLW\_ANBINDUNGS\_MODUS = InReihe) ist die Nutzung des SIS bei (ANLW\_ANBINDUNGS\_MODUS = Parallel) optional. Alternativ können auch die Clients, die den Konnektor als Default Gateway verwenden, per Redirect direkt ins Internet verwiesen werden (ANLW\_INTERNET\_MODUS=IAG).



## 1265 2.8 Lokale und entfernte Kartenterminals

1266 Gemäß [gemKPT\_Arch\_TIP] ermöglicht die Telematikinfrastruktur dem Anwender die  
1267 PIN-Eingabe zur Freischaltung eines HBAs oder einer SMC-B wahlweise lokal oder über  
1268 das Remote-PIN-Eingabeverfahren durchzuführen. Deshalb unterscheidet auch der  
1269 Konnektor zwischen einem lokalen Kartenterminal – räumlich („in Sichtweite“) dem  
1270 Arbeitsplatz zugeordnet – und einem entfernten Kartenterminal.

1271 Ein lokales Kartenterminal befindet sich lokal an einem Arbeitsplatz und kann von diesem  
1272 aus genutzt werden. Hingegen ist das entfernte Kartenterminal einem entfernten oder  
1273 auch – für zentral steckende Karten – keinem Arbeitsplatz fest zugewiesen. Ein lokales  
1274 Kartenterminal kann als sogenanntes Remote-PIN-KT verwendet werden, um die PIN für  
1275 eine in einem entfernten Kartenterminal steckende Karte einzugeben.

## 1276 2.9 Standalone-Szenario

1277 Gemäß § 291 SGB V Absatz 2b müssen „Diese Dienste [zur Online-Aktualisierung der  
1278 Versichertendaten auf der eGK] [...] auch ohne Netzanbindung an die  
1279 Praxisverwaltungssysteme der Leistungserbringer online genutzt werden können.“

1280 Dies bedeutet, dass der Konnektor ohne ein steuerndes Clientsystem ereignisgetrieben  
1281 Fachanwendungen ausführen können muss. Aus Fachsicht „steht der Konnektor alleine“,  
1282 ohne Clientsysteme. Die konkreten Aktionen, die Fachanwendungen in diesen Fällen  
1283 ausführen, sowie deren Auslöser werden in den jeweiligen Fachmodulspezifikationen  
1284 beschrieben.

1285 Ein solcher alleinstehender Konnektor mit Zugang zur TI muss zur Durchführung der  
1286 Fachanwendungen durch einen weiteren Konnektor unterstützt werden, der in direkter  
1287 Verbindung zum Clientsystem steht, selbst aber keine Online-Anbindung besitzt.

1288

### 3 Übergreifende Festlegungen

1289 Für die folgenden Inhalte bitte die Hinweise in Kapitel 1.5.3 „Erläuterungen zur  
1290 Spezifikation des Außenverhaltens,“ sowie Kapitel 1.5.4 Erläuterungen zur  
1291 Dokumentenstruktur und „Dokumentenmechanismen“ beachten.

1292 In diesem Kapitel werden die Aspekte des Konnektors behandelt, die  
1293 Funktionsmerkmalübergreifend geregelt werden müssen.

1294 Die Managementschnittstelle/Administrationsoberfläche des Konnektors wird dabei nicht  
1295 als übergreifender Aspekt, sondern als eigenes Funktionsmerkmal gewertet. Die  
1296 Festlegungen hierzu finden sich entsprechend in Kapitel 4.3.

#### 1297 **A\_18605 - Option Basisdienst TBAuth**

1298 Der Konnektor SOLL den Basisdienst TBAuth [gemSpec\_Kon\_TBAuth] unterstützen. [≤]

1299 Wird die SOLL-Anforderung A\_18605 nicht umgesetzt, so ist die Umsetzung mit einem  
1300 Firmwareupdate im Jahr 2021 nachzuholen.

#### 1301 **Dokumentformate**

1302 Mit dem Aufruf einer Operation, die Dokumente verarbeitet, muss durch den Aufrufer  
1303 festgelegt werden können, um welches Dokumentenformat es sich handelt, damit die  
1304 unterschiedlichen Formate zur Verarbeitung und etwaigen Anzeige unterschieden werden  
1305 können. Die nicht-XML-Formate werden dabei nach MIME-Typ-Klassen unterschieden:

- 1306 • „PDF/A“ für MIME-Typ „application/pdf-a“ gemäß [ISO 19005],
- 1307 • „Text“ für MIME-Typ „text/plain“,
- 1308 • „TIFF“ für MIME-Typ „image/tiff“ gemäß [TIFF6]
- 1309 • „Binär“ für alle übrigen MIME-Typen.

1310 Folgende Bezeichner werden verwendet:

1311 Alle\_DocFormate: XML, PDF/A, Text, TIFF, Binär

1312 nonQES\_DocFormate: XML, PDF/A, Text, TIFF, Binär

1313 QES\_DocFormate: XML, PDF/A, Text, TIFF

1314 Für nonQES\_DocFormate wird, trotz Gleichheit zu Alle\_DocFormate, ein eigener  
1315 Referenzbezeichner verwendet, da sich diese Liste noch ändern könnte. TIFF wird durch  
1316 [gemKPT\_Arch\_TIP] nicht für die nonQES verlangt. Die Unterstützung dieses Formats für  
1317 nonQES bedeutet jedoch keinen Mehraufwand, da die Routinen durch QES bereits  
1318 implementiert sind und nachgenutzt werden können.

#### 1319 **TIP1-A\_4500 - Dokumentgrößen von 25 MB**

1320 Der Konnektor MUSS für alle Außenschnittstellen, in denen ein Dokument verarbeitet  
1321 wird, Dokumente mit einer Größe ≤ 25 MB unterstützen. Der Konnektor KANN  
1322 Dokumente mit einer Größe > 25 MB unterstützen. [≤]

#### 1323 **A\_19052 - Vorgaben für Dokumentformate und Nachrichten**

1324 Der Konnektor MUSS für die Verarbeitung von Dokumenten und Nachrichten die  
1325 Vorgaben aus TAB\_KON\_775 erfüllen. [≤]

#### 1326 **TIP1-A\_4502 - Zeichensatzcodierungen UTF-8 und ISO-8859-15**

1327 Der Konnektor MUSS bei der Verarbeitung von Dokumenten der Formate XML und Text  
1328 die Zeichensatzcodierungen UTF-8 und ISO-8859-15 unterstützen. Das verarbeitete

1329 Dokument MUSS der Konnektor mit demselben Zeichensatz kodieren, in dem das  
1330 Eingangsdokument kodiert war. [≤=]

1331 **TIP1-A\_5541-01 - Referenzen in Dokumenten nicht dynamisch auflösen**

1332 Der Konnektor DARF in Dokumenten eventuell vorhandene Referenzen auf externe  
1333 Ressourcen NICHT auflösen, es sei denn es sind Verweise auf im Konnektor sicher  
1334 eingebrachte vorliegende Schemata oder dies wird im Einzelfall normativ gefordert. [≤=]

1335 **Kartentypen**

1336 Der Konnektor unterstützt eine Reihe von Kartentypen. Die folgende Tabelle enthält die  
1337 Liste der Referenzbezeichner für die verschiedenen Kartentypen, wie sie im weiteren  
1338 Verlauf verwendet werden. Die Unterstützung von Karten der Generation 2 (G2.x: G2.0,  
1339 G2.1 und höher) beschränkt sich bei diesen auf die Datenstrukturen und Schlüssel, die  
1340 aus Gründen der Abwärtskompatibilität zu den Karten der Generation 1+ vorhanden sind.  
1341 Eine Ausnahme hiervon bilden die Geräte-CVCs, die bereits für dieses Release basierend  
1342 auf ECC verwendet werden.

1344 **Tabelle 1: TAB\_KON\_500 Wertetabelle Kartentypen**

| ReferenzID<br>Kartentyp | Karten-<br>generatio<br>n | Beschreibung   |
|-------------------------|---------------------------|--|
| EGK                     | G1+                       | Die elektronische Gesundheitskarte gemäß [gemSpec_eGK_P1] und [gemSpec_eGK_P2]   |
| EGK                     | G2                        | Die elektronische Gesundheitskarte gemäß [gemSpec_COS] und [gemSpec_eGK_ObjSys] bzw. [gemSpec_eGK_ObjSys_G2.1]   |
| HBA-qSig                | -                         | HBA-Vorläuferkarte gemäß [HPC-P1] und [HPC-P2]   |
| HBA                     | G2                        | Der elektronische Heilberufsausweis (HBA) gemäß [gemSpec_COS] und [gemSpec_HBA_ObjSys]   |
| SMC-B                   | G2                        | Die Institutionskarte Typ B (Secure Module Card) gemäß [gemSpec_COS] und [gemSpec_SMC-B_ObjSys]  |
| HSM-B                   |                           | HSM-Variante einer SM-B.<br>Das HSM-B wird in dieser Fassung als ein oder mehrere virtuelle Kartenterminals verstanden, in denen virtuelle Karten stecken. |
| SMC-KT                  | G2                        | Die Karte Typ KT (Secure Module Card) gemäß [gemSpec_COS] und [gemSpec_gSMC-KT_ObjSys]   |
| KVK                     | -                         | Die Krankenversichertenkarte gemäß der Spezifikation [KVK]   |
| ZOD_2.0                 | -                         | HBA-Vorläuferkarte gemäß [HPC-P1] und [HPC-P2]   |
| UNKNOWN                 |                           | Eine nicht erkannte Karte oder nicht lesbare Karte   |
|                         |                           | Zusammenfassende ReferenzIDs   |

|        |  |  |
|--------|--|--|
| HBA-VK |  | Adressiert die HBA-Vorläuferkarten HBA-qSig und ZOD_2.0.<br>Wird dieser Referenzbezeichner verwendet, gelten die zugehörigen Aussagen und Festlegungen für beide Kartentypen.                              |
| HBAx   |  | Adressiert sowohl den HBA, als auch die HBA-Vorläuferkarten (HBA-VK)<br>Wird dieser Referenzbezeichner verwendet, gelten die zugehörigen Aussagen und Festlegungen für alle drei Kartentypen.              |
| SM-B   |  | Adressiert sowohl eine echte SMC-B als auch eine in einem HSM-B enthaltene virtuelle SMC-B.<br>Wird dieser Referenzbezeichner verwendet, gelten die zugehörigen Aussagen und Festlegungen für beide Typen. |

## Übergreifende Festlegungen zum Aufbau von sicheren Verbindungen

### TIP1-A\_7254 - Reaktion auf OCSP-Abfrage beim TLS-Verbindungsaufbau

Der Konnektor MUSS beim Aufbau von TLS-gesicherten Verbindungen zu einem zentralen Dienst der TI-Plattform oder zu einem fachanwendungsspezifischen Dienst, bei denen eine OCSP-Abfrage des Serverzertifikats nach TUC\_PKI\_006 erfolgt, neben Fehlerfällen bei folgenden Warnungen gemäß [gemSpec\_PKI#Tab\_PKI\_274]

- CERT\_REVOKED
- CERT\_UNKNOWN
- OCSP\_CHECK\_REVOCATION\_FAILED

mit Abbruch des Verbindungsaufbaus reagieren. [≤]

In [gemSpec\_Krypt#6] wird das Kommunikationsprotokoll zwischen einem Client und einer Vertrauenswürdigen Ausführungsumgebung (VAU) spezifiziert. Dabei wird ein sicherer Kanal auf HTTP-Anwendungsschicht zwischen dem Client und der VAU (Server) aufgebaut. Der Client ist hier ein Fachmodul des Konnektors; der Server ist ein Fachdienst.

### A\_17225-01A\_17225 - Aufbau einer sicheren Verbindung zur Vertrauenswürdigen Ausführungsumgebung (VAU)

Der Konnektor MUSS für Fachmodule den Aufbau einer sicheren Verbindung zur Vertrauenswürdigen Ausführungsumgebung (VAU) gemäß Kommunikationsprotokoll [gemSpec\_Krypt#6] unterstützen und das vom Server übergebene Zertifikat wie folgt prüfen:

```
TUC_KON_037 „Zertifikat prüfen“ {
    certificate = C.FD.AUT;
    qualifiedCheck = not_required;
    offlineAllowNoCheck = false;
    policyList = oid_fd_aut;
    intendedKeyUsage= intendedKeyUsage(C.FD.AUT);
    intendedExtendedKeyUsage = id_kp_serverAuth;
    validationMode = OCSP}
```

Der Konnektor MUSS die vom Fachmodul übergebene Rolle gegen die aus dem Zertifikat ermittelte Rolle prüfen. [≤]

### **A\_17777 - sicherheitstechnische Festlegungen zum Abruf von kryptographischen Schlüsseln von einem Schlüsselgenerierungsdienst**

Der Konnektor MUSS für Fachmodule für die Nutzung der Schlüsselableitungsfunktionalität die sicherheitstechnischen Festlegungen gemäß [gemSpec\_Krypt#3.15.5 Schlüsselableitungsfunktionalität ePA] und [gemSpec\_SGD] bereitstellen. [≤]

Der Gesamtablauf der Schlüsselableitungsfunktionalität gemäß [gemSpec\_SGD#2.3] für den Konnektor als Client ist aufgeteilt zwischen Basiskonnektor und Fachmodul. Die kryptographischen Vorgaben (u.a. Durchführung des ECDH, Schlüsselerzeugung, Ver- und Entschlüsselung, Signaturerzeugung und -prüfung) werden dabei durch den Basiskonnektor realisiert.

## **3.1 Konnektoridentität und gSMC-K**

### **TIP1-A\_4503 - Verpflichtung zur Nutzung von gSMC-K**

Der Konnektor MUSS das geheime Schlüsselmaterial zur Geräteidentität (ID.NK.VPN, ID.AK.AUT, ID.SAK.AUT) und der Rolle SAK (C.SAK.AUTD\_CVC) über Smartcards des Typs gSMC-K gemäß [gemSpec\_gSMC-K\_ObjSys] nutzen. Der Konnektor MUSS mit einer gSMC-K bestückt sein. Er KANN mit mehr als einer gSMC-K bestückt sein. [≤]

Die Notwendigkeit, den Konnektor mit mehr als einer gSMC-K zu bestücken, kann sich aus den Lastanforderungen aus [gemSpec\_Perf#4.1.2] ergeben.

### **TIP1-A\_4504 - Keine Administratorinteraktion bei Einsatz mehrerer gSMC-Ks**

Verwendet der Konnektor mehrere gSMC-Ks, DARF eine Administratorinteraktion für diese Belange NICHT erforderlich sein. [≤]

### **TIP1-A\_5543 - Keine manuelle PIN-Eingabe für gSMC-K**

Der Konnektor DARF Anwender und Administratoren außer bei der Inbetriebnahme (erstmalig oder nach Werksreset) NICHT auffordern, eine PIN für eine gSMC-K einzugeben. [≤]

### **TIP1-A\_4505 - Schutz vor physischer Manipulation gSMC-K (Sichere Verbundenheit der gSMC-K)**

Die gSMC-K des Konnektors MÜSSEN durch den Einsatz physikalischer Sperren oder manipulationssicherer Siegel so mit dem Konnektor verbunden sein, dass physischer Missbrauch oder physische Manipulation erkennbar ist. [≤]

gSMC-Ks gemäß [gemSpec\_gSMC-K\_ObjSys] verfügen über die Möglichkeit zur nachträglichen Generierung von Schlüsselpaaren und dem Nachladen der zugehörigen Zertifikate. Dieser Mechanismus wird erst in kommenden Releases durch den Konnektor unterstützt. Initial sind alle Identitäten bereits einmal auf der gSMC-K vorhanden.

### **TIP1-A\_4506 - Initiale Identitäten der gSMC-K**

In Abhängigkeit vom kryptographischen Verfahren MUSS der Konnektor folgende Objekte der gSMC-K als Quelle seiner Identitäten verwenden:

**Tabelle 2: TAB\_KON\_856: Identitäten des Konnektors auf der gSMC-K**

| Identifizier | Verzeichnis | Objekt der gSMC-K in Abhängigkeit vom kryptographischen Verfahren |
|--------------|-------------|---|
|--------------|-------------|---|

|                |           | RSA                | ECC                    |
|----------------|-----------|--------------------|------------------------|
| ID.NK.VPN      | MF/DF.NK  | EF.C.NK.VPN.R2048  | EF.C.NK.VPN2.XXXX      |
| ID.AK.AUT      | MF/DF.AK  | EF.C.AK.AUT.R2048  | EF.C.AK.AUT2.XXXX      |
| ID.SAK.AUT     | MF/DF.SAK | EF.C.SAK.AUT.R2048 | EF.C.SAK.AUT2.XXXX     |
| C.SAK.AUTD_CVC | MF/DF.SAK | -                  | EF.C.SAK.AUTD_CVC.E256 |

1420

1421 [ $\leq$ ]

1422

### 1423 3.1.1 Organisatorische Anforderungen und Sperrprozesse

#### 1424 **TIP1-A\_5392 - gSMC-K-Verantwortung durch den Hersteller des Konnektors**

1425 Der Hersteller des Konnektors MUSS die Rolle des Kartenherausgebers für in seinen  
1426 Konnektoren verbauten gSMC-Ks einnehmen.

1427 Der Hersteller des Konnektors KANN die von ihm verantwortete Personalisierung der  
1428 gSMC-K durch einen von ihm zu beauftragenden Dienstleister in seinem Namen  
1429 vornehmen lassen.

1430 [ $\leq$ ]

#### 1431 **TIP1-A\_5696 - Prüfung der personalisierten gSMC-K**

1432 Der Hersteller des Konnektors MUSS sich von der korrekten Personalisierung der  
1433 herausgegebenen gSMC-K überzeugen.

1434 [ $\leq$ ]

#### 1435 **A\_18928 - Ausstattung mit dual-personalisierten gSMC-K-X.509-Zertifikaten**

1436 Der Hersteller des Konnektors MUSS die Konnektoren mit einer gSMC-K mit  
1437 personalisierten RSA- und ECC-Zertifikaten gemäß TAB\_KON\_856 ausstatten. [ $\leq$ ]

1438

#### 1439 **A\_18930 - Unterstützung von gSMC-K Personalisierungsvarianten**

1440 Der Konnektor MUSS unterschiedliche gSMC-K-Personalisierungsvarianten sowohl mit als auch ohne  
1441 ECC-Zertifikate für ID.NK.VPN, ID.AK.AUT und ID.SAK.AUT unterstützen. [ $\leq$ ]

1442 Die Anforderung ist für die Anwendungsfälle Registrierung, IPsec-Authentisierung und  
1443 Autorisierung beim VPN-Zugangsdienst, TLS-Authentisierung zum eHealth-  
1444 Kartenterminal, TLS-Authentisierung zum Primärsystem nachzuweisen. Wenn RSA-2048  
1445 in der TI abgekündigt wird, entfällt dadurch die Anforderung.

1446

#### 1447 **TIP1-A\_5393 - Dokumentation der Konnektorzertifikatszuordnungen**

1448 Der Hersteller des Konnektors MUSS die Zuordnung von Konnektor und jeweils  
1449 eingebrachtem C.NK.VPN-Zertifikat mit dem Ziel dokumentieren, anhand eines  
1450 Sperrauftrages für einen Konnektor, das zu sperrende C.NK.VPN-Zertifikat identifizieren  
1451 zu können.

1452 [ $\leq$ ]

1453 Das bedeutet, dass der Konnektorhersteller je Konnektor die für die Identifikation des  
1454 C.NK.VPN-Zertifikates relevanten Daten wie z. B. Seriennummer des Konnektors und Art  
1455 der verbauten Komponenten, Seriennummer der gSMC-K, etc. für seinen Sperrprozesse  
1456 dokumentieren muss.



**TIP1-A\_5394 - Bereitstellen eines Konnektorsperrprozesses**

Der Hersteller des Konnektors MUSS für die von ihm verantworteten Konnektoren einen Sperrprozess etablieren, unterhalten und der gematik zugänglich machen.

Der Hersteller des Konnektors KANN die operative Durchführung des Sperrprozesses an Dritte delegieren.

[<=]

Sperrberechtigt ist die gematik im Rahmen des Change-Verfahrens (siehe [gemRL\_Betr\_TI#5.4]).

**TIP1-A\_5395 - Sperrberechtigung der gematik gegenüber Konnektorhersteller**

Der Hersteller des Konnektors MUSS im Rahmen der Change-Durchführung erteilte Sperraufträge der gematik fristgemäß (gemäß Change-Auftrag) bei dem TSP X.509 nonQES (Zertifikatsaussteller) umsetzen.

[<=]

Dazu bedient er die standardmäßige Schnittstelle zum TSP (siehe [gemSpec\_X.509\_TSP#TIP1-A\_3643]).

**TIP1-A\_5396 - Prüfung des Sperrauftrages für Konnektoren**

Der Hersteller des Konnektors MUSS vor der Umsetzung des Sperrauftrages für einen Konnektor die Sperrberechtigung des Beauftragenden prüfen und verhindern, dass Konnektoren missbräuchlich gesperrt werden.

[<=]

**TIP1-A\_5397 - Umsetzung von Sperraufträgen für Konnektoren**

Der Hersteller des Konnektors MUSS nach erfolgreicher Prüfung der Sperrberechtigung des Beauftragenden die Sperrung der entsprechenden C.NK.VPN-Zertifikate unverzüglich bei dem TSP X.509 nonQES (Zertifikatsaussteller) beauftragen.

[<=]

**TIP1-A\_5398 - Beschränkung der Sperrberechtigung des Konnektorherstellers**

Der Hersteller des Konnektors DARF NICHT die Sperrung von C.NK.VPN-Zertifikaten bei dem TSP X.509 nonQES (Zertifikatsaussteller) beauftragen, wenn er nicht durch einen für den Konnektor Sperrberechtigten dazu beauftragt wurde.

[<=]

**TIP1-A\_5399 - Protokollierung der Sperrung von Konnektoren**

Der Hersteller des Konnektors MUSS die Durchführung der Sperrung eines Konnektors protokollieren und der gematik auf Anfrage übermitteln.

Dabei MÜSSEN folgende Informationen protokolliert werden:

- Zeitpunkt der Beantragung und Umsetzung der Sperrung
- Grund der Sperrung
- Konnektoridentifikation

[<=]

Der Hersteller des Konnektors übernimmt im Rahmen der organisatorischen Sperrung die Aufgabe der Anwenderkommunikation gegenüber den betroffenen Anwendern. Die Eckpunkte zur Kommunikation sind Bestandteil des Beschlusses zur Außerbetriebnahme einer Konnektor-Baureihe und im Rahmen des Change-Verfahrens zwischen den Beteiligten abgestimmt.

**TIP1-A\_5400 - Fortführen des Konnektor-Sperrprozesses**

Der Hersteller des Konnektors MUSS die Fortführung des Sperrprozesses über die Einstellung seiner Geschäftstätigkeit hinaus gewährleisten.

[<=]



1504 Dies kann bspw. durch Übertragung der Aufgabe an einen Dritten realisiert werden.  
 1505 Dabei sind die Zuordnungen Konnektor zu Zertifikat gemäß Anforderung „Dokumentation  
 1506 der Konnektorzertifikatszuordnungen“ zur Verfügung zu stellen.

1507 Bei der Schlüsselerzeugung für die gSMC-K muss insbesondere auch mit technischen  
 1508 Maßnahmen die Vertraulichkeit der relevanten Schlüssel sichergestellt werden:

1509 **TIP1-A\_7225 - Schlüsselerzeugung bei einer Schlüsselspeicherpersonalisierung**

1510 Der Hersteller des Konnektors, der Schlüssel für die gSMC-K erzeugt, MUSS diese  
 1511 Schlüssel mittels eines technischen Sicherheitsmoduls (HSM, Chipkarte, TPM etc.)  
 1512 erzeugen, welches

- 1513 1. über einen Zugriffsschutz verfügt, sodass nur Berechtigte Schlüssel darauf nutzen  
 1514 können,
- 1515 2. in einem zutrittsgeschützten Bereich aufbewahrt wird und
- 1516 3. mindestens nach FIPS 140-2 Level 3 oder [COS-G2] (CC-zertifizierte Chipkarte  
 1517 der TI) zertifiziert ist.

1518 Wird für die Schlüsselerzeugung eine Schlüsselableitung verwendet, so MUSS die  
 1519 Schlüsselableitung die fachlichen Anforderungen aus GS-A\_5386 erfüllen.

1520 Es ist zulässig, dass asymmetrische Schlüssel bei der Personalisierung auf der gSMC-K  
 1521 selbst erzeugt werden und symmetrische Schlüssel mittels einer Schlüsselableitung  
 1522 erzeugt werden, bei dem sich der Ableitungsschlüssel (Masterkey) innerhalb eines nach  
 1523 3. zulässigen Hardwaresicherheitsmoduls befindet.

1524 Es ist zulässig, sicherheitstechnisch geeignete Maßnahmen zur Sicherstellung der  
 1525 Verfügbarkeit der Ableitungsschlüssel (Masterkey) umzusetzen (bspw. Shamir Secret-  
 1526 Sharing-Verfahren).

1527 Der Hersteller des Konnektors MUSS die Schlüsselerzeugung und die Schlüsselverwaltung  
 1528 in einem Konzept darstellen, das die technischen und organisatorischen Maßnahmen  
 1529 beschreibt, die den Schutzbedarf der verarbeiteten Informationsobjekte befriedigen. Der  
 1530 Hersteller des Konnektors MUSS dieses Konzept der gematik zur Verfügung stellen.[<=]

1531 **TIP1-A\_5703 - Geschützte Übertragung von Daten zum Kartenpersonalisierer**

1532 Der Hersteller des Konnektors, der Daten für die gSMC-K erzeugt (bspw. Schlüssel),  
 1533 MUSS diese Daten bei der Übertragung zum Kartenpersonalisierer hinsichtlich  
 1534 Vertraulichkeit, Authentizität und Integrität mit einem Verfahren nach [gemSpec\_Krypt]  
 1535 schützen.  
 1536 [<=]

1537 **3.2 Bootup-Phase**

1538 **TIP1-A\_4507 - Isolation während der Bootup-Phase**

1539 Da während der Bootup-Phase des Konnektors noch nicht alle Sicherheitsmechanismen  
 1540 ihre Leistung erbringen können, DÜRFEN die Dienste des Konnektors während dem  
 1541 Bootup über physikalische Schnittstellen von außen NICHT erreichbar sein.  
 1542 [<=]

1543 **TIP1-A\_4508 - Konnektorzustand nach Bootup**

1544 Der Konnektor MUSS nach Beendigung der Bootup-Phase die Initialisierung der  
 1545 Funktionsmerkmale durchlaufen haben. Die Startreihenfolge der Funktionsmerkmale  
 1546 kann unter Berücksichtigung von TIP1-A\_4507 herstellerspezifisch gestaltet werden.

1547 Im Rahmen der Bootup-Phase MÜSSEN folgende TUCs ausgeführt werden:

1548 TUC\_KON\_025, TUC\_KON\_035, TUC\_KON\_272, TUC\_KON\_341, TUC\_KON\_343,  
 1549 TUC\_KON\_352 (die Reihenfolge der TUC-Ausführung ist herstellerspezifisch).

1550 Treten während der Bootup-Phase Fehler auf, so MUSS die Bootup-Phase, sofern

1551 möglich, abgeschlossen werden.  
 1552 Sobald die Bootup-Phase abgeschlossen ist, MUSS TUC\_KON\_256 „Systemereignis  
 1553 absetzen“ mit folgenden Parameter aufgerufen werden:

```
1554 TUC_KON_256 {
1555     topic = "BOOTUP/BOOTUP_COMPLETE";
1556     eventType = Op;
1557     severity = Info;
1558 }
1559 [<=]
```

1560 Die hier gelisteten TUCs bilden nicht die abschließende Menge der während der Bootup-  
 1561 Phase zu erfüllenden Anforderungen. In den einzelnen Funktionsmerkmalen werden  
 1562 weitere Einzelanforderungen erhoben, die als Ausführungszeitpunkt die Bootup-Phase  
 1563 benennen (siehe Unterkapitel „Betriebsaspekte“ der einzelnen Funktionsmerkmal-  
 1564 Kapiteln, sowie Kapitel 4.3 Konnektormanagement).

### 1565 3.3 Betriebszustand

#### 1566 TIP1-A\_4509 - Betriebszustand erfassen

1567 Der Konnektor MUSS seinen Betriebszustand gemäß Tabelle TAB\_KON\_503  
 1568 Betriebszustand\_Fehlerzustandsliste über Fehlerzustände \$EC erfassen.  
 1569 Tritt die in Spalte „Beschreibung“ charakterisierte Fehlersituation eines Fehlerzustandes  
 1570 \$EC ein, wird sein Wert \$EC.value = true. Sobald die Fehlersituation beendet ist, springt  
 1571 der Wert auf \$EC.value = false. Die Fehlerzustände müssen dabei innerhalb der „max.  
 1572 Feststellungszeit“ (Tabellenspalte) erfasst werden. Eine maximale Feststellungszeit von  
 1573 einen Tag (1 day) verlangt, dass einmal am Tag der Zustand geprüft werden muss,  
 1574 unabhängig davon, welche TUCs aufgerufen werden. Eine maximale Feststellungszeit von  
 1575 1 sec, 10 sec, 1 min und 300 sec verlangt, dass nach der Feststellung einer Fehlfunktion  
 1576 innerhalb eines TUCs die Zustandsänderung innerhalb der angegebenen Zeit stattfinden  
 1577 muss.  
 1578 Nach Abschluss des Boot-Vorgangs müssen sämtliche Fehlerzustände mit einer „max.  
 1579 Feststellungszeit“ von „1 day“ erfasst worden sein.  
 1580 [<=]

#### 1581 TIP1-A\_4597 - Unterstützung von Missbrauchserkennungen

1582 Der Konnektor MUSS zur Unterstützung von Missbrauchserkennungen für alle  
 1583 Operationen, die in EVT\_MONITOR\_OPERATIONS gelistet sind und deren Alarmwert > 0  
 1584 ist, kontinuierlich folgende Aktivitäten durchlaufen:

- 1585 1. Minütlich gleitende 10-Minuten-Summe je in EVT\_MONITOR\_OPERATIONS  
 1586 gelistete Operation berechnen. Dazu gehen
  - 1587 • erfolgreiche Abschlüsse der Operation mit dem OK\_Val der Operation ein
  - 1588 • eine fehlerhaft beendete Operation mit dem NOK\_Val der Operation ein
- 1589 2. Überschreitet der gleitende 10-Minuten-Summenwert einer in  
 1590 EVT\_MONITOR\_OPERATIONS gelisteten Operation den zugehörigen Alarmwert, so  
 1591 setze EC\_CRYPTOPERATION\_ALARM auf True.

1592 [<=]

1593 Erklärung „Minütlich gleitende 10-Minuten-Summe“: Für die jeweilige Operation wird die  
 1594 Summe aller OK\_Val und NOK\_Val der letzten 10 Minuten gebildet. Diese Summe wird  
 1595 jede Minute neu berechnet.

**TIP1-A\_4512-02 - Ereignis bei Änderung des Betriebszustandes**

Der Konnektor MUSS per Ereignisdienst TUC\_KON\_256 über Änderungen des Betriebszustandes (Tabelle TAB\_KON\_503 Betriebszustand\_Fehlerzustandsliste) informieren.

Der Konnektor muss dazu für jeden Fehlerzustand \$EC mit Error Condition \$EC.errorcondition mit verändertem Wert \$EC.value den technischen Anwendungsfall TUC\_KON\_256 „Systemereignis absetzen“ mit folgenden Parametern aufrufen:

```
TUC_KON_256 {
    topic = "OPERATIONAL_STATE/$EC.errorcondition";
    eventType = $EC.type;
    severity = $EC.severity;
    parameters = („Value=$EC.value, $EC.parameterlist“)
}
```

**Tabelle 3: TAB\_KON\_503 Betriebszustand\_Fehlerzustandsliste**

| ErrorCondition<br>(siehe Hinweis 1)                               | Beschreibung   | Type | Sev<br>e<br>rity | max.<br>Fest<br>stell<br>ungs<br>-<br>zeit | Parameterlist<br>(siehe Hinweis 2)             |
|---|--|------|------------------|--|--|
| EC_CardTerminal_<br>Software_Out_Of_<br>Date (\$ctId)             | Software auf<br>Kartenterminal(\$ctId)<br>ist nicht aktuell  | Op   | Info             | 1<br>day                                   | CtID=\$ctId;<br>Bedeutung=<br>\$EC.description |
| EC_CardTerminal_<br>gSMC-KT_Certificate_<br>Expires_Soon (\$ctId) | Das Zertifikat der gSMC-KT<br>im<br>Kartenterminal(\$ctId)<br>läuft in weniger als 5 Wochen<br>ab  | Op   | Info             | 7<br>day<br>s                              | CtID=\$ctId;<br>Bedeutung=<br>\$EC.description |
| EC_Connector_<br>Software_Out_<br>Of_Date                         | I_KSRS_Download::list_<br>Updates<br>liefert mindestens eine<br>UpdateInformation mit einer<br>UpdateInformation/Firmware/<br>FWVersion > aktuelle Version<br>der Konnektorsoftware, deren<br>UpdateInformation/Firmware/<br>FWPriority = „Kritisch“ | Op   | Info             | 1<br>day                                   | Bedeutung=<br>\$EC.description                 |
| EC_FW_Update_Availa<br>ble  | I_KSRS_Download::list_<br>Updates<br>liefert mindestens eine<br>UpdateInformation mit einer<br>UpdateInformation/Firmware/<br>FWVersion > aktuelle Version<br>der Konnektor- oder<br>Kartenterminalsoftware  | Op   | Info             | 1<br>day                                   | Bedeutung=<br>\$EC.description                 |

|                                    |  |     |         |       |  |
|------------------------------------|--|-----|---------|-------|--|
| EC_FW_Not_Valid_Status_Blocked     | Konnektor Firmware muss aktualisiert werden. Zugang zur TI momentan nicht erlaubt.                                       | Sec | Fatal   | 1 day | Bedeutung=\$EC.description   |
| EC_Time_Sync_Not_Successful        | der letzte Synchronisationsversuch der Systemzeit war nicht erfolgreich.   | Op  | Info    | 1 sec | LastSyncAttempt=\$lastSyncAttemptTimestamp;<br>LastSyncSuccess=\$lastSyncSuccessTimestamp;<br>Bedeutung=\$EC.description |
| EC_TSL_Update_Not_Successful       | das letzte Update der TSL war nicht erfolgreich.   | Op  | Info    | 1 sec | Bedeutung=\$EC.description;<br>LastUpdateTSL=\$lastUpdateTSLTimestamp  |
| EC_TSL_Expiring                    | Systemzeit t mit $t > \text{NextUpdate-Element der TSL} - 7 \text{ Tage}$ und $t \leq \text{NextUpdate-Element der TSL}$ | Sec | Info    | 1 day | NextUpdateTSL=\$NextUpdate-Element der TSL;<br>Bedeutung=\$EC.description  |
| EC_BNetzA_VL_Update_Not_Successful | Das letzte Update der BNetzA-VL war nicht erfolgreich  | Op  | Info    | 1 sec | LastUpdateBNetzAVL=\$lastUpdateBNetzAVLTimestamp;<br>Bedeutung=\$EC.description  |
| EC_BNetzA_VL_not_valid             | Systemzeit t mit $t > \text{NextUpdate-Element der BNetzA-VL}$   | Sec | Warning | 1 day | NextUpdateBNetzAVL=\$NextUpdate-Element der BNetzA-VL;<br>Bedeutung=\$EC.description                                     |
| EC_TSL_Trust_Anchor_Expiring       | Gültigkeit des Vertrauensankers ist noch nicht abgelaufen, läuft aber innerhalb von 30 Tagen ab.                         | Sec | Info    | 1 day | ExpiringDateTrustAnchor=Ablaufdatum der Vertrauensanker gültigkeit;<br>Bedeutung=\$EC.description                        |

|  |   |     |         |       |  |
|--|---|-----|---------|-------|--|
| EC_LOG_OVERFLOW                        | <p>Wenn im Rahmen der Regeln für die rollierende Speicherung von Logging-Einträgen Einträge gelöscht werden, die nicht älter als</p> <p>SECURITY_LOG_DAYS, LOG_DAYS bzw. FM_&lt;fmName&gt;_LOG_DAYS sind, tritt der Fehlerzustand ein. Der Fehlerzustand kann nur durch einen Administrator wieder zurückgesetzt werden. Unter Protokoll wird die Liste der auslösenden Protokolle angegeben.</p> | Op  | Warning | 1 sec | <p>Protokoll=\$Protokoll ;</p> <p>Bedeutung=\$EC.description</p>   |
| EC_CRL_Expiring                        | <p>Systemzeit t &gt; NextUpdate der CRL – 3 Tage</p>  | Sec | Warning | 1 day | <p>ExpiringDateCRL=NextUpdate der CRL;</p> <p>Bedeutung=\$EC.description</p>   |
| EC_Time_Sync_Pending_Warning           | <p>MGM_LU_ONLINE=Enabled und keine erfolgreiche Synchronisation der Systemzeit seit d Tagen und d &gt; NTP_WARN_PERIOD und d &lt;= NTP_GRACE_PERIOD. Nach einer Korrektur oder Bestätigung der Systemzeit durch einen Administrator muss der Konnektor wie nach einer erfolgreichen Zeitsynchronisation verfahren, d.h., der Tagezähler wird auf 0 zurückgesetzt.</p>                             | Sec | Warning | 1 day | <p>LastSyncSuccess=\$lastSyncSuccess Timestamp;</p> <p>Bedeutung=\$EC.description</p>  |
| EC_TSL_Out_Of_Date_Within_Grace_Period | <p>Systemzeit t mit t &gt; NextUpdate-Element der TSL und t &lt;= NextUpdate-Element der TSL + CERT_TSL_DEFAULT_GRACE_PERIOD_DAYS und eine neue TSL ist nicht verfügbar</p>   | Sec | Warning | 1 day | <p>NextUpdateTSL=\$NextUpdate-Element der TSL;</p> <p>GracePeriodTSL=CERT_TSL_DEFAULT_GRACE_PERIOD_DAYS;</p> <p>Bedeutung=\$EC.description</p> |

|   |  |     |       |         |   |
|---|--|-----|-------|---------|---|
| EC_CardTerminal_<br>Not_Available<br>(\$ctId) | Kartenterminal(\$ctId) ist nicht verfügbar. Dieser Betriebszustand bezieht sich auf die als „aktiv“ gekennzeichneten KTs.  | Op  | Error | 1 sec   | CtID=\$ctId;<br>Bedeutung=\$EC.description                        |
| EC_No_VPN_TI_<br>Connection                   | Kein sicherer Kanal (VPN) in die Telematikinfrastruktur aufgebaut.<br>Der Wert 300 sec ist abgeleitet aus der maximalen Verbindungsaufbauzeit bei einem Standortausfall des VPN-Zugangsdienstes.     | Op  | Error | 300 sec | Bedeutung=\$EC.description  |
| EC_No_VPN_<br>SIS_Connection                  | Kein sicherer Kanal (VPN) zu den Sicheren Internet Services aufgebaut.<br>Der Wert 300 sec ist abgeleitet aus der maximalen Verbindungsaufbauzeit bei einem Standortausfall des VPN-Zugangsdienstes. | Op  | Error | 300 sec | Bedeutung=\$EC.description  |
| EC_No_Online_<br>Connection                   | Konnektor kann Dienste im Transportnetz nicht erreichen.   | Op  | Error | 10 sec  | Bedeutung=\$EC.description  |
| EC_IP_Addresses_<br>Not_Available             | Die IP-Adressen des Netzkonnektors sind nicht oder falsch gesetzt.   | Sec | Error | 1 sec   | Bedeutung=\$EC.description  |
| EC_CRL_Out_Of_<br>Date                        | Systemzeit t > Next Update der CRL   | Sec | Fatal | 1 day   | NextUpdateCRL=\$NextUpdate der CRL;<br>Bedeutung=\$EC.description |
| EC_Firewall_Not_<br>Reliable                  | Firewall-Regeln konnten nicht fehlerfrei generiert werden oder beim Laden der Firewall-Regeln ist ein Fehler aufgetreten.  | Sec | Fatal | 1 sec   | Bedeutung=\$EC.description  |

|                                    |   |     |       |       |  |
|------------------------------------|---|-----|-------|-------|--|
| EC_Random_Generator_Not_Reliable   | Der Zufallszahlengenerator kann die Anforderungen an die zu erzeugende Entropie nicht erfüllen.   | Sec | Fatal | 1 sec | Bedeutung= \$EC.description  |
| EC_Secure_KeyStore_Not_Available   | Sicherer Zertifikats- und Schlüsselspeicher des Konnektors (gSMC-K oder Truststore) nicht verfügbar   | Sec | Fatal | 1 sec | Bedeutung= \$EC.description  |
| EC_Security_Log_Not_Writable       | Das Sicherheitslog kann nicht geschrieben werden.   | Op  | Fatal | 1 sec | Bedeutung= \$EC.description  |
| EC_Software_Integrity_Check_Failed | Eine oder mehrere konnektorinterne Integritätsprüfungen der aktiven Konnektorbestandteile sind fehlgeschlagen.  | Sec | Fatal | 1 day | Bedeutung= \$EC.description  |
| EC_Time_Difference_Intolerable     | Abweichung zwischen der lokalen Zeit und der per NTP empfangenen Zeit bei der Zeitsynchronisation größer als NTP_MAX_TIMEDIFFERENCE.<br>Nach einer Korrektur oder Bestätigung der Systemzeit durch einen Administrator muss der Konnektor den Fehlerzustand zurücksetzen. | Sec | Fatal | 1 sec | NtpTimedifference= Zeitabweichung;<br>NtpMaxAllowed Timedifference =NTP_MAX_TIMEDIFFERENCE;<br>Bedeutung= \$EC.description |



|  |  |     |         |       |   |
|--|--|-----|---------|-------|---|
| EC_Time_Sync_Pending_Critical          | MGM_LU_ONLINE= Enabled und keine erfolgreiche Synchronisation der Systemzeit seit d Tagen und $d > \text{NTP\_GRACE\_PERIOD}$ Nach einer Korrektur oder Bestätigung der Systemzeit durch einen Administrator muss der Konnektor wie nach einer erfolgreichen Zeitsynchronisation verfahren, d.h., der Tagezähler wird auf 0 zurückgesetzt. | Sec | Fatal   | 1 day | LastSyncSuccess = \$lastSync<br>SuccessTimestamp;<br>NtpGracePeriod= NTP_GRACE_PERIOD;<br>Bedeutung= \$EC.description   |
| EC_TSL_Trust_Anchor_Out_Of_Date        | Gültigkeit des Vertrauensankers ist abgelaufen   | Sec | Fatal   | 1 day | ExpiringDateTrust Anchor= Ablaufdatum der Vertrauensanker gültigkeit;<br>Bedeutung= \$EC.description  |
| EC_TSL_Out_Of_Date_Beyond_Grace_Period | Systemzeit t mit $t > \text{NextUpdate-Element der TSL} + \text{CERT\_TSL\_DEFAULT\_GRACE\_PERIOD\_DAYS}$ und eine neue TSL ist nicht verfügbar  | Sec | Fatal   | 1 day | NextUpdateTSL = \$NextUpdate-Element der TSL;<br>GracePeriodTSL = CERT_TSL_DEFAULT_GRACE_PERIOD_DAYS;<br>Bedeutung= \$EC.description  |
| EC_CRYPTOPERATION_ALARM                | Gemäß TIP1-A_4597 wurde ein potentieller Missbrauch einer Kryptooperation erkannt. Nur der Administrator kann die Alarmmeldung zurücksetzen.   | Sec | Warning | 1 min | Operation= \$Operationsname;<br>Count= \$Summenwert;<br>Arbeitsplatz= \$<Liste operationsaufrufenden workplaceIDs>;<br>Meldung= 'Auffällige Häufung von Operationsaufrufen in den letzten 10 Minuten' |

|                                    |  |        |            |             |                                |
|------------------------------------|--|--------|------------|-------------|--------------------------------|
| EC_OTHER_<br>ERROR_<br>STATE(\$no) | Herstellerspezifische<br>Fehlerzustände, die per \$no<br>(von 1 aufsteigend<br>nummeriert)<br>identifiziert werden.<br>\$Type, \$Severity und<br>\$ParameterList legt<br>der Hersteller<br>nach Bedarf fest. | \$Type | \$Severity | <= 1<br>day | Bedeutung=<br>\$EC.description |
|------------------------------------|--|--------|------------|-------------|--------------------------------|

1610 **Erläuterungen zu TAB\_KON\_503:**

1611 Hinweis 1:  
1612 Jeder Fehlerzustand wird durch einen eindeutigen ErrorCondition identifiziert. Dieser kann  
1613 einen Parameter enthalten. Sind etwa die Kartenterminals mit ctId=47 und das mit ctId=93  
1614 nicht erreichbar, so lauten die ErrorCondition „EC\_CardTerminal\_Not\_Available(47)“ und  
1615 „EC\_CardTerminal\_Not\_Available(93)“.

1616 Hinweis 2:  
1617 EC.description referenziert den Text, der in der Spalte „Beschreibung“ des Zustandes  
1618 spezifiziert wurde.

1619 **[<=]**

1620 Unter „kartenbasiert“ sind nicht nur Lösungen mit Smartcards sondern auch solche mit  
1621 HSMs (Hardware Security Modules) zu verstehen.

1622 **A\_17085 - Bedingung für den Fehlerzustand EC\_No\_VPN\_TI\_Connection**

1623 Wenn MGM\_LU\_ONLINE=Enabled nicht erfüllt ist, DARF der Konnektor den  
1624 Zustand EC\_No\_VPN\_TI\_Connection NICHT annehmen. [**<=**]

1625 **A\_17086 - Bedingung für den Fehlerzustand EC\_No\_VPN\_SIS\_Connection**

1626 Wenn MGM\_LU\_ONLINE=Enabled oder ANLW\_INTERNET\_MODUS=SIS nicht erfüllt ist,  
1627 DARF der Konnektor den Zustand EC\_No\_VPN\_SIS\_Connection NICHT annehmen. [**<=**]

1628 **A\_17087 - Bedingung für den Fehlerzustand EC\_No\_Online\_Connection**

1629 Wenn MGM\_LU\_ONLINE=Enabled nicht erfüllt ist, DARF der Konnektor den  
1630 Zustand EC\_No\_Online\_Connection NICHT annehmen. [**<=**]

1631

1632 **Tabelle 4: TAB\_KON\_504 Ausführungserlaubnis für Dienste in kritischen**  
1633 **Fehlerzuständen**

|  |  |  |  |   |   |  |  |   |  |  |
|--|--|--|--|---|---|--|--|---|--|--|
|  | EC_<br>Soft<br>ware<br>_<br>Inte<br>grity<br>_<br>Chec<br>k_<br>Faile<br>d | EC_<br>Rand<br>om_<br>Gene<br>rator<br>_<br>Not_<br>Relia<br>ble | EC_<br>Sec<br>urity<br>_<br>Log<br>_<br>Not_<br>Writ<br>able | EC_<br>Tim<br>e_<br>Syn<br>c_<br>Pen<br>ding<br>_<br>Critic<br>al | EC_<br>_Ti<br>me<br>_<br>Dif<br>fe<br>ren<br>ce<br>_<br>Int<br>ole<br>r<br>abl<br>e | E<br>C<br>_<br>C<br>RL<br>_<br>O<br>ut<br>_<br>Of<br>_<br>D<br>at<br>e | EC_<br>TSL<br>_<br>Out<br>_<br>Of_<br>Dat<br>e_<br>Bey<br>ond<br>_<br>Gra<br>ce_<br>Peri<br>od | EC_<br>TSL<br>_<br>Tru<br>st_<br>Anc<br>hor<br>_<br>Out<br>_<br>Of_<br>Dat<br>e | EC_<br>Secu<br>re_<br>KeyS<br>tore<br>_<br>Not_<br>Avail<br>able | EC_<br>FW<br>_<br>Not<br>_<br>Vali<br>d_<br>Sta<br>tus<br>_<br>Blo<br>cke<br>d |
|--|--|--|--|---|---|--|--|---|--|--|

| Technische Use Cases (TUCs) der Basisdienste relevant für Fachanwendung und die Kommunikation mit Weiteren Anwendungen und SIS |   |   |   |   |   |   |   |   |   |   |
|--|---|---|---|---|---|---|---|---|---|---|
| Zugriffsberechtigungsdienst  |   |   |   |   |   |   |   |   |   |   |
| TUC_KON_000 Prüfe Zugriffsberechtigung   | - | x | x | x | x | x | x | x | x | x |
| Dienstverzeichnisdienst  |   |   |   |   |   |   |   |   |   |   |
| TUC_KON_041 Einbringen der Endpunktinformationen während der Bootup-Phase  | - | - | - | x | x | x | x | x | x | x |
| Kartenterminaldienst   |   |   |   |   |   |   |   |   |   |   |
| TUC_KON_051 Mit Anwender über Kartenterminal interagieren  | - | - | - | - | - | x | x | x | - | x |
| Kartendienst   |   |   |   |   |   |   |   |   |   |   |
| TUC_KON_005 Card-to-Card authentisieren  | - | - | - | - | - | x | x | x | - | x |
| TUC_KON_006 Datenzugriffsaudit eGK schreiben   | - | - | - | - | - | x | x | x | - | x |
| TUC_KON_018 eGK-Sperrung prüfen  | - | - | - | - | - | x | x | x | - | x |
| TUC_KON_024 Karte zurücksetzen   | - | - | - | - | - | x | x | x | - | x |
| TUC_KON_026 Liefere CardSession  | - | - | - | - | - | x | - | x | - | - |
| TUC_KON_200 SendeAPDU  | - | - | - | - | - | x | x | x | - | x |
| TUC_KON_202 LeseDatei  | - | - | - | - | - | x | x | x | - | x |

|  |   |   |   |   |   |   |   |   |   |   |
|--|---|---|---|---|---|---|---|---|---|---|
| TUC_KON_203<br>SchreibeDatei                               | - | - | - | - | - | x | x | x | - | x |
| TUC_KON_209<br>LeseRecord                                  | - | - | - | - | - | x | x | x | - | x |
| Systeminformationsdienst                                   |   |   |   |   |   |   |   |   |   |   |
| TUC_KON_256<br>Systemereignis absetzen                     | - | x | x | x | x | x | x | x | x | x |
| Verschlüsselungsdienst                                     |   |   |   |   |   |   |   |   |   |   |
| TUC_KON_072 Daten<br>symmetrisch<br>verschlüsseln          | - | - | - | x | x | x | x | x | - | x |
| TUC_KON_073 Daten<br>symmetrisch<br>entschlüsseln          | - | - | - | x | x | x | x | x | - | x |
| Zertifikatsdienst  |   |   |   |   |   |   |   |   |   |   |
| TUC_KON_034 Zertifikatsi<br>nformationen extrahieren       | - | - | - | x | x | x | x | x | - | x |
| Protokollierungsdienst                                     |   |   |   |   |   |   |   |   |   |   |
| TUC_KON_271 Schreibe<br>Protokolleintrag                   | - | x | x | x | x | x | x | x | x | x |
| TLS-Dienst   |   |   |   |   |   |   |   |   |   |   |
| TUC_KON_110 Kartenbasi<br>erte TLS-<br>Verbindung aufbauen | - | - | - | - | - | - | - | - | - | - |
| Verbindung zum VPN-<br>Konzentrator                        |   |   |   |   |   |   |   |   |   |   |
| TUC_VPN-ZD_0001<br>„IPsec Tunnel TI<br>aufbauen“           | - | - | - | - | - | - | - | - | - | - |

|   |   |   |   |   |   |   |   |   |   |   |
|---|---|---|---|---|---|---|---|---|---|---|
| TUC_VPN-ZD_0002<br>„IPsec Tunnel SIS<br>aufbauen“ | - | - | - | - | - | - | - | - | - | - |
| <b>Operationen der Basisdienste</b>               |   |   |   |   |   |   |   |   |   |   |
| Kartendienst                                      |   |   |   |   |   |   |   |   |   |   |
| VerifyPin   | - | - | - | - | - | x | x | x | - | x |
| UnblockPin  | - | - | - | - | - | x | x | x | - | x |
| ChangePin   | - | - | - | - | - | x | x | x | - | x |
| GetPinStatus                                      | - | - | - | - | - | x | x | x | - | x |
| Systeminformationsdienst                          |   |   |   |   |   |   |   |   |   |   |
| Schnittstelle der<br>Ereignissenke                | - | x | x | x | x | x | x | x | x | x |
| GetCardTerminals                                  | - | x | x | x | x | x | x | x | x | x |
| GetCards  | - | x | x | x | x | x | x | x | x | x |
| GetResourceInformation                            | - | x | x | x | x | x | x | x | x | x |
| Subscribe   | - | x | x | x | x | x | x | x | x | x |
| RenewSubscription                                 | - | x | x | x | x | x | x | x | x | x |
| Unsubscribe                                       | - | x | x | x | x | x | x | x | x | x |
| GetSubscription                                   | - | x | x | x | x | x | x | x | x | x |
| Verschlüsselungsdienst                            |   |   |   |   |   |   |   |   |   |   |
| EncryptDocument                                   | - | - | - | - | - | x | x | x | - | x |
| DecryptDocument                                   | - | - | - | - | - | x | x | x | - | x |
| Signaturdienst                                    |   |   |   |   |   |   |   |   |   |   |
| SignDocument                                      | - | - | - | - | - | x | x | x | - | x |

|                            |   |   |   |   |   |   |   |   |   |   |
|----------------------------|---|---|---|---|---|---|---|---|---|---|
| VerifyDocument             | - | - | - | - | - | x | x | x | - | x |
| GetJobNumber               | - | - | - | - | - | x | x | x | - | x |
| StopSignature              | - | - | - | - | - | x | x | x | - | x |
| ActivateComfortSignature   | - | - | - | - | - | x | x | x | - | x |
| DeactivateComfortSignature | - | - | - | - | - | x | x | x | - | x |
| GetSignatureMode           | - | - | - | - | - | x | x | x | - | x |
| Authentifizierungsdienst   |   |   |   |   |   |   |   |   |   |   |
| ExternalAuthenticate       | - | - | - | - | - | x | x | x | - | x |
| Zertifikatsdienst          |   |   |   |   |   |   |   |   |   |   |
| ReadCardCertificate        | - | - | - | - | - | x | x | x | x | x |
| CheckCertificateExpiration | - | - | - | - | - | x | x | x | x | x |
| VerifyCertificate          | - | - | - | - | - | x | - | x | x | x |
| Zeitdienst                 |   |   |   |   |   |   |   |   |   |   |
| I_NTP_Time_Information     | - | - | - | - | - | x | x | x | x | - |
| Konnektormanagement        |   |   |   |   |   |   |   |   |   |   |
| Softwareaktualisierung     | x | x | x | x | x | x | x | x | x | x |
| Protokolleinsicht          | x | x | x | x | x | x | x | x | x | x |
| Werksreset                 | x | x | x | x | x | x | x | x | x | x |
| Sonstiges                  | - | x | x | x | x | x | x | x | x | x |

1634 In den kritischen Fehlerzuständen, in denen keine TLS-Verbindung ins LAN aufgebaut  
1635 werden (EC\_Random\_Generator\_Not\_Reliable, EC\_Software\_Integrity\_Check\_Failed,  
1636 EC\_Security\_Log\_Not\_Writable, EC\_Time\_Sync\_Pending\_Critical,  
1637 EC\_Time\_Difference\_Intolerable), kann keine Verbindung zu den Kartenterminals  
1638 aufgebaut werden. Infolge sind hier keine Kartenoperationen zugelassen.

1639 Wenn keine Verbindung zum VPN-Konzentrator des SIS aufgebaut werden kann, ist  
1640 dadurch das Internet nicht über den Konnektor erreichbar. Wenn keine Verbindung zum  
1641 VPN-Konzentrator der TI aufgebaut werden kann, sind Bestandsnetze nicht erreichbar.

1642 Bezüglich der Administration des Konnektors im Zustand EC\_FIREWALL\_NOT\_RELIABLE  
1643 ist eine Abstimmung mit der Prüfstelle und der Zertifizierungsstelle notwendig.

#### 1644 **A\_16203 - Nutzbarkeit im Zustand EC\_FIREWALL\_NOT\_RELIABLE**

1645 Im Zustand EC\_Firewall\_Not\_Reliable DARF der Konnektor NICHT nutzbar sein.  
1646 Möglichkeiten zur Behebung des Zustandes EC\_Firewall\_Not\_Reliable sind mit dem CC -  
1647 Evaluierer und Zertifizierer abzustimmen. [ $\leq$ ]

1648 Die Architektur der TI ist so angelegt, dass die Fehlerzustände mit Severity=Fatal in den  
1649 Tabellen TAB\_KON\_504 und TAB\_KON\_503 mit vernachlässigbarer Wahrscheinlichkeit  
1650 von externen Einflüssen abhängen. Die SLAs für Dienste der zentralen TI-Plattform sind  
1651 so gefasst, dass diese schwerwiegend verletzt werden müssten, um dadurch einen  
1652 Konnektor in einen solchen kritischen Zustand zu bringen (externer Fehler aus Sicht des  
1653 Konnektors). Dass beispielsweise der TSL-Dienst über den Zeitraum der Grace-Period-  
1654 TSL (typisch: 7 Tage) nicht erreichbar ist (ErrorCondition EC\_TSL\_Out\_Of\_Date  
1655 \_Beyond\_Grace\_Period), kann nur bei massiver Verletzung der für zentrale Dienste  
1656 festgelegten SLAs eintreten.

1657 Um die konnektorinternen Fehlerquellen zu erfassen, die dazu führen, dass ein  
1658 Fehlerzustand mit Severity=Fatal eintritt oder ein anderer Zustand, in dem der  
1659 Konnektor nicht verwendbar ist, wird Folgendes gefordert:

#### 1660 **TIP1-A\_5148 - Performance - Konnektor - Mittlerer Abstand zwischen Ausfällen**

1661 Der Konnektorhersteller MUSS den mittleren Zeitabstand zwischen Ausfällen (MTBF) als  
1662 Produkteigenschaft ausweisen. Der Konnektor soll einen mittleren Zeitabstand zwischen  
1663 Ausfällen (MTBF) von mindestens 50 Jahren haben.  
1664 Ein „Ausfall“ gilt dann als eingetreten, wenn

- 1665 • der Konnektor nicht mehr gebootet werden kann, d. h. kein  
1666 „BOOTUP/BOOTUP\_COMPLETE“ Event ausgelöst wird, und dies nicht auf einen  
1667 externen Fehler zurückzuführen ist,
- 1668 • oder sich der Konnektor in einem Fehlerzustand mit Severity=Fatal befindet, der  
1669 nicht auf einen externen Fehler zurückzuführen ist,
- 1670 • oder Funktionen des Konnektors ausgefallen sind, ohne dass dies auf externe  
1671 Fehler zurückzuführen ist.

1672 [ $\leq$ ]

1673 Bei einem mittleren Zeitabstand zwischen Ausfällen (MTBF) von 50 Jahren ist die  
1674 Wahrscheinlichkeit, dass ein Fehlerzustand mit Severity=Fatal auftritt, kleiner 2 % pro  
1675 Jahr.

#### 1676 **TIP1-A\_4510-03 - Sicherheitskritische Fehlerzustände**

1677 Der Konnektor MUSS bei eingetretenem Fehlerzustand aus Tabelle Tab\_Kon\_503  
1678 Betriebszustand\_Fehlerzustandsliste mit Severity=Fatal dafür sorgen, dass von den  
1679 Operationen der Basisdienste und Technische Use Cases (TUCs) der Basisdienste, die  
1680 relevant für Fachanwendungen sind, nur erlaubte Operationen und TUCs gestartet und  
1681 ausgeführt werden.

1682 Welche Operationen und TUCs je eingetretenem Fehlerzustand ausgeführt werden  
1683 dürfen, legt Tabelle „TAB\_KON\_504 Ausführungserlaubnis für Dienste in kritischen  
1684 Fehlerzuständen“ fest: Jede Erlaubnis ist dort durch ein „x“ definiert.

1685 Abweichend zu Angaben in der Tabelle TAB\_KON\_504 DÜRFEN folgende Operationen und  
1686 TUCs NICHT im Zustand EC\_Firewall\_Not\_Reliable ausgeführt werden:



- 1687 • TUC\_KON\_000 PrüfeAufrufkontext
- 1688 • TUC\_KON\_041 Einbringen der Endpunktinformationen während der Bootup-Phase
- 1689 • GetCardTerminals
- 1690 • GetCards
- 1691 • GetResourceInformation
- 1692 • Subscribe
- 1693 • RenewSubscription
- 1694 • Unsubscribe
- 1695 • GetSubscription
- 1696 • ReadCardCertificate
- 1697 • CheckCertificateExpiration
- 1698 • VerifyCertificate

1699 Sind mehrere Fehlerzustände gleichzeitig eingetreten, dürfen nur die Operationen und  
1700 TUCs ausgeführt werden, die für alle eingetretenen Fehlerzustände erlaubt sind. Der  
1701 Konnektor muss Anfragen, die auf Grund eines kritischen Fehlerzustandes nicht  
1702 ausgeführt oder abgebrochen werden, mit einem Fehler (Fehlercode 4002) beantworten.  
1703

1704 **Tabelle 5: TAB\_KON\_502 Fehlercodes „Betriebszustand“**

| Fehlercode | ErrorType | Severity | Fehlertext  |
|------------|-----------|----------|---|
| 4002       | Security  | Fatal    | Der Konnektor befindet sich in einem kritischen Betriebszustand |

1705  
1706 [ $\leq$ ]

### 1707 3.3.1 Betriebsaspekte

1708 Der Konnektor soll per Signaleinrichtung am Konnektor die Fehlerzustände mit Severity  
1709 „Error“ und „Fatal“ anzeigen (siehe [TIP1-A\_4843]).

#### 1710 **TIP1-A\_4513 - Betriebszustände anzeigen und Fehlerzustände zurücksetzen**

1711 Der Konnektor MUSS es dem Administrator ermöglichen, den aktuellen Betriebszustand  
1712 einzusehen und Fehlerzustände zurückzusetzen, soweit diese Möglichkeit in Tabelle  
1713 „TAB\_KON\_503 Betriebszustand\_Fehlerzustandsliste“ für den jeweiligen Fehlerzustand  
1714 festgelegt ist.

1715 Ferner MUSS es die Managementschnittstelle dem Administrator ermöglichen,  
1716 Konfigurationsänderungen gemäß Tabelle TAB\_KON\_505 vorzunehmen:

1717 **Tabelle 6: TAB\_KON\_505 Konfigurationswerte Missbrauchserkennung**

| ReferenzID             | Belegung   | Bedeutung und Administrator-Interaktion  |
|------------------------|--|--|
| EVT_MONITOR_OPERATIONS | Liste von:<br>-<br>Operationsname<br>- OK_Val<br>(Nummer)<br>- NOK_Val | Der Administrator MUSS in der Liste der zur Missbrauchserkennung überwachbaren Operationen alle Listeneinträge einsehen können. Er MUSS den jeweiligen Alarmwert |

|  |                                     |   |
|--|-------------------------------------|---|
|  | (Nummer)<br>- Alarmwert<br>(Nummer) | editieren können (0-9999,<br>0=deaktiviert).<br>OK_VAL und NOK_VAL DÜRFEN durch<br>den Administrator NICHT veränderbar<br>sein. |
|--|-------------------------------------|---|

1718  
1719 [ $\leq$ ]

### 1720 3.4 Fachliche Anbindung der Clientsysteme

1721 Für die Schnittstellen des Konnektors zu den Clientsystemen kann gesteuert werden:

- 1722 • ob die Kommunikation zwischen Konnektor und Clientsystemen hinsichtlich
- 1723 Vertraulichkeit, Integrität und Authentizität zwingend durch TLS gesichert
- 1724 werden muss
- 1725 • ob sich Clientsysteme zwingend authentisieren müssen
- 1726 • welche Clientsysteme auf den Konnektor zugreifen dürfen (Whitelisting)

1727 Dabei werden die folgenden zwei Nutzungsszenarien nicht unterschieden:

- 1728 • Nutzung von Fachanwendungen (in Form von Fachmodulen)
- 1729 • Nutzung von Basisdiensten des Konnektors

1730 Sowohl die Anbindung zur Administration des Konnektors, als auch die Anbindung zur

1731 Nutzung von Bestandsnetzen oder dem gesicherten Internetzugang sind nicht

1732 Gegenstand dieser Schnittstellenfestlegungen. Für die Anbindung zu Administration wird

1733 diese im Kapitel Konnektormanagement beschrieben, für die Anbindung von

1734 Bestandsnetzen bzw. dem gesicherten Internetzugang ist diese Art der Regelung nicht

1735 erforderlich, da es sich dort um Routing-Funktionen handelt.

1736 Die seitens des Administrators einstellbaren Werte und Listen sind, der allgemeinen

1737 Struktur dieses Dokuments folgend, im Unterkapitel 3.4.1 Betriebsaspekte beschrieben.

1738

#### 1739 TIP1-A\_4514 - Verfügbarkeit einer TLS-Schnittstelle

1740 Der Konnektor MUSS TLS in Richtung der Clientsysteme für alle Außenschnittstellen der

1741 Basisdienste:

- 1742 • Dienstverzeichnisdienst
- 1743 • Kartenterminaldienst
- 1744 • Systeminformationsdienst
- 1745 • Verschlüsselungsdienst
- 1746 • Signaturdienst
- 1747 • Zertifikatsdienst
- 1748 • Kartendienst
- 1749 • LDAP-Proxy

1750 unterstützen.

1751 Ferner MUSS der Konnektor für die SOAP-Endpunkte der Fachmodule TLS unterstützen.

1752 Der Konnektor MUSS sich mittels ID.AK.AUT gegenüber dem Client authentisieren.

1753 [ $\leq$ ]

**TIP1-A\_4515 - Verpflichtung zur Nutzung der TLS-Verbindung**

Der Konnektor MUSS immer TLS-Verbindungsanfragen von Clientsystemen annehmen.  
Der Konnektor MUSS bei gesetzter Variable ANCL\_TLS\_MANDATORY = Enabled den Verbindungsversuch von Clientsystemen ohne TLS ablehnen. Ausgenommen hiervon sind Anfragen an den Dienstverzeichnisdienst bei gesetzter Variable ANCL\_DVD\_OPEN = Enabled.  
[<=]

**TIP1-A\_4516 - Authentifizierung der Clients über Basic-Auth und X.509-Zertifikate**

Der Konnektor MUSS zur Client-Authentifizierung die Verfahren Basic Authentication (Username/Password) [RFC2617] über HTTP/TLS [RFC2818] und zertifikatsbasierte Client-Authentifizierung (X.509) [gemSpec\_PKI#8.3.1.4] über TLS anbieten.  
Dabei MUSS für eine erfolgreiche Prüfung bei Basic Authentication:

- das seitens des Clientsystems präsentierte Credential in ANCL\_CUP\_LIST enthalten sein

Für eine erfolgreiche Prüfung mit zertifikatsbasierter Client-Authentifizierung MUSS:

- das seitens des Clientsystems präsentierte Zertifikat in ANCL\_CCERT\_LIST enthalten sein
- die Zertifikatsprüfung (nur Prüfung auf „mathematische Korrektheit“ und „Gültigkeit nicht abgelaufen“) erfolgreich durchlaufen werden

Schlägt die Prüfung fehl, MUSS der Verbindungsversuch des Clientsystem abgelehnt werden.[<=]

Bei der Authentisierung des Clientsystems geht es um eine Authentisierung in zwei Richtungen:

1. Authentisierung des Clientsystems in der Rolle eines Clients gegenüber dem Konnektor für die Übertragung von SOAP-Requests.
2. Authentisierung des Clientsystems in der Rolle eines Servers gegenüber dem Konnektor zum Empfang von CETP-Ereignismitteilungen des Systeminformationsdienstes.

Für beide Richtungen kann das Clientsystem dasselbe Zertifikat verwenden.

**TIP1-A\_5009 - Authentifizierungsvarianten für Verbindungen zwischen Konnektor und Clientsystemen**

Der Konnektor MUSS für Verbindungen zu Clientsystemen als Authentifizierungsmethode ausschließlich folgende Varianten erlauben:

1. Für Verbindungen mit dem Konnektor in der Rolle des Servers (SOAP-Requests):
  - TLS-Server-Authentifizierung des Konnektors und TLS-Client-Authentifizierung des Clientsystems
  - TLS-Server-Authentifizierung des Konnektors und BasicAuthentifizierung des Clientsystems
  - TLS-Server-Authentifizierung des Konnektors ohne TLS-Client-Authentifizierung des Clientsystems
  - Keine Authentifizierung des Konnektors und des Clientsystems
2. Für Verbindungen mit dem Konnektor in der Rolle des Clients (CETP-Protokoll):
  - TLS-Server-Authentifizierung des Clientsystems und TLS-Client-Authentifizierung des Konnektors

- 1799 • TLS-Server-Authentifizierung des Clientsystems ohne TLS-Client-Authentifizierung des Konnektors

- 1801 • Keine Authentifizierung des Konnektors und des Clientsystems

1802 Alle anderen Verbindungsversuche von Clientsystemen MÜSSEN vom Konnektor  
1803 abgelehnt werden.

1804 [ $\leq$ ]

1805 Für die Anbindung der Clientsysteme ergeben sich verschiedene Konfigurationsvarianten  
1806 bezüglich der Absicherung der Verbindungen zwischen Konnektor und Clientsystemen.  
1807 Tabelle TAB\_KON\_852 listet die Varianten für die Verbindungen zum Aufruf der  
1808 Webservice-Schnittstellen (Varianten SOAP1 bis SOAP4), für die Verbindungen zum  
1809 Senden von Events (Varianten CETP1 und CETP2) und für Verbindungen zum Abruf des  
1810 Dienstverzeichnisses (Varianten DVD1, DVD2 und DVD3).

1811 **Tabelle 7: TAB\_KON\_852 Konfigurationsvarianten der Verbindungen zwischen Konnektor**  
1812 **und Clientsystemen**

| Konfigu-<br>rations-<br>variant<br>e | ANCL_<br>TLS_<br>MAN-<br>DATORY | ANCL_<br>CAUT_<br>MAN-<br>DATORY | ANCL_<br>CAUT_<br>MODE | ANCL_<br>DVD_<br>OPEN | Bedeutung   |
|--------------------------------------|---------------------------------|----------------------------------|------------------------|-----------------------|---|
| CETP1                                | Enabled                         | Irrelevant                       | Irrelevant             | Irrelevant            | Der Konnektor sendet Events ausschließlich über TLS. Er authentisiert sich, wenn ihn das Clientsystem im Rahmen des TLS-Handshakes dazu auffordert.                         |
| CETP2                                | Disabled                        | Irrelevant                       | Irrelevant             | Irrelevant            | Der Konnektor sendet Events immer über eine TCP-Verbindung ohne TLS.  |
| SOAP1                                | Enabled                         | Enabled                          | CERTIFICATE            | Irrelevant            | Der Konnektor akzeptiert vom Clientsystem nur Aufrufe über TLS. Der Konnektor verlangt beim TLS-Handshake die Authentisierung des Clientsystems per Zertifikat.             |
| SOAP2                                | Enabled                         | Enabled                          | PASSWORD               | Irrelevant            | Der Konnektor akzeptiert vom Clientsystem nur Aufrufe über TLS. Der Konnektor prüft auf Anwendungsebene, dass Aufrufer sich per Username/Password [RFC2617] authentisieren. |
| SOAP3                                | Enabled                         | Disabled                         | Irrelevant             | Irrelevant            | Der Konnektor akzeptiert vom Clientsystem nur Aufrufe über TLS. Der Konnektor nimmt keine Clientauthentifizierung vor.  |

|       |            |            |            |            |  |
|-------|------------|------------|------------|------------|--|
| SOAP4 | Disabled   | Irrelevant | Irrelevant | Irrelevant | Der Konnektor akzeptiert vom Clientsystem sowohl Aufrufe ohne TLS als auch über TLS. Im zweiten Fall sollte der Konnektor das Clientsystem nicht authentifizieren, wenn er es aber für den Sonderfall ANCL_CAUT_MANDATORY=Enabled aktuell tut, sehen wir das nicht als Fehler. |
| DVD1  | Irrelevant | Irrelevant | Irrelevant | Enabled    | Zugriff auf Dienstverzeichnisdienst kann über HTTP und HTTPS erfolgen.   |
| DVD2  | Enabled    | *          | *          | Disabled   | Zugriff auf Dienstverzeichnisdienst kann nur über HTTPS erfolgen.<br>*) Bzgl. Clientauthentisierung wirken die Schalter wie in SOAP 1, SOAP 2, SOAP 3  |
| DVD3  | Disabled   | Irrelevant | Irrelevant | Disabled   | Zugriff auf Dienstverzeichnisdienst kann über HTTP und HTTPS erfolgen.   |

#### A\_21224 - Authentifizierung für Verbindungen zwischen Konnektor und Clientsystemen bei LDAPS

Bei der Verwendung des LDAPS-Proxies im Konnektor, MUSS sich der Konnektor abhängig von der Stellung der Schalter ANCL\_TLS\_MANDATORY, ANCL\_CAUT\_MANDATORY und ANCL\_CAUT\_MODE gemäß der Tabelle TAB\_KON\_860 verhalten.

**Tabelle 8: TAB\_KON\_860 Konfigurationsvarianten der Verbindungen zwischen Konnektor und Clientsystemen bei LDAPS**

| Konfigurationsvariante | ANCL_TLS_MANDATORY | ANCL_CAUT_MANDATORY | ANCL_CAUT_MODE | Bedeutung   |
|------------------------|--------------------|---------------------|----------------|---|
| LDAPS1                 | Enabled            | Enabled             | CERTIFICATE    | Der Konnektor akzeptiert vom Clientsystem nur Aufrufe über TLS. Der Konnektor verlangt beim TLS-Handshake die Authentisierung des Clientsystems per Zertifikat. |
| LDAPS2                 | Enabled            | Enabled             | PASSWORD       | Der Konnektor akzeptiert vom Clientsystem nur Aufrufe über TLS. Der Konnektor nimmt keine Clientauthentifizierung vor.  |

|        |          |            |            |  |
|--------|----------|------------|------------|--|
| LDAPS3 | Enabled  | Disabled   | Irrelevant | Der Konnektor akzeptiert vom Clientsystem nur Aufrufe über TLS. Der Konnektor nimmt keine Clientauthentifizierung vor.   |
| LDAPS4 | Disabled | Irrelevant | Irrelevant | Der Konnektor akzeptiert vom Clientsystem sowohl Aufrufe ohne TLS als auch über TLS. Im zweiten Fall sollte der Konnektor das Clientsystem nicht authentifizieren, wenn er es aber für den Sonderfall ANCL_CAUT_MANDATORY=Enabled, ANCL_CAUT_MODE=CERTIFICATE aktuell tut, sehen wir das nicht als Fehler. |

[<=]

Aus A\_21224 resultiert direkt, dass als Client-Authentisierung für LDAPS nur Client-Zertifikate unterstützt werden müssen. Die Authentisierung mit Username/Passwort wird bei LDAPS nicht unterstützt.

Es sei noch einmal betont, dass TIP1-A\_5009 sich nur auf SOAP und CETP bezieht und TIP1-A\_4516 das Basic-Authentication Verfahren nur für HTTP fordert.

### 3.4.1 Betriebsaspekte

Damit sich ein Clientsystem mittels X.509 authentisieren kann, muss es über ein entsprechendes Zertifikat verfügen. Diese Zertifikate kann der Administrator entweder mit seinen lokalen Mitteln selbst oder mittels des Konnektors erzeugen. In beiden Fällen müssen diese Zertifikate sowohl im Clientsystemen (hier zusammen mit ihren privaten Schlüsseln), als auch im Konnektor vorhanden sein.

Da es sich um eine lokal begrenzte Authentisierung im Verantwortungsbereich des Betreibers des lokalen Netzes handelt, werden keine weiteren Vorgaben zu den Schlüsselspeichern auf Clientsystemseite erhoben. Auch hinsichtlich der außerhalb des Konnektors erzeugten Zertifikate gelten keine weiteren Vorgaben. Ferner ist eine Online-Prüfung dieser Zertifikate nicht erforderlich.

#### **TIP1-A\_4517 - Schlüssel und X.509-Zertifikate für die Authentisierung des Clientsystems erzeugen und exportieren sowie X.509-Zertifikate importieren**

Der Konnektor MUSS die Erstellung und den Export von X.509-Zertifikaten für Clientsysteme und der zugehörigen privaten Schlüssel durch den Administrator über das Managementinterface ermöglichen. Hierbei MUSS der Konnektor dem Administrator die Möglichkeit geben, das kryptographische Verfahren RSA-2048 oder ECC-256 auszuwählen. Als Exportformat MUSS PKCS#12 verwendet werden. Die so erstellten Zertifikate werden zu ANCL\_CCERT\_LIST angefügt.

Der Konnektor MUSS dem Administrator ferner den Import von konnektorfremden X.509-Zertifikaten für Clientsysteme über das Managementinterface ermöglichen. Die so importierten Zertifikate werden zu ANCL\_CCERT\_LIST angefügt.

[<=]

#### **TIP1-A\_4518 - Konfiguration der Anbindung Clientsysteme**

Der Administrator MUSS in der Managementoberfläche die in TAB\_KON\_506 genannten Parameter im Managementinterface konfigurieren können.

- 1853 Wird ANCL\_TLS\_MANDATORY auf ENABLED gewechselt, MÜSSEN alle nicht per TLS  
1854 gesicherten http-Verbindungen geschlossen werden, sobald die in den Verbindungen  
1855 jeweils aktuell laufenden Außenschnittstelle-Operationen abgeschlossen wurden, mit  
1856 Ausnahme von http-Verbindungen zum Dienstverzeichnisdienst.  
1857 Der Konnektor MUSS den Administrator geeignet und verständlich auf seine  
1858 Verantwortung für die Sicherung der Kommunikation hinweisen.

1859 **Tabelle 9: TAB\_KON\_506 Konfigurationsparameter der Clientsystem-Authentisierung**

| ReferenzID          | Belegung  | Bedeutung und Administrator-Interaktion  |
|---------------------|---|--|
| ANCL_TLS_MANDATORY  | Enabled/Disabled                                    | Der Administrator MUSS die verpflichtende Verwendung eines TLS gesicherten Kanals an- oder abschalten können.<br>Wenn ANLW_ANBINDUNGS_MODUS = Parallel MUSS der Administrator vor dem Disablen von ANCL_TLS_MANDATORY einen Warnhinweis bestätigen, der ihn über die mit der Abschaltung verbundenen Risiken informiert und darlegt, dass in diesem Fall der Nutzer die Verantwortung für die Sicherstellung der vertraulichen Übertragung übernimmt.<br>Default-Wert: Enabled |
| ANCL_CAUT_MANDATORY | Enabled/Disabled                                    | Der Administrator MUSS die verpflichtende Authentifizierung der Clientsysteme an- oder abschalten können.<br>Default-Wert: Enabled   |
| ANCL_CAUT_MODE      | CERTIFICATE / PASSWORD                              | Der Administrator MUSS konfigurieren können, welcher Client Authentifizierungsmodus genutzt werden kann bzw. genutzt werden muss.<br>Default-Wert: CERTIFICATE   |
| ANCL_CCERT_LIST     | Liste von X.509-Zertifikaten zugeordnet zu ClientID | Whitelist an importierten oder vom Konnektor erzeugten X.509-Zertifikaten und dazugehörigen Clientsystem IDs. Der Administrator MUSS die Liste der Zertifikate und den zugehörigen Clientsystemen verwalten können, der Inhalt der Zertifikate MUSS menschlich lesbar dargestellt werden.<br>Es muss für den Administrator erkennbar sein, welches kryptographische Verfahren (RSA-2048 oder ECC -256) dem jeweiligen Zertifikat zugrunde liegt.                               |



|               |   |  |
|---------------|---|--|
| ANCL_CUP_LIST | Liste von Benutzer/Passwort Kombinationen, zugeordnet zu ClientID | Whitelist an UserCredentials und dazugehörigen Clientsystem IDs. Der Administrator MUSS eine Liste von Credentials und zugehörigem Clientsystem verwalten können. Bei diesen Benutzer-/Passwortkombinationen handelt es sich nicht um personenbezogene Credentials, sondern um clientbezogene. |
| ANCL_DVD_OPEN | Enabled/Disabled  | Der Administrator MUSS konfigurieren können, ob der Zugriff auf den Dienstverzeichnisdienst auch dann über einen ungesicherten http-Kanal erfolgen kann (ENABLED), wenn ANCL_TLS_MANDATORY = ENABLED ist.<br>Default-Wert: Enabled   |

1860

1861 [ $\leq$ ]1862 **3.5 Clientsystemschnittstelle**1863 **TIP1-A\_5401 - Parallele Nutzbarkeit Clientsystemschnittstelle**

1864 Alle Schnittstellen, die der Konnektor den Clientsystemen zur Verfügung stellt, MÜSSEN  
 1865 parallel durch mehrere Aufrufer nutzbar sein.

1866 [ $\leq$ ]1867 **3.5.1 SOAP-Schnittstelle**

1868 Für die Beschreibung der SOAP-Schnittstelle zum Clientsystem wird in dieser  
 1869 Spezifikation WSDL Version 1.1 [WSDL1.1] eingesetzt. Die Interoperabilität zwischen  
 1870 verschiedenen SOAP-Implementierungen wird durch die Vorgaben des WS-I Basic Profile  
 1871 erreicht.

1872 **A\_15601 - SOAP für Web-Services der Basisdienste**

1873 Der Konnektor MUSS für die an der Clientsystemschnittstelle definierten Web-Services  
 1874 der Basisdienste [SOAP1.1] verwenden. [ $\leq$ ]

1875 **TIP1-A\_4519 - Web-Services konform zu [BasicProfile1.2]**

1876 Der Konnektor MUSS die für die Clientsystemschnittstelle definierten Web-Services  
 1877 konform zu [BasicProfile1.2] anbieten.

1878 Abweichend von R1012 in [BasicProfile1.2] MUSS der Konnektor nur das Character  
 1879 Encoding UTF-8 unterstützen. Andere Kodierungen MUSS der Konnektor mit einem Fehler  
 1880 beantworten. [ $\leq$ ]

1881 **TIP1-A\_4519-01 - ab PTV4: Web-Services konform zu [BasicProfile1.2]**

1882 Der Konnektor MUSS die für die Clientsystemschnittstelle definierten Web-Services der  
 1883 Basisdienste konform zu [BasicProfile1.2] anbieten.

1884 [ $\leq$ ]

**A\_15606 - Character Encoding für Web-Services**

Abweichend von R1012 in [BasicProfile1.2] und [BasicProfile2.0] MUSS der Konnektor nur das Character Encoding UTF-8 unterstützen. Andere Kodierungen MUSS der Konnektor mit einem Fehler beantworten. [≤]

Da der Konnektor UTF-16 nicht unterstützt, muss das Clientsystem den Request in UTF-8 kodieren. Diese Festlegungen gelten nur für die eigentliche SOAP-Nachricht. Sind in der SOAP-Nachricht base64-encodierte XML-Elemente vorhanden, so können diese XML-Elemente andere Zeichencodierungen aufweisen.

**Fachmodule**

Fachmodule können für Web-Services, die Clientsystemen bereitgestellt werden, entweder [SOAP1.1] mit [BasicProfile1.2] oder [SOAP1.2] mit [BasicProfile2.0] verwenden. Die genaue Ausprägung erfolgt in der jeweiligen Interfacebeschreibung des Web-Services für das Fachmodul.

**A\_15607 - SOAP für Web-Services der Fachmodule**

Der Konnektor MUSS für die an der Clientsystemschnittstelle definierten Web-Services der Fachmodule [SOAP1.1] und [SOAP1.2] unterstützen. Die SOAP-Version pro Web-Service Endpunkt wird durch die WSDL des jeweiligen Web-Service definiert. [≤]

**A\_15608 - Web-Services der Fachmodule konform zu [BasicProfile1.2]**

Der Konnektor MUSS für die an der Clientsystemschnittstelle definierten Web-Services der Fachmodule bei [SOAP1.1] die Profilierung konform zu [BasicProfile1.2] anbieten. [≤]

**A\_15609 - Web-Services der Fachmodule konform zu [BasicProfile2.0]**

Der Konnektor MUSS für die an der Clientsystemschnittstelle definierten Web-Services der Fachmodule bei [SOAP1.2] die Profilierung konform zu [BasicProfile2.0] anbieten. [≤]

**3.5.2 Statusrückmeldung und Fehlerbehandlung**

Der Konnektor bietet Operationen an der Außenschnittstelle über SOAP-Webservices an. Treten bei der Ausführung einer Operation Fehler auf, so werden diese an das aufrufende System gemeldet. Die von den Basisdiensten des Konnektors angebotenen SOAP-Webservices melden Fehler, die bei der Ausführung einer Operation auftreten, über eine SOAP-Fault-Nachricht (siehe auch [gemSpec\_OM#3.2.3]).

**TIP1-A\_5058 - Fehlerübermittlung durch gematik-SOAP-Fault**

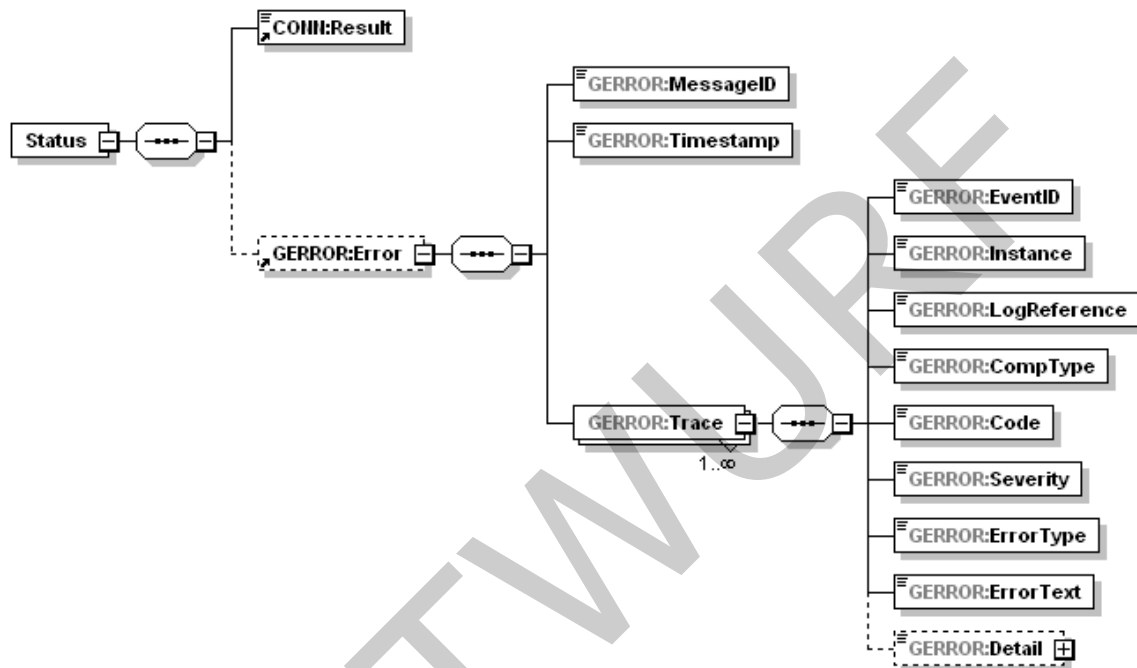
Der Konnektor MUSS Fehlermeldungen, die im Rahmen einer über die Außenschnittstelle aufgerufenen Operation auftreten, an das Clientsystem mittels gematik-SOAP-Faults melden. [≤]

**TIP1-A\_5058-01 - ab PTV4: Fehlerübermittlung durch gematik-SOAP-Fault**

Der Konnektor MUSS Fehlermeldungen, die im Rahmen einer über die Außenschnittstelle aufgerufenen Operation eines Basisdienst-SOAP-Webservices auftreten, an das Clientsystem mittels gematik-SOAP-Faults melden. [≤]

Treten bei konnektorinternen Operationen (TUCs) Fehler auf, so werden diese an den Aufrufer (aufrufender TUC oder aufrufende Operation) zurückgegeben. Der Aufrufer kann den aufgetretenen Fehler in seinem Kontext neu interpretieren. Das bedeutet insbesondere, dass ein Error eines aufgerufenen TUCs nicht zwingend zum Abbruch des aufrufenden TUCs bzw. der aufrufenden Operation führen muss. So ist es dem Aufrufer

- 1932 möglich, einen Error als Warnung zu interpretieren und an den eigenen internen oder  
1933 externen Aufrufer zurückzumelden. Diese dabei erzeugte Fehlerkette wird in Form einer  
1934 Fehler-Trace-Struktur abgebildet, um eine Nachverfolgung von Fehlern zu ermöglichen.
- 1935 Operationen an der Außenschnittstelle können die Fehlerkette zu Informationszwecken in  
1936 der SOAP-Antwort an das Clientsystem senden. Dazu enthält jede SOAP-Antwort das  
1937 Element Status, dass gemäß dem XML-Schema [ConnectorCommon.xsd] aufgebaut ist  
1938 (siehe auch Abbildung PIC\_KON\_107 XML-Struktur des Status-Elements einer SOAP-  
1939 Antwort).
- 1940



- 1941  
1942

1943 **Abbildung 3: PIC\_KON\_107 XML-Struktur des Status-Elements einer SOAP-Antwort**

- 1944 Schlägt eine Operation fehl, so wird eine SOAP-Fault-Meldung an das Clientsystem  
1945 versendet. Im Erfolgsfall wird das Status-Element in die Antwortnachricht an das  
1946 Clientsystem aufgenommen. Ist der Fehler-Trace leer (Element GERROR:Error ist nicht  
1947 vorhanden), so wird CONN:Result auf OK gesetzt. Andernfalls, d. h. wenn in  
1948 GERROR:Trace Fehler der Schwere Info oder Warning (zu Informationszwecken)  
1949 enthalten sind, wird CONN:Result auf Warning gesetzt.

#### 1950 **TIP1-A\_4521 - Protokollierung von Fehlern inkl. Trace-Struktur**

- 1951 Der Konnektor MUSS Fehler protokollieren, die in fachlichen und technischen Abläufen  
1952 von der gematik spezifiziert oder herstellerspezifisch definiert sind und den Schweregrad  
1953 (Severity) Warning, Error oder Fatal haben. Zur Nachvollziehbarkeit des Fehlers MÜSSEN  
1954 Fehlerursache, fachliche und technische Auslöser des Fehlverhaltens aus den  
1955 Protokolleinträgen erkennbar sein.  
1956 [ $\leq$ ]

#### 1957 **A\_14159 - Rückgabe von Fehlermeldungen an der Außenschnittstelle**

- 1958 Der Konnektor MUSS bei der Rückgabe von Fehlermeldungen an der Außenschnittstelle  
1959 sicherstellen, dass im letzten "GERROR:Trace"-Element der GERROR-Struktur ein von der  
1960 gematik spezifizierter Fehler steht. Die GERROR-Struktur kann weitere gematik- und  
1961 herstellerspezifische Fehler enthalten.  
1962 [ $\leq$ ]

- 1963 In der Regel ist es ausreichend, wenn die GERROR-Struktur an der Außenschnittstelle nur  
 1964 ein Element „GERROR:Trace“ mit einem gematik-Fehler enthält.
- 1965 Wenn für eine Fehlersituation kein Fehlercode spezifiziert ist, kann ein  
 1966 herstellerspezifischer Fehler zur Detaillierung verwendet werden. In diesem Fall muss ein  
 1967 passender gematik-Fehler als letztes GERROR:Trace-Element gewählt werden. Bei  
 1968 Fehlern in technischen Abläufen kann Fehlercode 4001 als letztes GERROR:Trace-Element  
 1969 verwendet werden. Die Wahl des letzten GERROR:Trace-Elements ist mit der gematik  
 1970 abzustimmen.
- 1971 Zur Struktur von Fehlermeldungen siehe auch [gemSpec\_OM#GS-A\_3856].

### 1972 3.5.3 Transport großer Dokumente

- 1973 SOAP Message Transmission Optimization Mechanism (MTOM) ermöglicht den direkten  
 1974 Transport von binären Daten in Webservices, d.h. ohne dass eine zur Laufzeit aufwändige  
 1975 Verpackung der binären Daten in ein Base64-XML-Element notwendig wird. Auf die  
 1976 Definition der Webservices und ihre Funktionalität hat dieser Optimierungsmechanismus  
 1977 keinen Einfluss. Der Einsatz von MTOM dient der Verbesserung des Performance-  
 1978 Verhaltens für große Dokumente.
- 1979 Das Clientsystem kann die Optimierung des Transports großer Dokumente per SOAP  
 1980 Message Transmission Optimization Mechanism (MTOM) anstoßen. In den WSDL-Dateien  
 1981 werden keine MTOM Serialization Policy Assertion [WS-MTOMPolicy] eingebettet.
- 1982 **TIP1-A\_5694 - SOAP Message Transmission Optimization Mechanism für**  
 1983 **Basisdienste**
- 1984 Der Konnektor KANN SOAP Message Transmission Optimization Mechanism (MTOM)  
 1985 gemäß [MTOM] unterstützen.
- 1986 Wenn der Konnektor MTOM unterstützt, MUSS er MTOM für Signatur- und  
 1987 Verschlüsselungsdienst unterstützen, DARF aber NICHT MTOM für andere Dienste  
 1988 unterstützen.
- 1989 Wenn der Konnektor MTOM unterstützt, MUSS er, vergleichbar dem Einsatz des Attributs  
 1990 `wsp:Optional="true"` einer MTOM Serialization Policy Assertion [WS-MTOMPolicy], genau  
 1991 dann MTOM für die Antwortnachricht verwenden, wenn entweder
- 1992 • die Aufrufnachricht eine `application/xop+xml` Nachricht ist
  - 1993 • oder der `Accept` HTTP header der Aufrufnachricht folgenden Wert hat:  
 1994 `multipart/related; type=application/xop+xml`
- 1995 [`<=`]
- 1996 **TIP1-A\_5694-02 - ab PTV4: SOAP Message Transmission Optimization**  
 1997 **Mechanism für Basisdienste**
- 1998 Der Konnektor KANN SOAP Message Transmission Optimization Mechanism (MTOM)  
 1999 gemäß [MTOM-SOAP1.1] für Basisdienste unterstützen.[`<=`]
- 2000 **A\_15786 - SOAP Message Transmission Optimization Mechanism für**  
 2001 **Basisdienste - Einschränkung**
- 2002 Wenn der Konnektor MTOM für Basisdienste unterstützt, MUSS er MTOM für Signatur-  
 2003 und Verschlüsselungsdienst unterstützen, DARF aber NICHT MTOM für andere Dienste  
 2004 unterstützen.[`<=`]
- 2005 **A\_15610 - Verwendung von MTOM für Antwortnachricht**
- 2006 Wenn der Konnektor MTOM unterstützt, MUSS er, vergleichbar dem Einsatz des Attributs  
 2007 `wsp:Optional="true"` einer MTOM Serialization Policy Assertion [WS-MTOMPolicy], genau  
 2008 dann MTOM für die Antwortnachricht verwenden, wenn entweder

- 2009 • die Aufrufnachricht eine application/xop+xml Nachricht ist
- 2010 • oder der Accept HTTP header der Aufrufnachricht folgenden Wert hat:
- 2011 multipart/related; type=application/xop+xml.

2012 [**<=**]

### 2013 **A\_15611 - SOAP Message Transmission Optimization Mechanism für**

### 2014 **Fachmodule**

2015 Der Konnektor MUSS SOAP Message Transmission Optimization Mechanism (MTOM)

2016 gemäß [MTOM] für Fachmodule unterstützen, wenn es in der Schnittstellenbeschreibung

2017 des Fachmodules explizit gefordert wird. [**<=**]

## 2018 **3.6 Verwendung manuell importierter CA-Zertifikate**

2019 TI-fremde X.509-Zertifikate werden im Rahmen des Verschlüsselungsdienstes verwendet.

2020 Um den Vertrauensraum für diese Zertifikate abzubilden, erlaubt der Konnektor, X.509-

2021 CA-Zertifikate zu diesen TI-fremden X.509-Zertifikaten in eine interne Liste

2022 (CERT\_IMPORTED\_CA\_LIST) zu importieren.

2023 Der Konnektor kann dann im Rahmen der Hybridverschlüsselung den symmetrischen

2024 Schlüssel empfangerspezifisch mit diesem TI-fremden X.509-Zertifikat verschlüsseln. Die

2025 TI-fremden Zertifikate dürfen nicht zu einem anderen Zweck als diesem eingesetzt

2026 werden.

### 2027 **TIP1-A\_5433 - Manuell importierte X.509-CA-Zertifikate nur für hybride**

### 2028 **Verschlüsselung**

2029 Der Konnektor DARF End-Entity-Zertifikate, die lediglich gegen manuell importierte

2030 X.509-CA-Zertifikate geprüft werden, die von CAs außerhalb der TI stammen

2031 (CERT\_IMPORTED\_CA\_LIST), NICHT für andere Zwecke als zur hybriden Verschlüsselung

2032 von Dokumenten verwenden.

2033 [**<=**]

2034 Die Berücksichtigung der CA-Zertifikate aus CERT\_IMPORTED\_CA\_LIST muss auf

2035 folgende Anwendungsfälle beschränkt werden:

- 2036 1. Prüfung eines Zertifikates im Rahmen der hybriden Verschlüsselung
- 2037 2. Prüfung eines Zertifikates im Rahmen eines Aufrufes der Operation "VerifyCertificate"

2038

### 2039 **TIP1-A\_5660 - Hinweise im Handbuch für manuell importierte X.509-CA-**

### 2040 **Zertifikate**

2041 Das Handbuch des Konnektors MUSS sinngemäß folgende Hinweise enthalten:

- 2042 • Der Administrator übernimmt die Verantwortung für die Verlässlichkeit der
- 2043 importierten CA-Zertifikate.
- 2044 • Der Administrator kann sich bei seiner Entscheidung für einen Import von CA-
- 2045 Zertifikaten auf die Informationen der gematik stützen.
- 2046 • Die gematik veröffentlicht dazu Informationen über CA-Betreiber, welche die
- 2047 Erfüllung der Sicherheitsanforderungen der gematik nachgewiesen haben.

2048 [**<=**]

### 2049 3.7 Testunterstützung

2050 Gemäß Testkonzept Online-Rollout (Stufe 1) [gemKPT\_Test\_ORIS1#TIP1-A\_2839] muss  
 2051 ein Hersteller eines Konnektors seine Modelle in drei Ausführungen vorsehen: Eine für die  
 2052 Testumgebung, eine für die Referenzumgebung und eine für die Produktivumgebung.

2053 Damit trotz dieser Forderung die Firmware je Konnektorversion für alle Umgebungen  
 2054 identisch ist, wird die Erkennung der Umgebung an die gSMC-K gebunden. Die  
 2055 Konnektor-Firmware muss zwischen den Umgebungen PU und RU/TU unterscheiden. Die  
 2056 gSMC-K besitzt hierzu den Datencontainer MF/EF.EnvironmentSettings, der die jeweilige  
 2057 Umgebungskennung enthält (PU bzw. TU/RU). Die Umgebungskennung wird read-only  
 2058 auf der gSMC-K gespeichert.

#### 2059 **TIP1-A\_4981 - Steuerung der Betriebsumgebung via gSMC-K**

2060 Der Produkttyp Konnektor MUSS sowohl in der Testumgebung (TU), der  
 2061 Referenzumgebung (RU) wie auch der Produktivumgebung (PU) betreibbar sein.  
 2062 Die Information, ob eine Konnektorinstanz in der TU/RU oder PU betrieben wird, MUSS  
 2063 der Konnektor dem File MF/EF.EnvironmentSettings der gSMC-K entnehmen.  
 2064 Abhängig von der ermittelten Betriebsumgebung MÜSSEN die Konfigurationswerte gemäß  
 2065 Tabelle TAB\_KON\_812 verwendet werden.  
 2066

2067 **Tabelle 10: TAB\_KON\_812 Umgebungsabhängige Konfigurationsparameter**

| Betriebsumgebung | Konfigurationsparameter | Konfigurationswert                           | Beschreibung   |
|------------------|-------------------------|--|--|
| PU               | NET_TI_ZENTRAL          | siehe [gemSpec_Net#Tab_Adrkonzept_Produktiv] | Siehe TAB_KON_680. Dieser Wert MUSS für den Administrator über die Managementschnittstelle einsehbar, DARF aber NICHT änderbar sein. |
|                  | NET_TI_GESICHERTE_FD    | siehe [gemSpec_Net#Tab_Adrkonzept_Produktiv] | Siehe TAB_KON_680. Dieser Wert MUSS für den Administrator über die Managementschnittstelle einsehbar, DARF aber NICHT änderbar sein. |
|                  | NET_TI_OFFENE_FD        | siehe [gemSpec_Net#Tab_Adrkonzept_Produktiv] | Siehe TAB_KON_680. Dieser Wert MUSS für den Administrator über die Managementschnittstelle einsehbar, DARF aber NICHT änderbar sein. |
|                  | DNS_TOP_LEVEL_DOMAIN_TI | telematik.                                   | Siehe TAB_KON_731. Dieser Wert MUSS für den Administrator über die Managementschnittstelle einsehbar, DARF aber NICHT änderbar sein. |



|       |                         |   |  |
|-------|-------------------------|---|--|
| RU/TU | NET_TI_ZENTRAL          | siehe [gemSpec_Net#<br>Tab_Adrkonzept_Test] | Siehe TAB_KON_680.<br>Dieser Wert MUSS für den Administrator über die Managementschnittstelle mit dem Konfigurationswert voreingestellt und änderbar sein. |
|       | NET_TI_GESICHERTE_FD    | siehe [gemSpec_Net#<br>Tab_Adrkonzept_Test] | Siehe TAB_KON_680.<br>Dieser Wert MUSS für den Administrator über die Managementschnittstelle mit dem Konfigurationswert voreingestellt und änderbar sein. |
|       | NET_TI_OFFENE_FD        | siehe [gemSpec_Net#<br>Tab_Adrkonzept_Test] | Siehe TAB_KON_680.<br>Dieser Wert MUSS für den Administrator über die Managementschnittstelle mit dem Konfigurationswert voreingestellt und änderbar sein. |
|       | DNS_TOP_LEVEL_DOMAIN_TI | telematik-test.                             | Siehe TAB_KON_731.<br>Dieser Wert MUSS für den Administrator über die Managementschnittstelle einsehbar, aber nicht änderbar sein.                         |

2068  
2069  
2070

[<=]

#### 2071 **TIP1-A\_4707 - Betrieb in Test- und Referenzumgebung**

2072 Der Produkttyp Konnektor MUSS auch in der Test- und Referenzumgebung betrieben  
2073 werden können. Dafür MUSS der Vertrauensanker des Konnektors für diese Umgebung  
2074 ausgetauscht werden können. Dies KANN durch Lieferung eines neuen Konnektors oder  
2075 durch Austausch der gSMC-K durch den Hersteller ermöglicht werden. Der Hersteller  
2076 MUSS sicherstellen, dass Konnektoren ausschließlich mit den zu ihrer Einsatzumgebung  
2077 gehörenden Vertrauensankern ausgestattet werden.

2078 [≤]

#### 2079 **TIP1-A\_4982 - Anzeige von TU/RU in der Managementschnittstelle**

2080 Die Administrationsoberfläche MUSS, wenn der Konnektor in der Testumgebung (TU)  
2081 oder Referenzumgebung (RU) betrieben wird, die Umgebungsbezeichnung zu jeder Zeit  
2082 erkennbar in der Managementschnittstelle anzeigen.

2083 Die Anzeige eines Betriebs in der Produktivumgebung DARF NICHT explizit angezeigt  
2084 werden.

2085 [≤]



2086

## 4 Funktionsmerkmale

2087 Für die folgenden Inhalte bitte die Hinweise in Kapitel 1.5.3 „Erläuterungen zur  
2088 Spezifikation des Außenverhaltens,“ sowie Kapitel 1.5.4 Erläuterungen zur  
2089 Dokumentenstruktur und „Dokumentenmechanismen“ beachten.

### 2090 4.1 Anwendungskonnektor

#### 2091 4.1.1 Zugriffsberechtigungsdienst

2092 Der Zugriffsberechtigungsdienst ist ein interner Dienst. Er ermöglicht es Operationen eine  
2093 Prüfung auf Zugriffsberechtigung für die von ihnen benötigten Ressourcen  
2094 durchzuführen. Die Prüfung erfolgt direkt nach Aufruf einer Operation des Konnektors  
2095 durch das Clientsystem und basiert auf den im Clientaufruf enthaltenen Parametern.

2096 Der Zugriffsberechtigungsdienst definiert über ein Informationsmodell die erlaubten  
2097 Zugriffsmöglichkeiten. Um dies zu erreichen, modelliert es Mandanten und ordnet ihnen  
2098 Clientsysteme sowie die vom Konnektor verwalteten externen Ressourcen  
2099 (Kartenterminal mit Slots, Arbeitsplatz mit Signaturproxy und SMC-Bs) zu. Diese durch  
2100 einen Administrator persistent zu modellierenden Entitäten und Beziehungen beinhalten  
2101 die erlaubten Zugriffswege vom Clientsystem über Arbeitsplatz zum Kartenterminal und  
2102 dessen Slots. Sie werden im Konnektor administrativ konfiguriert. Der Signaturproxy hat  
2103 keine eigene Identität im Informationsmodell, da er den Kontext des aufrufenden  
2104 Clientsystems verwendet.

2105 Neben diesen persistenten Entitäten und Beziehungen bildet das Modell auch die in den  
2106 Slots temporär gesteckten Karten und die zugehörigen Kartensitzungen als transiente  
2107 Entitäten und Beziehungen ab.

2108 Abbildung PIC\_Kon\_100 stellt das Informationsmodell dar. Die persistenten Entitäten  
2109 haben einen grünen Hintergrund, die transienten einen weißen.

2110 Tabelle TAB\_KON\_507 beschreibt die Entitäten und legt ihren Identitätsschlüssel fest.  
2111 Tabelle TAB\_KON\_508 beschreibt die Attribute. Tabelle TAB\_KON\_509 beschreibt die  
2112 Entitätsbeziehungen und referenziert dabei die in Abbildung PIC\_Kon\_100 durch Zahlen  
2113 in eckigen Klammern markierten Beziehungen. Tabelle TAB\_KON\_510 definiert  
2114 Constraints, die zusätzlich zu den in Abbildung PIC\_Kon\_100 definierten Kardinalitäten  
2115 gelten. Die Constraints werden mittels Object Constraint Language (OCL) definiert.

##### 2116 4.1.1.1 Funktionsmerkmalweite Aspekte

###### 2117 TIP1-A\_4522 - Zugriffsberechtigungs-Informationsmodell des Konnektors

2118 Der Konnektor MUSS die Entitäten, Attribute und Beziehungen des Informationsmodells  
2119 intern vorhalten, dabei für die Einhaltung der definierten Constraints sorgen und die  
2120 persistenten Entitäten und Beziehungen dauerhaft speichern. Der Konnektor MUSS dabei  
2121 eine Mindestanzahl von 999 Mandanten unterstützen.

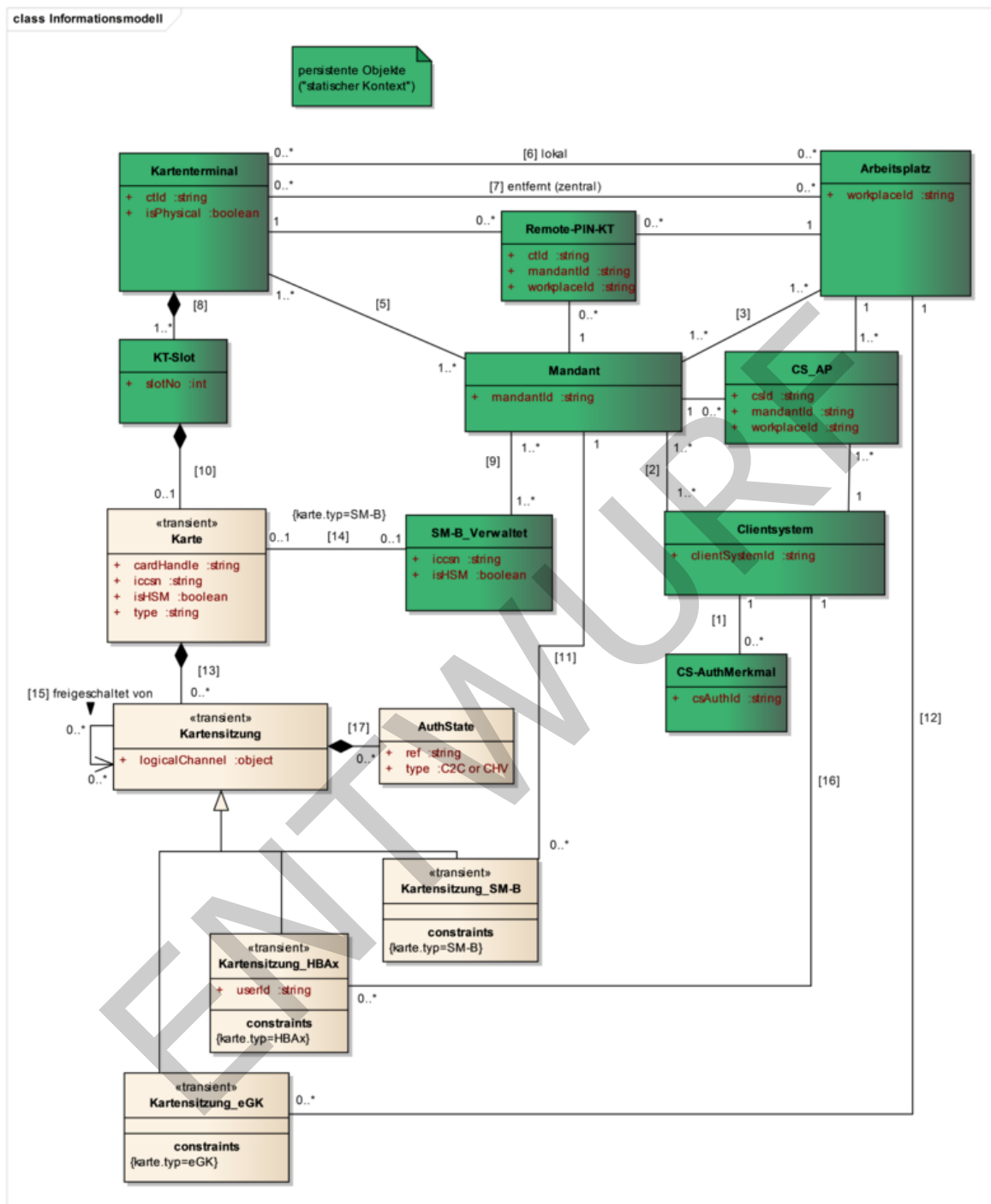
2122 Das Informationsmodell ist definiert durch das UML-Diagramm „PIC\_Kon\_100  
2123 Informationsmodell des Konnektors,“ und die Tabelle „TAB\_KON\_510 Informationsmodell  
2124 Constraints“. Der Konnektor darf nur Daten in sein Informationsmodell übernehmen, die  
2125 alle Eigenschaften des Informationsmodells, insbesondere die Constraints, erfüllen.  
2126 Die Entitäten werden in Tabelle „TAB\_KON\_507 Informationsmodell Entitäten“  
2127 beschrieben, die Attribute in Tabelle „TAB\_KON\_508 Informationsmodell Attribute“ und

2128 die Beziehungen in Tabelle „TAB\_KON\_509 Informationsmodell Entitätenbeziehungen“.  
2129 [**<=**]

2130 *Hinweis zu den Bezeichnern der Entitäten und ihrer Attribute: Im Folgenden beginnen*  
2131 *Entitäten mit einem Großbuchstaben, Attribute mit einem Kleinbuchstaben. Werden die*  
2132 *Entitäten und Attribute in XML-Dokumenten verwendet, so beginnen die zugeordneten*  
2133 *XML-Elementbezeichner grundsätzlich mit einem Großbuchstaben und verwenden den*  
2134 *englischen Begriff, der im Folgenden in Klammern angegeben ist, wenn zur besseren*  
2135 *Lesbarkeit im Modell ein deutscher Begriff verwendet wird.*

ENTWURF

2136



2137

2138

2139

Abbildung 4: PIC\_Kon\_100 Informationsmodell des Konnektors

2140

2141

2142

Tabelle 11: TAB\_KON\_507 Informationsmodell Entitäten

| Entität | persistent/<br>transient | Identitätsschlüssel | Beschreibung |
|---------|--------------------------|---------------------|--------------|
|---------|--------------------------|---------------------|--------------|

|                                  |            |                       |  |
|----------------------------------|------------|-----------------------|--|
| Mandant                          | persistent | mandantId             | Zu Mandanten und Mandantenfähigkeit siehe Kapitel Mandantenfähigkeit.  |
| Clientsystem                     | persistent | clientSystemId        | Unter einem Clientsystem wird hier ein einzelnes oder eine Gruppe von Systemen verstanden, welche im LAN der Einsatzumgebung auf die Clientsystem-Schnittstelle des Konnektors zugreifen.  |
| CS-AuthMerkmal (CS-AuthProperty) | persistent | csAuthId              | Das Authentifizierungsmerkmal dient der Authentifizierung, wenn sich das Clientsystem gegenüber dem Konnektor authentisiert. Der Identitätsschlüssel csAuthId wird bei der Administration vergeben   |
| Arbeitsplatz (Workplace)         | persistent | workplaceId           | alle dem Konnektor bekannten Arbeitsplätze   |
| Kartenterminal (CardTerminal)    | persistent | ctId                  | alle dem Konnektor bekannten Kartenterminals.  |
| KT-Slot (CT-Slot)                | persistent | ctId, slotNo          | Die sich in den Kartenterminals befindenden Chipkartenslots (Functional Unit Type 00)  |
| Karte (Card)                     | transient  | cardHandle oder iccsn | Die in den Kartenterminals steckenden Smartcards des Gesundheitswesens, die persönliche Identitäten oder Rollen repräsentieren (eGK, HBA, SMC-B). Karten, die nur Geräteidentitäten tragen (gSMC-K, gSMC-KT) werden in diesem Modell nicht betrachtet. Karten im Sinne dieses Informationsmodells existieren maximal so lange, wie sie im Kartenterminal stecken. Die aktuell im System steckenden Karten werden |

|                                       |           |                                |   |
|---------------------------------------|-----------|--------------------------------|---|
|                                       |           |                                | <p>vom Clientsystem über das cardHandle adressiert. Die iccsn erlaubt eine dauerhafte Adressierung einer Karte.</p> <p>Für den Kartentyp „SM-B“ kann hier auch eine in einem HSM-B enthaltene virtuelle SMC-B abgebildet werden.</p>  |
| Kartensitzung (CardSession)           | transient | siehe konkrete Kartensitzungen | <p>Kartensitzungen stellen ein wesentliches Konzept im Sicherheitsmodell des Konnektors dar. Eine Kartensitzung verwaltet einen aktuellen logischen Sicherheitsstatus einer Karte. Die Kartensitzungen sind einer Karte fest zugewiesen.</p> <p>Zu einer Karte kann es mehrere Kartensitzungen geben, die voneinander logisch unabhängige Sicherheitsstatus einer Karte verwalten.</p> <p>Der Konnektor führt alle Zugriffe auf eine Karte im Kontext einer Kartensitzung zu dieser Karte aus.</p> <p>Das Attribut logischerKanal bezeichnet den logischen Kanal zur Karte, der im Rahmen der Kartensitzung verwendet wird (gemäß Standard [7816-4]).</p> |
| Kartensitzung_eGK (CardSession_eGK)   | transient | cardHandle                     | <p>Kartensitzung für eine eGK. Die KVK ist im Modell nicht explizit dargestellt. Soweit anwendbar, gelten für die KVK die gleichen Aussagen wie für die eGK.</p>  |
| Kartensitzung_SM-B (CardSession_SM-B) | transient | cardHandle, mandantId          | <p>Kartensitzung für eine SM-B</p>  |

|  |            |  |  |
|--|------------|--|--|
| Kartensitzung_HBAx<br>(CardSession_HBAx) | transient  | cardHandle,<br>clientSystemId,<br>userId             | Kartensitzung für einen HBAx.<br>Unter dem Typ „HBAx“ sind auch die Vorläuferkarten wie „HBA-qSig“ und „ZOD_2.0“ inkludiert.   |
| SM-B_Verwaltet<br>(SM-B_managed)         | persistent | iccsn  | SM-Bs müssen im Gegensatz zu den übrigen Karten im Konnektor vor ihrer Verwendung persistent im Informationsmodell als „SM-B_Verwaltet“ per Administration aufgenommen werden. Dies gilt auch für die in einem HSM-B enthaltenen virtuellen SMC-Bs.  |
| CS_AP                                    | persistent | mandantId,<br>clientSystemId,<br>workplaceId         | CS_AP legt die von einem Clientsystem pro Mandanten nutzbaren Arbeitsplätze fest. Ein Clientsystem kann dabei mehrere Arbeitsplätze bedienen. Ebenso können Arbeitsplätze von mehreren Clientsystemen, auch gleichzeitig, genutzt werden, z. B. bei zwei unterschiedlichen, voneinander unabhängigen Praxisprogrammen. |
| Remote-PIN-KT                            | persistent | mandantId,<br>workplaceId, ctId                      | Remote-PIN-KT legt pro Mandant und Arbeitsplatz fest, über welches Kartenterminal eine Remote PIN-Eingabe erfolgen soll, wenn an diesem Arbeitsplatz die PIN-Eingabe für eine Karte erforderlich ist, die nicht in einem dem Arbeitsplatz lokal zugeordneten Kartenterminal steckt.                                    |
| AuthState                                | transient  | cardHandle,<br>(clientSystemId),<br>(userId),<br>ref | Zu einer Kartensitzung gibt es höhere AuthorizationStates, die durch (type =C2C) Freischaltung oder durch PIN-Eingabe (type=CHV) erreicht werden können.   |

|  |  |  |  |
|--|--|--|--|
|  |  |  | Für eGK G2.0 wird der Zustand der MRPINs nicht in AuthState gespeichert. |
|--|--|--|--|

2143

2144

2145

**Tabelle 12: TAB\_KON\_508 Informationsmodell Attribute**

| Attribut       | Beschreibung   |
|----------------|--|
| cardHandle     | Das Identifikationsmerkmal einer Karte für die Dauer eines Steckzyklusses. Es wird mit dem Entfernen der Karte aus dem Kartenterminal ungültig. Es wird automatisch vom Konnektor vergeben.  |
| clientSystemId | Das Identifikationsmerkmal eines Clientsystems. Es wird per Administration dem Mandanten im Clientsystem und im Konnektor zugeordnet.  |
| csAuthId       | Das Identifikationsmerkmal eines Authentifizierungsmerkmals.   |
| ctId           | Das Identifikationsmerkmal eines Terminals. Es ist eine fixe Eigenschaft des Kartenterminals.  |
| iccsn          | Die Seriennummer einer Karte. Sie identifiziert eine Karte dauerhaft.  |
| isHSM          | Attribut der Entitäten Karte und SM-B_Verwaltet. Es ist false, wenn eine echte Smardcard abgebildet wird und true, wenn es sich um eine virtuelle SMC-B handelt, die in einem HSM-B enthalten ist.                                   |
| isPhysical     | Attribut des Kartenterminals das den Wert „Ja“ hat, wenn es sich um ein tatsächlich existierendes Kartenterminal handelt. Ist der Wert „Nein“, dann handelt es sich um ein logisches Kartenterminal im Zusammenhang mit einem HSM-B. |
| logicalChannel | Referenz auf ein Objekt, das einen logischen Kanal repräsentiert.  |
| mandantId      | Das Identifikationsmerkmal eines Mandanten. Es wird per Administration dem Mandanten im Clientsystem und im Konnektor zugeordnet.  |
| ref            | Das Identifikationsmerkmal eines AuthState zu einer gegebenen Kartensitzung. Im Falle C2C handelt es sich um die   |



|             |   |
|-------------|---|
|             | KeyRef (mit einer bestimmten Rolle) und in Falle CHV um eine referenzierte PIN.   |
| slotNo      | Das Identifikationsmerkmal eines Slot für ein bestimmtes Kartenterminal. Diese fortlaufende Nummer ist eine fixe Eigenschaft des Kartenterminals. Sie beginnt bei 1.  |
| type        | Als Kartenattribut: Typ einer Karte. Im Folgenden berücksichtigte Werte: „HBAX“, „SM-B“, „EGK“.<br>Als Attribute eines AuthState: Typ des AuthState. „C2C“ steht für gegenseitige Kartenauthentisierung. „CHV“ steht für Card Holder Verification per PIN-Eingabe.                        |
| userId      | Das Identifikationsmerkmal des Nutzers im Clientsystem (Die userId wird durch das Clientsystem vergeben und verwaltet). Die userId wird im Kontext eine Kartensitzung_HBAX vom Konnektor verwendet, um als Bestandteil des Identitätsschlüssels die Kartensitzung_HBAX zu identifizieren. |
| workplaceId | Das Identifikationsmerkmal eines Arbeitsplatzes. Es wird per Administration dem Mandanten im Clientsystem und im Konnektor zugeordnet.  |

2146

2147

**Tabelle 13: TAB\_KON\_509 Informationsmodell Entitätenbeziehungen**

| Entitätenbeziehung                              | persistent/<br>transient | Beschreibung  |
|---|--------------------------|---|
| Authentifikationsmerkmale des Clientsystems [1] | persistent               | Diese Relation legt für jedes Clientsystem eine Menge von Authentisierungsmerkmalen fest. Mit einem dieser Authentisierungsmerkmale muss sich ein Client gegenüber dem Konnektor authentisiert haben, um als das entsprechende Clientsystem vom Konnektor akzeptiert zu werden. |
| Clientsysteme des Mandanten [2]                 | persistent               | Diese Relation weist Clientsystemen Mandanten zu.   |
| Arbeitsplätze des Mandanten [3]                 | persistent               | Diese Relation weist Arbeitsplätze Mandanten zu.<br>Arbeitsplätze können von mehreren Mandanten genutzt werden. Z. B. kann ein von mehreren Mandanten genutzter gemeinsamer Empfang als ein Arbeitsplatz modelliert werden.   |
| Kartenterminals des Mandanten [5]               | persistent               | Diese Relation weist Kartenterminals Mandanten zu.  |

|   |            |  |
|---|------------|--|
| Lokale Kartenterminals [6]                          | persistent | Diese Relation erfasst die Kartenterminals, die sich lokal an einem Arbeitsplatz befinden und von diesem genutzt werden können. Die Modellierung lässt es zu, dass Kartenterminals mehreren Arbeitsplätzen lokal zugewiesen werden. Jeder an der TI teilnehmende Arbeitsplatz wird in der Regel mindestens ein lokales Kartenterminal benötigen. |
| Entfernte Kartenterminals [7]                       | persistent | Diese Relation beschreibt, auf welche Kartenterminals Arbeitsplätze (remote) zugreifen dürfen. Dies ist für zentral steckende Karten vorgesehen.   |
| Slot eines Kartenterminals [8]                      | persistent | Die Zuordnung von Slots zu einem Kartenterminal ergibt sich automatisch aus den Eigenschaften des Kartenterminals.   |
| SM-B_Verwaltet eines Mandanten [9]                  | persistent | Diese Relation legt fest, welche verwalteten SM-Bs einem Mandanten zugeordnet sind.  |
| Kartenterminal-Slot, in dem eine Karte steckt [10]  | transient  | Sobald eine Karte in ein Kartenterminal gesteckt wird, ergibt sich implizit eine Relation der Karte zu dem Slot, in dem sie steckt, [6] und indirekt über [4] zum Kartenterminal.  |
| Mandant der Kartensitzung SM-B [11]                 | transient  | Beim Anlegen einer Kartensitzung SM-B wird diese immer dem zugreifenden Mandanten zugeordnet.  |
| Arbeitsplatz der Kartensitzung eGK [12]             | transient  | Eine Kartensitzung eGK ist immer einem Arbeitsplatz zugeordnet.  |
| Karte einer Kartensitzung [13]                      | transient  | Jeder Kartensitzung ist genau einer Karte zugeordnet.  |
| Gesteckte SM-B [14]                                 | transient  | Wird eine SM-B gesteckt und handelt es sich um eine verwaltete SM-B, ergibt sich über die iccsn die Zuordnung.   |
| Freischaltung einer Karte [15]                      | transient  | Diese Relation erfasst die Freischaltung einer Karte durch eine andere Karte.  |
| Bindung der Kartensitzung_HBAx an Clientsystem [16] | transient  | Kartensitzungen HBAx sind einem Clientsystem zugeordnet.   |
| AuthState pro Kartensitzung [17]                    | transient  | Eine Kartensitzung kann erhöhte Sicherheitszustände (Authorization State) haben.   |

2149 **Tabelle 14: TAB\_KON\_510 Informationsmodell Constraints**

| #  | Beschreibung  | Definition mittels OCL<br>(Die Constraints werden im UML ergänzenden Standard OCL definiert.)   |
|----|---|---|
| C1 | Eine eGK muss eine oder keine Kartensitzung haben.  | <b>context</b> Karte<br><b>inv:</b> self.type = "eGK" implies<br>self.kartensitzung.size() <= 1   |
| C2 | Wenn zwei Kartensitzungen einer HBAX dem gleichen Clientsystem zugeordnet sind und ihre userIds gleich sind, dann müssen die beiden Kartensitzungen identisch sein. | <b>context</b> Kartensitzung-HBAX<br><b>inv:</b> forAll(k1, k2 : Kartensitzung-HBAX  <br>k1.karte = k2.karte<br>and k1.clientsystem = k2.clientsystem<br>and k1.userId = k2.userId<br>implies<br>k1 = k2) |
| C3 | Wenn zwei SM-B-Kartensitzungen einer Karte dem gleichen Mandanten zugeordnet sind, dann müssen die beiden Kartensitzungen identisch sein.                           | <b>context</b> Kartensitzung-SM-B<br><b>inv:</b> forAll(k1, k2 : Kartensitzung-SM-B  <br>k1.karte = k2.karte<br>and k1.mandant = k2.mandant implies<br>k1 = k2)   |
| C4 | Die Seriennummer iccsn einer Karte muss eindeutig sein.   | <b>context</b> Karte<br><b>inv:</b> Karte.allInstances -><br>isUnique(iccsn)  |
| C5 | Die Seriennummer iccsn einer Karte muss für die vom Konnektor verwalteten SM-Bs eindeutig sein.   | <b>context</b> SM-B_Verwaltet<br><b>inv:</b> SM-B_Verwaltet.allInstances -><br>isUnique(iccsn)  |
| C6 | Das CardHandle einer Karte muss eindeutig sein.   | <b>context</b> Karte<br><b>inv:</b> Karte.allInstances -><br>isUnique(cardHandle)   |
| C7 | Die Identifikationsnummer des Clientsystems muss eindeutig sein.  | <b>context</b> Clientsystem<br><b>inv:</b> Clientsystem.allInstances -><br>isUnique(clientSystemId)   |

|     |  |  |
|-----|--|--|
| C8  | Die Identifikationsnummer des Mandanten muss eindeutig sein.   | <b>context</b> Mandant<br><b>inv:</b> Mandant.allInstances -> isUnique (mandantId)   |
| C9  | Die Identifikationsnummer des Arbeitsplatzes muss eindeutig sein.  | <b>context</b> Arbeitsplatz<br><b>inv:</b> Arbeitsplatz.allInstances -> isUnique (workplaceId)   |
| C10 | Die Identifikationsnummer des Kartenterminals muss eindeutig sein.   | <b>context</b> Kartenterminal<br><b>inv:</b> Kartenterminal.allInstances -> isUnique (ctId)  |
| C11 | Die Identifikationsnummer (slotNo) des Kartenterminal-Slots für ein gegebenes Kartenterminal muss eindeutig sein.  | <b>context</b> Kartenterminal<br><b>inv:</b> self.kT-Slot -> isUnique (slotNo)   |
| C12 | Es muss gewährleistet sein, dass nur Arbeitsplätze und Clientsysteme einander im Rahmen eines Mandanten zugeordnet werden, die diesem Mandanten selbst zugeordnet sind.                          | <b>context</b> CS-AP<br><b>inv:</b> self.arbeitsplatz.mandant.includes (self.mandant)<br><b>inv:</b> self.clientsystem.mandant.includes (self.mandant)           |
| C13 | Es muss gewährleistet sein, dass nur Kartenterminals und Arbeitsplätze einander im Rahmen eines Mandanten zur Remote-PIN-Eingabe zugeordnet werden, die diesem Mandanten selbst zugeordnet sind. | <b>context</b> Remote-PIN-KT<br><b>inv:</b> self.arbeitsplatz.mandant.includes (self.mandant)<br><b>inv:</b> self.kartenterminal.mandant.includes (self.mandant) |

|     |   |   |
|-----|---|---|
| C14 | Zur Remote-PIN-Eingabe muss ein <u>lokales</u> Kartenterminal ausgewählt sein.                                  | <b>context</b> Remote-PIN-KT<br><b>inv:</b> self.arbeitsplatz<br>.localKartenterminal<br>.includes(self.kartenterminal)<br><b>inv:</b> not self.arbeitsplatz<br>.entferntKartenterminal<br>.includes(self.kartenterminal) |
| C15 | Zur Remote-PIN-Eingabe darf pro Mandanten und Arbeitsplatz nicht mehr als ein Kartenterminal ausgewählt werden. | <b>context</b> Remote-PIN-KT<br><b>inv:</b> forAll(r1, r2 : Remote-PIN-KT  <br>r1.arbeitsplatz = r2.arbeitsplatz<br>and r1.mandant = r2.mandant implies<br>r1 = r2)   |
| C16 | Eine Kartensitzung-HBAX muss immer eine zugehörige userId haben.  | <b>context</b> Kartensitzung-HBAX<br><b>inv:</b> self.userId <> null  |

2150 *Hinweis zur Remote-PIN-Eingabe: Constraints C14 und C15 legen fest, dass auch im Fall*  
 2151 *mehrerer lokaler Kartenterminals an einem Arbeitsplatz nur eines (oder keines) dieser*  
 2152 *Kartenterminals pro Mandant für die Remote-PIN-Eingabe im Informationsmodell*  
 2153 *konfiguriert wird.*

#### 2154 **TIP1-A\_4523 - Sicherung der Aktualität des Informationsmodells**

#### 2155 **Zugriffsberechtigungsdienst**

2156 Der Konnektor MUSS seine Entscheidungen zur Zugriffsberechtigung basierend auf den  
 2157 aktuellen, realen statischen wie transienten Entitäten und Beziehungen des  
 2158 Informationsmodells treffen. Veränderungen an der statischen Definition (durch den  
 2159 Administrator), sowie Veränderungen an den Entitäten (Änderung der Verfügbarkeit und  
 2160 Zustandsänderung von Karten, Kartenterminals und Clientsystemen) MÜSSEN bei  
 2161 Zugriffsanfragen unmittelbare Auswirkung auf die Entscheidung des  
 2162 Zugriffsberechtigungsdienstes zur Folge haben.

2163 [**<=**]

#### 2164 **4.1.1.2 Durch Ereignisse ausgelöste Reaktionen**

2165 Keine.

#### 2166 **4.1.1.3 Interne TUCs, nicht durch Fachmodule nutzbar**

2167 Keine.

#### 2168 **4.1.1.4 Interne TUCs, auch durch Fachmodule nutzbar**

##### 2169 **4.1.1.4.1 TUC\_KON\_000 „Prüfe Zugriffsberechtigung“**

2170 Vor Ausführung jeder Operation an der Außenschnittstelle muss der Konnektor prüfen, ob  
 2171 die Operation ausgeführt werden darf (Autorisierung). Diese Prüfung auf  
 2172 Zugriffsberechtigung wird in TUC\_KON\_000 „Prüfe Zugriffsberechtigung“ gekapselt.

TUC\_KON\_000 „Prüfe Zugriffsberechtigung“ hat als Aufrufparameter den Aufrufkontext der Operation (siehe Abbildung PIC\_KON\_101), optional das cardHandle einer Karte, optional eine Kartenterminal-ID ctId und optional die Steuerungsparameter „needCardSession“ sowie „allWorkplaces“. Über den Steuerungsparameter „needCardSession“ wird festgelegt, ob zu den CardHandles im Rahmen der Operationsausführung eine Kartensitzung benötigt wird. Über den Steuerungsparameter „allWorkplaces“ wird festgelegt, ob die Auswertung im Rahmen der Operation arbeitsplatzübergreifend für alle vom Mandanten für das angegebene Clientsystem erreichbaren Kartenterminals erfolgen soll.

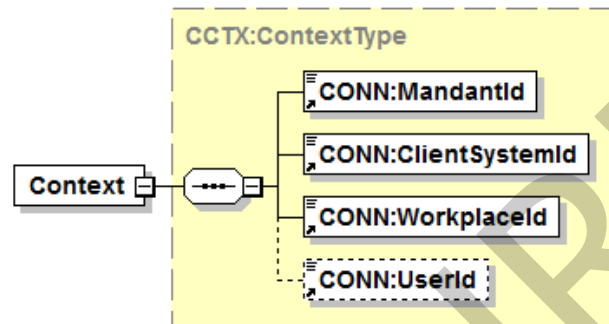


Abbildung 5: PIC\_KON\_101 Aufrufkontext der Operation

#### TIP1-A\_4524-02 - TUC\_KON\_000 „Prüfe Zugriffsberechtigung“

Der Konnektor MUSS den technischen Use Case TUC\_KON\_000 „Prüfe Zugriffsberechtigung“ umsetzen.

Tabelle 15: TAB\_KON\_511 – TUC\_KON\_000 „Prüfe Zugriffsberechtigung“

| Element                     | Beschreibung  |
|-----------------------------|---|
| Name                        | TUC_KON_000 "Prüfe Zugriffsberechtigung"  |
| Beschreibung                | Es wird geprüft, ob eine Autorisierung im Rahmen der angegebenen Eingangsdaten erteilt wird. Die Autorisierung wurde erteilt, wenn der TUC erfolgreich durchlaufen wurde (kein Abbruch durch Fehlermeldung)." |
| Eingangs-anforderungen      | keine   |
| Auslöser und Vorbedingungen | Aufruf einer Operation des Konnektors durch das Clientsystem.   |

|                 |   |
|-----------------|---|
| Eingangsdaten   | <ul style="list-style-type: none"> <li>• mandantId</li> <li>• clientSystemId</li> <li>• workplaceId</li> <li>• userId - <i>optional</i></li> <li>• ctId - <i>optional</i><br/>(Kartenterminalidentifikator)</li> <li>• cardHandle - <i>optional</i></li> <li>• needCardSession [Boolean] – <i>optional; default: true</i><br/>(„needCardSession“=true;<br/>„doNotNeedCardSession“=false)<br/>Dieser Schalter gibt an, ob eine Kartensitzung benötigt wird <ul style="list-style-type: none"> <li>- true, der aufrufende TUC verwendet eine Kartensitzung</li> <li>- false, der aufrufende TUC verwendet keine Kartensitzung</li> </ul> Die Berechtigungsprüfung geht im Default-Fall, davon aus, dass eine Kartensitzung benötigt wird, und prüft für diesen Fall die Berechtigung mit. </li> <li>• allWorkplaces [Boolean] – <i>optional; default: false</i><br/>Dieser Schalter gibt an, ob eine mandantenweite Zugriffsberechtigung gemeint ist.<br/>Dieser Parameter muss dann (true) gesetzt werden, wenn die Berechtigungsprüfung nicht auf die vom angegebenen Arbeitsplatz erreichbaren Kartenterminals beschränkt ist, sondern sich auf alle vom Clientsystem (clientSystemId) und dem Mandant (mandantId) insgesamt erreichbaren Kartenterminals beziehen soll. Ist dieser Schalter gleich true, wird die Berechtigung unabhängig vom Eingangsparameter workplaceId geprüft.</li> </ul> |
| Komponenten     | Konnektor   |
| Ausgangsdaten   | <ul style="list-style-type: none"> <li>• keine</li> </ul>   |
| Nachbedingungen | <ul style="list-style-type: none"> <li>• Autorisierung erteilt</li> </ul>   |



|                            |  |
|----------------------------|--|
| Standardablauf             | <ol style="list-style-type: none"> <li>1. Prüfe, ob die Pflichtparameter (mandantId, clientSystemId, workplaceId) vollständig gesetzt sind.</li> <li>2. Falls ANCL_CAUT_MANDATORY = Enabled, dann prüfe, ob die gemäß [TIP1-A_4516] durchgeführte Authentifizierung über ein dem Clientsystem zugeordnetes CS-AuthMerkmal erfolgte.</li> <li>3. Ermittle Zugriffsregel R zu den Aufrufparametern: <ol style="list-style-type: none"> <li>3.1. Falls der Parameter cardHandle nicht null ist, muss das Kartenobjekt des Informationsmodells Karte(cardHandle) ermittelt werden.</li> <li>3.2. Zu den Parametern (ctId, cardHandle, needCardSession, allWorkplaces) muss mittels Tabelle „TAB_KON_513 Zugriffsregeln Regelzuordnung“ die Zugriffsregel R ermittelt werden.</li> </ol> </li> <li>4. Prüfe die Bedingungen der in Schritt 3 ermittelten Regel R: <ol style="list-style-type: none"> <li>4.1. Zur Regel R muss die relevante Spalte in Tabelle „TAB_KON_514 Zugriffsregeln Definition“ ermittelt werden.</li> <li>4.2. Jede Zeile, die in der Spalte R ein „x“ hat, muss geprüft werden: <ol style="list-style-type: none"> <li>4.2.1 Prüfe, ob die in Spalte „Bedingung“ mittels OCL formulierte Bedingung für die Eingangsdaten erfüllt ist.</li> </ol> </li> </ol> </li> </ol> |
| Varianten/<br>Alternativen | <ol style="list-style-type: none"> <li>2. Bei einem Aufruf mit einem cardHandle zu den Kartentypen SMC-KT und UNKNOWN wird Schritt 3 in folgender Variante durchlaufen:<br/><br/>Ermittle Zugriffsregel R zu den Aufrufparametern: <ol style="list-style-type: none"> <li>3.1. ctId wird zum cardHandle bestimmt<br/>Zu den Parametern ( <ol style="list-style-type: none"> <li>ctId,</li> <li>cardHandle = null,</li> <li>needCardSession = false,</li> <li>allWorkplaces = false)</li> </ol> muss mittels Tabelle „TAB_KON_513 Zugriffsregeln Regelzuordnung“ die Zugriffsregel R ermittelt werden.</li> </ol> </li> </ol>   |
| Fehlerfälle                | <p>Fehler im Ablauf (Standardablauf oder Varianten) führen in den folgend ausgewiesenen Schritten zu einem Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes:</p> <ol style="list-style-type: none"> <li>(→1) Es sind nicht alle Pflichtparameter gesetzt, Fehlercode: 4021</li> <li>(→2) Clientsystem aus dem Aufrufkontext nicht authentifiziert, Fehlercode: 4204</li> <li>(→3.1) Karte nicht als gesteckt identifiziert,</li> </ol>   |

|                                |   |
|--------------------------------|---|
|                                | Fehlercode: 4008<br>(→3.2) Zu den Parametern konnte keine Regel ermittelt werden,<br>Fehlercode: 4019<br>(→4.2.1) Bedingung nicht erfüllt<br>Fehlercode: wie in Spalte „ErrorCode“ der geprüften<br>Zeile aus<br>Tabelle „TAB_KON_514-01 Zugriffsregeln Definition“ |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | PIC_KON_118 Aktivitätsdiagramm zu „TUC_KON_000 Prüfe Zugriffsberechtigung“  |

2191  
2192  
2193  
2194  
2195  
2196

[<=]

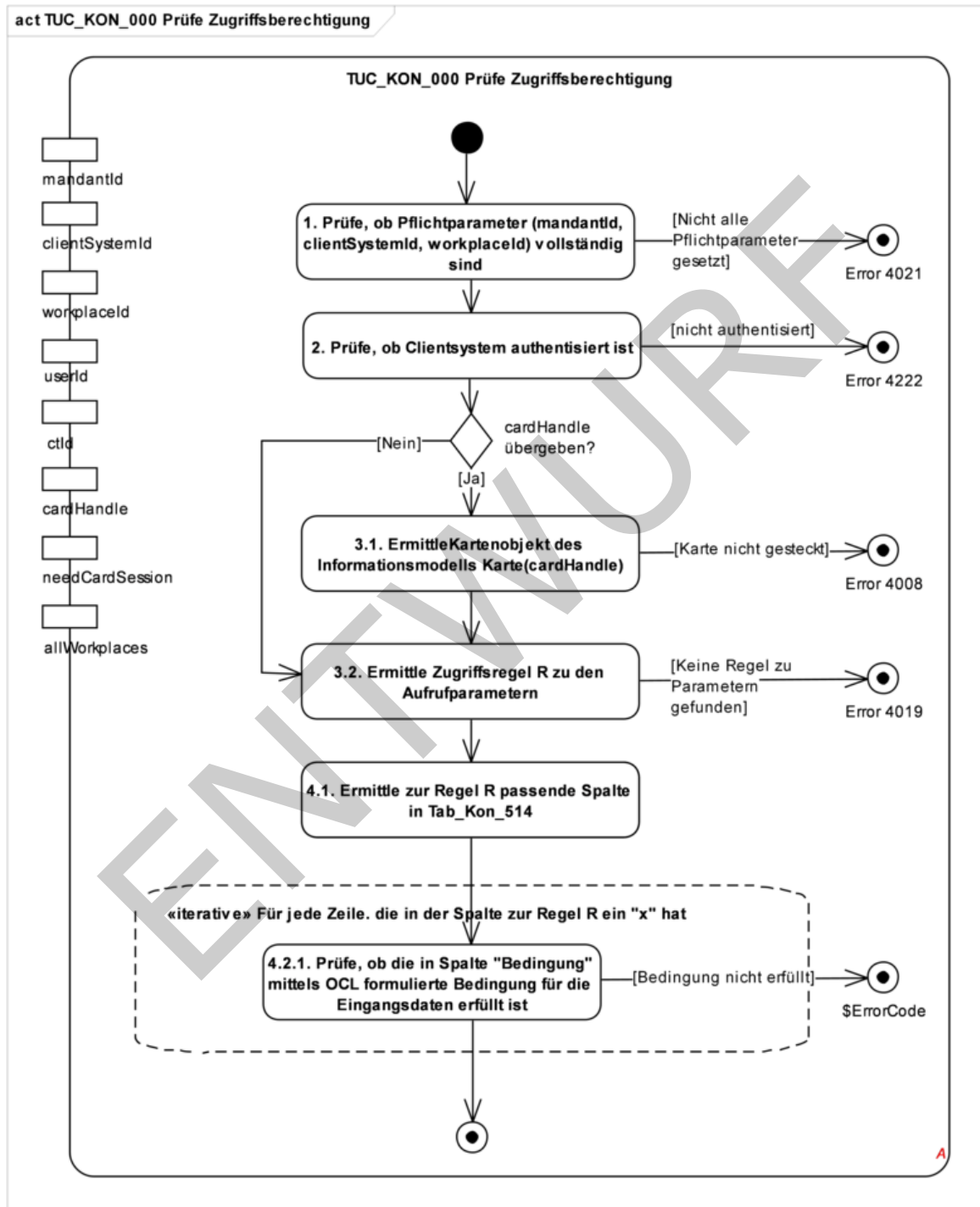
Eine Beschreibung aller Zugriffsregeln gibt Tabelle TAB\_KON\_512.

**Tabelle 16: TAB\_KON\_512 Zugriffsregeln Beschreibung**

| Regel | Beschreibung  |
|-------|---|
| R1    | Innerhalb des Mandanten m darf das Clientsystem cs verwendet werden.  |
| R2    | Innerhalb des Mandanten m darf das Clientsystem cs auf das Kartenterminal kt zugreifen.   |
| R3    | Innerhalb des Mandanten m darf das Clientsystem cs den Arbeitsplatz ap nutzen.  |
| R4    | Innerhalb des Mandanten m darf das Clientsystem cs über den Arbeitsplatz ap auf das Kartenterminal kt zugreifen.  |
| R5    | Innerhalb des Mandanten m darf das Clientsystem cs über den Arbeitsplatz ap auf die lokal gesteckte eGK zugreifen. Eine Kartensitzung wird nicht benötigt.  |
| R6    | Innerhalb des Mandanten m darf das Clientsystem cs über den Arbeitsplatz ap auf die lokal gesteckte eGK zugreifen. Eine Kartensitzung wird benötigt. Wenn bereits eine Kartensitzung besteht, ist sichergestellt, dass sie vom Arbeitsplatz ap gestartet wurde. |
| R7    | Innerhalb des Mandanten m darf das Clientsystem cs über den Arbeitsplatz ap auf die SM-B zugreifen. Es wird dabei sichergestellt, dass es sich um eine im Mandanten verwaltete SM-B handelt.  |
| R8    | Innerhalb des Mandanten m darf das Clientsystem cs auf den HBAX zugreifen. Eine Kartensitzung wird nicht benötigt.  |
| R9    | Innerhalb des Mandanten m darf das Clientsystem cs auf den HBAX zugreifen. Eine Kartensitzung wird benötigt. Wenn bereits Kartensitzungen zum HBAX bestehen, wird der Zugriff auf den HBAX verhindert, wenn es eine   |

Kartensitzung zum selben Clientsystem, aber einer anderen UserId gibt, deren Sicherheitszustand erhöht ist.

2197



2198

2199

2200

**Abbildung 6: PIC\_KON\_118 Aktivitätsdiagramm zu „TUC\_KON\_000 Prüfe Zugriffsberechtigung“**

2201

2202

Welche Zugriffsregel für einen gegebenen Satz an Aufrufparametern anzuwenden ist, wird in Tabelle TAB\_KON\_513 ermittelt. Die Pflichtfelder mandantId, clientSystemId und

workplaceId und das optionale Feld userId sind zwar für die Auswertung der Regeln wichtig, tragen aber nicht zur Auswahl der Regel bei und sind daher in der Tabelle nicht vorhanden. Zur Auswahl einer Regel ist relevant,

- ob ctId bzw. cardHandle als Aufrufparameter gesetzt sind (not null) oder leer sind (null),
- von welchem Typ eine Karte ist, falls der Aufrufparameter cardHandle gesetzt ist,
- und welchen Wert die Aufrufparameter „needCardSession“ und „allWorkplaces“ annehmen.

**Tabelle 17: TAB\_KON\_513 Zugriffsregeln Regelzuordnung**

| Parameter              | R1    | R2       | R3    | R4       | R5           | R6           | R7              | R8       | R9       |
|------------------------|-------|----------|-------|----------|--------------|--------------|-----------------|----------|----------|
| ctId                   | null  | not null | null  | not null |              |              |                 |          |          |
| cardHandle             | null  | null     | null  | null     | not null     | not null     | not null        | not null | not null |
| Karte(cardHandle).type |       |          |       |          | eGK oder KVK | eGK oder KVK |                 |          |          |
| Karte(cardHandle).type |       |          |       |          |              |              | SM-B            |          |          |
| Karte(cardHandle).type |       |          |       |          |              |              |                 | HBAX     | HBAX     |
| needCardSession        | false | false    | false | false    | false        | true         | true oder false | false    | true     |
| allWorkplaces          | true  | true     | false | false    | false        | false        | false           | false    | false    |

Tabelle TAB\_KON\_514 definiert einzelne Bedingungen, ordnet sie den Regeln zu und definiert ErrorCodes für den Fall, dass eine Bedingung nicht erfüllt ist.

Die Bedingungen in Tabelle TAB\_KON\_514 sind wie folgt gruppiert:

- Entitäten: Hier wird geprüft, ob die Entitäten, die mit den Aufrufparametern adressiert werden, im Informationsmodell existieren.
- Mandantenbezug: Hier wird geprüft, ob die adressierten Entitäten im Informationsmodell dem adressierten Mandanten zugeordnet sind.
- Relationen: Hier wird geprüft, ob die benötigten Zugriffsbeziehungen zum Zugriff auf die adressierten Entitäten im Informationsmodell existieren.
- Kartensitzungen: Hier wird geprüft, ob die benötigte Kartensitzung im Rahmen der bereits existierenden Kartenbeziehungen existieren darf.

Die Fehlercodes mit Beschreibung, ErrorType und Severity Tabelle TAB\_KON\_515.

2227 **Tabelle 18: TAB\_KON\_514-01 Zugriffsregeln Definition**

|                                    | Bedingung<br>(siehe Hinweis 1)   | R1 | R2 | R3 | R4 | R5 | R6 | R7 | R8 | R9 | Error Code  |
|------------------------------------|--|----|----|----|----|----|----|----|----|----|---|
| Entität<br>(siehe<br>Hinweis<br>2) | inv : userId <> null   |    |    |    |    |    |    |    |    | x  | 4003  |
|                                    | let m : Mandant =<br>Mandant(mandantId)<br>inv : m <> null   | x  | x  | x  | x  | x  | x  | x  | x  | x  | 4021 an der<br>Außenschnittstelle<br>4004 im Protokoll<br>(siehe Hinweis 3) |
|                                    | let cs : Clientsystem<br>= Clientsystem<br>(clientSystemId)<br>inv : cs <> null  | x  | x  | x  | x  | x  | x  | x  | x  | x  | 4021 an der<br>Außenschnittstelle<br>4005 im Protokoll<br>(siehe Hinweis 3) |
|                                    | let ap : Arbeitsplatz<br>= Arbeitsplatz<br>(workplaceId)<br>inv : ap <> null   |    |    | x  | x  | x  | x  | x  | x  | x  | 4021 an der<br>Außenschnittstelle<br>4006 im Protokoll<br>(siehe Hinweis 3) |
|                                    | let kt : Kartenterminal<br>= Kartenterminal<br>(ctId)<br>inv : kt <> null  |    | x  |    | x  |    |    |    |    |    | 4007  |
|                                    | let k : Karte = Karte<br>(cardHandle)<br>inv : k <> null   |    |    |    |    | x  | x  | x  | x  | x  | 4008  |
| Mandant<br>bezug                   | let m : Mandant =<br>Mandant(mandantId)<br>let cs : Clientsystem<br>= Clientsystem<br>(clientSystemId)<br>inv : cs.mandant.<br>includes(m) | x  | x  | x  | x  | x  | x  | x  | x  | x  | 4010  |
|                                    | let m : Mandant =<br>Mandant(mandantId)<br>let ap : Arbeitsplatz<br>= Arbeitsplatz<br>(workplaceId)<br>inv : ap.mandant.<br>includes(m)    |    |    | x  | x  | x  | x  | x  | x  | x  | 4011  |

|          |   |  |   |   |   |   |   |   |   |      |
|----------|---|--|---|---|---|---|---|---|---|------|
|          | <pre>let m : Mandant = Mandant(mandantId) let kt : Kartenterminal = Kartenterminal(ctId) inv : kt.mandant. includes(m)</pre>  |  | x | x |   |   |   |   |   | 4012 |
|          | <pre>let m : Mandant = Mandant(mandantId) let k : Karte = Karte(cardHandle) inv : k.kT-Slot. kartenterminal.mandant .includes(m)</pre>  |  |   |   | x | x | x | x | x | 4012 |
| Relation | <pre>let k : Karte = Karte(cardHandle) inv : k.SM-B_Verwaltet &lt;&gt; null</pre>   |  |   |   |   |   | x |   |   | 4009 |
|          | <pre>let m : Mandant = Mandant(mandantId) let k : Karte = Karte(cardHandle) inv : k.SM-B_Verwaltet .mandant -&gt; includes(m)</pre>   |  |   |   |   |   | x |   |   | 4013 |
|          | <pre>let m : Mandant = Mandant(mandantId) let cs : Clientsystem = Clientsystem (clientSystemId) let ap : Arbeitsplatz = Arbeitsplatz (workplaceId) inv : CS_AP.allInstances -&gt; exists(c : CS_AP   c.mandant = m and c.arbeitsplatz = ap and c.clientsystem = cs)</pre> |  |   | x | x | x | x | x | x | 4014 |

|  |   |  |  |   |  |   |   |   |  |      |
|--|---|--|--|---|--|---|---|---|--|------|
| <pre> let ap : Arbeitsplatz   = Arbeitsplatz   (workplaceId) let kt : Kartenterminal   = Kartenterminal   (ctId) inv :   ap.lokalKartenterminal   .includes(kt) or   ap.entferntKarten   terminal   .includes(kt) </pre>   |   |  |  | x |  |   |   |   |  | 4015 |
| <pre> let ap : Arbeitsplatz   = Arbeitsplatz   (workplaceId) let kt : Kartenterminal   = Karte(cardHandle).kT-   Slot.kartenterminal inv :   ap.lokalKartenterminal   .includes(kt) or   ap.entferntKarten   terminal   .includes(kt) </pre>   |   |  |  |   |  | x | x | x |  | 4015 |
| <pre> let m : Mandant = Mandant   (mandantId) let kt : Kartenterminal   = Kartenterminal(ctId) let cs : Clientsystem =   Clientsystem   (clientSystemId) inv : CS_AP.allInstances   -&gt; exists(c : CS_AP     c.arbeitsplatz   .lokalKartenterminal   .includes(kt) or   c.arbeitsplatz   .entferntKartenterminal   .includes(kt)   and c.mandant = m   and c.arbeitsplatz.mandant   .includes(m)   and c.clientsystem = cs) </pre> | x |  |  |   |  |   |   |   |  | 4020 |



|                     |   |  |  |  |  |   |   |  |  |   |  |      |
|---------------------|---|--|--|--|--|---|---|--|--|---|--|------|
|                     | <pre> let ap : Arbeitsplatz     = Arbeitsplatz     (workplaceId) let kt : Kartenterminal     = Karte(cardHandle).kT-     Slot.kartenterminal inv :     ap.lokalKartenterminal     .includes(kt) </pre>  |  |  |  |  | x | x |  |  |   |  | 4016 |
| Karten<br>sitzungen | <pre> let ap : Arbeitsplatz     = Arbeitsplatz     (workplaceId) let k : Karte =     Karte(cardHandle) inv : k.kartensitzung     -&gt; not exists(ks :         Kartensitzung               ks.arbeitsplatz     &lt;&gt; ap) </pre>  |  |  |  |  |   | x |  |  |   |  | 4017 |
|                     | <pre> let k : Karte = Karte     (cardHandle) let cs : Clientsystem     = Clientsystem     (clientSystemId) inv : k.kartensitzung     -&gt; not exists     (ks : Kartensitzung           ks         .clientsystem = cs and         ks         .userId &lt;&gt; userId and         ks         .authState.size() &gt; 0     ) </pre> |  |  |  |  |   |   |  |  | x |  | 4018 |

## 2228 Erläuterungen zu TAB\_KON\_514-01:

- 2229 Hinweis 1:  
 2230 Jede Bedingung ist als Constraint mittels OCL definiert, ist einzeln prüfbar und hat als  
 2231 Eingangsparameter mandantId, clientSystemId, workplaceId, ctId, cardHandle und userId.
- 2232 Hinweis 2:  
 2233 Zur Bezeichnung einer Objektinstanz, die im Informationsmodell vorhanden ist, wird die  
 2234 Notation <<Entitätsbezeichner>>(<<Komma separierte Liste der Identitätsschlüssel>>  
 2235 verwendet.
- 2236 Hinweis 3:  
 2237 Bei manchen Bedingungen gibt es unterschiedliche Fehlermeldungen für die  
 2238 Außenschnittstelle und für die interne Protokollierung. Dann wird folgende Notation in  
 2239 Spalte "Error Code" verwendet:

- 2240 " <<Fehlercode>> an der Außenschnittstelle" für den Fehlercode, der über die  
2241 Außenschnittstelle zurückgegeben werden muss  
2242 " <<Fehlercode>> im Protokoll" für den Fehlercode, der für die interne Protokollierung  
2243 verwendet werden muss.

2244

2245 **Tabelle 19: TAB\_KON\_515 Fehlercodes TUC\_KON\_000 „Prüfe Zugriffsberechtigung“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4003  | Technical | Error    | Keine User-ID angegeben, die zur Identifikation der Kartensitzung_HBAX benötigt wird. |
| 4004  | Technical | Error    | Ungültige Mandanten-ID  |
| 4005  | Technical | Error    | Ungültige Clientsystem-ID   |
| 4006  | Technical | Error    | Ungültige Arbeitsplatz-ID   |
| 4007  | Technical | Error    | Ungültige Kartenterminal-ID   |
| 4008  | Technical | Error    | Karte nicht als gesteckt identifiziert  |
| 4009  | Security  | Error    | SM-B ist dem Konnektor nicht als SM-B_Verwaltet bekannt                               |
| 4010  | Security  | Error    | Clientsystem ist dem Mandanten nicht zugeordnet                                       |
| 4011  | Security  | Error    | Arbeitsplatz ist dem Mandanten nicht zugeordnet                                       |
| 4012  | Security  | Error    | Kartenterminal ist dem Mandanten nicht zugeordnet                                     |
| 4013  | Security  | Error    | SM-B_Verwaltet ist dem Mandanten nicht zugeordnet                                     |
| 4014  | Security  | Error    | Für den Mandanten ist der Arbeitsplatz nicht dem Clientsystem zugeordnet              |
| 4015  | Security  | Error    | Kartenterminal ist weder lokal noch entfernt vom Arbeitsplatz aus zugreifbar          |

|      |           |       |  |
|------|-----------|-------|--|
| 4016 | Security  | Error | Kartenterminal ist nicht lokal vom Arbeitsplatz aus zugreifbar   |
| 4017 | Security  | Error | Die eGK hat bereits eine Kartensitzung, die einem anderen Arbeitsplatz zugeordnet ist.   |
| 4018 | Security  | Error | Der HBAX hat mindestens eine Kartensitzung zu einer anderen UserId, deren Sicherheitszustand erhöht ist. (Sicherheitszustand wird bei PIN-Eingabe erhöht.) |
| 4019 | Technical | Error | Zu den Parametern konnte keine Regel ermittelt werden.   |
| 4020 | Security  | Error | Kartenterminal ist weder lokal noch entfernt über irgendeinen dem Clientsystem zugeordneten Arbeitsplatz aus zugreifbar                                    |
| 4021 | Technical | Error | Es sind nicht alle Pflichtparameter mandantId, clientSystemId, workplaceId gefüllt.  |
| 4204 | Security  | Error | Clientsystem aus dem Aufrufkontext konnte nicht authentifiziert werden.  |

2246 Hinweis zu Fehler 4018: Sicherheitszustand wird bei PIN-Eingabe erhöht.

#### 2247 4.1.1.5 Operationen an der Außenschnittstelle

2248 Keine

#### 2249 4.1.1.6 Betriebsaspekte

##### 2250 TIP1-A\_4525 - Initialisierung Zugriffsberechtigungsdiens

2251 Der Konnektor MUSS mit Abschluss der Bootup-Phase den Ist-Zustand transienter  
2252 Entitäten und Beziehungen des Informationsmodells erfasst haben.

2253 [ $\leq$ ]

2254

##### 2255 TIP1-A\_4526 - Bearbeitung Informationsmodell Zugriffsberechtigungsdiens

2256 Für die Administration MUSS der Konnektor eine Administrationsoberfläche zur Pflege des  
2257 Informationsmodells zur Verfügung stellen. Die Oberfläche muss es ermöglichen,  
2258 sämtliche persistente Entitäten und Beziehungen des durch Abbildung „PIC\_Kon\_100  
2259 Informationsmodell des Konnektors“ und Tabelle „TAB\_KON\_510 Informationsmodell  
2260 Constraints“ definierten Informationsmodells initial anzulegen, zu ändern und zu löschen.

2261 [ $\leq$ ]

2262 Im Anhang I „Umsetzungshinweise“ werden Empfehlungen zur Umsetzung der  
2263 Administration des Informationsmodells gegeben.

#### 2264 4.1.2 Dokumentvalidierungsdienst

2265 Der Dokumentvalidierungsdienst ist ein Dienst, der nur intern genutzt wird, d. h., dass  
2266 dessen definierte Verhaltensweisen nur in anderen TUCs des Konnektors nachgenutzt

2267 werden. Er bietet Schnittstellen zum Validieren von Dokumenten an. Dabei werden  
 2268 diejenigen spezifischen Dokumentformate unterstützt, die an den Außenschnittstellen  
 2269 anderer Dienste wie Signatur- und Verschlüsselungsdienst auftreten können  
 2270 (`Alle_DocFormate` gemäß Kapitel 3).

2271 Die jeweils gültigen XML-Schemas der Fachmodule werden den Herstellern von der  
 2272 gematik bereitgestellt.

#### 2273 **4.1.2.1 Funktionsmerkmalweite Aspekte**

##### 2274 **A\_18780 - PDF/A-3 DARF NICHT unterstützt werden**

2275 Der Konnektor DARF Dokumente im PDF/A-3 Format NICHT unterstützen.  
 2276 [`<=`]

#### 2277 **4.1.2.2 Durch Ereignisse ausgelöste Reaktionen**

2278 Keine.

#### 2279 **4.1.2.3 Interne TUCs, nicht durch Fachmodule nutzbar**

2280 Keine

#### 2281 **4.1.2.4 Interne TUCs, auch durch Fachmodule nutzbar**

##### 2282 *4.1.2.4.1 TUC\_KON\_080 „Dokument validieren“*

2283

##### 2284 **TIP1-A\_4527-01 - TUC\_KON\_080 „Dokument validieren“**

2285 Der Konnektor MUSS den technischen Use Case TUC\_KON\_080 „Dokument validieren“  
 2286 umsetzen.

2287

##### 2288 **Tabelle 20: TAB\_KON\_143 – TUC\_KON\_080 „Dokument validieren“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_080 „Dokument validieren“   |
| Beschreibung   | Dieser TUC prüft das Format eines Dokuments und führt dokumententyp-spezifische Validierungen durch. Unterstützt werden <code>Alle_DocFormate</code> (außer „Binär“).   |
| Auslöser       | <ul style="list-style-type: none"> <li>• Aufruf durch Fachmodul</li> <li>• Aufruf durch Basisdienst</li> </ul>  |
| Vorbedingungen | Keine   |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• <code>documentToBeValidated</code><br/>(Zu validierendes Dokument.)</li> <li>• <code>documentFormat</code><br/>(mögliche Werte siehe Definition <code>Alle_DocFormate</code>;<br/>Formatangabe für das Dokument)</li> </ul> <p>Optional für XML-Dokumente:</p> |

|                 |   |
|-----------------|---|
|                 | <ul style="list-style-type: none"> <li>xmlSchemas – optional/nur für XML-Dokumente (XML-Schema und ggf. weitere vom Hauptschema benutzte Schemata)</li> <li>signaturePolicyIdentifier – optional/nur für XML-Formate gemäß einer referenzierten Signaturreichtlinie (URI identifiziert die Signaturreichtlinie)</li> </ul>  |
| Komponenten     | Konnektor   |
| Ausgangsdaten   | <ul style="list-style-type: none"> <li>documentValidationProtocol (Prüfprotokoll)<br/>Die Ausprägung dieses Konnektor-internen Parameters erfolgt herstellerspezifisch.</li> </ul>  |
| Nachbedingungen | Keine   |
| Standardablauf  | <p><b>Validierung der Dokumente auf Typkonformität</b><br/>Der Konnektor führt je nach Format des Dokuments (documentFormat) eine der folgenden Prüfungen durch:</p> <p>A) XML-Dokumentvalidierung<br/>Im Fall eines XML-Dokuments prüft der Konnektor:</p> <ul style="list-style-type: none"> <li>Prüfe die XML-Wohlgeformtheit des Dokumentes (documentToBeValidated)</li> <li>Wenn signaturePolicyIdentifier vorhanden ist, dann ermittle das xmlSchema aus der referenzierten Signaturreichtlinie und prüfe die Validität von documentToBeValidated in Bezug auf das hinterlegte XML-Schema. Der Eingangsparameter xmlSchemas wird ignoriert.</li> <li>Wenn signaturePolicyIdentifier nicht vorhanden ist und xmlSchemas übergeben wurden, dann prüfe die Wohlgeformtheit von xmlSchemas und die Validität von documentToBeValidated in Bezug auf xmlSchemas.</li> <li>Wenn nicht durch Prüfung gegen XML-Schema bereits erfolgt, dann prüfe die Eindeutigkeit der ID-Attributwerte im XML-Dokument.</li> </ul> <p>B) PDF/A-Dokumentvalidierung<br/>PDF/A-Dokumente werden geprüft, ob sie sich als PDF/A Dokumente in ihren PDF/A-Metadaten ausweisen: Es wird geprüft, ob diese eines der folgenden Elemente enthalten</p> <pre> &lt;pdfaid:part xmlns:pdfaid="http://www.aiim.org/pdfa/ns/id/"&gt;1&lt;/pdfaid:part&gt;  &lt;pdfaid:part xmlns:pdfaid="http://www.aiim.org/pdfa/ns/id/"&gt;2&lt;/pdfaid:part&gt;  &lt;pdfaid:part xmlns:pdfaid="http://www.aiim.org/pdfa/ns/id/"&gt;3&lt;/pdfaid:part&gt; </pre> |

|                                |   |
|--------------------------------|---|
|                                | <p>C) TIFF-Dokumentvalidierung<br/>Der Konnektor prüft, ob das Dokument an Hand seiner ersten 8 Byte als TIFF-Dokument [TIFF6] zu identifizieren ist.</p> <p>D) MIME-Dokumentvalidierung<br/>Die Struktur von MIME-Dokumenten wird entsprechend [MIME] validiert.</p> <p>E) Text-Dokumentvalidierung<br/>Der Konnektor prüft die Konformität zum im Dokumentenformat vorgegebenen Character-Encoding.<br/>Für Binärdokumente findet keine Validierung statt.<br/>Hinweis: Byte-order-marks (BOM) sind im Rahmen von UTF-8 kodierten Dokumenten gemäß UTF8 Standard ([RFC3629], Kapitel 6) erlaubt, aber nicht notwendigerweise im Dokument vorhanden.</p>   |
| Varianten/<br>Alternativen     |   |
| Fehlerfälle                    | <p><b>Standardablauf:</b><br/>Bei der Dokumentenvalidierung protokolliert der TUC alle aufgetretenen Fehler im Rückgabewert documentValidationProtocol.</p> <p>(→A) Fehlerfälle bei XML-Dokumentvalidierung<br/>Wenn keine Schemata übergeben wurden (xmlSchemas oder signaturePolicyIdentifier nicht vorhanden): Fehlercode 4193<br/>Wenn eines der übergebenen Schemata selbst nicht wohlgeformt oder invalide ist, wird Fehlercode 4026 gemeldet.<br/>Wenn das XML-Dokument nicht wohlgeformt ist, wird Fehlercode 4022 gemeldet.<br/>Das XML-Dokument ist nicht valide in Bezug auf das zur Validierung benutzte Schema (xmlSchemas bzw. signaturePolicyIdentifier): Fehlercode 4023.</p> <p>(→B) Fehlerfälle bei PDF/A-Dokumentvalidierung<br/>Bei fehlgeschlagener Validierung: Fehlercode 4024, Dokumentformat = PDF/A</p> <p>(→C) Fehlerfälle bei TIFF-Dokumentvalidierung<br/>Bei fehlgeschlagener Validierung: Fehlercode 4024, Dokumentformat = TIFF</p> <p>(→D) Fehlerfälle bei MIME-Dokumentvalidierung<br/>Bei fehlgeschlagener Validierung: Fehlercode 4024, Dokumentformat = MIME</p> <p>(→E) Fehlerfälle bei Text-Dokumentvalidierung<br/>Bei fehlgeschlagener Validierung: Fehlercode 4024, Dokumentformat = Text</p> |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | keine   |

**Tabelle 21: TAB\_KON\_144 Fehlercodes TUC\_KON\_080 „Dokument validieren“**

| Fehlercode | ErrorType | Severity | Fehlertext |
|------------|-----------|----------|------------|
|------------|-----------|----------|------------|

|   |          |         |  |
|---|----------|---------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |          |         |  |
| 4022  | Security | Error   | XML-Dokument nicht wohlgeformt   |
| 4023  | Security | Error   | XML-Dokument nicht valide in Bezug auf XML-Schema  |
| 4024  | Security | Error   | Formatvalidierung fehlgeschlagen (<Dokumentformat>)<br>Der Parameter Dokumentformat kann die Werte XML, PDF/A, TIFF, MIME und Text annehmen. |
| 4026  | Security | Error   | XML-Schema nicht valide  |
| 4193  | Security | Warning | kein XML-Schema für XML-Dokument vorhanden   |

2292

2293 [ $\leq$ ]2294 **4.1.2.5 Operationen an der Außenschnittstelle**

2295 Keine

2296 **4.1.2.6 Betriebsaspekte**

2297 Keine

2298 **4.1.3 Dienstverzeichnisdienst**

2299 Der Dienstverzeichnisdienst liefert dem aufrufenden Clientsystem sowohl Informationen  
 2300 über die Version und Produktkenndaten des Konnektors, als auch die SOAP-Endpunkte,  
 2301 über die das Clientsystem die einzelnen Dienstoperationen erreichen kann.

2302 **4.1.3.1 Funktionsmerkmalweite Aspekte**

2303 Die Endpunkte der Basisdienste werden in WSDL spezifiziert. Diese Endpunkte und  
 2304 weitere konnektormodellspezifische Informationen werden dem Clientsystem in Form  
 2305 eines Dienstverzeichnisdienstes gesammelt angeboten.

2306 Der prinzipielle Ablauf sieht dabei folgendermaßen aus:

2307 Das Clientsystem ruft beim Initialisieren des Systems mit HTTP-GET die vordefinierte  
 2308 URL: `https://<ANLW_LAN_IP_ADDRESS`  
 2309 oder `MGM_KONN_HOSTNAME>/connector.sds` oder `http://<ANLW_LAN_IP_ADDRESS`  
 2310 oder `MGM_KONN_HOSTNAME>/connector.sds` des Konnektors auf.

2311 Der Konnektor stellt die Liste der Dienste, der Versionen und die Endpunkte der Dienste  
 2312 in einem XML-Dokument zusammen. Jeder über SOAP erreichbare Basisdienst des  
 2313 Konnektors wird in dieser Liste geführt. Ferner können Fachmodule ihre eigenen  
 2314 Endpunkte über TUC\_KON\_041 „Einbringen der Endpunktinformationen während der  
 2315 Bootup-Phase“ einbringen. Die so erstellte Liste der Dienste wird als Antwort an das  
 2316 Clientsystem übergeben.

2317 Das Clientsystem prüft, ob die gewünschten Dienste und Versionen unterstützt werden  
 2318 und merkt sich die Endpunkte der Dienste für die späteren Aufrufe. Danach kann das  
 2319 Clientsystem diese Dienstendpunkte nach Bedarf aufrufen.



**TIP1-A\_4528 - Bereitstellen des Dienstverzeichnisdienst**

Der Konnektor MUSS den Dienstverzeichnisdienst anbieten. Dieser Dienst veröffentlicht auf: `https://$ANLW_LAN_IP_ADDRESS` oder `$MGM_KONN_HOSTNAME>/connector.sds` oder `http://$ANLW_LAN_IP_ADDRESS` oder `$MGM_KONN_HOSTNAME>/connector.sds`. Die Datei MUSS über https erreichbar sein. Wenn (ANCL\_DVD\_OPEN = Enabled) oder (ANCL\_TLS\_MANDATORY = Disabled) MUSS die Datei auch über http erreichbar sein. [**<=**]

**TIP1-A\_4529-02TIP1-A\_4529 - Formatierung der Ausgabedatei**

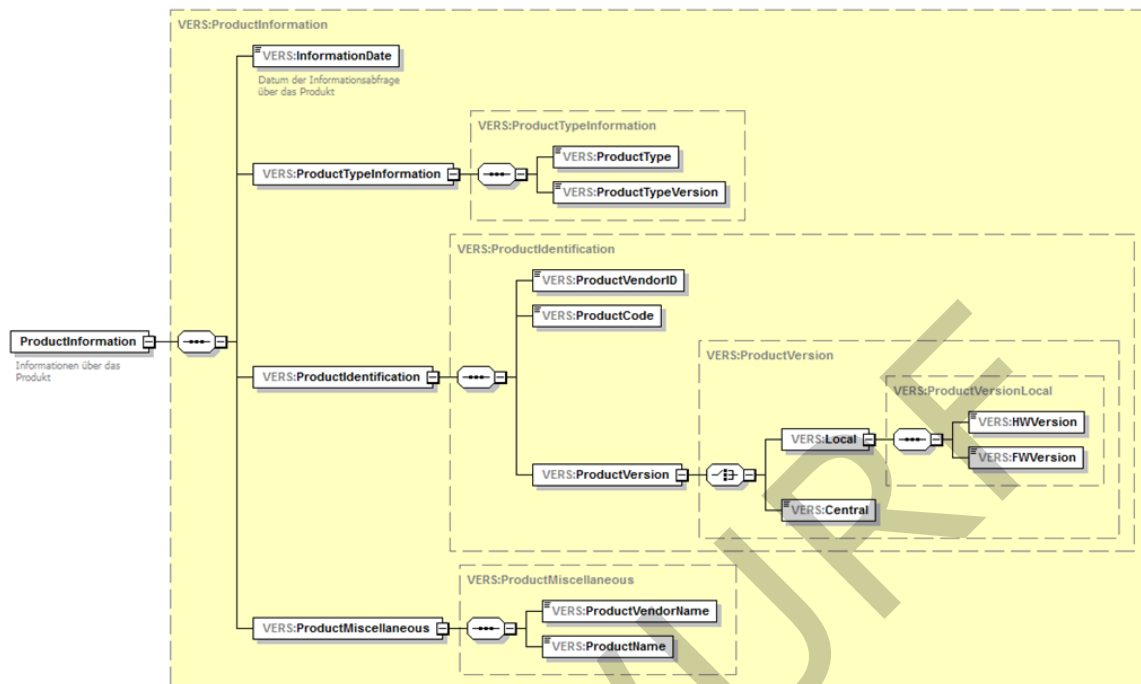
Das XML-Dokument, welches als „connector.sds“ dem Aufrufer zurückgeliefert wird, MUSS gemäß dem Schema „conn/ServiceDirectory.xsd“ formatiert sein. conn/ServiceDirectory.xsd referenziert die Schemata „tel/version/ProductInformation.xsd“ (siehe [gemSpec\_OM]) und „conn/ServiceInformation.xsd“. TAB\_KON\_516, TAB\_KON\_517 und TAB\_KON\_518 beschreiben die Elemente der zu verwendenden Schemastruktur.

**Tabelle 22: TAB\_KON\_516 Basisanwendung Dienstverzeichnisdienst**

|                          |   |
|--------------------------|---|
| <b>Name</b>              | ConnectorServiceDirectory                         |
| <b>Version</b>           | <a href="#">Siehe Anhang D3.1.0 (XSD-Version)</a> |
| <b>Namensraum</b>        | Siehe <a href="#">Anhang D GitHub</a>             |
| <b>Namensraum-Kürzel</b> | CONN  |
| <b>Operationen</b>       | Lesen der vom Konnektor unterstützten Dienste     |
| <b>WSDL</b>              | Keine   |
| <b>Schema</b>            | ServiceDirectory.xsd                              |

2340  
2341

**Tabelle 23: TAB\_KON\_517 Schemabeschreibung Produktinformation (ProductInformation.xsd)**

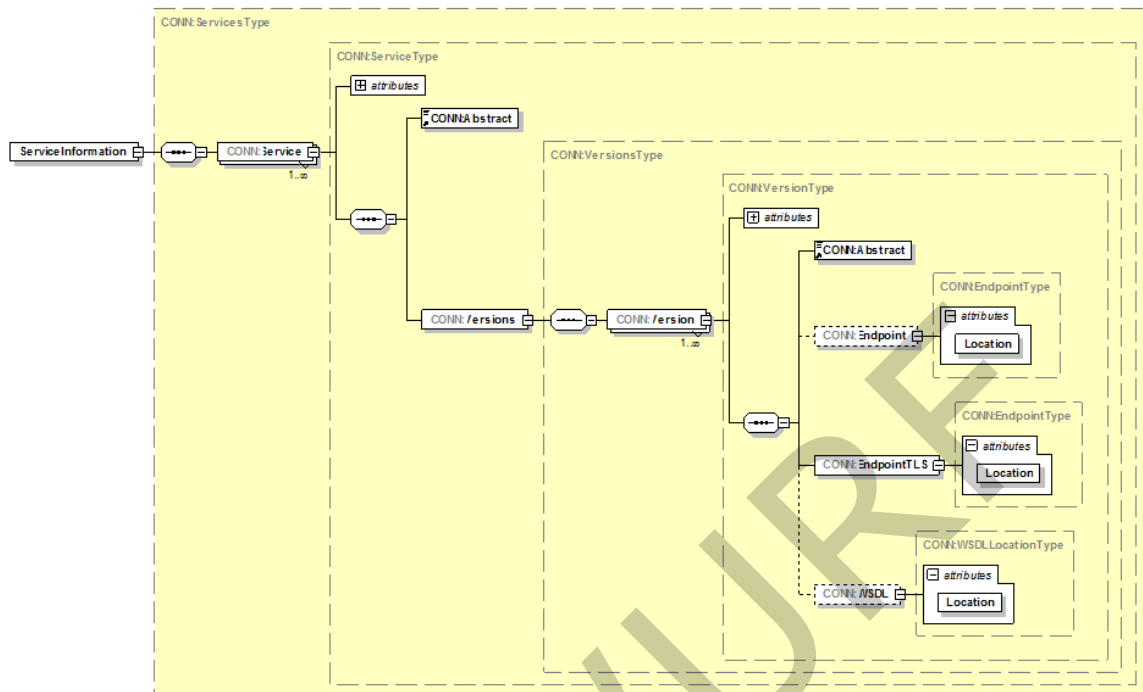


2342  
2343

| Element   | Bedeutung                                      |
|---|--|
| ProductInformation/InformationDate                                      | Datum der Informationsabfrage über das Produkt |
| ProductInformation/ProductTypeInfo/ProductType                          | Produkttyp (Konnektor)                         |
| ProductInformation/ProductTypeInfo/ProductTypeVersion                   | Produkttypversion des Konnektormodells         |
| ProductInformation/ProductIdentification/ProductVendorID                | ID des Konnektorherstellers                    |
| ProductInformation/ProductIdentification/ProductCode                    | Produktkürzel                                  |
| ProductInformation/ProductIdentification/ProductVersion/Local/HWVersion | Hardwareversion des Konnektormodells           |
| ProductInformation/ProductIdentification/ProductVersion/Local/FWVersion | Firmwareversion des Konnektormodells           |
| ProductInformation/ProductMiscellaneous/ProductVendorName               | Name des Konnektorherstellers                  |
| ProductInformation/ProductMiscellaneous/ProductName                     | Produktname                                    |

2344  
2345

**Tabelle 24: TAB\_KON\_518 Schemabeschreibung Serviceinformation (Serviceinformation.xsd)**



| Element  | Bedeutung  |
|--|--|
| ServiceInformation/Service                                   | Element beschreibt einen Dienst oder ein Fachmodul   |
| ServiceInformation/Service/@Name                             | Name des Dienstes. Dieser Wert korrespondiert mit dem Feld „Name“ aus der jeweiligen Basisanwendung/Dienstbeschreibung (englischer Dienstname in Tabelle TAB_KON_798).       |
| ServiceInformation/Service/Abstract                          | eine kurze textuelle Beschreibung des Dienstes/Fachmoduls  |
| ServiceInformation/Service/Versions                          | die Liste der unterstützten Versionen  |
| ServiceInformation/Service/Versions/Version                  | Beschreibung der Dienstversion/Fachmodulversion  |
| ServiceInformation/Service/Versions/Version/@TargetNamespace | der Namensraum der Dienst-/Fachmodulversion  |
| ServiceInformation/Service/Versions/Version/@Version         | Vollständige Versionsnummer (Konnektordienstversion) des Dienstes/Fachmoduls. Dieser Wert entspricht dem Wert „WSDL-Version“ des jeweiligen Dienstes in Tabelle TAB_KON_798. |

|   |  |
|---|--|
| ServiceInformation/Service/Versions/Version/Abstract              | Eine kurze textuelle Beschreibung dieser Version des Dienstes/Fachmoduls |
| ServiceInformation/Service/Versions/Version/EndpointTLS/@Location | Absoluter URL des über TLS erreichbaren Dienstes.                        |
| ServiceInformation/Service/Versions/Version/Endpoint/@Location    | Absoluter URL des erreichbaren Dienstes (ohne TLS).                      |
| ServiceInformation/Service/Versions/Version/WSDL/@Location        | (optional) Absoluter URL der WSDL-Beschreibung                           |

[<=]

#### TIP1-A\_4530 - Aufbau Dienst URLs

Die URLs der Dienste KÖNNEN herstellerspezifisch aufgebaut werden.

[<=]

#### 4.1.3.2 Durch Ereignisse ausgelöste Reaktionen

Keine.

#### 4.1.3.3 Interne TUCs, nicht durch Fachmodule nutzbar

Keine

#### 4.1.3.4 Interne TUCs, auch durch Fachmodule nutzbar

Da der Konnektor als Black-Box mit inkludierten Fachmodulen ohne erkennbare Innenschnittstellen spezifiziert wird, stellt der folgende TUC lediglich einen Mechanismus zur editoriellen Kopplung der Fachmodulspezifikationen mit der Konnektorspezifikation dar:

4.1.3.4.1 TUC\_KON\_041 „Einbringen der Endpunkthinformationen während der Bootup-Phase“

#### TIP1-A\_4531 - TUC\_KON\_041 „Einbringen der Endpunkthinformationen während der Bootup-Phase“

Der Dienstverzeichnisdienst des Konnektors MUSS es den Fachmodulen ermöglichen, die zum jeweiligen Fachmodul gehörenden Endpunkte während der Bootup-Phase des Konnektors in den Dienstverzeichnisdienst einzubringen.

**Tabelle 25: TAB\_KON\_519 - TUC\_KON\_041 „Einbringen der Endpunkthinformationen während der Bootup-Phase“**

| Element      | Beschreibung   |
|--------------|--|
| Name         | TUC_KON_041 „Einbringen der Endpunkthinformationen während der Bootup-Phase“   |
| Beschreibung | Fachmodule MÜSSEN ihre Endpunkthinformationen während der Bootup-Phase in den Dienstverzeichnisdienst einbringen können. |

|                                |   |
|--------------------------------|---|
| Auslöser und Vorbedingungen    | Keine   |
| Eingangsdaten                  | <ul style="list-style-type: none"> <li>serviceInformation (Ein XML-Dokument mit dem Wurzelement „ServiceInformation“ gemäß dem Schema „ServiceInformation.xsd“. Eine Beschreibung des Schemas befindet sich in TAB_KON_518.)</li> </ul> |
| Komponenten                    | Konnektor, Fachmodule   |
| Ausgangsdaten                  | <ul style="list-style-type: none"> <li>Keine</li> </ul>   |
| Standardablauf                 | Die übergebenen Serviceinformationen des Fachmoduls werden in die Gesamtstruktur „connector.sds“ aufgenommen.<br>Falls beim Speichern der eingebrachten Endpunkthinformationen ein Fehler auftritt, wird Fehler 4027 ausgelöst.         |
| Varianten/Alternativen         | Keine   |
| Fehlerfälle                    | 4027: Die Endpunkthinformationen konnten nicht übernommen werden.   |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

**Tabelle 26: TAB\_KON\_520 Fehlercodes TUC\_KON\_041 „Einbringen der Endpunkthinformationen während der Bootup-Phase“**

| Fehlercode | ErrorType | Severity | Fehlertext  |
|------------|-----------|----------|---|
| 4027       | Technical | Error    | Die Endpunkthinformationen konnten nicht übernommen werden. |

[<=]

#### 4.1.3.5 Operationen an der Außenschnittstelle

##### **TIP1-A\_4532 - Schnittstelle der Basisanwendung Dienstverzeichnisdienst**

Der Dienstverzeichnisdienst des Konnektors MUSS die in Tabelle TAB\_KON\_521 Schnittstelle der Basisanwendung Dienstverzeichnisdienst beschriebene Schnittstelle anbieten.

**Tabelle 27: TAB\_KON\_521 Schnittstelle der Basisanwendung Dienstverzeichnisdienst**

|                     |   |
|---------------------|---|
| <b>Dienstname</b>   | ConnectorServiceDirectory   |
| <b>Beschreibung</b> | Der Aufruf liefert Angaben über den Hersteller, über das Konnektormodell und die Liste der Dienste, Konnektordienstversionen (KDV) und Endpunkte der Dienste. |
| <b>Aufruf</b>       | GET /connector.sds HTTP/1.1<br>Host: ANLW_LAN_IP_ADDRESS oder MGM_KONN_HOSTNAME   |

|                        |  |  |
|------------------------|--|--|
| <b>Rückgabe</b>        | Das Antwortdokument ist in der Schemadatei <code>ServiceDirectory.xsd</code> beschrieben.  |  |
|                        | ConnectorServices  |  |
|                        |    |  |
|                        | <b>Name</b>  | <b>Beschreibung</b>                        |
|                        | ProductInformation   | Kurzbeschreibung des Konnektormodells      |
|                        | ServiceInformation   | Beschreibung der Dienste                   |
|                        | <b>ProductInformation:</b><br>Das Schema wird in TAB_KON_517 beschrieben. Die Felder sind gemäß [gemSpec_OM] zu befüllen und gemäß dem Schema „ <code>ProductInformation.xsd</code> “ zu formatieren.  |  |
|                        | <b>TLS-Mandatory:</b> Boolean Wert der festlegt, ob die Verwendung eines TLS-Kanals für Dienstaufrufe verpflichtend ist.<br>Definierende Variable ist: ANCL_TLS_MANDATORY<br><b>ClientAutMandatory:</b> Boolean Wert der festlegt, ob Client Authentifizierung verpflichtend ist.<br>Definierende Variable ist: ANCL_CAUT_MANDATORY. |  |
|                        | <b>ServiceInformation:</b><br>Das Schema wird in TAB_KON_518 beschrieben. Die Felder sind gemäß dem Schema <code>ServiceInformation.xsd</code> zu formatieren.<br>Falls (ANCL_CAUT_MANDATORY = Enabled) oder (ANCL_TLS_MANDATORY = Enabled), MUSS die Rückgabedatei ausschließlich https-Endpunkte enthalten.                        |  |
|                        | <b>Fehlercodes</b>   | Die Standard HTTP1.1 Fehlercodes [RFC2616] |
| <b>Vorbedingungen</b>  | Keine  |  |
| <b>Nachbedingungen</b> | Keine  |  |
| <b>Hinweise</b>        | Keine  |  |

[&lt;=]

#### 4.1.3.6 Betriebsaspekte

##### TIP1-A\_4533 - Dienstverzeichnisdienst initialisieren.

Mit Abschluss der Bootup-Phase MUSS der Dienstverzeichnisdienst an der Außenschnittstelle die vollständige Liste aller Services bereitstellen, die der

2389 Anwendungskonnektor den Clientsystemen anbietet, inklusive der Services der  
 2390 Fachmodule.  
 2391 [ $\leq$ ]

#### 2392 4.1.4 Kartenterminaldienst

2393 Die Aufgabe des Kartenterminaldienstes ist das Management aller vom Konnektor  
 2394 adressierbaren Kartenterminals. Dies umfasst alle administrativen Prozesse  
 2395 (insbesondere das Pairing, vgl. [gemSpec\_KT#2.5.2]). Ferner kapselt der  
 2396 Kartenterminaldienst die Zugriffe auf Kartenterminals durch Basisdienste und  
 2397 Fachmodule.

2398 Für die TLS-Verbindungen zu den Kartenterminals muss der Konnektor die Vorgaben aus  
 2399 [gemSpec\_Krypt#3.3.2] und hinsichtlich ECC-Migration die Vorgaben aus  
 2400 [gemSpec\_Krypt#5] befolgen.

2401 Innerhalb des Kartenterminaldienstes werden folgende Präfixe für Bezeichner verwendet:

- 2402 • Events (Topic Ebene 1): „CT“
- 2403 • Konfigurationsparameter: „CTM\_“

2404 Der Kartenterminaldienst verwaltet hinsichtlich der Kartenterminals mindestens die in der  
 2405 informativen Tabelle TAB\_KON\_522 Parameterübersicht des Kartenterminaldienstes  
 2406 ausgewiesenen Parameter, weitere herstellerspezifische Parameter sind möglich. Die  
 2407 normative Festlegung wann welche Parameter mit welchen Werten belegt werden, erfolgt  
 2408 in den folgenden Abschnitten und Unterkapiteln.

2409 Dabei beschrieben CTM\_xyz-Bezeichner Parameter, die den Dienst als Ganzes betreffen.  
 2410 Zu jedem Kartenterminal selbst werden dessen Parameter in einem CT-Object gekapselt.  
 2411 Die folgende Tabelle zeigt die Attribute der jeweiligen CT-Objekte über  
 2412 Punktschreibweise.

2413 **Tabelle 28: TAB\_KON\_522 Parameterübersicht des Kartenterminaldienstes**

| ReferenzID               | Belegung              | Zustandswerte  |
|--------------------------|-----------------------|--|
| CTM_CT_LIST              | Liste von CT-Objekten | Eine Liste von Repräsentanzen (CT-Objects) der dem Konnektor bekannten Kartenterminals.  |
| Pro CTM_CT_LIST Eintrag: |                       |  |
| Gerätekenndaten          |                       |  |
| CT.CTID                  | Identifizier          | Eindeutige, statische Identifikation des Kartenterminals   |
| CT.IS_PHYSICAL           | Ja/Nein               | Kennzeichnung, ob es sich um ein physisches oder logisches Kartenterminal handelt, zur Unterscheidung von eHealth-Kartenterminals und HSM-Bs. Da dieser Unterschied gemäß der aktuellen HSM-B-Lösung für den Konnektor transparent ist, wird der Parameter in dieser Spezifikation immer auf „Ja“ gesetzt. |

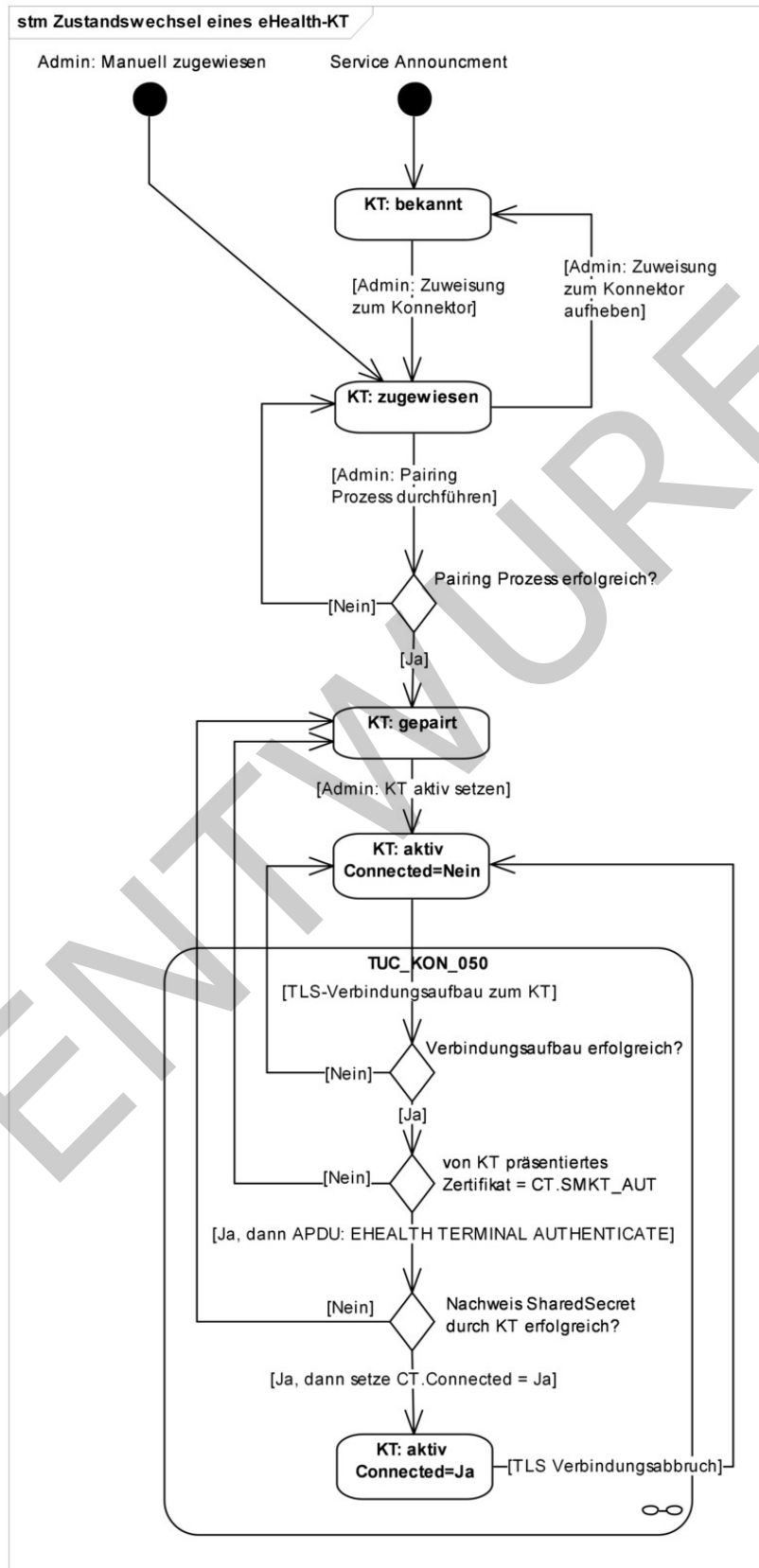


|                              |                                  |  |
|------------------------------|----------------------------------|--|
|                              |                                  | Der Parameterwert „Nein“ ist für zukünftige Nutzung vorgesehen.  |
| CT.MAC_ADRESS                | MAC-Adresse                      | Die MAC-Adresse des Kartenterminals  |
| CT.HOSTNAME                  | String                           | SICCT-Terminalname des Kartenterminals, auch als FriendlyName bezeichnet   |
| CT.IP_ADRESS                 | IP-Adresse                       | Die IP-Adresse des Kartenterminals   |
| CT.TCP_PORT                  | Portnummer                       | Der TCP-Port des SICCT-Kommandointerpreters des Kartenterminals  |
| CT.SLOT_COUNT                | Nummer                           | Anzahl der Slots des Kartenterminals   |
| CT.SLOTS_USED                | Liste                            | Liste der aktuell mit Karten belegten Slots  |
| CT.PRODUCT_INFORMATION       | Inhalt<br>ProductInformation.xsd | Die Herstellerinformationen zum Kartenterminal gemäß [gemSpec_OM]  |
| CT.EHEALTH_INTERFACE_VERSION | Version                          | Die EHEALTH-Interface-Version des Kartenterminals, die mittels GET STATUS aus dem Element VER des Discretionary Data Objects ermittelt wurde.  |
| CT.VALID_VERSION             | Boolean                          | True, wenn die Version des Kartenterminals (CT.EHEALTH_INTERFACE_VERSION) durch den Konnektor unterstützt wird, d.h. zu den in CTM_SUPPORTED_KT_VERSIONS passt<br>Default-Wert = false |
| CT.DISPLAY_CAPABILITIES      | Datenstruktur                    | Displayeigenschaften wie in der Struktur Display Capabilities Data Object in [SICCT#5.5.10.17] beschrieben   |
| Pairingdaten                 |                                  |  |
| CT.SMKT_AUT                  | X.509-Cert                       | C.SMKT.AUT-Zertifikat des Kartenterminals, gespeichert im Rahmen des Pairings  |
| CT.SHARED_SECRET             |                                  | ShS.KT.AUT, gespeichert im Rahmen des Pairings   |
| Verbindungsdaten             |                                  |  |
| CT.CORRELATION               | bekannt<br>zugewiesen<br>gepairt | Der Korrelationsstatus zum Konnektor:  |

|                      |                         |   |
|----------------------|-------------------------|---|
|                      | aktiv<br>aktualisierend | <ul style="list-style-type: none"> <li>• bekannt (über Service Announcement/Service Discovery gelernte Kartenterminals),</li> <li>• zugewiesen (durch den Administrator aus dem Bereich der bekannten Kartenterminals oder manuell konfigurierte Kartenterminals),</li> <li>• gepairt (Pairing erfolgreich aber noch nicht zum Verbindungsaufbau freigegeben)</li> <li>• aktiv (durch Administrator zum Verbindungsaufbau freigegeben),</li> <li>• aktualisierend (ein laufender Updatevorgang, ausgelöst durch den Konnektor; Der Zustand tritt ein, wenn der Kartenterminaldienst das Event „KSR/UPDATE/START“ fängt und endet mit dem Event „KSR/UPDATE/END“)</li> </ul> |
| CT.CONNECTED         | Ja/Nein                 | Der Verfügbarkeitsstatus des Kartenterminals (Ja = nach Aufbau der TLS-Verbindung und erfolgter zweiter Authentifizierung)  |
| CT.ACTIVEROLE        | User/Admin              | Benutzerrolle, die für die aktuelle Session verwendet wird  |
| KT-Admin-Credentials |                         |   |
| CT.ADMIN_USERNAME    | String                  | Username des Administrators am KT   |
| CT.ADMIN_PASSWORD    | String                  | Password des Administrators am KT   |

2414

2415 Zum besseren Verständnis sind die Zustände, die ein Kartenterminal einnehmen kann, im  
2416 nachfolgenden Zustandsdiagramm PIC\_KON\_071 dargestellt.



2417

2418 **Abbildung 7: PIC\_KON\_071 Korrelationszustände eines eHealth-KT**2419 **4.1.4.1 Funktionsmerkmalweite Aspekte**2420 **TIP1-A\_4534 - Kartenterminals nach eHealth-KT-Spezifikation**

2421 Der Kartenterminaldienst MUSS Kartenterminals nach der eHealth-Kartenterminal  
2422 Spezifikation [gemSpec\_KT] unterstützen.

2423 [ $\leq$ ]

2424 Zur Unterstützung von HSM-Bs benötigt der Konnektor virtuelle Kartenterminals  
2425 (CT.IS\_PHYSICAL=Nein), in denen virtuelle SMC-Bs „stecken“ können (siehe Kapitel  
2426 4.1.4). Diese Kartenterminals werden innerhalb des Zugriffsberechtigungsdienstes sowie  
2427 des Systeminformationsdienstes wie normale Kartenterminals berücksichtigt. Weitere  
2428 Details zu den logischen Kartenterminals finden sich im Kapitel Betriebsaspekte.

2429 **TIP1-A\_4535 - Unterstützung logischer Kartenterminals für HSMs**

2430 Der Kartenterminaldienst MUSS logische Kartenterminals mit logischen Slots  
2431 unterstützen. Zu jedem verwalteten HSM (siehe Kartendienst) MUSS der Konnektor ein  
2432 oder mehrere logische Kartenterminal mit folgenden Bedingungen vorhalten:

- 2433 • Jedes logische KT MUSS als CT-Object mit eindeutiger CTID in CTM\_CT\_LIST  
2434 enthalten sein
- 2435 • Die CT-Attribute MÜSSEN gemäß TAB\_KON\_522 Parameterübersicht des  
2436 Kartenterminaldienstes gesetzt werden.

2437 [ $\leq$ ]

2438 **TIP1-A\_4536 - TLS-Verbindung zu Kartenterminals halten**

2439 Der Kartenterminaldienst MUSS jede mit einem Kartenterminal etablierte Verbindung  
2440 durch Nutzung des in [SICCT#6.1.4.5] definierten Keep-Alive Mechanismus halten. Der  
2441 Konnektor MUSS für das Heartbeat-Intervall gemäß [SICCT#6.1.4.5] den Wert  
2442 CTM\_KEEP\_ALIVE\_INTERVAL verwenden und beim Ausbleiben von Terminal-Antworten  
2443 eines Kartenterminals nach der Anzahl von CTM\_KEEP\_ALIVE\_TRY\_COUNT Versuchen  
2444 die Netzwerkverbindung zu diesem Kartenterminal beenden.

2445 [ $\leq$ ]

2446 **TIP1-A\_6725 - Lebensdauer von Textanzeigen am Kartenterminal**

2447 Der Konnektor MUSS steuern, dass Textanzeigen am Kartenterminal nur so lange  
2448 angezeigt werden, wie sie im jeweiligen Anwendungskontext benötigt werden.

2449 [ $\leq$ ]

2450 Ziel der Textanzeigen am Kartenterminal ist die Kommunikation mit dem Benutzer zur  
2451 Unterstützung der Anwendungsfälle. Die Anzeige am Kartenterminal muss daher einen  
2452 engen zeitlichen Bezug zum jeweiligen Anwendungskontext haben.

2453 Nachrichten, deren Anwendungskontext mit einem Event beendet werden, wie etwa die  
2454 Aufforderung zum Stecken der Karte im Kontext von TUC\_KON\_056, deren inhaltliche  
2455 Berechtigung mit dem Stecken der Karte erlischt, (ebenso zum Ziehen der Karte im  
2456 Rahmen von TUC\_KON\_057) müssen sofort gelöscht werden, wenn das Event eintritt.

2457 Nachrichten, deren Lebensdauer nicht durch ein natürliches Event beendet wird, müssen  
2458 eine vordefinierte Lebensdauer erhalten, die per Konfiguration an die Bedürfnisse der  
2459 Leistungserbringer anpassbar sein sollte. Das gilt für Ergebnisanzeigen oder Anzeigen  
2460 von Fehlern.

### TIP1-A\_4537 - KT-Statusanpassung bei Beendigung oder Timeout einer Netzwerkverbindung

Tritt ein Timeout ein oder wird eine Netzwerkverbindung zu einem Kartenterminal (oder zu einem HSM, welches einem logischen Kartenterminal zugeordnet ist) beendet oder zurückgesetzt und ist CT.CONNECTED = Ja, so MUSS der Konnektor:

- CT.CONNECTED für das Kartenterminal auf „Nein“ setzen
- Für jeden in CT.SLOTS\_USED gelisteten Slot X zur weiteren internen Bearbeitung  
TUC\_KON\_256 {  
    topic = „CT/SLOT\_FREE“;  
    eventType = Op;  
    severity = Info;  
    parameters = („CtID=\$CT.CTID, SlotNo=\$X“);  
    doLog = false;  
    doDisp = false }  
    rufen
- TUC\_KON\_256 {  
    topic = „CT/DISCONNECTED“;  
    eventType = Op;  
    severity = Info;  
    parameters = („CtID=\$CT.CTID, Hostname=\$CT.HOSTNAME“) }  
    auslösen
- CT.SLOTS\_USED leeren

[<=]

### TIP1-A\_4538 - Wiederholter Verbindungsversuch zu den KTs

Sind in CTM\_CT\_LIST Kartenterminals mit CT.CONNECTED=Nein und CT.VALID\_VERSION = True und CT.CORRELATION = „aktiv“ und ist CTM\_SERVICE\_DISCOVERY\_CYCLE>0, MUSS der Konnektor im Zeitabstand CTM\_SERVICE\_DISCOVERY\_CYCLE-Minuten entweder eine Service Discovery-Nachricht an alle KTs als Broadcast oder an jedes einzelne dieser unverbundenen KTs als Unicast senden.

[<=]

### TIP1-A\_4538-02 - ab PTV4: Wiederholter Verbindungsversuch zu den KTs

Sind in CTM\_CT\_LIST Kartenterminals mit CT.CONNECTED=Nein und CT.VALID\_VERSION = True und CT.CORRELATION = „aktiv“ und ist CTM\_SERVICE\_DISCOVERY\_CYCLE>0, MUSS der Konnektor im Zeitabstand CTM\_SERVICE\_DISCOVERY\_CYCLE-Minuten an jedes einzelne dieser unverbundenen KTs eine Service-Discovery-Nachricht als Unicast senden.

[<=]

### TIP1-A\_6478 - Erlaubte SICCT-Kommandos bei CT.CONNECTED=Nein

Der Kartenterminaldienst DARF SICCT-bzw. EHEALTH-Kommandos NICHT an ein Kartenterminal senden, wenn für dieses Kartenterminal CT.CONNECTED=Nein gesetzt ist. Ausgenommen hiervon sind die in TAB\_KON\_785 gelisteten EHEALTH- bzw. SICCT-Kommandos.

[<=]

**Tabelle 29: TAB\_KON\_785 Erlaubte SICCT-Kommandos bei CT.CONNECTED=Nein**

| SICCT-Kommando           |
|--------------------------|
| SICCT CT INIT CT SESSION |
| SICCT CT CLOSE SESSION   |
| SICCT GET STATUS         |

|                               |
|-------------------------------|
| SICCT SET STATUS              |
| SICCT CT DOWNLOAD INIT        |
| SICCT CT DOWNLOAD DATA        |
| SICCT CT DOWNLOAD FINISH      |
| EHEALTH TERMINAL AUTHENTICATE |

#### TIP1-A\_4539 - Robuster Kartenterminaldienst

Das Ziehen einer Karte während einer Kartenaktion DARF NICHT dazu führen, dass das verwaltete Kartenterminal im Anschluss durch den Konnektor nicht weiter genutzt werden kann. Die entsprechende Ressource MUSS nach Erkennung der Fehlersituation freigegeben werden. Ein manuelles Eingreifen DARF NICHT erforderlich sein.

[<=]

#### TIP1-A\_5408 - Terminal-Anzeigen beim Anfordern und Auswerfen von Karten

Der Konnektor MUSS beim Anfordern und Auswerfen von Karten die folgenden Display-Nachrichten für die Anzeige im Kartenterminal verwenden, wenn der Aufrufer keine konkrete Display-Nachricht übergeben hat. Der Verweis auf den Kartenterminal-Slot SOLL in der Display-Nachricht entfallen, wenn es keine Slot-Auswahl am Kartenterminal gibt.

**Tabelle 30: TAB\_KON\_727 Terminalanzeigen beim Anfordern und Auswerfen von Karten**

| Kontext         | Kartentyp   | Terminal-Anzeige  |
|-----------------|---|---|
| Karte anfordern | EGK   | Bitte • 0x0BeGK • 0x0Bin<br>• 0x0BSLOT X • 0x0Bstecken      |
|                 | HBA,<br>HBAX,<br>HBA-qSig   | Bitte • 0x0BHBA • 0x0Bin<br>• 0x0BSLOT X • 0x0Bstecken      |
|                 | SMC-B   | Bitte • 0x0BSMC-B • 0x0Bin<br>• 0x0BSLOT X • 0x0Bstecken    |
|                 | sonstiger<br>Kartentyp<br>oder kein<br>explizit<br>angegebener<br>Kartentyp | Bitte • 0x0BKarte • 0x0Bin<br>• 0x0BSLOT X • 0x0Bstecken    |
| Karte auswerfen | EGK   | Bitte • 0x0BeGK • 0x0Baus<br>• 0x0BSLOT X • 0x0Bentnehmen   |
|                 | HBA,<br>HBAX,<br>HBA-qSig   | Bitte • 0x0BHBA • 0x0Baus<br>• 0x0BSLOT X • 0x0Bentnehmen   |
|                 | SMC-B   | Bitte • 0x0BSMC-B • 0x0Baus<br>• 0x0BSLOT X • 0x0Bentnehmen |
|                 | sonstiger<br>Kartentyp<br>oder kein<br>explizit                             | Bitte • 0x0BKarte • 0x0Bentnehmen                           |

|  |                       |  |
|--|-----------------------|--|
|  | angegebener Kartentyp |  |
|--|-----------------------|--|

2518 [ $\leq$ ]

#### 2519 4.1.4.2 Durch Ereignisse ausgelöste Reaktionen

##### 2520 TIP1-A\_4540 - Reaktion auf Dienstbeschreibungspakete

2521 Der Konnektor MUSS Service Announcement für das Auffinden von Kartenterminals  
2522 entsprechend [SICCT] und [gemSpec\_KT] unterstützen. Der Konnektor MUSS

2523 Dienstbeschreibungspakete auf UDP Port `CTM_SERVICE_ANNOUNCEMENT_PORT`

2524 entgegennehmen.

2525 Erhält er ein solches Dienstbeschreibungspaket, MUSS er

2526 • TUC\_KON\_054 mit Mode=AutoAdded und IP-Adresse; TCP-Port; MAC-Adresse;  
2527 Hostname aus dem Dienstbeschreibungspaket, aufrufen

2528 • für das mit der MAC-Adressen in CTM\_CT\_LIST korrelierende CT-Object, wenn  
2529 CT.CORRELATION > "bekannt" und CT.VALID\_VERSION = true ist,  
2530 TUC\_KON\_050 { ctId = CT.CtId; role = „User“ } aufrufen.

2531 [ $\leq$ ]

##### 2532 TIP1-A\_4541 - Reaktion auf KT-Slot-Ereignis – Karte eingesteckt

2533 Der Kartenterminaldienst MUSS auf SICCT-Ereignisnachrichten „Slot-Ereignis – Karte  
2534 eingesteckt“ ([SICCT#6.1.4.4], TAG ,84') wie folgt reagieren:

2535 • das meldende Kartenterminal CT in CTM\_CT\_LIST ermitteln,

2536 • den in der Ereignisnachricht benannten Slot (FU-Nummer) in CT.SLOTS\_USED  
2537 aufnehmen,

2538 • zur weiteren internen Bearbeitung rufe TUC\_KON\_256 {  
2539 topic = „CT/SLOT\_IN\_USE“;  
2540 eventType = Op;  
2541 severity = Info;  
2542 parameters = („CtID=\$CT.CTID,  
2543 SlotNo=<FU-Nummer aus Ereignisnachricht>„);  
2544 doLog = false;  
2545 doDisp = false } auf.

2546 [ $\leq$ ]

##### 2547 TIP1-A\_4542 - Reaktion auf KT-Slot-Ereignis – Karte entfernt

2548 Der Kartenterminaldienst MUSS auf SICCT-Ereignisnachrichten „Slot-Ereignis – Karte  
2549 entfernt“ ([SICCT#6.1.4.4], TAG ,85') wie folgt reagieren:

2550 • das meldende Kartenterminal CT in CTM\_CT\_LIST ermitteln,

2551 • den in der Ereignisnachricht benannten Slot (FU-Nummer) aus CT.SLOTS\_USED  
2552 entfernen,

2553 • zur weiteren internen Bearbeitung rufe TUC\_KON\_256 {  
2554 topic = „CT/SLOT\_FREE“;  
2555 eventType = Op;  
2556 severity = Info;  
2557 parameters = („CtID=\$CT.CTID,  
2558 SlotNo=<FU-Nummer aus Ereignisnachricht>„);  
2559 doLog = false;  
2560 doDisp = false }  
2561 auf.



2562 [**<=**]

2563 **TIP1-A\_4543 - KT-Statusanpassung bei Beginn eines Updatevorgangs**

2564 Tritt der Event "KSR/UPDATE/START" mit „Target=KT“ ein, MUSS der Konnektor:

- 2565 • Setze CT = CTM\_CT\_LIST(CTID-Parameter des Ereignisses)
- 2566 • CT.CORRELATION für das Kartenterminal merken und auf „aktualisierend“ setzen
- 2567 • Für jeden in CT.SLOTS\_USED gelisteten Slot X zur weiteren internen Bearbeitung
- 2568 TUC\_KON\_256 {
- 2569     topic = „CT/SLOT\_FREE“;
- 2570     eventType = Op;
- 2571     severity = Info;
- 2572     parameters = („CtID=\$CT.CTID, SlotNo=\$CT.SLOTS\_USED[X]“);
- 2573     doLog = false;
- 2574     doDisp = false
- 2575 } aufrufen

2576 [**<=**]

2577 **TIP1-A\_4544 - KT-Statusanpassung bei Ende eines Updatevorgangs**

2578 Tritt der Event "KSR/UPDATE/END" mit „Target=KT“ ein, MUSS der Konnektor:

- 2579 • Setze CT = CTM\_CT\_LIST(CTID-Parameter des Ereignisses)
- 2580 • CT.CORRELATION auf den beim „KSR/UPDATE/START“ gemerkten Wert setzen
- 2581 • Aktualisiere Gerätedaten durch Aufruf TUC\_KON\_055 „Befülle CT-Object“ {ctId =
- 2582 CTID}
- 2583 • Wenn CT.VALID\_VERSION = true, Rufe TUC\_KON\_050 „Beginne
- 2584 Kartenterminalsitzung“ {ctId = CTID; role = „User“}
- 2585 • Wenn CT.VALID\_VERSION = false und CT.CORRELATION = „aktiv“, setze
- 2586 CT.CORRELATION=„gepairt“

2587 [**<=**]

2588 **4.1.4.3 Interne TUCs, nicht durch Fachmodule nutzbar**

2589 *4.1.4.3.1 TUC\_KON\_050 „Beginne Kartenterminalsitzung“*

2590 **TIP1-A\_4545-03 - TUC\_KON\_050 „Beginne Kartenterminalsitzung“**

2591 Der Konnektor MUSS den technischen Use Case „Beginne Kartenterminalsitzung“ gemäß

2592 TUC\_KON\_050 umsetzen.

2593

2594 **Tabelle 31: TAB\_KON\_039 – TUC\_KON\_050 „Beginne Kartenterminalsitzung“**

| Element      | Beschreibung  |
|--------------|---|
| Name         | TUC_KON_050 „Beginne Kartenterminalsitzung“   |
| Beschreibung | TUC_KON_050 baut eine TLS-Verbindung vom Konnektor zum Kartenterminal auf und beginnt eine SICCT-Sitzung. Anschließend erfolgt die 2. Authentifizierung des Kartenterminals (Prüfung SharedSecret). |

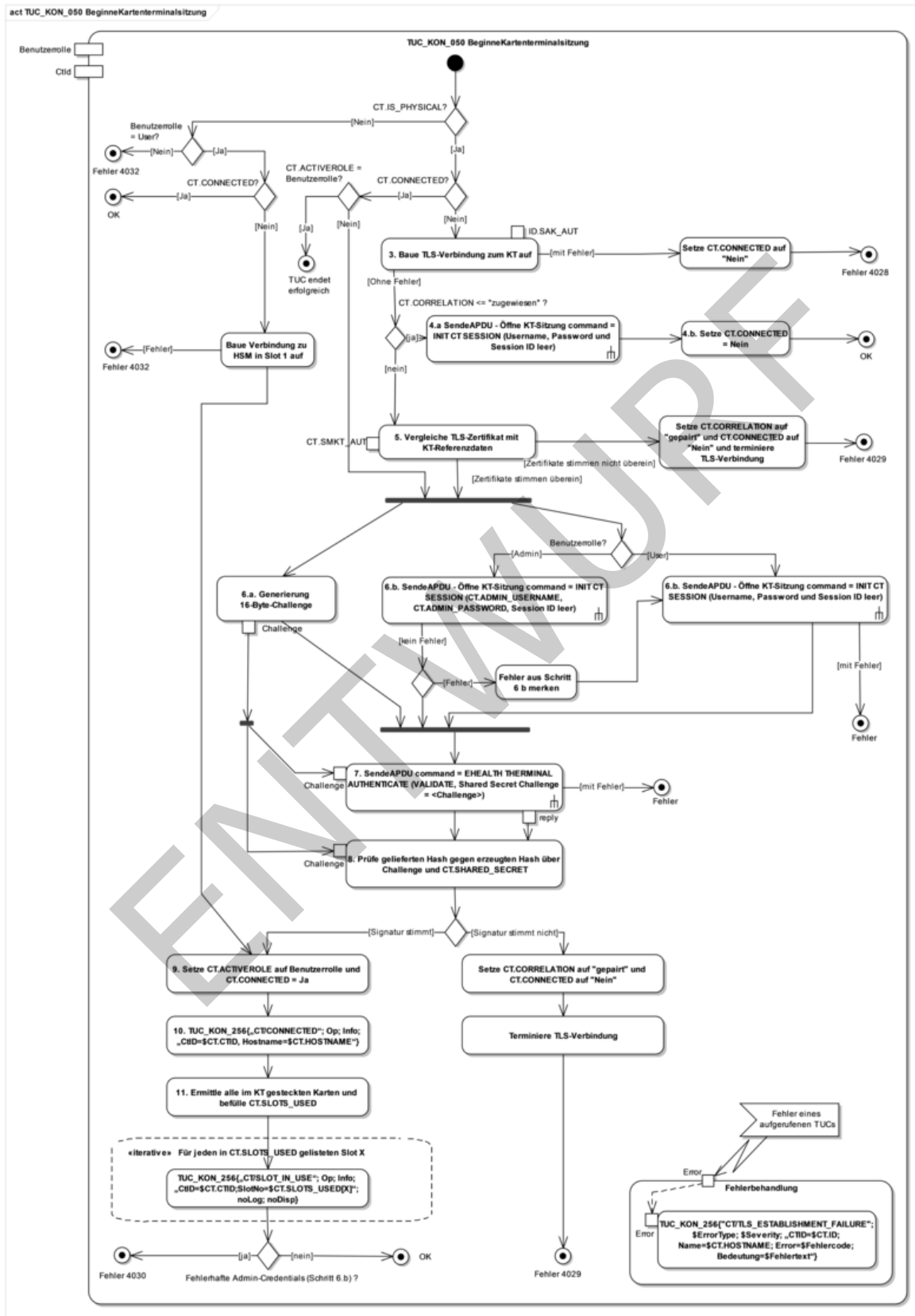
|                 |  |
|-----------------|--|
| Auslöser        | <ul style="list-style-type: none"> <li>• Neustart des Konnektors</li> <li>• nach dem Setzen eines Kartenterminals auf „aktiv“</li> <li>• im Rahmen eines erneuten Verbindungsversuchs</li> </ul>   |
| Vorbedingungen  | ctId ist in CTM_CT_LIST vorhanden  |
| Eingangsdaten   | <ul style="list-style-type: none"> <li>• ctId<br/>(zu verbindendes Kartenterminal)</li> <li>• role<br/>(Benutzerrolle; gültig sind: „User“ und „Admin“)</li> </ul>   |
| Komponenten     | Konnektor, Kartenterminal  |
| Ausgangsdaten   | keine  |
| Nachbedingungen | <ul style="list-style-type: none"> <li>• TLS-Kanal und SICCT-Session mit gewünschter Benutzerrolle aufgebaut, wenn CT.CORRELATION &gt;= "gepairt"</li> <li>• TLS-Kanal und SICCT-Session mit leerem Username, Password und Session ID aufgebaut, wenn CT.CORRELATION &lt;= „zugewiesen“</li> <li>• Steck-Ereignisse für alle im KT befindlichen Karten ausgelöst, wenn CT.CORRELATION &gt;= „gepairt“</li> </ul> |

|                |  |
|----------------|--|
| Standardablauf | <p>Setze CT = CTM_CT_LIST(ctId)</p> <ol style="list-style-type: none"> <li>1. Wenn CT.IS_PHYSICAL = Nein:<br/>prüfen, ob role = „User“<br/>Wenn CT.CONNECTED =<br/>Ja: TUC endet erfolgreich<br/>Nein:<br/>- Verbindung zu HSM in Slot 1 aufbauen<br/>- weiter mit Schritt 9</li> <li>2. Wenn CT.CONNECTED = Ja<br/>prüfen, ob CT.ACTIVEROLE = role<br/>Ja: TUC endet erfolgreich<br/>Nein:<br/>- Schließen der Cardterminal Session mit dem Kartenterminalkommando SICCT CLOSE CT SESSION,<br/>- weiter ab Schritt 6 (halten der TLS-Verbindung und<br/>nur Wechsel der Session)</li> <li>3. Aufbau einer TLS-Verbindung mit dem Kartenterminal unter Verwendung von ID.SAK.AUT. Dabei Prüfung des KT-Zertifikats mittels<br/>TUC_KON_037 {<br/>certificate= C.SMKT.AUT;<br/>qualifiedCheck=not_required;<br/>offlineAllowNoCheck=true;<br/>policyList= oid_smkt_aut;<br/>intendedKeyUsage= intendedKeyUsage(C.SMKT.AUT);<br/>intendedExtendedKeyUsage=id-kp-serverAuth;<br/>validationMode=NONE }</li> <li>4. Wenn CT.CORRELATION &lt;= „zugewiesen“:<br/>a. Öffne eine Cardterminal Session mit dem Kartenterminalkommando SICCT INIT CT SESSION (siehe [SICCT#5.10]) mit leerem Username, Password und Session ID<br/>b. Nur Verbindung in niedriger Korrelation, daher setze CT.CONNECTED = Nein, um fachliche Nutzung des KT zu verhindern<br/>c. beende TUC erfolgreich</li> <li>5. Prüfe, ob das Zertifikat aus der TLS-Verbindung mit den zum Kartenterminal gespeicherten Referenzdaten (CT.SMKT_AUT) übereinstimmt.<br/>a. Läuft das Zertifikat CT.SMKT_AUT (oder C.SMKT.AUT, sie müssen hier identisch sein), dann geht der Konnektor über in den Betriebszustand EC_CardTerminal_SMC-KT_Certificate_Expires_Soon (ctId).</li> <li>6. Parallelisierung<br/>a. Generierung eines zufälligen Werts (Challenge) mit mindestens 16 Byte Länge gemäß [gemSpec_Krypt#2.2] (siehe [gemSpec_KT#DO_KT_0004]),<br/>b. Öffnen einer Cardterminal Session mit dem Kartenterminalkommando SICCT INIT CT SESSION (siehe</li> </ol> |
|----------------|--|

|  |   |
|--|---|
|  | <p>[SICCT#5.10]) mit</p> <ul style="list-style-type: none"> <li>- ctId als Adressat</li> <li>- Wenn role = User<br/>dann mit leerem Username, Password und Session ID</li> <li>Wenn role = „Admin<br/>dann mit leerer Session ID aber mit CT.ADMIN_USERNAME und CT.ADMIN_PASSWORD</li> </ul> <p>7. Senden der Challenge mittels Kartenterminalkommando EHEALTH TERMINAL AUTHENTICATE (siehe [gemSpec_KT#3.7.2]) in der Ausprägung VALIDATE mit:</p> <ul style="list-style-type: none"> <li>- Kartenterminal als Empfänger</li> <li>- und mit der in Schritt 6a generierten Challenge im Shared Secret Challenge DO</li> </ul> <p>8. Prüfe Antwort des Kartenterminals, ob sie einen korrekten Hashwert über Challenge und angehängtes CT.SHARED_SECRET gemäß [gemSpec_KT#SEQ_KT_0002] Schritt 4-5 enthält</p> <p>9. Setze:</p> <ul style="list-style-type: none"> <li>a. CT.ACTIVEROLE = \$role</li> <li>b. CT.CONNECTED = Ja</li> </ul> <p>10. Wenn TLS-Verbindung neu aufgebaut werden musste, rufe TUC_KON_256 {<br/>topic = „CT/CONNECTED“;<br/>eventType = „Op“;<br/>severity = Info;<br/>parameters = („CtID=\$CT.CTID,<br/>Hostname=\$CT.HOSTNAME“) }</p> <p>11. Ermittle alle im KT gesteckten Karten und befülle entsprechend CT.SLOTS_USED</p> <p>Für jeden in CT.SLOTS_USED gelisteten Slot X zur weiteren internen Bearbeitung TUC_KON_256{<br/>topic = „CT/SLOT_IN_USE“;<br/>eventType = Op;<br/>severity = Info;<br/>parameters = („CtID=\$CT.CTID,<br/>SlotNo=\$CT.SLOTS_USED[X]“);<br/>doLog = false;<br/>doDisp = false }<br/>rufen.</p> |
|--|---|

|                                |  |
|--------------------------------|--|
| Varianten/<br>Alternativen     | Keine.   |
| Fehlerfälle                    | <p>Fehler in den folgenden Schritten des Ablaufs führen zu:<br/>         Aufruf von TUC_KON_256 {<br/>           topic = "CT/TLS_ESTABLISHMENT_FAILURE";<br/>           eventType = \$ErrorType;<br/>           severity = \$Severity;<br/>           parameters = („CtID=\$CT.ID, Name=\$CT.HOSTNAME,<br/>                           Error=\$Fehlercode, Bedeutung=\$Fehlertext“) }<br/>         Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes</p> <p>(→1): Admin-Rolle für logische KTs nicht möglich (hätte bei korrekter Implementierung nicht stattfinden dürfen), Fehlercode: 4032<br/>         (→1): Verbindungsaufbau zu HSM fehlgeschlagen, Fehlercode: 4032<br/>         (→3): Fehler im TLS-Verbindungsaufbau bzw. Zertifikatsprüfung, Fehlercode: 4028<br/>         und setze CT.CONNECTED auf „Nein“<br/>         (→3): TLS-Verbindung konnte nicht innerhalb von CTM_TLS_HS_TIMEOUT Sekunden aufgebaut werden , Fehlercode: 4028 und setze CT.CONNECTED auf „Nein“<br/>         (→5): Präsentiertes Zertifikat nicht das aus dem Pairing, Fehlercode: 4029<br/>         und setze CT.CORRELATION auf „gepairt“<br/>         und setze CT.CONNECTED auf „Nein“<br/>         und terminiere TLS-Verbindung<br/>         (→6b): Hinterlegte KT-Admin-Credentials fehlerhaft, Fehlercode: 4030<br/>         und in die User-Session zurückzuwechseln (damit das KT für den normalen Fachbetrieb weiterhin zur Verfügung steht)<br/>         (→8): Prüfung auf Nachweis SharedSecret fehlgeschlagen, Fehlercode 4029<br/>         und setze CT.CORRELATION auf „gepairt“<br/>         und setze CT.CONNECTED auf „Nein“<br/>         und terminiere TLS-Verbindung</p> |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | Abbildung PIC_KON_110 Aktivitätsdiagramm zu „Beginne Kartenterminalsitzung   |

2595  
2596



2597  
2598

Abbildung 8: PIC\_KON\_110 Aktivitätsdiagramm zu „Beginne Kartenterminalsitzung“

2599 **Tabelle 32: TAB\_KON\_523 Fehlercodes TUC\_KON\_050 „Beginne Kartenterminalsitzung“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4028  | Technical | Error    | Fehler beim Versuch eines Verbindungsaufbaus zu KT               |
| 4029  | Security  | Error    | Fehler bei der KT-Authentisierung. KT möglicherweise manipuliert |
| 4030  | Security  | Error    | Admin-Werte für KT fehlerhaft                                    |
| 4032  | Technical | Error    | Verbindung zu HSM konnte nicht aufgebaut werden                  |

2600  
2601  
2602  
2603 [≤]  
2604

2605 **4.1.4.3.2 TUC\_KON\_054 „Kartenterminal hinzufügen“**

2606 **TIP1-A\_4546 - TUC\_KON\_054 „Kartenterminal hinzufügen“**

2607 Der Konnektor MUSS den technischen Use Case TUC\_KON\_054 „Kartenterminal  
2608 hinzufügen“ umsetzen.

2609 **Tabelle 33: TAB\_KON\_524 – TUC\_KON\_054 „Kartenterminal hinzufügen“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_054 „Kartenterminal hinzufügen“   |
| Beschreibung   | Dieser TUC nimmt ein neues Kartenterminal in die zentrale Verwaltung auf (CTM_CT_LIST) oder aktualisiert die Einträge zu einem bereits erfassten Kartenterminal.  |
| Auslöser       | <ul style="list-style-type: none"> <li>• ein empfangenes Dienstbeschreibungspaket ohne vorheriges Service Discovery</li> <li>• manuelles Hinzufügen eines KT-Eintrags</li> <li>• ein empfangenes Dienstbeschreibungspaket nach vorherigem Auslösen eines manuellen Service Discovery</li> </ul> |
| Vorbedingungen | <ul style="list-style-type: none"> <li>• entweder ist das KT noch nicht in CTM_CT_LIST enthalten</li> <li>• oder das KT ist unter anderer IP/anderem Hostnamen schon gelistet</li> </ul>  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• Mode (AutoAdded, ManuallyAdded, ManuallyModified)</li> <li>• IP-Adresse (IPv4)</li> <li>• TCP-Port (optional)</li> <li>• MAC-Adresse (optional)</li> <li>• Hostname (optional)</li> </ul>  |



|                            |   |
|----------------------------|---|
| Komponenten                | Konnektor, Kartenterminal   |
| Ausgangsdaten              | <ul style="list-style-type: none"> <li>keine</li> </ul>   |
| Nachbedingungen            | <ul style="list-style-type: none"> <li>Das Kartenterminal ist mit allen Gerätekenndaten in CTM_CT_LIST vorhanden</li> </ul>   |
| Standardablauf             | <ol style="list-style-type: none"> <li>Sofern optionale Parameter nicht übergeben wurden oder Mode&lt;&gt;AutoAdded, ermittle Port, MAC und Hostname via Service<br/>Discovery als UDP/IP-Unicast an IP-Adresse und Port<br/>CTM_SERVICE_DISCOVERY_PORT</li> <li>Finde CT in CTM_CT_LIST über MAC-Adresse</li> <li>Wenn MAC-Adresse nicht in CTM_CT_LIST: <ol style="list-style-type: none"> <li>Erzeuge neuen CT-Object-Eintrag in CTM_CT_LIST und <ul style="list-style-type: none"> <li>Generiere eindeutige CT.CTID</li> <li>setze CT.MAC_ADRESS auf MAC-Adresse</li> <li>Setze CT.CORRELATION = „bekannt“</li> <li>Setze CT.CONNECTED = „Nein“</li> </ul> </li> <li>Wenn Mode= ManuallyAdded setze CT.CORRELATION = „zugewiesen“</li> </ol> </li> <li>Wenn CT.CONNECTED = Ja und (IP-Adresse &lt;&gt; CT.IP_ADRESS oder TCP-Port &lt;&gt; CT.TCP_PORT), beende TLS-Verbindung und setze CT.CONNECTED = „Nein“</li> <li>Befülle: CT.IP_ADRESS, CT.Hostname, CT.TCP_PORT</li> <li>Wenn CT.CORRELATION&gt;=„zugewiesen“ rufe TUC_KON_055 „Befülle CT-Object“</li> </ol> |
| Varianten/<br>Alternativen | Keine   |
| Fehlerfälle                | <p>Fehler im Ablauf (Standardablauf oder Varianten) führen in den folgend ausgewiesenen Schritten zu einem Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes:</p> <p>(→1) Keine Antwort innerhalb<br/>CTM_SERVICE_DISCOVERY_TIMEOUT,<br/>Fehlercode: 4033</p> <p>(→1) Ermittelte MAC-Adresse weicht von übergebener MAC-Adresse<br/>ab, Fehlercode: 4035</p> <p>(→1) Ermittelter Hostname-Adresse weicht von übergebenem Hostname<br/>ab, Fehlercode: 4036</p> <p>(→2) Wenn Mode=ManuallyModified und nicht gefunden,<br/>Fehlercode:<br/>4037</p> <p>Zusätzlich im Abbruchfall:</p> <ul style="list-style-type: none"> <li>Aufruf von TUC_KON_256 {<br/>topic = "CT/CT_ADDING_ERROR";<br/>eventType = \$ErrorType;<br/>severity = \$Severity;<br/>parameters = („IP=\$IP-Adresse, Name=\$HOSTNAME,<br/>Error=\$Fehlercode, Bedeutung=\$Fehlertext“) }</li> </ul>  |

|                                |                                    |
|--------------------------------|------------------------------------|
|                                | - Keine Veränderung an CTM_CT_LIST |
| Nichtfunktionale Anforderungen | Keine                              |
| Zugehörige Diagramme           | keine                              |

2610 **Tabelle 34: TAB\_KON\_525 Fehlercodes TUC\_KON\_054 „Kartenterminal hinzufügen“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4033  | Technical | Error    | Kartenterminal antwortet nicht, Zufügen fehlgeschlagen  |
| 4035  | Technical | Error    | Angegebener IP-Adresse gehört zu einer anderen MAC-Adresse als die, die übergeben wurde. Angaben zur MAC prüfen   |
| 4036  | Technical | Error    | Angegebener IP-Adresse gehört zu einem anderen Hostname als der, der übergeben wurde. Angaben zum Hostname prüfen |
| 4037  | Technical | Error    | Verwaltung der Kartenterminals inkonsistent   |

2611  
2612  
2613 **[<=]**

2614 **4.1.4.3.3 TUC\_KON\_053 „Paire Kartenterminal“**

2615 **TIP1-A\_4548-02 - TUC\_KON\_053 „Paire Kartenterminal“**

2616 Der Konnektor MUSS den technischen Use Case „Paire Kartenterminal“ gemäß  
2617 TUC\_KON\_053 umsetzen.

2618 **Tabelle 35: TAB\_KON\_041 – TUC\_KON\_053 „Paire Kartenterminal“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_053 „Paire Kartenterminal“   |
| Beschreibung   | TUC_KON_053 führt das Pairing zwischen dem Konnektor und einem eHealth-Kartenterminal durch.   |
| Auslöser       | Dialoge zur Administration des Konnektors. Der Administrator hat ein Kartenterminal im Dialog der Managementschnittstelle ausgewählt und das Pairing aufgerufen. |
| Vorbedingungen | <ul style="list-style-type: none"> <li>KT ist in CTM_CT_LIST vorhanden</li> <li>CT.CORRELATION = „zugewiesen“</li> <li>CT.IS_PHYSICAL = Ja</li> </ul>            |

|                     |  |
|---------------------|--|
| Eingangsdaten       | <ul style="list-style-type: none"> <li>ctId</li> </ul>   |
| Komponenten         | Konnektor, Kartenterminal  |
| Ausgangsdaten       | <ul style="list-style-type: none"> <li>Keine</li> </ul>  |
| Nachbedingunge<br>n | <ul style="list-style-type: none"> <li>CT.CORRELATION = „aktiv“, wenn Pairing erfolgreich</li> <li>CT.CORRELATION = „zugewiesen“, wenn Pairing nicht erfolgreich</li> <li>CT.CONNECTED = „Ja“, wenn Pairing erfolgreich</li> </ul>   |
| Standardablauf      | <p>Setze CT = CTM_CT_LIST(ctId)</p> <ol style="list-style-type: none"> <li>1. Prüfe CT.VALID_VERSION = true</li> <li>2. Aufbau einer TLS-Verbindung mit dem Kartenterminal unter Verwendung von ID.SAK.AUT. Dabei: <ol style="list-style-type: none"> <li>a. Speichern des präsentierten KT-Zertifikats in CT.SMKT_AUT</li> <li>b. Prüfung des KT-Zertifikats mittels TUC_KON_037{ <pre>certificate = C.SMKT.AUT; qualifiedCheck = not_required; offlineAllowNoCheck = true; policyList = oid _smkt_aut; intendedKeyUsage= intendedKeyUsage(C.SMKT.AUT)</pre> ; <pre>intendedExtendedKeyUsage = id-kp-serverAuth; validationMode = NONE }</pre> </li> </ol> </li> <li>3. Der Konnektor entnimmt den Fingerprint dem KT-Zertifikat und stellt dies dem Administrator im Dialog der Managementschnittstelle dar.<br/>Der Konnektor fordert den Administrator auf, den Fingerprint zu akzeptieren oder zurückzuweisen.</li> <li>4. Wenn der Administrator den Fingerprint bestätigt, <ol style="list-style-type: none"> <li>a. generiert der Konnektor einen neuen Schlüssel, das Shared Secret ShS.KT.AUT gemäß [gemSpec_Krypt#2.2] (siehe [gemSpec_KT#3.7]) und speichert es in CT.SHARED_SECRET</li> <li>b. und eröffnet der Konnektor mit dem Kartenterminalkommando SICCT INIT CT SESSION (siehe [SICCT#5.10]) mit <ul style="list-style-type: none"> <li>- ctId als Adressat</li> <li>- und mit leerem Username, Password und Session ID</li> </ul> eine Cardterminal Session.</li> </ol> </li> <li>5. Der Konnektor sendet mittels Kartenterminalkommando EHEALTH<br/> <pre>TERMINAL AUTHENTICATE</pre> (siehe [gemSpec_KT#3.7.2]) in der Ausprägung CREATE mit <ul style="list-style-type: none"> <li>- ctId als Empfänger</li> <li>- und mit dem in Schritt 4.a generierten Schlüssel im Shared Secret DO und der Display Message „KT:\$CT.MAC_ADRESS MIT</li> </ul> </li> </ol> |

|                            |   |
|----------------------------|---|
|                            | <p>KON:\$MGM_KONN_HOSTNAME PAIREN OK?", wobei die MAC-Adresse mit Trenner im folgenden Format dargestellt werden MUSS: „AABBCC:DDEEFF“</p> <p>das Shared Secret an das Kartenterminal.</p> <p>6. Der Konnektor prüft ob in der Antwort des Kartenterminals eine</p> <p>korrekte Signatur des Shared Secrets gemäß [gemSpec_KT#SEQ_KT_0001] Schritt 7, ausgeführt mit dem Schlüssel zum Zertifikat CT.SMKT_AUT vorliegt.</p> <p>7. CT.CORRELATION wird auf „gepairt“ gesetzt</p> <p>8. TLS-Verbindung, die zum Pairen diente, beenden und zuvor das Kartenterminalkommando SICCT CLOSE CT SESSION mit ctId als Adressat senden</p> <p>9. Automatischer Zustandsübergang CT.CORRELATION = „gepairt“ nach „aktiv“ (implizite Aktion des Administrators durch Aufruf von TUC_KON_053).</p> <p>10. „Arbeits“-TLS-Verbindung neu aufbauen durch Aufruf TUC_KON_050 { ctId; role = „User“}</p>   |
| Varianten/<br>Alternativen | <p>(→4): weist der Administrator den Fingerprint in Schritt 3 ab, wird TUC_KON_053 nach Ausführung folgender Aktivitäten beendet:</p> <p>4.1.a) Löschen von CT.SMKT_AUT</p> <p>4.1.b) Abbau der TLS-Verbindung, Setzen von CT.CONNECTED auf „Nein“</p>  |
| Fehlerfälle                | <p>Fehler in den folgenden Schritten des Ablaufs führen zu:</p> <p>a) Aufruf von TUC_KON_256 {<br/>topic = "CT/ERROR";<br/>eventType = \$ErrorType;<br/>severity = \$Severity;<br/>parameters = („CtId=\$ctId, Name=\$CT.HOSTNAME, Error=\$Fehlercode, Bedeutung=\$Fehlertext");<br/>doDisp = false }</p> <p>b) Löschen von CT.SMKT_AUT, CT.SHARED_SECRET</p> <p>c) Direkte Anzeige der Fehlermeldung für den Administrator</p> <p>d) Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes</p> <p>(→1) Version des KT wird nicht unterstützt, Fehlercode: 4042</p> <p>(→2b) Zertifikat ist zeitlich nicht gültig, Fehlercode: 1021 (CERTIFICATE_NOT_VALID_TIME)</p> <p>(→2) Fehler im TLS Verbindungsaufbau bzw. Zertifikatsprüfung, Fehlercode: 4040</p> <p>(→4b) Fehler in SICCT INIT CT SESSION, Fehlercode: 4041 mit Angabe des SICCT-Fehlers</p> <p>(→5) Fehler in EHEALTH TERMINAL AUTHENTICATE, Fehlercode: 4041 mit Angabe des SICCT-Fehlers</p> <p>(→6) Signaturprüfung fehlgeschlagen, Fehlercode: 4041</p> |

|                      |                   |
|----------------------|-------------------|
| Zugehörige Diagramme | Siehe PIC_KON_057 |
|----------------------|-------------------|

2619 **Tabelle 36: TAB\_KON\_113 Fehlercodes TUC\_KON\_053 „Paire Kartenterminal“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4040  | Security  | Error    | Fehler beim Versuch eines Verbindungsaufbaus zum KT  |
| 4041  | Technical | Error    | Fehler im Pairing, SICCT-Fehler <sup>(Nur wenn dieser Fehler wegen eines Fehlers auf der SICCT-Schnittstelle auftritt, ist der SICCT-Fehlercode mit anzugeben.)</sup> : <SICCT-Fehler> |
| 4042  | Technical | Error    | Die Version des Kartenterminals wird nicht unterstützt   |

2620 Hinweis zu Fehler 4041:

2621 Nur wenn dieser Fehler wegen eines Fehlers auf der SICCT-Schnittstelle auftritt, ist der SICCT-

2622 Fehlercode mit anzugeben.

```

graph TD
    Start(( )) --> 0[0. Setze CT=CTM_CT_LIST[C8D)]
    0 --> 1[1. Prüfe CT.VALID_VERSION=TRUE]
    1 --> Fehler4042((Fehler 4042))
    1 --> 2[2. Aufbau TLS-Verbindung zum KT mittels ID.SAK.AUT]
    2 --> Fehler4040((Fehler 4040))
    2 --> 2a[2.a Speicherung KT-Zertifikat in CT.SMKT.AUT]
    2a --> 2b[2.b Prüfung KT-Zertifikat (TUC_PKI_018)]
    2b --> FehlerTUC_PKI_018((Fehler aus TUC_PKI_018))
    2b --> 3[3. Fingerabdruck des Zertifikats anzeigen]
    3 -- "[Fingerabdruck durch Administrator zurückgewiesen]" --> FehlerTUCs[Fehler eines aufgerufenen TUCs]
    3 -- "[Fingerabdruck vom Administrator bestätigt]" --> 4a[4.a Shared Secret generieren u. speichern in CT.SHARED_SECRET]
    3 -- "[Fingerabdruck vom Administrator bestätigt]" --> 4b[4.b Eröffnung Card Terminal Session  
command = INIT CT SESSION (Username, Password und Session ID leer)]
    4a -- "secret, d. h. ShS.KT.AUT" --> 5[5. Übersendung Shared Secret an KT  
command = EHEALTH THERMINAL AUTHENTICATE (CREATE, Shared Secret = <secret>, ...)]
    4b -- "[ohne Fehler]" --> 5
    4b -- "[Fehler 4041]" --> Fehler4041_1((Fehler 4041))
    5 -- "Antwort" --> 6[6. KT Signatur des Shared Secret prüfen]
    6 -- "[Signatur stimmt nicht]" --> Fehler4041_2((Fehler 4041))
    6 -- "ShS.KT.AUT" --> 6
    6 --> 7[7. CT.CORRELATION auf "gepairt" setzen]
    7 --> 8[8. TLS-Pairing-Verbindung beenden]
    8 --> 9[9. CT.CORRELATION auf "aktiv" setzen]
    9 --> 10[10. Arbeits-TLS-Verbindung neu aufbauen (TUC_KON_050)]
    10 -- "Ctid, User" --> End(( ))
    10 --> 4c[4.1.c Abbau TLS-Verbindung, CT.CONNECTED="Nein"]
    4c --> 4b
    4b --> 4a1[4.1.a Löschen von CT.SMKT_AUT]
    4a1 --> Fehler4041_3((Fehler 4041))
    4a1 --> 4b
    FehlerTUCs --> Fehlerbehandlung[Fehlerbehandlung]
    Fehlerbehandlung --> TUC256[TUC_KON_256 ("CTERROR"; $ErrorType; $Severity; „Ctid=$Ctid; Name=$CT.HOSTNAME; Error=$Fehlercode; Bedeutung=$Fehlertext"; noDisp)]
    TUC256 --> Loeschen[Löschen von CT.SMKT_AUT, CT.FINGERPRINT, CT.SHARED_SECRET]
    Loeschen --> Anzeige[Direkte Anzeige Fehlermeldung für Administrator]
    Anzeige --> Abbruch[Abruch Verarbeitung mit ausgewiesenen Fehlercodes]
    Abbruch --> End

```

**[<=]**

2628 4.1.4.3.4 TUC\_KON\_055 „Befülle CT-Object“

2629 **TIP1-A\_4985 - TUC\_KON\_055 „Befülle CT-Object“**

2630 Der Konnektor MUSS den technischen Use Case TUC\_KON\_055 „Befülle CT-Object“  
2631 umsetzen.  
2632

2633 **Tabelle 37: TAB\_KON\_526 – TUC\_KON\_055 „Befülle CT-Object“**

| Element         | Beschreibung   |
|-----------------|--|
| Name            | TUC_KON_055 „Befülle CT-Object“  |
| Beschreibung    | Dieser TUC befüllt ein vorhandenes CT-Object aus CTM_CT_LIST mit den aktuellen Produktinformationen, die vom Kartenterminal bezogen werden und prüft, ob die Version des Kartenterminals unterstützt wird.   |
| Auslöser        | <ul style="list-style-type: none"> <li>• TUC_KON_054</li> <li>• Ereignis „KSR/UPDATE/END“ mit „Target=KT“</li> <li>• Verändern von CT.CORRELATION auf „zugewiesen“</li> <li>• Administratoraktion</li> </ul>   |
| Vorbedingungen  | <ul style="list-style-type: none"> <li>• ctId ist in CTM_CT_LIST vorhanden</li> </ul>  |
| Eingangsdaten   | <ul style="list-style-type: none"> <li>• ctId</li> </ul>   |
| Komponenten     | Konnektor, Kartenterminal  |
| Ausgangsdaten   | <ul style="list-style-type: none"> <li>• Supported (Boolean, True, wenn die Version des KT unterstützt wird)</li> </ul>  |
| Nachbedingungen | <ul style="list-style-type: none"> <li>• Die Gerätekenndaten des Kartenterminals in CTM_CT_LIST sind aktualisiert</li> </ul>   |
| Standardablauf  | <p>Setze CT = CTM_CT_LIST(ctId)</p> <ol style="list-style-type: none"> <li>1. Wenn CT.CONNECTED=Nein: Rufe TUC_KON_050 {<br/>ctId,<br/>role = „User“ }</li> <li>2. Folgende CT.Werte via SICCT GET STATUS ermitteln und befüllen: <ul style="list-style-type: none"> <li>• CT.SLOTCOUNT</li> <li>• CT.PRODUCTINFORMATION</li> <li>• CT.EHEALTH_INTERFACE_VERSION (Element VER aus Discretionary Data Data Object (DD DO))</li> <li>• CT.DISPLAY_CAPABILITIES (aus Display Capabilities Data Object in [SICCT#5.5.10.17])</li> </ul> </li> <li>3. Finde Eintrag in CTM_SUPPORTED_KT_VERSIONS anhand der ersten beiden Stellen (Haupt- und Nebenversionsnummer) aus CT.EHEALTH_INTERFACE_VERSION</li> </ol> <p><u>Eintrag gefunden:</u> Die dritte Stelle der KT-Version ist im Vergleich zur dritten Stelle im gefundenen</p> |





|                 |   |
|-----------------|---|
|                 | <ul style="list-style-type: none"> <li>• <i>displayMessage</i> – <i>optional/nicht erforderlich bei opmode= OutputErase, sonst mandatory</i> (Text zur Darstellung am KT, Länge durch KT begrenzt);</li> <li>• <i>opMode</i> [KtOutputMode] (Kommando-Modus)</li> <li>• <i>inputLength</i> – <i>optional/nur bei opMode=Input</i> (erwartete Eingabelänge, 0 für „beliebig“ lang)</li> <li>• <i>waitTimer</i> – <i>optional/nur bei opMode=OutputWait</i> (Wartezeit bis zur ersten Eingabe in Sekunden)</li> </ul>   |
| Komponenten     | Konnektor, Kartenterminal   |
| Ausgangsdaten   | <ul style="list-style-type: none"> <li>• <i>opResult</i> [OK   ABBRUCH ] – <i>optional/verpflichtend, wenn opMode=Input oder opMode=OutputConfirm</i> (Nutzertastendruck)</li> <li>• <i>inputData</i> – <i>optional/nur bei opMode = Input</i> (Zifferneingabe des Benutzers)</li> </ul>  |
| Nachbedingungen | Wenn Mode=OutputKeep bleibt Data auf dem Display des KT stehen  |
| Standardablauf  | <p>Setze CT = CTM_CT_LIST(ctId)</p> <ol style="list-style-type: none"> <li>1. <i>opMode</i>= <ol style="list-style-type: none"> <li>a. <u>Input:</u><br/>Der Konnektor MUSS via SICCT INPUT am CT zur Eingabe auffordern. Dabei MUSS die KT-Ansteuerung so erfolgen, dass: <ul style="list-style-type: none"> <li>• <i>displayMessage</i> zur Anzeige gebracht wird</li> <li>• maximal <i>inputLength</i> Ziffern akzeptiert werden. Bei <i>inputLength</i>=0 werden 1-n Zeichen akzeptiert (n=Maximalwert, definiert durch KT)</li> <li>• die eingegebenen Zeichen am Display angezeigt werden</li> <li>• die Eingabe explizit mit OK oder ABBRUCH beendet werden muss</li> </ul> </li> <li>b. <u>OutputWait:</u><br/>Der Konnektor MUSS via SICCT OUTPUT am CT <i>displayMessage</i> zur Anzeige bringen. Nach einer Wartezeit von <i>waitTimer</i> Sekunden MUSS der Konnektor die Anzeige des KT leeren.</li> <li>c. <u>OutputConfirm:</u><br/>Der Konnektor MUSS via SICCT INPUT am CT <i>displayMessage</i> zur Anzeige bringen und auf eine Bestätigung durch den Nutzer warten (zulässig: OK, ABBRUCH)</li> <li>d. <u>OutputKeep:</u><br/>Der Konnektor MUSS via SICCT OUTPUT am CT <i>displayMessage</i> zur Anzeige bringen. Die Anzeige</li> </ol> </li> </ol> |

|                                |   |
|--------------------------------|---|
|                                | bleibt erhalten, bis das KT neue Informationen am Display darstellen muss/soll.<br><br>e. <u>OutputErase</u> :<br>Der Konnektor MUSS via SICCT OUTPUT am CT die Anzeige leeren.   |
| Varianten/<br>Alternativen     | <ul style="list-style-type: none"> <li>Ist das Kartenterminal-Display durch einen anderen, zeitgleich im Konnektor ablaufenden Vorgang reserviert, so muss der Konnektor zunächst maximal 10 Sekunden lang versuchen, Zugriff auf das Display zu erhalten (und somit parallele Zugriffe auf das Display zu serialisieren). Erst nach Ablauf der Wartezeit wird Fehler 4039 geworfen.</li> </ul>   |
| Fehlerfälle                    | Fehler in den folgenden Schritten des Ablaufs führen zum Aufruf von TUC_KON_256 {<br>topic = "CT/ERROR";<br>eventType = \$ErrorType;<br>severity = \$Severity;<br>parameters = („CtID=\$ctId, Name=\$CT.HOSTNAME,<br>Error=\$Fehlercode, Bedeutung=\$Fehlertext")<br>}<br><br>(→1) Display und PinPad des Kartenterminals sind aktuell belegt (PIN, Eingabe, andere Ausgabe etc.), Fehlercode: 4039<br>(→1) Kartenterminal antwortet mit einer spezifischen Fehlermeldung, Fehlercode <gemäß [SICCT]> |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

**Tabelle 39: TAB\_KON\_114 Fehlercodes TUC\_KON\_051 „Mit Anwender über Kartenterminal interagieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4039  | Technical | Error    | Kartenterminal durch andere Nutzung aktuell belegt |

[<=]

#### 4.1.4.4.2 TUC\_KON\_056 „Karte anfordern“

##### **TIP1-A\_5409 - TUC\_KON\_056 „Karte anfordern“**

Der Konnektor MUSS den technischen Use Case „Karte anfordern“ gemäß TUC\_KON\_056 umsetzen.

2653 **Tabelle 40: TAB\_KON\_723 - TUC\_KON\_056 „Karte anfordern“**

| Element         | Beschreibung   |
|-----------------|--|
| Name            | TUC_KON_056 „Karte anfordern“  |
| Beschreibung    | Der TUC ermöglicht es, die Aufforderung zum Karte-Stecken an das Kartenterminal zu senden und dabei eine Meldung zum Anzeigen im Display des Kartenterminals mitzugeben.   |
| Auslöser        | Fachmodul im Konnektor oder Operation RequestCard ruft diesen Use Case auf.  |
| Vorbedingungen  |  |
| Eingangsdaten   | <ul style="list-style-type: none"> <li>- ctId<br/>(Kartenterminalidentifikator)</li> <li>- slotId<br/>(Nummer des Kartenslots)</li> <li>- cardType - <i>optional</i></li> <li>- displayMessage - <i>optional</i><br/>(Text zur Darstellung am Kartenterminal, Länge durch KT begrenzt)</li> <li>- timeOut<br/>(Wartezeit in Sekunden)</li> </ul>   |
| Komponenten     | Konnektor, Kartenterminal  |
| Ausgangsdaten   | <ul style="list-style-type: none"> <li>• cardObject<br/>(Informationsobjekt der Karte)</li> </ul>  |
| Nachbedingungen | Im Erfolgsfall enthält die CM_CARD_LIST ein neues CARD-Objekt des geforderten Typs.  |
| Standardablauf  | <p>Setze CT = CTM_CT_LIST(ctId)</p> <ol style="list-style-type: none"> <li>Falls displayMessage nicht explizit angegeben ist, MUSS sie gemäß Anforderung [TIP1-A_5408] erstellt werden.</li> <li>Der Konnektor MUSS das Kommando SICCT REQUEST ICC an das Kartenterminal CT senden. Die verfügbaren Optionen (Optical Signal, Acoustic Signal) können herstellerspezifisch sein bzw. über die Konfigurationsschnittstelle des Konnektors eingestellt werden. displayMessage wird als Eingabeaufforderung mitgegeben. Der übergebene timeOut-Wert wird dem SICCT-Kommando als Parameter übergeben.</li> </ol> |

|                            |   |
|----------------------------|---|
|                            | <p>3. Falls die Karte noch nicht gesteckt war, wird durch das Stecken der Karte das Ereignis „Karte gesteckt“ ausgelöst, worauf der Konnektor reagiert [TIP1-A_4563].</p> <p>4. Die Verarbeitung wird fortgesetzt, wenn eines der Ereignisse eingetreten ist:</p> <ol style="list-style-type: none"> <li>SICCT REQUEST ICC kehrt mit '6201' zurück (eine aktivierte Karte steckte bereits)</li> <li>SICCT REQUEST ICC kehrt mit '9000' oder '9001' zurück und das Ereignis "Karte gesteckt" wurde gemäß [TIP1-A_4563] verarbeitet</li> <li>SICCT REQUEST ICC kehrt mit '9000' oder '9001' zurück und das Ereignis "Karte gesteckt" wurde nicht empfangen (eine deaktivierte Karte steckte bereits), die Karte wurde durch<br/> Rufe TUC_KON_001 {<br/> ctId; slotId }<br/> geöffnet.</li> </ol> <p>In allen Fällen liegt in CM_CARD_LIST ein neues CARD-Objekt vor.</p> <p>5. Falls cardType angegeben ist, wird nach erfolgreicher Ausführung von SICCT REQUEST ICC der AID des MF der gesteckten Karte gelesen und mit dem gewünschten Kartentyp in cardType verglichen. Bei fehlender Übereinstimmung wird der Ablauf mit dem Fehler 4051 abgebrochen.</p> <p>6. Es wird cardObject (ein Informationsobjekt der Karte, die sich in dem Slot mit der Nummer slotId befindet) zurückgegeben.</p> |
| Varianten/<br>Alternativen | Die Ausgabe einer Display-Nachricht entfällt, wenn der adressierte Slot bereits eine gesteckte Karte enthält.   |
| Fehlerfälle                | <p>Fehler in den folgenden Schritten des Ablaufs führen zu Aufruf von TUC_KON_256 {<br/> topic = "CT/ERROR";<br/> eventType = \$ErrorType;<br/> severity = \$Severity;<br/> parameters = („CtID=\$ctId, Name=\$CT.HOSTNAME,<br/> Error=\$Fehlercode,<br/> Bedeutung=\$Fehlertext“) }</p> <p>(→2) Display des Kartenterminals ist aktuell belegt, Fehlercode: 4039<br/> (→2) Fehler beim Zugriff auf das Kartenterminal, Fehlercode: 4044<br/> (→2) Ungültige Kartenterminal-ID: Fehlercode: 4007<br/> (→2) Ungültige Kartenslot-ID: Fehlercode: 4097<br/> (→2) Kartenterminal nicht aktiv, Fehlercode: 4221<br/> (→2) Kartenterminal ist nicht verbunden, Fehlercode: 4222<br/> (→2) Kartenterminal antwortet mit einer spezifischen</p>  |

|                                |   |
|--------------------------------|---|
|                                | Fehlermeldung,<br>Fehlercode <gemäß [SICCT]><br>(→4) Timeout. Es wurde keine Karte innerhalb der angegebenen Zeitspanne gesteckt, Fehlercode: 4202<br>(→5) Falscher Kartentyp, Fehlercode: 4051 |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

2654 **Tabelle 41: TAB\_KON\_724 Fehlercodes TUC\_KON\_056 „Karte anfordern“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4039  | Technical | Error    | Kartenterminal durch andere Nutzung aktuell belegt                           |
| 4044  | Technical | Error    | Fehler beim Zugriff auf das Kartenterminal                                   |
| 4051  | Technical | Error    | Falscher Kartentyp   |
| 4007  | Technical | Error    | Ungültige Kartenterminal-ID  |
| 4097  | Technical | Error    | Ungültige Kartenslot-ID  |
| 4202  | Technical | Error    | Timeout. Es wurde keine Karte innerhalb der angegebenen Zeitspanne gesteckt. |
| 4221  | Technical | Error    | Kartenterminal nicht aktiv   |
| 4222  | Technical | Error    | Kartenterminal ist nicht verbunden   |

2655  
2656 [ $\leq$ ]

2657 4.1.4.4.3 TUC\_KON\_057 „Karte auswerfen“

2658 **TIP1-A\_5410 - TUC\_KON\_057 „Karte auswerfen“**

2659 Der Konnektor MUSS den technischen Use Case „Karte auswerfen“ gemäß TUC\_KON\_057  
2660 umsetzen.

2661  
2662 **Tabelle 42: TAB\_KON\_725 – TUC\_KON\_057 „Karte auswerfen“**

| Element      | Beschreibung   |
|--------------|--|
| Name         | TUC_KON_057 „Karte auswerfen“  |
| Beschreibung | Der TUC ermöglicht es, das SICCT-Kommando zum Auswerfen der Karte an das Kartenterminal zu senden und dabei eine Meldung zum Anzeigen im Display des Kartenterminals |

|                            |   |
|----------------------------|---|
|                            | mitzugeben, die den Benutzer zum Entnehmen der Karte auffordert.  |
| Auslöser                   | Fachmodul im Konnektor oder Operation EjectCard ruft diesen Use Case auf.   |
| Vorbedingungen             |   |
| Eingangsdaten              | <ul style="list-style-type: none"> <li>• ctId<br/>(Kartenterminalidentifikator)</li> <li>• slotId<br/>(Nummer des Kartenslots)</li> <li>• displayMessage – <i>optional</i><br/>(Text zur Darstellung am Kartenterminal, Länge durch KT begrenzt)</li> <li>• timeOut<br/>(Wartezeit in Sekunden)</li> </ul>  |
| Komponenten                | Konnektor, Kartenterminal   |
| Ausgangsdaten              | keine   |
| Nachbedingungen            | Durch das Entfernen der Karte wird das Ereignis „Karte entfernt“ ausgelöst, worauf der Konnektor reagiert [TIP1-A_4562].  |
| Standardablauf             | <p>Setze CT = CTM_CT_LIST(ctId)</p> <ol style="list-style-type: none"> <li>1. Der Konnektor prüft, dass entweder die Karte nicht reserviert ist oder der Aufrufer im Besitz des Karten-Locks ist.</li> <li>2. Falls displayMessage nicht explizit angegeben ist, MUSS sie gemäß Anforderung [TIP1-A_5408] erstellt werden.</li> <li>3. Der Konnektor MUSS das Kommando <code>SICCT EJECT ICC</code> an das Kartenterminal CT senden. Der Aufruf MUSS mit der Option „Delivery: Mechanical Throwout“ erfolgen. Die anderen Optionen (Optical Signal, Acoustic Signal) können herstellerspezifisch eingestellt werden bzw. können über die Konfigurationsschnittstelle des Konnektors eingestellt werden. Der übergebene Wert timeOut wird dem SICCT-Kommando als Parameter übergeben.</li> </ol> |
| Varianten/<br>Alternativen | Auch im Falle, dass nach der internen Buchführung des Konnektors in dem angegebenen Slot des Kartenterminals keine Karte steckt, MUSS der Konnektor das SICCT-Kommando <code>SICCT EJECT ICC</code> an das Kartenterminal senden. Meldet das Kartenterminal keinen Fehler, so meldet auch der Konnektor keinen Fehler und es kann davon ausgegangen werden, dass sich keine Karte mehr in dem Slot befindet.  |
| Fehlerfälle                | Fehler in den folgenden Schritten des Ablaufs führen zu Aufruf von TUC_KON_256 {<br>topic = "CT/ERROR";<br>eventType = \$ErrorType;   |

|                                |   |
|--------------------------------|---|
|                                | <pre>severity = \$Severity; parameters = („CtID=\$CtID, Name=\$CT.HOSTNAME,               Error=\$Fehlercode,               Bedeutung=\$Fehlertext“) }</pre> <p>(→1) Die Karte ist fremdreserviert, Fehlercode 4093<br/> (→3) Display des Kartenterminals ist aktuell belegt, Fehlercode: 4039<br/> (→3) Fehler beim Zugriff auf das Kartenterminal, Fehlercode: 4044<br/> (→3) Karte deaktiviert, aber nicht entnommen, Fehlercode: 4203<br/> (→3) Ungültige Kartenterminal-ID: Fehlercode: 4007<br/> (→3) Ungültige Kartenslot-ID: Fehlercode: 4097<br/> (→3) Kartenterminal nicht aktiv, Fehlercode: 4221<br/> (→3) Kartenterminal ist nicht verbunden, Fehlercode: 4222<br/> (→3) Kartenterminal antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;gemäß [SICCT]&gt;</p> |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

2663 **Tabelle 43: TAB\_KON\_796 Fehlercodes TUC\_KON\_057 „Karte auswerfen“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4039  | Technical | Error    | Kartenterminal durch andere Nutzung aktuell belegt           |
| 4044  | Technical | Error    | Fehler beim Zugriff auf das Kartenterminal                   |
| 4203  | Technical | Error    | Karte deaktiviert, aber nicht entnommen                      |
| 4093  | Technical | Error    | Karte wird in einer anderen Kartensitzung exklusiv verwendet |
| 4007  | Technical | Error    | Ungültige Kartenterminal-ID                                  |
| 4097  | Technical | Error    | Ungültige Kartenslot-ID                                      |
| 4221  | Technical | Error    | Kartenterminal nicht aktiv                                   |
| 4222  | Technical | Error    | Kartenterminal ist nicht verbunden                           |

2664  
2665 **[<=]**

2666 **4.1.4.4.4 TUC\_KON\_058 „Displaygröße ermitteln“**

2667 **A\_17473 - TUC\_KON\_058 „Displaygröße ermitteln“**

2668 Der Konnektor MUSS den technischen Use Case „Displaygröße ermitteln“ gemäß  
2669 TUC\_KON\_058 umsetzen.

2670

**Tabelle 44: TAB\_KON\_854 – TUC\_KON\_058 „Displaygröße ermitteln“**

| Element                        | Beschreibung  |
|--------------------------------|---|
| Name                           | TUC_KON_058 „Displaygröße ermitteln“  |
| Beschreibung                   | Der TUC liefert den Inhalt der Variable CT.DISPLAY_CAPABILITIES zurück.   |
| Auslöser                       | Fachmodul im Konnektor ruft diesen Use Case auf.  |
| Vorbedingungen                 |   |
| Eingangsdaten                  | <ul style="list-style-type: none"> <li>ctId<br/>(Kartenterminalidentifikator)</li> </ul>  |
| Komponenten                    | Konnektor   |
| Ausgangsdaten                  | CT.DISPLAY_CAPABILITIES   |
| Nachbedingungen                | Keine   |
| Standardablauf                 | Setze CT = CTM_CT_LIST(ctId) <ol style="list-style-type: none"> <li>Der Konnektor prüft, ob CT.DISPLAY_CAPABILITIES belegt ist.</li> <li>Falls CT.DISPLAY_CAPABILITIES belegt ist, wird der Inhalt der Variable zurückgegeben.</li> </ol>   |
| Varianten/<br>Alternativen     | Keine   |
| Fehlerfälle                    | Fehler in den folgenden Schritten des Ablaufs führen zu Aufruf von TUC_KON_256 { <pre> topic = "CT/ERROR"; eventType = \$ErrorType; severity = \$Severity; parameters = („CtID=\$CtID, Name=\$CT.HOSTNAME,                Error=\$Fehlercode,                Bedeutung=\$Fehlertext") } (→2) CT.DISPLAY_CAPABILITIES ist nicht belegt, Fehlercode 4254 </pre> |
| Nichtfunktionale Anforderungen | Keine   |



|                      |       |
|----------------------|-------|
| Zugehörige Diagramme | Keine |
|----------------------|-------|

2671 **Tabelle 45: TAB\_KON\_855 Fehlercodes TUC\_KON\_058 „Displaygröße ermitteln“**

| Fehlercode | ErrorType | Severity | Fehlertext  |
|------------|-----------|----------|---|
| 4254       | Technical | Error    | Keine Displaygröße für das Kartenterminal definiert |

2672

2673

2674 [ $\leq$ ]

#### 2675 4.1.4.5 Operationen an der Außenschnittstelle

##### 2676 **TIP1-A\_5411-02**~~TIP1-A\_5411~~ - Basisdienst Kartenterminaldienst

2677 Der Konnektor MUSS Clientsystemen den Basisdienst Kartenterminaldienst anbieten.

2678

2679

2680 **Tabelle 46: TAB\_KON\_722 Basisdienst Kartenterminaldienst**

|                          |  |                         |
|--------------------------|--|-------------------------|
| <b>Name</b>              | CardTerminalService  |                         |
| <b>Version (KDV)</b>     | Siehe <del>Anhang D</del> 1.1.0 (WSDL-Version) 1.1.2 (XSD-Version) |                         |
| <b>Namensraum</b>        | Siehe <del>Anhang D</del> GitHub                                   |                         |
| <b>Namensraum-Kürzel</b> | CT für Schema und CTW für WSDL                                     |                         |
| <b>Operationen</b>       | <b>Name</b>  | <b>Kurzbeschreibung</b> |
|                          | RequestCard  | Karte anfordern         |
|                          | EjectCard  | Karte auswerfen         |
| <b>WSDL</b>              | CardTerminalService.wsdl   |                         |
| <b>Schema</b>            | CardTerminalService.xsd  |                         |

2681

2682 [ $\leq$ ]

##### 2683 4.1.4.5.1 RequestCard

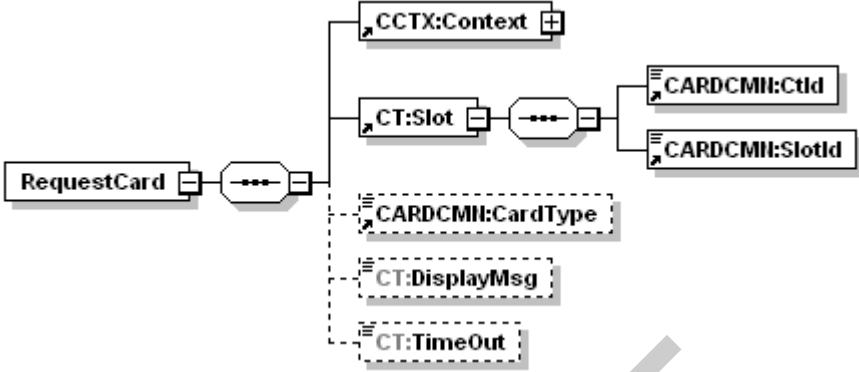
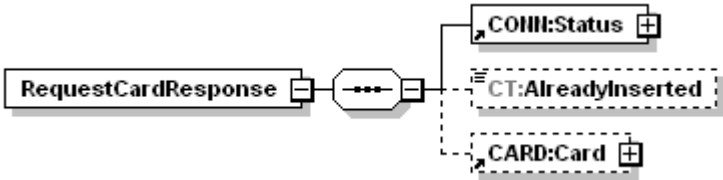
##### 2684 **TIP1-A\_5412 - Operation RequestCard**

2685 Der Konnektor MUSS an der Außenschnittstelle eine Operation RequestCard, wie in  
2686 Tabelle TAB\_KON\_716 Operation RequestCard beschrieben, anbieten.

2687

2688 **Tabelle 47: TAB\_KON\_716 Operation RequestCard**

|                     |  |
|---------------------|--|
| <b>Name</b>         | RequestCard  |
| <b>Beschreibung</b> | Liefert die Information zu einer Karte, die in dem Slot eines Kartenterminals steckt oder innerhalb einer bestimmten Zeit (Timeout) gesteckt wird. |

|                              |  |  |
|------------------------------|--|--|
| <b>Aufruf-<br/>parameter</b> |    |  |
|                              | Name   | Beschreibung   |
|                              | CCTX:Context   | MandantId, CsId, WorkplaceId verpflichtend   |
|                              | CT:Slot  | Adressiert den Slot eines Kartenterminals über die Identifikation des Kartenterminal CARDCMN:CtId und die Nummer des Slots CARDCMN:SlotId  |
|                              | CARDCMN:CardType   | Ein Kartentyp aus {EGK, KVK, HBAX, SM-B} als optionaler Filter. Wenn angegeben, werden nur Karten vom spezifizierten Typ zurückgegeben.  |
|                              | CT:DisplayMsg  | Diese Nachricht wird am Display des Kartenterminals angezeigt, um den Nutzer zum Stecken der Karte aufzufordern.   |
|                              | CT:TimeOut   | Die Zeit in sec, die maximal gewartet wird bis zum Stecken einer Karte. Wird dieser Parameter nicht übergeben, SOLL der Konnektor den Wert 20 sec verwenden. Optional KANN dieser Default-Wert im Konnektor konfigurierbar sein. |
| <b>Rückgabe</b>              |  |  |
|                              | Name   | Beschreibung   |
|                              | CONN:Status  | Enthält den Ausführungsstatus der Operation (siehe 3.5.2)  |
|                              | CT:AlreadyInserted   | Dieses optionale Flag gibt an, ob die Karte bereits vor der Anfrage steckte (Wert true) oder erst auf Anforderung dieses Aufrufs gesteckt wurde (Wert false oder Element nicht vorhanden).                                       |

|                      |           |   |
|----------------------|-----------|---|
|                      | CARD:Card | Falls eine Karte gesteckt ist, werden Information zur Karte zurückgegeben (siehe 4.1.6.5.2) |
| <b>Vorbedingung</b>  | keine     |   |
| <b>Nachbedingung</b> | keine     |   |

2689 Der Ablauf der Operation RequestCard ist in Tabelle TAB\_KON\_717 Ablauf RequestCard  
 2690 beschrieben.  
 2691

2692 **Tabelle 48: TAB\_KON\_717 Ablauf RequestCard**

| Nr. | Aufruf<br>Technischer Use<br>Case oder<br>Interne<br>Operation | Beschreibung   |
|-----|--|--|
| 1.  | checkArguments   | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab. Wird die Operation für einen nicht unterstützten Kartentypen aufgerufen, so bricht die Operation mit Fehler 4058 ab.  |
| 2.  | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“               | Prüfung der Zugriffsberechtigung durch den Aufruf<br>TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientSystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>ctId = \$Slot.CtId;<br>needCardSession=false;<br>allWorkplaces=false }<br>Tritt bei der Prüfung ein Fehler auf, so bricht die Operation mit dem Fehlercode aus TUC_KON_000 ab. |
| 3.  | TUC_KON_056<br>„Karte anfordern“                               | Anforderung der Karte vom Kartenterminal durch Aufruf<br>TUC_KON_056(<br>ctId = \$Slot.CtId;<br>slotId = \$Slot.SlotId;<br>\$cardType;<br>displayMessage = \$DisplayMsg;<br>\$timeout)   |

2693 **Tabelle 49: TAB\_KON\_718 Fehlercodes „RequestCard“**

| Fehlercode  | ErrorType | Severity | Fehlertext            |
|---|-----------|----------|-----------------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |                       |
| 4000  | Technical | Error    | Syntaxfehler          |
| 4058  | Security  | Error    | Aufruf nicht zulässig |

2694  
2695  
2696 [ $\leq$ ]

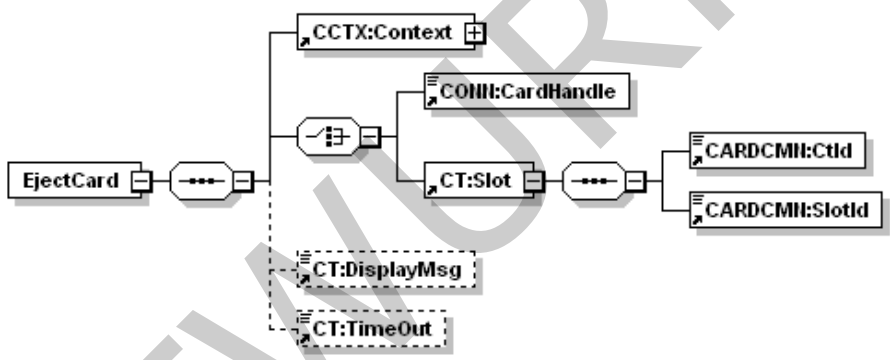
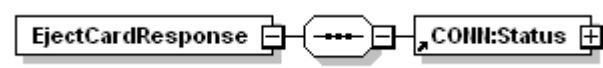
2697 4.1.4.5.2 EjectCard

2698 **TIP1-A\_5413 - Operation EjectCard**

2699 Der Konnektor MUSS an der Außenschnittstelle eine Operation EjectCard, wie in Tabelle  
2700 TAB\_KON\_719 Operation EjectCard beschrieben, anbieten.

2701

2702 **Tabelle 50: TAB\_KON\_719 Operation EjectCard**

| Name                 | EjectCard   |   |
|----------------------|---|---|
| Beschreibung         | Beendet die Kommunikation mit der Karte und wirft sie aus, falls das Kartenterminal eine solche mechanische Funktion hat. |   |
| Aufruf-<br>parameter |                                        |   |
|                      | Name  | Beschreibung  |
|                      | Context   | MandantId, CsId, WorkplaceId verpflichtend  |
|                      | CONN:<br>CardHandle   | Adressiert die Karte, die ausgeworfen werden soll.  |
|                      | CT:Slot   | Adressiert alternativ den Slot eines Kartenterminals, aus dem die Karte ausgeworfen werden soll. Die Adressierung erfolgt über die Identifikation des Kartenterminal <code>CARDCMN:CtId</code> und die Nummer des Slots <code>CARDCMN:SlotId</code> . |
|                      | CT:<br>DisplayMsg   | Diese Nachricht wird am Display des Kartenterminals angezeigt, um den Nutzer zum entnehmen der Karte aufzufordern.  |
|                      | CT:TimeOut  | Die Zeit in msec, die maximal gewartet wird bis eine Karte gezogen ist. Wird dieser Parameter nicht übergeben, SOLL der Konnektor den Wert 20 sec verwenden. Optional KANN dieser Default-Wert im Konnektor konfigurierbar sein.                      |
| Rückgabe             |                                       |   |

|                      |        |   |
|----------------------|--------|---|
|                      | Name   | Beschreibung  |
|                      | Status | Enthält den Ausführungsstatus der Operation (siehe 3.5.2) |
| <b>Vorbedingung</b>  | keine. |   |
| <b>Nachbedingung</b> | keine. |   |

2703 Der Ablauf der Operation EjectCard ist in Tabelle TAB\_KON\_720 Ablauf EjectCard  
 2704 beschrieben.  
 2705

2706 **Tabelle 51: TAB\_KON\_720 Ablauf EjectCard**

| Nr. | Aufruf Technischer Use Case oder Interne Operation | Beschreibung   |
|-----|--|--|
| 1.  | checkArguments                                     | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.   |
| 2.  | TUC_KON_000 „Prüfe Zugriffsberechtigung“           | Ist \$cardHandle vorgegeben, so wird \$ctId als Id des Kartenterminals bestimmt, in dem die Karte steckt. Prüfung der Zugriffsberechtigung durch den Aufruf TUC_KON_000 {<br>mandantId = \$Context.MandantId;<br>clientSystemId = \$Context.ClientSystemId;<br>workplaceId = \$Context.WorkplaceId;<br>ctId = \$Slot.CtId<br>bzw. ctId =<br>CM_CARD_LIST(\$CardHandle).CTID;<br>needCardSession = false;<br>allWorkplaces = false }<br>Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab. |
| 3.  | TUC_KON_057 „Karte auswerfen“                      | Wurde EjectCard mit dem Parameter Slot aufgerufen: Veranlassen des Kartenauswurfs am Kartenterminal durch Aufruf TUC_KON_057 {<br>ctId = \$Slot.CtId;<br>slotId = \$Slot.Slotid;<br>displayMessage = \$DisplayMsg;<br>\$timeOut }<br>Wurde EjectCard mit dem Parameter CardHandle aufgerufen: Veranlassen des Kartenauswurfs am Kartenterminal durch Aufruf TUC_KON_057 {<br>ctId = CM_CARD_LIST(\$CardHandle).CTID;<br>slotId = CM_CARD_LIST (\$CardHandle).SLOTNO; ;<br>displayMessage = \$DisplayMsg;<br>\$timeOut }              |

2707 **Tabelle 52: TAB\_KON\_721 Fehlercodes Operation „EjectCard“**

| Fehlercode  | ErrorType | Severity | Fehlertext                              |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4000  | Technical | Error    | Syntaxfehler                            |
| 4203  | Technical | Error    | Karte deaktiviert, aber nicht entnommen |
| 4101  | Technical | Error    | Karten-Handle ungültig                  |

2708

2709

2710 [ $\leq$ ]

#### 2711 4.1.4.6 Betriebsaspekte

##### 2712 4.1.4.6.1 Allgemeine Betriebsaspekte

##### 2713 **TIP1-A\_4549 - Initialisierung Kartenterminaldienst**

2714 Während des Bootvorgangs, nach dem Einlesen der persistierten Informationen des  
2715 Kartenterminaldienstes MUSS der Konnektor für jedes Kartenterminal CT in  
2716 CTM\_CT\_LIST:

- 2717 • die zugehörigen Attribute CT.SLOTS\_USED, CT.VALID\_VERSION und ggf. (bei  
2718 dynamischer Adressvergabe) CT.IP\_ADRESS aktualisieren
- 2719 • für jedes CT mit CT.CORRELATION = „aktiv“:
  - 2720 • Wenn CT.VALID\_VERSION = True: TUC\_KON\_050 „Beginne  
2721 Kartenterminalsitzung“ {ctId=CT.CtID; role=„User“} aufrufen
  - 2722 • Wenn CT.VALID\_VERSION = False: CT.CORRELATION=„gepairt“  
2723 setzen

2724 [ $\leq$ ]

2725 Hinweis: Bei der Initialisierung des Kartenterminaldienstes liest der Konnektor noch nicht  
2726 die Karten, um zu ermitteln, welche Karten gesteckt sind. Dies erfolgt erst bei  
2727 Initialisierung des Kartendienstes.

##### 2728 **TIP1-A\_4550 - Konfigurationsparameter des Kartenterminaldienstes**

2729 Die Managementschnittstelle MUSS es einem Administrator ermöglichen  
2730 Konfigurationsänderungen gemäß Tabelle TAB\_KON\_527 vorzunehmen:  
2731

2732 **Tabelle 53: TAB\_KON\_527 Konfigurationswerte eines Kartenterminalobjekts**

| ReferenzID                     | Belegung   | Bedeutung und Administrator-Interaktion   |
|--------------------------------|------------|---|
| CTM_SERVICE_DISCO<br>VERY_PORT | Portnummer | Der Administrator MUSS die Portnummer eingeben können, auf der die KTs im lokalen Netz auf Dienstanfragen hören.<br>Default-Wert=4742 |

|                               |                     |  |
|-------------------------------|---------------------|--|
| CTM_SERVICE_DISCOVERY_TIMEOUT | X Sekunden          | Der Administrator MUSS die Anzahl Sekunden eingeben können, die der Konnektor auf Antworten zu Service-Discovery-Anfragen wartet.<br>Default-Wert=3  |
| CTM_SERVICE_ANNOUNCEMENT_PORT | Portnummer          | Der Administrator MUSS die Portnummer eingeben können, auf der der Konnektor auf Dienstbeschreibungspakete hört.<br>Default-Wert=4742  |
| CTM_SERVICE_DISCOVERY_CYCLE   | X Minuten           | Der Administrator MUSS die Anzahl Minuten einstellen können, in denen der Konnektor wiederholt Service Discovery Nachrichten absetzt.<br>Default-Wert=10,<br>0=Deaktiviert   |
| CTM_KEEP_ALIVE_INTERVAL       | X Sekunden          | Intervall in Sekunden in den Keep-Alive-Nachrichten an das Kartenterminal gesendet werden<br>Der Administrator MUSS diesen Wert im vorgegebenen Bereich anpassen können.<br>Wertebereich: 1-10<br>Default-Wert=10  |
| CTM_KEEP_ALIVE_TRY_COUNT      | Anzahl der Versuche | Anzahl von aufeinander folgenden, nicht beantworteten Keep-Alive-Nachrichten, nach denen ein Timeout der TLS-Verbindung festgestellt wird<br>Der Administrator MUSS diesen Wert im vorgegebenen Bereich anpassen können.<br>Wertebereich: 3-10<br>Default-Wert=3 |
| CTM_TLS_HS_TIMEOUT            | X Sekunden          | Der Administrator MUSS die Anzahl Sekunden eingeben können, die der Konnektor auf den TLS-Verbindungsaufbau zum Kartenterminal wartet (Handshake-Timeout).<br>Wertebereich: 1-60<br>Default-Wert=10  |

2733

2734 [ $\leq$ ]2735 **TIP1-A\_4986 - Informationsparameter des Kartenterminaldienstes**

2736 Die Managementschnittstelle MUSS es einem Administrator ermöglichen die  
 2737 Informationsparameter gemäß Tabelle TAB\_KON\_528 einzusehen:

2738

2739 **Tabelle 54: TAB\_KON\_528 Informationsparameter des Kartenterminaldienstes**

| ReferenzID                | Belegung           | Bedeutung und Administrator-Interaktion   |
|---------------------------|--------------------|---|
| CTM_SUPPORTED_KT_VERSIONS | Liste von EHEALTH- | Der Administrator MUSS die Liste der vom Konnektor unterstützten modellunabhängigen |

|  |                     |  |
|--|---------------------|--|
|  | Interface-Versionen | EHEALTH-Interface-Versionen einsehen können. |
|--|---------------------|--|

2740

2741 [ $\leq$ ]

## 2742 4.1.4.6.2 Kartenterminals pflegen

2743 Im Folgenden werden die Administratorinteraktionen beschrieben, die zum Hinzufügen,  
 2744 Pairen, Bearbeiten und Löschen von Kartenterminals innerhalb der CTM\_CT\_LIST  
 2745 angeboten werden müssen. Eine Aktualisierung der Kartenterminals mit neuer Firmware  
 2746 wird in Kapitel 4.3.9 beschrieben.

2747 **TIP1-A\_4551 - Einsichtnahme von Kartenterminaleinträgen**

2748 Die Managementschnittstelle MUSS es einem Administrator ermöglichen die Liste der  
 2749 verwalteten und neu entdeckten Kartenterminals einzusehen (CTM\_CT\_LIST).

2750 [ $\leq$ ]2751 **TIP1-A\_4552 - Manueller Verbindungsversuch zu Kartenterminals**

2752 Die Managementschnittstelle MUSS es einem Administrator ermöglichen zu jedem CT-  
 2753 Object-Eintrag in CTM\_CT\_LIST mit CT.CONNECTED=Nein und CT.CORRELATION=aktiv  
 2754 einen manuellen Verbindungsaufbau über TUC\_KON\_050 {ctId=CtID; role=„User“}  
 2755 auszulösen.

2756 [ $\leq$ ]2757 **TIP1-A\_4553 - Einsichtnahme in und Aktualisierung der Kartenterminaleinträge**

2758 Die Managementschnittstelle MUSS es einem Administrator ermöglichen zu jedem CT-  
 2759 Object-Eintrag in CTM\_CT\_LIST die Werte gemäß Tabelle TAB\_KON\_529 einsehen zu  
 2760 können:

2761 Zu jedem Eintrag MUSS der Administrator TUC\_KON\_055 „Befülle CT-Object“ auslösen  
 2762 können.

2763

2764 **Tabelle 55: TAB\_KON\_529 Anzeigewerte zu einem Kartenterminalobjekt**

| ReferenzID          | Belegung     | Bedeutung und Administrator-Interaktion  |
|---------------------|--------------|--|
| Geräte<br>kenndaten |              |  |
| CT.CTID             | Identifizier | Eindeutige, statische Identifikation des Kartenterminals   |
| CT.IS_PHYSICAL      | Ja/Nein      | Kennzeichnung, ob es sich um ein logisches oder physisches Kartenterminal handelt (siehe auch TAB_KON_522 Parameterübersicht des Kartenterminaldienstes) |
| CT.MAC_ADRESS       | MAC-Adresse  | die MAC-Adresse des Kartenterminals  |
| CT.HOSTNAME         | String       | SICCT-Terminalname des Kartenterminals, auch als FriendlyName bezeichnet   |
| CT.IP_ADRESS        | IP-Adresse   | die IP-Adresse des Kartenterminals   |
| CT.TCP_PORT         | Portnummer   | der TCP-Port des SICCT-Kommandointerpreters des Kartenterminals  |



|                                      |   |  |
|--------------------------------------|---|--|
| CT.SLOTCOUNT                         | Nummer  | Anzahl der Slots des Kartenterminals   |
| CT.SLOTS_USED                        | Liste   | Liste der mit Karten belegten Slots  |
| CT.PRODUCT<br>INFORMATION            | Inhalt Product<br>Information.xsd                           | die Herstellerinformationen zum<br>Kartenterminal gemäß [gemSpec_OM]   |
| CT.EHEALTH_<br>INTERFACE_<br>VERSION | Version   | Die EHEALTH-Interface-Version des<br>Kartenterminals, die mittels des SICCT-<br>Kommandos GET STATUS aus dem Element<br>VER des Discretionary Data Objects ermittelt<br>wurde  |
| CT.VALID_<br>VERSION                 | Boolean   | True, wenn die Version des Kartenterminals<br>(CT.EHEALTH_INTERFACE_VERSION) durch<br>den Konnektor unterstützt wird, d.h. zu den<br>in CTM_SUPPORTED_KT_VERSIONS passt  |
| Pairingdaten                         |   |  |
| CT.SMKT_AUT                          | X.509-Cert  | C.SMKT.AUT-Zertifikat des Kartenterminals,<br>gespeichert im Rahmen des Pairings   |
| Verbindungs<br>daten                 |   |  |
| CT.<br>CORRELATION                   | bekannt<br>zugewiesen<br>gepairt<br>aktiv<br>aktualisierend | Der Korrelationsstatus zum Konnektor: <ul style="list-style-type: none"> <li>• bekannt (über Service Announcement/Service Discovery gelernte Kartenterminals),</li> <li>• zugewiesen (durch den Administrator aus dem Bereich der bekannten Kartenterminals oder manuell konfigurierte Kartenterminals),</li> <li>• gepairt (Pairing erfolgreich aber noch nicht zum Verbindungsaufbau freigegeben)</li> <li>• aktiv (durch Administrator zum Verbindungsaufbau freigegeben),</li> <li>• aktualisierend (ein laufender Updatevorgang, ausgelöst durch den Konnektor; Der Zustand tritt ein, wenn der Kartenterminaldienst das Event „KSR/UPDATE/START“ fängt und endet mit dem Event „KSR/UPDATE/END“),</li> </ul> |
| CT.CONNECTED                         | Ja/Nein   | Der Verfügbarkeitsstatus des<br>Kartenterminals (Ja = nach Aufbau der TLS-<br>Verbindung und erfolgter zweiter<br>Authentifizierung)   |
| CT.ACTIVEROLE                        | User/Admin  | Benutzerrolle, die für die aktuelle Session<br>verwendet wird  |

|                      |        |                                   |
|----------------------|--------|-----------------------------------|
| KT-Admin-Credentials |        |                                   |
| CT.ADMIN_USERNAME    | String | Username des Administrators am KT |

[<=]

#### TIP1-A\_4554 - Bearbeitung von Kartenterminaleinträgen

Die Managementschnittstelle MUSS es einem Administrator ermöglichen zu jedem CT-Object-Eintrag in CTM\_CT\_LIST die Werte gemäß Tabelle TAB\_KON\_530 ändern zu können:

Zur Überprüfung der veränderten Parameter auf Korrektheit MUSS nach Änderung von CT.IP\_ADRESS, TCP\_PORT oder HOSTNAME TUC\_KON\_054 mit Mode= ManuallyModified und allen vorhandenen CT-Parametern aufgerufen werden. Endet der Aufruf von TUC\_KON\_054 mit einem Fehler, MUSS der Konnektor die geänderten Konfigurationswerte auf ihren Ausgangswert zurücksetzen.

**Tabelle 56: TAB\_KON\_530 Konfigurationswerte eines Kartenterminalobjekts**

| ReferenzID        | Belegung   | Bedeutung und Administrator-Interaktion   |
|-------------------|------------|---|
| CT.IP_ADRESS      | IP-Adresse | Der Administrator MUSS für KTs mit CT.IS_PHYSICAL=Ja die IPv4-Adresse des Kartenterminals eingeben können.                            |
| CT.TCP_PORT       | Portnummer | Der Administrator MUSS für KTs mit CT.IS_PHYSICAL=Ja den TCP-Port des SICCT-Kommandointerpreters des Kartenterminals eingeben können. |
| CT.HOSTNAME       | String     | Der Administrator MUSS den SICCT-Terminalnamen (Hostname) - auch als FriendlyName bezeichnet - des Kartenterminals eingeben können.   |
| CT.ADMIN_USERNAME | String     | Der Administrator MUSS für KTs mit CT.IS_PHYSICAL=Ja den Username des KT-Administrators des Kartenterminals eingeben können.          |
| CT.ADMIN_PASSWORD | String     | Der Administrator MUSS für KTs mit CT.IS_PHYSICAL=Ja das Password des KT-Administrators des Kartenterminals eingeben können.          |

[<=]

#### TIP1-A\_6477 - Manuelles Service Discovery

Die Managementschnittstelle MUSS es einem Administrator ermöglichen, ein Service Discovery entsprechend [SICCT] auszulösen, um neue Kartenterminals hinzuzufügen.

[<=]

#### TIP1-A\_4555 - Manuelles Hinzufügen eines Kartenterminals

Die Managementschnittstelle MUSS es einem Administrator ermöglichen für neue Kartenterminals CT-Objects manuell in CTM\_CT\_LIST aufzunehmen.

2787 Hierzu MUSS der Administrator für das neue Kartenterminal folgende Werte eingeben  
2788 können:

- 2789 • IP-Adresse (Eingabe verpflichtend)
- 2790 • TCP-Port (Eingabe optional)
- 2791 • MAC-Adresse (Eingabe optional)
- 2792 • Hostname (Eingabe optional)

2793 Bestätigt der Administrator seine Eingaben, MUSS TUC\_KON\_054 mit  
2794 Mode=ManuallyAdded und allen eingegebenen Parametern aufgerufen werden.  
2795 [**<=**]

2796 Als Sicherung gegen den unbemerkten Austausch von Kartenterminals oder deren  
2797 Identitäten wird das gSMC-KT über den Konnektor logisch an das eHealth-Kartenterminal  
2798 gebunden. Dieser Vorgang wird als Pairing von Kartenterminal und gSMC-KT bezeichnet  
2799 und ist ausführlich in [gemSpec\_KT] beschrieben.

## 2800 **TIP1-A\_4556 - Pairing mit Kartenterminal durchführen**

2801 Die Managementschnittstelle MUSS es einem Administrator ermöglichen alle  
2802 Kartenterminals mit CT.CORRELATION = „zugewiesen“ in einer Liste einzusehen und für  
2803 einen ausgewählten Eintrag mit CT.VALID\_VERSION=True TUC\_KON\_053 auslösen zu  
2804 können.  
2805 [**<=**]

## 2806 **TIP1-A\_4557 - Ändern der Korrelationswerte eines Kartenterminals**

2807 Die Managementschnittstelle MUSS es einem Administrator ermöglichen zu einem  
2808 Kartenterminal aus CTM\_CT\_LIST für KT's mit CT.IS\_PHYSICAL=Ja den Wert für  
2809 CT.CORRELATION nach folgenden Mustern zu ändern:

- 2810 • CT.CORRELATION = „bekannt“  
2811 Das Kartenterminal gilt als nicht durch den Konnektor verwaltet.
- 2812 • → „zugewiesen“:  
2813 Ein (per Service Announcement entdecktes) Kartenterminal dem Konnektor  
2814 zuweisen.  
2815 Folgende Schritte MUSS der Konnektor für diesen Zustandswechsel zuvor  
2816 erfolgreich durchlaufen:  
2817 - Rufe TUC\_KON\_055 „Befülle CT-Object“  
2818 - Prüfen, ob CT.HOSTNAME bereits für ein anderes  
2819 Kartenterminal in CTM\_CT\_LIST verwendet wird. Wenn ja  
2820 MUSS dieser Änderungsversuch fehlschlagen (Prinzip der  
2821 Eindeutigkeit verletzt). Der Administrator MUSS eine  
2822 entsprechende Fehlermeldung erhalten.
- 2823 • CT.CORRELATION = „zugewiesen“  
2824 Das Kartenterminal gilt als durch den Konnektor verwaltet.
- 2825 • → „bekannt“
- 2826 • → „gepairt“:  
2827 Das Pairing wurde erfolgreich durchgeführt; die Werte  
2828 CT.SMKT\_AUT, CT.SHARED\_SECRET sind im CT-Objekt  
2829 eingetragen.
- 2830 • CT.CORRELATION = „gepairt“  
2831 Verbundenheit zwischen Kartenterminal und gesteckter gSMC-KT wurde  
2832 nachgewiesen

- 2833 • → „zugewiesen“:
- 2834 Die Werte CT.SMKT\_AUT, CT.SHARED\_SECRET werden gelöscht
- 2835 • → „aktiv“:
- 2836 Wechsel nur möglich, wenn CT.VALID\_VERSION=True. Dann Aufruf
- 2837 von TUC\_KON\_050 „Beginne Kartenterminalsitzung“ {ctId=CT.CtID;
- 2838 role=„User“}
- 2839 • CT.CORRELATION = „aktiv“
- 2840 Das Kartenterminal kann fachlich genutzt werden
- 2841 • → „gepairt“:
- 2842 Eventuelle TLS-Verbindung wird beendet, CT.CONNECTED auf Nein
- 2843 gesetzt.

2844 [ $\leq$ ]

#### 2845 **TIP1-A\_5698 - Löschen von Kartenterminaleinträgen**

2846 Die Managementschnittstelle MUSS einem Administrator die Möglichkeit bieten,  
2847 Kartenterminals aus der Liste der Kartenterminals (CTM\_CT\_LIST) zu entfernen.  
2848 [ $\leq$ ]

#### 2849 *4.1.4.6.3 Import der Kartenterminal-Informationen*

2850 Im Rahmen des Konnektormanagements müssen die Konfigurationsdaten des Konnektors  
2851 ex- und importiert werden können (siehe Kapitel 4.3.3). Eine Sonderstellung nimmt  
2852 dabei der Import von Kartenterminalinformationen ein, da hier im Rahmen des Imports  
2853 folgende Interaktion mit dem Administrator erforderlich ist:

2854

#### 2855 **TIP1-A\_5011 - Import von Kartenterminal-Informationen**

2856 Der Konnektor MUSS vor der endgültigen Aktivierung der importierten  
2857 Kartenterminalkonfiguration folgende zusätzliche Schritte ausführen:

- 2858 1. Die Liste der zu importierenden Kartenterminals MUSS dem Administrator  
2859 angezeigt werden. Er MUSS die Möglichkeit erhalten, einzelne Kartenterminals aus  
2860 dieser Liste zu löschen.
- 2861 2. Erst nach Bestätigung durch den Administrator werden die  
2862 Kartenterminalinformationen in die Kartenterminalverwaltung übernommen.
- 2863 3. Sofern die Kartenterminal-Konfiguration in einen Konnektor mit neuer Identität  
2864 importiert werden soll (neuer Konnektor oder neuer privater Schlüssel und neues  
2865 Zertifikat C.SAK.AUT auf der gSMC-K), muss die neue Identität des Konnektors  
2866 allen importierten Kartenterminals bekannt gemacht werden (Wartungs-Pairing,  
2867 siehe auch [gemSpec\_KT#2.5.2.4]).
  - 2868 a. Dazu baut der Konnektor unter der Nutzung von C.SAK.AUT eine temporäre  
2869 TLS-Verbindung auf und sendet das eHealth-Kartenterminal-Kommando  
2870 EHEALTH TERMINAL AUTHENTICATE in der Variante „ADD“ an jedes in der  
2871 Liste aufgeführte Kartenterminal. Mit dem Kommando und P2=03 holt sich der  
2872 Konnektor eine Challenge.
  - 2873 b. Der eigentliche Austausch bzw. die Aufnahme des neuen Zertifikates erfolgt im  
2874 KT erst, nachdem diese Challenge mit dem Kommando EHEALTH TERMINAL  
2875 AUTHENTICATE im Modus P2=04 vom Konnektor korrekt beantwortet wurde.  
2876 Dieses Kommando sowie die Erzeugung der Challenge-Antwort wird in  
2877 [gemSpec\_KT#3.7.2.4] und [gemSpec\_KT#3.7.2] beschrieben.

- c. Nach erfolgreicher Abarbeitung des Kommandos wird der Eintrag in die interne Liste der gepairten Kartenterminals übernommen und die temporäre Verbindung zum Kartenterminal abgebaut. Kann ein Kartenterminal nicht erreicht werden, so MUSS die Befehlskette nachgeholt werden, sobald das Kartenterminal vom Konnektor wieder als verfügbar erkannt wird.

4. Zur abschließenden Kontrolle und zur weiteren fachlichen Nutzung baut der Konnektor zu jedem der neu konfigurierten und aktiv gesetzten Kartenterminals via TUC\_KON\_050 eine Verbindung auf.

[<=]

#### 4.1.5 Kartendienst

Innerhalb des Kartendienstes werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): „CARD“
- Konfigurationsparameter: „CM\_“

Der Konnektor verwaltet eine Liste aller Karten (CM\_CARD\_LIST), die in die vom Konnektor verwalteten Kartenterminals gesteckt sind (CTM\_CT\_LIST). Alle Ereignisse und Operationen, die sich auf Karten beziehen, werden durch diesen Basisdienst gekapselt.

Für jede gesteckte Karte vergibt er einen eindeutigen Identifikator (im weiteren Text CardHandle bezeichnet), mit dem diese Karte adressiert werden kann, um zu diesen oder mit diesen Karten Operationen auszuführen. Dieses Handle ist gültig bis zum Entfernen der Karte aus dem Kartenterminal.

Um die in [gemSpec\_Perf] geforderten Zeiten für kartenbezogene Operationen erreichen zu können, kann es erforderlich sein, dass der Konnektor möglichst viele Informationen der Karten cached. Hierzu gehören Steuerdaten wie Extended Length, Version etc. aber auch Zertifikate der Karte (X.509 und CVC). Da es sich bei Caching um einen internen Mechanismus handelt, der sich nicht auf das funktionale Außenverhalten von TUCs oder Operationen auswirkt, wird das Caching nicht weiter beschrieben oder explizit gefordert. Es kann aber Anforderungen aus Sicherheitsicht bezüglich des Cachings geben (insbesondere hinsichtlich der erlaubten Caching-Dauer). Die Einhaltung dieser Vorgaben wird im Rahmen der CC-Evaluierung geprüft werden.

Der Kartendienst verwaltet mindestens die in der informativen Tabelle TAB\_KON\_531 ausgewiesenen Parameter, weitere herstellerspezifische Parameter sind möglich. Die normative Festlegung wann welche Parameter wie belegt werden, erfolgt in den folgenden Abschnitten und Unterkapiteln.

**Tabelle 57: TAB\_KON\_531 Parameterübersicht des Kartendienstes**

| ReferenzID      | Belegung                | Zustandswerte  |
|-----------------|-------------------------|--|
| CM_CARD_LIST    | Liste von Card-Objekten | Eine Liste von Repräsentanzen (CardObjects) der dem Konnektor bekannten Karten. Die Attribute der Card-Objekte sind im Folgenden gelistet. |
| CARD.CARDHANDLE |                         | vom Konnektor vergebenen eindeutigen Identifikator (Handle).   |

|                                       |                                |   |
|---------------------------------------|--------------------------------|---|
| CARD.CTID                             |                                | Kartenterminal, in dem die Karte steckt   |
| CARD.SLOTNO                           |                                | Slot, in dem die Karte steckt   |
| CARD.ICCSN                            |                                | ICCSN der Karte (sofern auslesbar),   |
| CARD.TYPE                             |                                | Typ der Karte gemäß Tabelle TAB_KON_500 Wertetabelle Kartentypen  |
| CARD.CARDVERSION                      |                                | die Versionsinformationen zum Produkttyp der Karte und den gespeicherten Datenstrukturen gemäß [gemSpec_Karten_Fach_TIP].   |
| CARD.CARDVERSION.COSVERSION           |                                | Produkttypversion des COS   |
| CARD.CARDVERSION.OBJECTSYSTEMVERSION  |                                | Produkttypversion des Objektsystems   |
| CARD.CARDVERSION.CARDPTPERSVERSION    |                                | Produkttypversion der Karte bei Personalisierung  |
| CARD.CARDVERSION.DATASTRUCTUREVERSION |                                | Version der Speicherstrukturen (aus EF.Version)   |
| CARD.CARDVERSION.LOGGINGVERSION       |                                | Version der Befüllvorschrift für EF.Logging   |
| CARD.CARDVERSION.ATRVERSION           |                                | Version der Befüllvorschrift für EF.ATR   |
| CARD.CARDVERSION.GDOVERSION           |                                | Version der Befüllvorschrift für EF.GDO   |
| CARD.CARDVERSION.KEYINFOVERSION       |                                | Version der Befüllvorschrift für KeyInfo  |
| CARD.INSERTTIME                       | Timestamp                      | Zeitpunkt, an dem die Karte gesteckt wurde  |
| CARD.CARDHOLDERNAME                   | String                         | Name des Karteninhabers bzw. der Institution/Organisation (subject.commonName)  |
| CARD.KVNR                             | String                         | Versicherten-ID (unveränderbarer Teil der KVNR)   |
| CARD.CERTEXPIRATIONDATE               |                                | Ablaufdatum des AUT-Zertifikats der Karte   |
| CARD.CARDSESSION_LIST                 | Liste von CardSession-Objekten | Eine Liste von Repräsentanzen (CardSession-Objects) der pro Karte vorhandenen Kartensitzungen. Die Attribute der CardSession-Objekte sind im Folgenden gelistet. Das Tripel aus MandantID, CSID und |

|                       |  |  |
|-----------------------|--|--|
|                       |  | UserID bildet den Kontext ab, in welchem diese Kartensitzung initiiert wurde.  |
| CARDSESSION.AUTHSTATE | Liste von Einträgen aus<br>a) C2C:KeyRef, Role<br>oder<br>b) CHV: PINRef | Liste von erreichten Sicherheitszuständen. Jeder einzelne Sicherheitszustand kann entweder über C2C gegen KeyRef (mit einer bestimmten Rolle gemäß [gemSpec_PKI_TI#Tab_PKI_918]) oder Card Holder Verification (CHV) gegen eine referenzierte PIN erreicht worden sein. Für eGK G2.0 wird der Zustand der MRPINs nicht in AuthState gespeichert. |
| CARDSESSION.MANDANTID |  | Mandant-ID   |
| CARDSESSION.CSID      |  | Clientsystem-ID  |
| CARDSESSION.USERID    |  | Nutzer-ID  |
| CARDSESSION.AUTHBY    | Referenz auf CardSession   | Kartensitzung, über die diese Karte freigeschaltet wurde (nur für eGK belegt)  |
| CARDSESSION.SIGNMODE  | „PIN“ oder „Comfort“   | Signaturmodus<br>„PIN“: Komfortsignaturmodus ist für die Karte ausgeschaltet<br>„Comfort“: Komfortsignaturmodus ist eingeschaltet<br>Default-Wert=„PIN“<br>Nur relevant für den HBA  |

#### 4.1.5.1 Funktionsmerkmalweite Aspekte

##### TIP1-A\_4988 - Unterstützung von Gen1 und Gen2 Karten

Der Konnektor MUSS eGKs der Generation 1+ unterstützen.

Der Konnektor DARF eGKs der Generation 1 NICHT unterstützen. eGKs der Generation 1 werden im Konnektor als CARD.TYPE = UNKNOWN geführt.

Der Konnektor MUSS für eGK, HBA, SMC-B, gSMC-KT und gSMC-K Karten der Generation 2 unterstützen. Karten der Generation 2 sind alle Karten, deren Version des dem aktiven Objektsystem zugrundeliegenden Produkttyps (Tag 'C1' in EF.Version2) den Wert 4.x.x hat, wobei x in {0, ..., 255}.

Bei Karten der Generation 2

- MUSS der Konnektor die ECC-basierten Geräte-CV-Zertifikate unterstützen.

##### [<=]

Es kann notwendig sein, Karten der Generation 2 (G2) näher zu bezeichnen. In diesem Fall wird in G2.0- und G2.1-Karten unterschieden. Wird von Karten der Generation 2 gesprochen, so gilt die Festlegung für G2.x (G2.0, G2.1 und höher) des betrachteten Kartentyps.



**TIP1-A\_4558 - Caching-Dauer von Kartendaten im Konnektor**

Sofern der Konnektor Daten gesteckter Karten cached, so DÜRFEN diese Daten von HBAX und SM-B NICHT länger als 24 Stunden gecached werden.

Der Konnektor DARF Daten der eGK NICHT über den Steckzyklus der Karte hinaus cachen.

Ausnahme: Eine Cachingdauer über den Steckzyklus der eGK hinaus wird von einer Fachanwendung gefordert und durch die Fachmodulspezifikation dieser Fachanwendung definiert.

[<=]

**TIP1-A\_6031 - Kein selbsttätiges Zurücksetzen der SM-B**

Der Konnektor DARF NICHT selbsttätig die SM-B und deren Sicherheitszustände zurücksetzen, auch nicht, wenn die Daten der SM-B nach Ablauf der 24-Stunden-Frist neu in den Cache eingelesen werden.

[<=]

**TIP1-A\_4559 - Konnektorzugriffsverbot auf DF.KT**

Der Konnektor DARF NICHT auf das DF.KT einer gSMC-KT zugreifen.

[<=]

**TIP1-A\_4560 - Rahmenbedingungen für Kartensitzungen**

Der Konnektor MUSS alle Zugriffe auf Karten aus CM\_CARD\_LIST, die den Sicherheitszustand der Karte erhöhen können oder einen erhöhten Sicherheitszustand der Karte voraussetzen, im Kontext einer Kartensitzung zu dieser Karte durchführen (CARD.CARDESESSION). Ausgenommen hiervon ist der Zugriff durch das CMS (bzw. VSDD) auf die eGK.

Der Konnektor MUSS sicherstellen, dass in einer Kartensitzung nur dann auf einen erhöhten Sicherheitszustand einer Karte zugegriffen werden kann, wenn die zur Erreichung dieses Sicherheitszustandes erforderlichen Authentisierungen (PIN-Verifikation, C2C-Rollen-Authentisierung etc.) in dieser verwendeten Kartensitzung erfolgreich durchgeführt wurden.

Der Konnektor MUSS Authentisierungen in einer Kartensitzung so durchführen, dass in anderen Kartensitzungen vorhandene Sicherheitszustände nicht beeinflusst werden. (Eine falsche PIN-Eingabe in einer Kartensitzung darf den erhöhten Sicherheitszustand einer anderen Sitzung nicht zurücksetzen etc.).

Der Konnektor SOLL die Verwaltung der Sicherheitsstatus der Kartensitzungen so über die Nutzung logischer Kartenkanäle umsetzen, dass PIN-Verifikationen, die für die Dauer der Kartensitzung Gültigkeit haben, nicht unnötig wiederholt werden müssen.

[<=]

Für die TUCs zur PIN-Eingabe, Änderung und Entsperrung sind Festlegungen hinsichtlich der auf dem Kartenterminal anzuzeigenden Meldungen erforderlich. Die folgende Tabelle definiert diese Terminalanzeigen gemäß [SICCT#5.5.10.19].

**TIP1-A\_4561-02 - Terminal-Anzeigen für PIN-Operationen**

Der Konnektor MUSS im Rahmen des interaktiven PIN-Handlings die folgenden Displaymessages für die Anzeige im Kartenterminal verwenden:

**Tabelle 58: TAB\_KON\_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal**

| Karte/<br>Kontext | PIN-Referenz | I/<br>O | Terminal-Anzeige | ANW<br>(max.Anz<br>·<br>Zeichen) |
|-------------------|--------------|---------|------------------|----------------------------------|
|                   |              |         |                  |                                  |



|  |  |   |   |    |
|--|--|---|---|----|
| <b>eGK</b><br>/PIN-Eingabe<br>für Vertreter-<br>PIN            | PIN.AMTS_REP   | I | Vertreter-<br>PIN • <b>0x0B</b> für • <b>0x0B</b> ANW<br><b>0x0F</b> Vertr-PIN:                         | 22 |
| <b>eGK</b><br>/PIN-Eingabe<br>für Vertreter-<br>PIN ändern     | PIN.AMTS_REP   | I | Vertreter-PIN • <b>0x0B</b> ändern<br><b>0x0F</b> PIN.eGK:  |    |
| <b>eGK</b><br>/PIN-Eingabe<br>für Vertreter-<br>PIN entsperren | PIN.AMTS_REP   | I | Vertreter-PIN • <b>0x0B</b> entsperren<br><b>0x0F</b> PIN.eGK:  |    |
| <b>eGK</b><br>/PIN-Eingabe<br>für PIN-Schutz<br>einschalten    | MRPIN.NFD,<br>MRPIN.DPE,<br>MRPIN.AMTS,<br>MRPIN.GDD | I | PIN-<br>Schutz • <b>0x0B</b> ANW • <b>0x0B</b> einschalt<br>en<br><b>0x0F</b> PIN.eGK:                  | 16 |
| <b>eGK</b><br>/PIN-Eingabe<br>für PIN-Schutz<br>abschalten     | MRPIN.NFD,<br>MRPIN.DPE,<br>MRPIN.AMTS,<br>MRPIN.GDD | I | PIN-<br>Schutz • <b>0x0B</b> ANW • <b>0x0B</b> abschalt<br>en<br><b>0x0F</b> PIN.eGK:                   | 16 |
| <b>eGK</b><br>/Sonstige  | ALLE (außer<br>PIN.AMTS_REP)                         | I | PIN • <b>0x0B</b> für • <b>0x0B</b> ANW<br><b>0x0F</b> PIN.eGK:   | 32 |
| HBax   | PIN.CH   | I | Eingabe • <b>0x0B</b> Freigabe-PIN • <b>0x0B</b> HBA<br><b>0x0F</b> PIN.HBA:                            |    |
|  | PIN.QES<br>(Signatur<br>auslösen)                    | I | #UVW-XYZ • <b>0x0B</b> Eingabe • <b>0x0B</b> Signatur-<br>PIN • <b>0x0B</b> HBA<br><b>0x0F</b> PIN.QES: |    |
| HBA  | PIN.QES<br>(Komfortsignatu<br>r aktivieren)          | I | Komfortsignatur • <b>0x0B</b> aktivieren • <b>0x0B</b> HBA<br><b>0x0F</b> PIN.QES:                      |    |
| SMC-B  | PIN.SMC  | I | Eingabe • <b>0x0B</b> PIN • SMC-B • <b>0x0B</b> SLOT:X<br><b>0x0F</b> PIN.SMC:                          |    |
| ANDERE   | BELIEBIG   | I | Herstellerspezifisch  |    |
| Erfolgreiche<br>PIN-Eingabe                                    | ALLE   | O | PIN • <b>0x0B</b> erfolgreich • <b>0x0B</b> verifiziert!  |    |
| Fehlerhafte<br>PIN-Eingabe                                     | ALLE   | O | PIN • <b>0x0B</b> falsch • <b>0x0B</b> oder • <b>0x0B</b> gesperrt!                                     |    |
| PUK-Eingabe  | eGK<br>PUK.CH  | I | Eingabe • <b>0x0B</b> Versicherten- <b>0x0B</b> PUK<br><b>0x0F</b> PUK.eGK:                             |    |
|  | HBax<br>PUK.CH                                       | I | Eingabe • <b>0x0B</b> Freigabe-PUK • <b>0x0B</b> HBA<br><b>0x0F</b> PUK.HBA:                            |    |
|  | HBax<br>PUK.QES                                      | I | Eingabe • <b>0x0B</b> Signatur-PUK • <b>0x0B</b> HBA<br><b>0x0F</b> PUK.QES:                            |    |

|                                     |                                     |   |  |
|-------------------------------------|-------------------------------------|---|--|
|                                     | SMC-B<br>PUK.SMC                    | I | Eingabe • <b>0x0B</b> PUK • SMC-B • <b>0x0B</b> SLOT:X<br><b>0x0F</b> PUK.SMC:   |
| Erfolgreiche<br>PUK-Eingabe         | ALLE                                | O | PIN • <b>0x0B</b> erfolgreich • <b>0x0B</b> entsperrt!   |
| Fehlerhafte<br>PUK-Eingabe          | ALLE                                | O | PUK • <b>0x0B</b> falsch • <b>0x0B</b> oder • <b>0x0B</b> gesperrt!  |
| Eingabe einer<br>neuen PIN          | eGK<br>ALLE (außer<br>PIN.AMTS_REP) | I | Eingabe •<br><b>0x0B</b> neue • <b>0x0B</b> Versicherten- <b>0x0B</b> PIN •<br><b>0x0B</b> (6-8 • Ziffern)<br><b>0x0F</b> PIN.eGK:   |
|                                     | eGK<br>PIN.AMTS_REP                 | I | Eingabe •<br><b>0x0B</b> neue • <b>0x0B</b> Vertreter-PIN •<br><b>0x0B</b> (6-8 • Ziffern)<br><b>0x0F</b> Vertr-PIN:                 |
|                                     | HBAx<br>PIN.CH                      | I | Eingabe • <b>0x0B</b> neue • <b>0x0B</b> Freigabe-<br>PIN • <b>0x0B</b> HBA • <b>0x0B</b> (6-8 • Ziffern)<br><b>0x0F</b> PIN.HBA:    |
|                                     | HBAx<br>PIN.QES                     | I | Eingabe •<br><b>0x0B</b> neue • <b>0x0B</b> Signatur-<br>PIN • <b>0x0B</b> HBA • <b>0x0B</b> (6-8 • Ziffern)<br><b>0x0F</b> PIN.QES: |
|                                     | SMC-B<br>PIN.SMC                    | I | Eingabe • <b>0x0B</b> neue • <b>0x0B</b> PIN • SMC-B •<br><b>0x0B</b> SLOT:X • <b>0x0B</b> (6-8 • Ziffern)<br><b>0x0F</b> PIN.SMC:   |
| Eingabe einer<br>Transport-PIN      | eGK<br>PIN.CH                       | I | Eingabe • <b>0x0B</b> Transport-<br><b>0x0B</b> Versicherten- <b>0x0B</b> PIN<br><b>0x0F</b> T-PIN.eGK:                              |
|                                     | HBAx<br>PIN.CH                      | I | Eingabe • <b>0x0B</b> Transport- <b>0x0B</b> PIN • <b>0x0B</b> HBA<br><b>0x0F</b> T-PIN.HBA:   |
|                                     | HBAx<br>PIN.QES                     | I | Eingabe • <b>0x0B</b> Transport- <b>0x0B</b> PIN • <b>0x0B</b> HBA<br><b>0x0F</b> T-PIN.QES:   |
|                                     | SMC-B<br>PIN.SMC                    | I | Eingabe • <b>0x0B</b> Transport- <b>0x0B</b> PIN • SMC-<br>B • <b>0x0B</b> SLOT:X<br><b>0x0F</b> T-PIN.SMC:                          |
| Wieder-holung<br>einer neuen<br>PIN | eGK<br>PIN.CH                       | I | Eingabe • <b>0x0B</b> Versicherten- <b>0x0B</b> PIN • <b>0x0B</b><br>wiederholen!<br><b>0x0F</b> PIN.eGK:                            |
|                                     | eGK<br>PIN.AMTS_REP                 | I | Eingabe •<br><b>0x0B</b> neue • <b>0x0B</b> Vertreter-PIN •<br><b>0x0B</b> wiederholen!<br><b>0x0F</b> Vertr-PIN:                    |
|                                     | HBAx<br>PIN.CH                      | I | Eingabe • <b>0x0B</b> für • HBA • <b>0x0B</b> wiederholen!<br><b>0x0F</b> PIN.HBA:   |
|                                     | HBAx<br>PIN.QES                     | I | Eingabe • <b>0x0B</b> für • HBA • <b>0x0B</b> wiederholen!<br><b>0x0F</b> PIN.QES:   |

|   |                  |   |   |
|---|------------------|---|---|
|   | SMC-B<br>PIN.SMC | I | Eingabe• <b>0x0B</b> PIN.SMC• <b>0x0B</b> SLOT:X•<br><b>0x0B</b> wiederholen!<br><b>0x0F</b> PIN.SMC: |
| Ungleichheit<br>bei der<br>Wieder-holung<br>der Eingabe<br>der neuen PIN                                      | ALLE             | O | PINs• <b>0x0B</b><br>nicht• <b>0x0B</b> identisch!• <b>0x0B</b> Abbruch!                              |
| Erfolgreiche<br>PIN-Änderung  | ALLE             | O | PIN•0x0Berfolgreich•0x0Bgeändert!   |
| Anzeigen am lokalen Terminal beim Remote-PIN-Verfahren für das Ergebnis der Verschlüsselung durch die gSMC-KT |                  |   |   |
| Erfolgreiche<br>Verschlüsselun<br>g   | ALLE             | O | Eingabe•0x0Bwird•0x0Bbearbeitet.  |
| Fehler bei der<br>Verschlüsselun<br>g   | ALLE             | O | Eingabe•0x0Bfehlgeschlagen.   |

[&lt;=]

Hinweise zu den Terminalanzeigen bei PIN-Eingaben und zu obiger Tabelle:

- ANW kennzeichnet den Anwendungsfall und wird durch den vom Aufrufer übergebenen String ersetzt (siehe z. B. TUC\_KON\_012 „PIN verifizieren“)
- Zu PIN.SMC: "Slot:X" im PIN-Prompt gibt die Slot-Nummer im Kartenterminal an, in der die SMC steckt, da in einem Kartenterminal mehr als eine SMC stecken kann.
- Variable Teile der Terminalanzeige (Job- und Slot-Nummer) sind kursiv formatiert.
- Zeichensatz gemäß ISO 646DE-/DIN 66003-Codierung
- max. 48 Zeichen Text + 10 Zeichen PIN-Prompt bei Input
- max. 48 Zeichen bei Output
- Leerzeichen werden als "•" dargestellt
- UVW-XYZ: zeigt die Jobnummer an (siehe Kapitel 4.1.8.1.4)
- #: Beginn der Jobnummer zur Verifizierung des korrekten Kartenterminals
- Weitere Details zur Gestaltung der Jobnummer finden sich im Kapitel 4.1.8.1.4.
- Die Zeilenumbrüche in der Spalte "Terminal-Anzeige" sind editorisch bedingt.
- 0x0B und 0x0F (Sollbruchstellen bzw. Trennung zwischen Nachricht und PIN-Prompt) sind Trennzeichen gemäß [SICCT#5.6.1].

In den Technischen Use Cases TUC\_KON\_012 „PIN verifizieren“, TUC\_KON\_019 „PIN ändern“, TUC\_KON\_021 „PIN entsperren“ wird das Remote-PIN-Verfahren verwendet, sofern die Zielkarte in einem als für den Arbeitsplatz entfernt definiertem Kartenterminal steckt (siehe Kap. 4.1.1.1, Relation [7]). In diesem Fall erfolgt die Nutzerinteraktion am

2999 Remote-PIN-KT von workplaceId (PinInputKT). Dabei wendet der Konnektor das folgende  
3000 Verfahren an:

3001 **TIP1-A\_5012 - Remote-PIN-Verfahren**

3002 Der Konnektor MUSS das Remote-PIN Verfahren im Sinne der BSI TR-03114  
3003 unterstützen. Abweichend von der TR-03114 MUSS statt der SMC-A eine gSMC-KT  
3004 verwendet werden.

3005 Der Konnektor MUSS für die PIN-Objekte: HBA.PIN.CH, HBA.PUK.CH, HBA.PIN.QES,  
3006 HBA.PUK.QES, SM-B.PIN.SMC und SM-B.PUK.SMC das Remote-PIN Verfahren  
3007 unterstützen. Für alle anderen Karten und PIN-Objekte DARF das Verfahren NICHT  
3008 verwendet werden.

3009 Für die Interaktion mit dem Anwender MÜSSEN die Display Messages entsprechend  
3010 TAB\_KON\_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal verwendet  
3011 werden.

3012 Der Ablauf für eine PIN-Operation gegen eine Zielkarte MUSS in diesen logischen  
3013 Schritten erfolgen:

- 3014 1. Aufruf TUC\_KON\_005 „Card-to-Card authentisieren“ mit eigens für diesen  
3015 Zweck erzeugten Cardsession sowohl für die „Sendekarte“ im PinInputKT (gSMC-  
3016 KT) sowie der Zielkarte. AuthMode ist „gegenseitig+TC“
- 3017 2. Der Benutzer wird mit dem SICCT-Kommando PERFORM VERIFICATION bzw.  
3018 MODIFY VERIFICATION DATA zur Eingabe der PIN am PinInputKT aufgefordert.  
3019 Als Display Messages für die erfolgreiche Bearbeitung bzw. Fehler in der  
3020 Bearbeitung dieser Kommandos müssen die Texte mitgesendet werden, die in  
3021 TAB\_KON\_090 für die Ergebnisse der Verschlüsselung durch die gSMC-KT  
3022 festgelegt sind.
- 3023 3. Im PinInputKT verschlüsselt die gSMC-KT die eingegebene PIN mit dem zuvor  
3024 erzeugten Sessionkey.
- 3025 4. Die verschlüsselte PIN wird in das zur intendierten PIN-Operation passende  
3026 Kommando eingebettet (PIN verifizieren, ändern oder entsperren - wird durch den  
3027 eigentlichen PIN-TUC festgelegt) und das Kommando vom Konnektor an die  
3028 Zielkarte zur Entschlüsselung und Verifikation übergeben. Dabei MUSS die  
3029 Übertragung im gleichen Logischen Kanal wie die SM Vereinbarung erfolgen.
- 3030 5. Der Konnektor zeigt das Resultat der Zielkarte mittels SICCT OUTPUT am  
3031 lokalen Kartenterminal an. Er verwendet dabei den in TAB\_KON\_090 für die  
3032 aktuelle PIN-Operation spezifizierten Ausgabetexte.
- 3033 6. Das Result der Zielkarte wird an den Aufrufer zurückgegeben

3034 Fehlermeldung: Ein Fehler in der Verarbeitung führt zum Abbruch mit Fehlercode 4053  
3035 „Remote-PIN nicht möglich“ (Security, Error).

3036 [**<=**]

3037 *Hinweis: Derzeit schlägt die Freischaltung der SMC-B durch Card-2-Card-Authentisierung*  
3038 *ohne Fehlermeldung fehl. Der Sicherheitszustand der SMC-B wird nicht verändert. Diese*  
3039 *Einschränkung betrifft TUC\_KON\_005 „Card-to-Card authentisieren“ (TAB\_KON\_096).*

3040 **4.1.5.2 Durch Ereignisse ausgelöste Reaktionen**

3041 **TIP1-A\_4562 - Reaktion auf „Karte entfernt“**

3042 Empfängt der Kartendienst das Ereignis „CT/SLOT\_FREE“, so MUSS der Konnektor:

- 3043 • das über die im Ereignis gemeldeten Parameter CtID und SlotNo in  
3044 CM\_CARD\_LIST adressierte CardObject CARD identifizieren

3045 • für dieses CardObject folgendes Ereignis absetzen:  
 3046 TUC\_KON\_256{  
 3047     topic = „CARD/REMOVED“;  
 3048     eventType = Op;  
 3049     Severity = Info;  
 3050     parameters = <Params>}  
 3051 wobei <Params> mit folgenden Werten belegt werden MUSS:

- 3052 • „CardHandle=\$CARD.CARDHANDLE,
- 3053 • Type=\$CARD.TYP,
- 3054 • CardVersion=\$CARD.VER,
- 3055 • ICCSN=\$CARD.ICCSN,
- 3056 • CtID=\$CARD.CTID,
- 3057 • SlotID=\$CARD.SLOTID,
- 3058 • InsertTime=\$CARD.INSERTTIME,
- 3059 • CardHolderName=\$CARD.CARDHOLDERNAME,
- 3060 • KVNVR=\$CARD.KVNVR
- 3061 • das zugehörige CardObject aus CM\_CARD\_LIST entfernen.

3062  
 3063 [**<=**]

#### 3064 **TIP1-A\_4563 - Reaktion auf „Karte gesteckt“**

3065 Empfängt der Kartendienst das Ereignis „CT/SLOT\_IN\_USE“, so MUSS der Konnektor für  
 3066 die Karte, die über die im Ereignis gemeldeten Parameter CtID und SlotNo adressiert ist,  
 3067 über TUC\_KON\_001 ein neues CardObject in CM\_CARD\_LIST anlegen.

3068 [**<=**]

### 3069 **4.1.5.3 Interne TUCs, nicht durch Fachmodule nutzbar**

#### 3070 *4.1.5.3.1 TUC\_KON\_001 „Karte öffnen“*

#### 3071 **TIP1-A\_4565 - TUC\_KON\_001 „Karte öffnen“**

3072 Der Konnektor MUSS den technischen Use Case „Karte öffnen“ gemäß TUC\_KON\_001  
 3073 umsetzen.

3074

3075 **Tabelle 59: TAB\_KON\_734 – TUC\_KON\_001 „Karte öffnen“**

| Element      | Beschreibung  |
|--------------|---|
| Name         | TUC_KON_001 „Karte öffnen“  |
| Beschreibung | Der TUC initialisiert ein Card-Object basierend auf einer physikalischen Karte und fügt es CM_CARD_LIST zu. Die Karte kann erst im Anschluss unter Verwendung des erzeugten CardHandles verwendet werden. |
| Auslöser     | Der Kartenterminaldienst meldet das Belegen eines KT-Slots  |

|                |   |
|----------------|---|
| Vorbedingungen | <ul style="list-style-type: none"> <li>In ctId/slotId steckt eine Karte</li> </ul>  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>ctId<br/>(Kartenterminalidentifikator)</li> <li>slotId<br/>(Nummer des Kartenslots)</li> </ul>   |
| Komponenten    | Karte, Kartenterminal, Konnektor  |
| Ausgangsdaten  | Keine   |
| Standardablauf | <p>1. Prüfe, ob unter ctId und slotId ein Eintrag in CM_CARD_LIST vorhanden ist. Wenn bereits ein Eintrag vorhanden ist, lösche diesen.</p> <p>2. Erzeuge neuen Card-Object-Eintrag in CM_CARD_LIST und</p> <p>a) Generiere CARD.CARDHANDLE. mit folgenden Anforderungen:</p> <ul style="list-style-type: none"> <li>- Das CardHandle MUSS innerhalb CM_CARD_LIST eindeutig sein.</li> <li>- Ein ungültig gewordenes CardHandle DARF innerhalb von 48h NICHT als neues CardHandle vergeben werden.</li> </ul> <p>b) Befülle CARD.CTID und CARD.SLOTNO mit den Eingangsdaten</p> <p>c) Ermittle und befülle (soweit durch Karte unterstützt) die folgenden Daten:</p> <ul style="list-style-type: none"> <li>- CARD.ICCSN</li> <li>- CARD.TYPE (mögliche Werte siehe Tabelle TAB_KON_500 Wertetabelle Kartentypen)</li> <li>- CARD.CARDVERSION</li> <li>- CARD.INSERTTIME (=aktuelle Systemzeit)</li> <li>- CARD.CARDHOLDERNAME (aus X.509-AUT-Zertifikat)</li> <li>- CARD.KVNR (nur für eGK, aus C.CH.AUT: unveränderbarer Teil der KVNR)</li> <li>- CARD.CERTEXPIRATIONDATE (=validity aus X.509-AUT-Zertifikat)</li> </ul> <p>X.509-AUT-Zertifikat bezeichnet für eGK das C.CH.AUT-Zertifikat, für HBAX das C.HP.AUT-Zertifikat und für SMC-B das C.HCI.AUT-Zertifikat.</p> <p>3. Rufe TUC_KON_256{<br/> topic = „CARD/INSERTED“;<br/> eventType = Op;<br/> severity = Info;<br/> parameters = &lt;Params&gt;} </p> |

|                                |   |
|--------------------------------|---|
|                                | <p>mit &lt;Params&gt; belegt aus dem CARD-Object:<br/>         „CardHandle=\$, CardType=\$, CardVersion=\$,<br/>         ICCSN=\$, CtID=\$,<br/>         SlotID=\$, InsertTime=\$, CardHolderName=\$, KVN=\$,<br/>         CertExpirationDate=\$“</p> <p>In CardVersion sind die Werte</p> <ul style="list-style-type: none"> <li>- COSVERSION und</li> <li>- OBJECTSYSTEMVERSION</li> </ul> <p>aus CARD.CARDVERSION einzutragen. Für eGK G1+ ist zusätzlich die</p> <ul style="list-style-type: none"> <li>- DATASTRUCTUREVERSION</li> </ul> <p>aus CARD.CARDVERSION einzutragen. CardVersion kann weitere Werte aus CARD.CARDVERSION enthalten.</p> |
| Varianten/<br>Alternativen     | <p>Im Falle der KVK gibt es kein EF.ATR, EF.GDO und EF.DIR. Es wird daher lediglich der ATR ausgewertet, den das Kartenterminal beim Stecken der Karte liefert.</p>   |
| Fehlerfälle                    | <p>(-&gt; 2c) Karte/Kartenterminal antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;gemäß [gemSpec_COS]/[SICCT]&gt;</p> <p>Auch im Fehlerfall wird Schritt 3 durchlaufen. Wenn nicht alle zu einem Kartentyp notwendigen Daten von der Karte gelesen werden konnten, dann wird Schritt 3 mit CardType=UNKNOWN ausgeführt.</p> <p>Auch im Fehlerfall wird Schritt 3 durchlaufen. Wenn nicht alle zu einem Kartentyp notwendigen Daten von der Karte gelesen werden konnten, dann wird Schritt 3 mit CardType=UNKNOWN ausgeführt.</p>   |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

3076

3077 [ $\leq$ ]

#### 3078 4.1.5.4 Interne TUCs, auch durch Fachmodule nutzbar

##### 3079 4.1.5.4.1 TUC\_KON\_026 „Liefere CardSession“

##### 3080 TIP1-A\_4566 - TUC\_KON\_026 „Liefere CardSession“

3081 Der Konnektor MUSS den technischen Use Case „Liefere CardSession“ gemäß

3082 TUC\_KON\_26 umsetzen.

3083 **Tabelle 60: TAB\_KON\_735 - TUC\_KON\_026**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_026 „Liefere CardSession“  |
| Beschreibung   | Dieser Use Case gibt auf Grund der übergebenen Parameter die zugehörige CardSession zurück. Ist für die Parameterkombination noch keine CardSession vorhanden, wird eine neue erzeugt und im zugehörigen Card-Object hinterlegt.   |
| Auslöser       | <ul style="list-style-type: none"> <li>• Indirekter Aufruf über durch Clientsysteme ausgeführte Operationen.</li> <li>• Aufruf durch Fachmodul</li> </ul>  |
| Vorbedingungen | Keine  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• mandantId</li> <li>• clientSystemId</li> <li>• cardHandle</li> <li>• userId - <i>optional/verpflichtend, wenn cardType = HBAX</i></li> </ul>  |
| Komponenten    | Konnektor  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>• cardSession</li> </ul>  |
| Standardablauf | <ol style="list-style-type: none"> <li>1. Ermittle Card in CM_CARD_LIST über cardHandle</li> <li>2. Prüfe dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>3. Ermittle cardSession in Card.CARDSESSION_LIST über mandantId, clientSystemId und userId</li> </ol> |



|                                   |  |
|-----------------------------------|--|
| Varianten/<br>Alternativen        | (→3) Wenn keine CardSession für diese Parameter vorhanden:<br>1.                   erzeuge neue CardSession in Card.<br>CARDSESSION_LIST<br>2.                   Befülle CARDSESSION.MANDANTID,<br>.CSID und<br>.USERID mit Übergabeparametern |
| Fehlerfälle                       | (→2) Karte bereits reserviert, Fehlercode 4093   |
| Nichtfunktionale<br>Anforderungen | keine  |
| Zugehörige<br>Diagramme           | keine  |

3084 **Tabelle 61: TAB\_KON\_824 Fehlercodes TUC\_KON\_026 „Liefere CardSession“**

| Fehlercode | ErrorType | Severity | Fehlertext   |
|------------|-----------|----------|--|
| 4093       | Technical | Error    | Karte wird in einer anderen Kartensitzung exklusiv verwendet |

3085  
3086 **[<=]**

3087 *Hinweis zu TAB\_KON\_735 - TUC\_KON\_026: Die WorkplaceId wird als Eingangsparameter*  
3088 *nicht benötigt. Bereits TUC\_KON\_000 stellt sicher, dass eine eGK jeweils nur von einem*  
3089 *einzigem Arbeitsplatz aus angesprochen werden kann.*

3090 **4.1.5.4.2 TUC\_KON\_012 „PIN verifizieren“**

3091 **TIP1-A\_4567 - TUC\_KON\_012 „PIN verifizieren“**

3092 Der Konnektor MUSS den technischen Use Case „PIN verifizieren“ gemäß TUC\_KON\_012  
3093 umsetzen.

3094

3095 **Tabelle 62: TAB\_KON\_087 – TUC\_KON\_012 „PIN verifizieren“**

| Element      | Beschreibung  |
|--------------|---|
| Name         | TUC_KON_012 „PIN verifizieren“  |
| Beschreibung | Dieser Use Case führt die Verifikation einer PIN einer Karte durch. Dabei wird der Anwender am Display des Kartenterminals aufgefordert, die PIN einzugeben. Dies erfolgt am PIN-Pad des Kartenterminals.<br>Remote-PIN-Eingabe wird dabei automatisch unterstützt. |
| Auslöser     | <ul style="list-style-type: none"> <li>Aufruf des Use Case durch Basisdienste des Konnektors</li> <li>Aufruf des Use Cases durch ein Fachmodul im Konnektor</li> </ul>  |

|                |  |
|----------------|--|
|                | <ul style="list-style-type: none"> <li>Aufruf der Operation VerifyPin des CardService (siehe 4.1.5.5.1) durch das Clientsystem.</li> </ul>   |
| Vorbedingungen | Karte unterstützt die übergebene pinRef  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>cardSession<br/>(Kartensitzung der Karte, deren PIN verifiziert werden soll)</li> <li>workplaceId</li> <li>pinRef<br/>(Referenz auf die zu verifizierende PIN, gemäß der Objektsystem-Spezifikation des jeweiligen Kartentyps.)</li> <li>actionName – <i>optional/verpflichtend, wenn cardType = eGK</i><br/>(Zeichenkette, max. 32 bzw. 22 Zeichen PIN.AMTS_REP mit dem Namen der zugreifenden Fachanwendung bzw. des zu nutzenden Datenobjekts und der Zugriffsart, die mit dieser PIN freigeschaltet werden soll, z. B. für MRPIN.NFD: actionName = „Notfalldaten schreiben“;<br/>Positionen in der Zeichenkette, an denen ein Zeilenumbruch bei der Ausgabe am Kartenterminal erlaubt ist, werden mit `0x0B` gekennzeichnet. `0x0B` zählt bei der Länge der Zeichenkette nicht.)</li> <li>verificationType [Mandatorisch   Sitzung]<br/>(Art der PIN-Verifikation: <ul style="list-style-type: none"> <li>Mandatorisch: PIN wird immer verifiziert.</li> <li>Sitzung: PIN wird nicht erneut verifiziert, falls dies für die cardSession zuvor bereits geschehen ist und der dadurch erreichte Sicherheitszustand nicht zurückgesetzt wurde.)</li> </ul> </li> </ul> |
| Komponenten    | Karte, Kartenterminal, Konnektor   |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>pinResult [PinResult]<br/>(Ergebnis der PIN-Verifikation)</li> <li>leftTries – <i>optional/verpflichtend, wenn pinResult = REJECTED</i><br/>(Anzahl der verbleibenden Versuche)</li> </ul>  |
| Standardablauf | <ol style="list-style-type: none"> <li>Ermittle Card = CM_CARD_LIST(CardSession)</li> <li>Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>Wenn PinTyp(pinRef) = PIN.QES oder VerificationType = Mandatorisch 6.</li> <li>Wenn pinRef in CARDESSION.AUTHSTATE vorhanden: pinResult = OK;</li> <li>Prüfe TUC_KON_022 „Liefere PIN-Status“ <ol style="list-style-type: none"> <li>„VERIFYABLE“;</li> <li>„DISABLED“: pinResult = OK;</li> </ol> </li> <li>Ermittle PinInputKT: Wenn Card.ctId ein dem Arbeitsplatz(workplaceId) lokal zugeordnetes Kartenterminal ist</li> </ol>  |

|                            |   |
|----------------------------|---|
|                            | <p>(siehe Relation [6], Kapitel 4.1.1.1)</p> <ol style="list-style-type: none"> <li>a. Setze PinInputKT = Card.CtID</li> <li>b. sonst „lokales Kartenterminal, das für die Remote-PIN-Eingabe zu verwenden ist“: PinInputKT = Arbeitsplatz(workplaceId).remote-PIN-KT(mandantId)</li> </ol> <p>7. Atomare Operation: PIN verifizieren inkl. Eventing und Ergebnisvermerk</p> <ol style="list-style-type: none"> <li>a. Rufe TUC_KON_256 {<br/> topic = „CARD/PIN/VERIFY_STARTED“;<br/> eventType = Op;<br/> severity = Info;<br/> parameters = („CardHandle=\$, CardType=\$, ICCSN=\$, CtID=\$, SlotID=\$, PinRef=\$, PinInputCtID=\$PinInputKT“, doLog=false)}</li> <li>b. Pin-Verifikation über „Perform Verification“ ([SICCT]) mit<br/> Display Messages gemäß Kontext in TAB_KON_090<br/> Terminalanzeigen beim Eingeben der PIN am Kartenterminal, bei eGK ersetze „ANW“ durch actionName in Display Message.<br/> Wenn PinInputKT=Card.CtID dann PIN Verifikation direkt an<br/> Card.CtID, ansonsten Remote-PIN-Eingabe gemäß (TIP1-A_5012)</li> <li>c. Setze pinResult in Abhängigkeit von Ergebnis Perform Verification:<br/> <ul style="list-style-type: none"> <li>- pinResult = OK für erfolgreiche Prüfung</li> <li>- pinResult = ERROR für Nutzer-Abbruch oder Bearbeitungsfehler (siehe Fehlerfälle)</li> <li>- pinResult = REJECTED für falsche PIN;<br/> leftTries = x (bei Kartenantwort '63 Cx', x &gt; 0)</li> <li>- pinResult = BLOCKED für gesperrte PIN (bei Kartenantwort '63 C0')</li> </ul> </li> <li>d. Rufe TUC_KON_256 {<br/> topic = „CARD/PIN/VERIFY_FINISHED“;<br/> eventType = Op;<br/> severity = Info;<br/> parameters = („CardHandle=\$, CardType=\$, ICCSN=\$, CtID=\$, SlotID=\$, PinRef=\$, PinInputCtID=\$PinInputKT, Result=\$pinResult“);<br/> doLog = false }</li> <li>e. befülle CARDSESSION.AUTHSTATE mit pinRef und Ergebnis der PIN-Prüfung</li> </ol> <p>8. Liefere pinResult zurück</p> |
| Varianten/<br>Alternativen | Schritt 7e: Für eGK G2.0 wird der Zustand der MRPINs nicht in AuthState gespeichert.  |
| Fehlerfälle                | Schritt 7 wird mit allen Teilschritten durchlaufen. Ein als Fehler ausgewiesener Zustand führt erst nach Schritt 7e zum Abbruch des TUCs. Fehleingaben zählen explizit nicht zu den Fehlerzuständen, sondern werden auf das Ergebnis REJECTED   |

|                                |   |
|--------------------------------|---|
|                                | <p>oder BLOCKED abgebildet.</p> <ul style="list-style-type: none"> <li>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094</li> <li>(→2) Karte ist fremd reserviert, Fehlercode 4093</li> <li>(→5) Rückgabewert= <ul style="list-style-type: none"> <li>- VERIFIED, Fehlercode 4001</li> <li>- TRANSPORT_PIN oder EMPTY_PIN, Fehlercode 4065</li> <li>- BLOCKED, Fehlercode 4063</li> </ul> </li> <li>(→6b) kein Remote-PIN-KT zugeordnet, Fehlercode 4092</li> <li>(-&gt;6b) Card.TYP=eGK und Card.CtID ist nicht dem durch workplaceId bezeichneten Arbeitsplatz lokal zugeordnet: Fehlercode 4053</li> <li>(→7) Timeout bei PIN Eingabe: Fehlercode 4043</li> <li>(→7) Abbruch durch Nutzer: Fehlercode 4049</li> <li>(→7) Sind das für die PIN-Eingabe benötigte Kartenterminal oder benötigte Teile davon (PIN Pad, Display) durch einen anderen zeitgleich im Konnektor ablaufenden Vorgang reserviert, so bricht der Use Case mit Fehler 4060 ab.</li> <li>(→7) Rückgabewert= <ul style="list-style-type: none"> <li>- transportgeschützt (Transport-PIN oder Leer-PIN), Fehlercode 4065</li> </ul> </li> <li>(→7b) Ungültige PIN-Referenz; Fehlercode 4072</li> <li>(→7b) Karte/Kartenterminal antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;gemäß [gemSpec_COS]/[SICCT]&gt;</li> </ul> <p>Zusätzliche Fehlerfälle ergeben sich aus der Remote-PIN-Eingabe gemäß (TIP1-A_5012)</p> |
| Nichtfunktionale Anforderungen |   |
| Zugehörige Diagramme           | Abbildung PIC_KON_111 Aktivitätsdiagramm zu „PIN verifizieren“  |

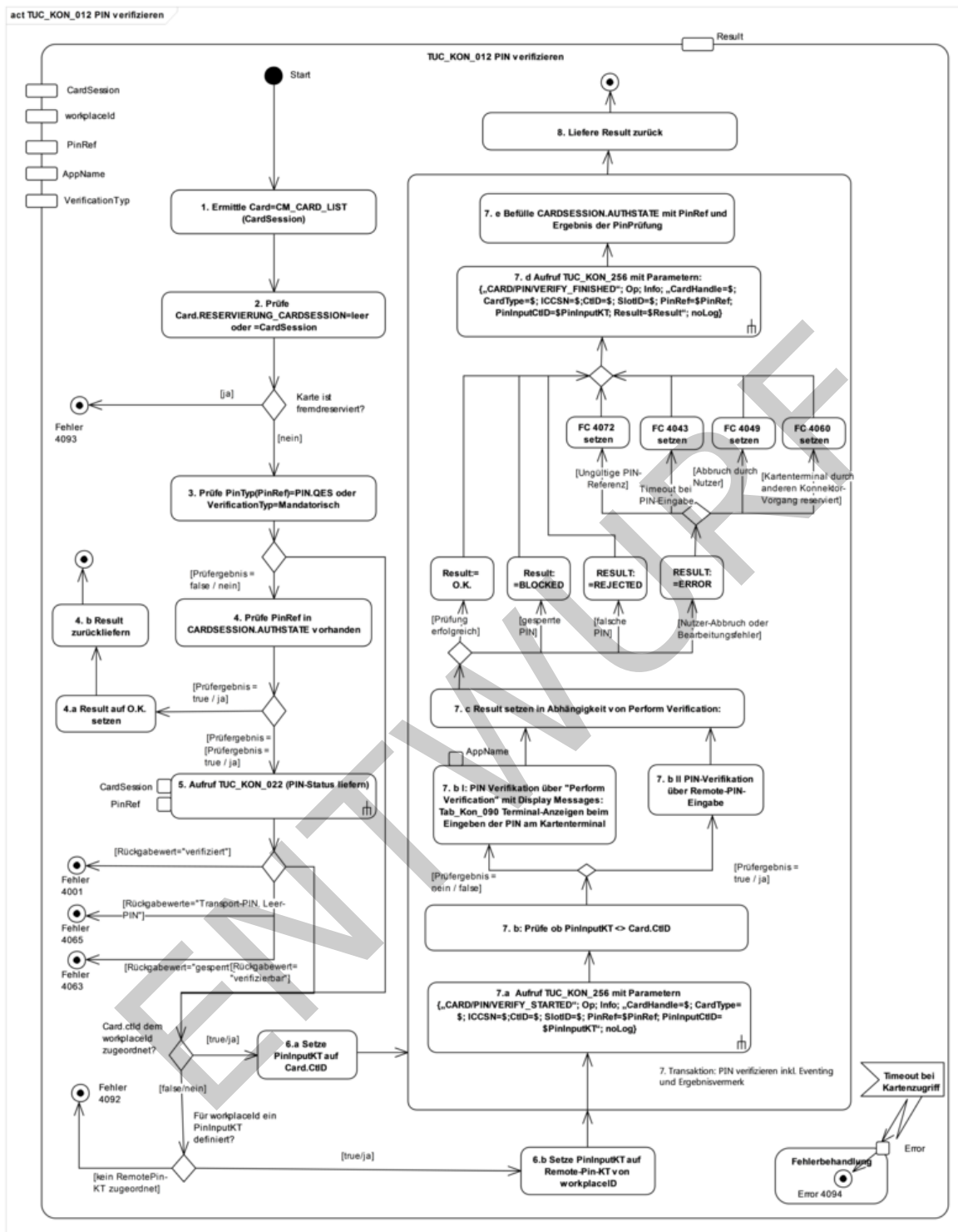


Abbildung 10: PIC\_KON\_111 Aktivitätsdiagramm zu „PIN verifizieren“

Tabelle 63: TAB\_KON\_089 Fehlercodes TUC\_KON\_012 „PIN verifizieren“

| Fehlercode  | ErrorType | Severity | Fehlertext |
|---|-----------|----------|------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |            |

|      |           |         |   |
|------|-----------|---------|---|
| 4001 | Technical | Error   | Interner Fehler   |
| 4043 | Technical | Warning | Timeout bei der PIN-Eingabe   |
| 4049 | Technical | Error   | Abbruch durch den Benutzer  |
| 4053 | Security  | Error   | Remote-PIN nicht möglich  |
| 4060 | Technical | Error   | Ressource belegt  |
| 4063 | Security  | Error   | PIN bereits gesperrt (BLOCKED)                                      |
| 4065 | Technical | Warning | PIN ist transportgeschützt, Änderung erforderlich                   |
| 4072 | Technical | Error   | Ungültige PIN-Referenz <code>pinRef</code>                          |
| 4092 | Technical | Error   | Remote-PIN-KT benötigt aber für diesen Arbeitsplatz nicht definiert |
| 4093 | Technical | Error   | Karte wird in einer anderen Kartensitzung exklusiv verwendet        |
| 4094 | Technical | Error   | Timeout beim Kartenzugriff aufgetreten                              |

3099  
3100  
3101  
3102

[<=]

3103 4.1.5.4.3 TUC\_KON\_019 „PIN ändern“

3104 **TIP1-A\_4568 - TUC\_KON\_019 „PIN ändern“**

3105 Der Konnektor MUSS den technischen Use Case „PIN ändern“ gemäß TUC\_KON\_019  
3106 umsetzen.  
3107

3108 **Tabelle 64: TAB\_KON\_736 – TUC\_KON\_019 „PIN ändern“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_019 „PIN ändern“  |
| Beschreibung   | Dieser Use Case führt die Änderung einer PIN einer Karte durch. Dabei wird der Anwender am Display des Kartenterminals aufgefordert, alte und neue PIN einzugeben. Remote-PIN-Eingabe wird dabei automatisch unterstützt. |
| Auslöser       | <ul style="list-style-type: none"> <li>Aufruf der Operation <code>ChangePin</code> des <code>CardService</code> (siehe 4.1.5.5.2) durch das Clientsystem.</li> <li>Aufruf durch Fachmodul</li> </ul>                      |
| Vorbedingungen | Karte unterstützt die übergebene <code>pinRef</code>  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li><code>cardSession</code></li> <li><code>workplaceId</code> (Arbeitsplatz-Identifikator)</li> </ul>   |

|                |   |
|----------------|---|
|                | <ul style="list-style-type: none"> <li>• pinRef<br/>(Referenz auf die zu ändernde PIN, gemäß der Objektsystem-Spezifikation des jeweiligen Kartentyps)</li> <li>• sourceCardSession – <i>optional/verpflichtend, wenn C2C erforderlich ist</i><br/>(CardSession der Karte, die für die Card-to-Card-Authentisierung bei Änderung der PIN einer eGK der Generation 1+ verwendet werden soll.)</li> </ul>   |
| Komponenten    | Karte, Kartenterminal, Konnektor  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>• pinResult [PinResult]<br/>(Ergebnis der PIN-Verifikation)</li> <li>• leftTries – <i>optional/verpflichtend, wenn pinStatus = REJECTED</i><br/>(verbleibende Versuche)</li> </ul>   |
| Standardablauf | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(CardSession)</li> <li>2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>3. Prüfe TUC_KON_022 „Liefere PIN-Status“ {cardSession; pinRef}&lt;&gt;BLOCKED</li> <li>4. Wenn pinRef=PIN.AMTS_REP, dann rufe TUC_KON_012 „PIN verifizieren“ {<br/>cardSession;<br/>workplaceId;<br/>pinRef=PIN.CH;<br/>actionName= „“;<br/>mandatorisch}</li> <li>5. Wenn Card.TYP=eGK UND Card.Version=Generation1+, dann Aufruf TUC_KON_005 „Card-to-Card authentisieren“{<br/>sourceCardSession;<br/>targetCardSession=cardSession;<br/>AuthMode =einseitig}.</li> </ol> <p>Falls keine sourceCardSession angegeben ist, kann die CardSession der für den Mandanten verwalteten SMC-B verwendet werden.</p> <ol style="list-style-type: none"> <li>6. Ermittle PinInputKT: Wenn Card.ctId ein dem Arbeitsplatz (workplaceId) lokal zugeordnetes Kartenterminal ist (siehe Relation [6], Kapitel 4.1.1.1) <ol style="list-style-type: none"> <li>a. Setze PinInputKT = Card.CtID</li> <li>b. sonst „lokales Kartenterminal, das für die Remote-PIN-Eingabe zu verwenden ist“: PinInputKT = Arbeitsplatz(workplaceId).remote-PIN-KT(mandantId)</li> </ol> </li> <li>7. Atomare Operation: PIN ändern inkl. Eventing und Ergebnisvermerk <ol style="list-style-type: none"> <li>a. Rufe TUC_KON_256 {<br/>topic = „CARD/PIN/CHANGE_STARTED“;</li> </ol> </li> </ol> |

|                            |  |
|----------------------------|--|
|                            | <pre> eventType = Op; severity = Info; parameters = („CardHandle=\$, CardType=\$,               ICCSN=\$, CtID=\$, SlotID=\$, PinRef=\$,               PinInputCtID=\$PinInputKT“); doLog = false } </pre> <p>b. Pin-Änderung über „MODIFY VERIFICATION DATA“ ([SICCT]) mit Display Messages entsprechend TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal. Bei Änderung der Versicherten-PIN der eGK ist dabei der Platzhalter „ANW“ durch den String „Änderung“ zu ersetzen. Der Platzhalter „#UVW-XYZ“ entfällt für die PIN.QES des HBA.</p> <p>Wenn PinInputKT=Card.CtID, dann PIN-Änderung direkt an Card.CtID, ansonsten Remote-PIN-Eingabe gemäß (TIP1-A_5012)</p> <p>Dabei sowohl Unterstützung normaler PIN-Änderung als auch Umsetzens eines Transportschutzes (alle Varianten gemäß Kartenspec sind zu unterstützen)</p> <p>c. Setze pinResult in Abhängigkeit von Ergebnis MODIFY VERIFICATION DATA:</p> <ul style="list-style-type: none"> <li>- pinResult = OK für erfolgreiche Änderung</li> <li>- pinResult = ERROR für Nutzer-Abbruch oder Bearbeitungsfehler (siehe Fehlerfälle)</li> </ul> <p>pinResult = REJECTED für falsche PIN-Eingaben;<br/>leftTries = x<br/>(bei Kartenantwort '63 Cx', x &gt; 0)</p> <ul style="list-style-type: none"> <li>- pinResult = BLOCKED für gesperrte PIN (bei Kartenantwort '63 C0')</li> </ul> <p>d. Rufe TUC_KON_256 {<br/>topic = „CARD/PIN/CHANGE_FINISHED“;<br/>eventType = Op;<br/>severity = Info;<br/>parameters = („CardHandle=\$, CardType=\$, ICCSN=\$;CtID=\$, SlotID=\$, PinRef=\$, PinInputCtID=\$PinInputKT, Result=pinStatus“);<br/>doLog = false}</p> <p>e. Wenn Result = REJECTED oder BLOCKED , dann entferne PinRef aus CARDESESSION.AUTHSTATE</p> <p>8. Liefere pinResult und ggf. leftTries zurück</p> |
| Varianten/<br>Alternativen | <p>Schritt 4: Für eGK G2.0 gilt:</p> <p>Wenn pinRef=PIN.AMTS_REP, dann rufe TUC_KON_012 „PIN verifizieren“ {<br/>cardSession;<br/>workplaceId;<br/>pinRef=MRPIN.AMTS;<br/>actionName= „“;<br/>mandatorisch}</p>  |



|                                |  |
|--------------------------------|--|
|                                | Schritt 7e: Für eGK G2.0 wird der Zustand der MRPINs nicht in AuthState gespeichert.   |
| Fehlerfälle                    | <p>Schritt 7 wird mit allen Teilschritten durchlaufen. Ein als Fehler ausgewiesener Zustand führt erst nach Schritt 7e zum Abbruch des TUCs.</p> <p>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden,</p> <p>Fehlercode 4094</p> <p>(→2) Karte ist fremd reserviert, Fehlercode 4093</p> <p>(→3) pinStatus=BLOCKED: Fehlercode 4063</p> <p>(→5) sourceCardSession benötigt aber leer, Fehlercode 4071</p> <p>(→6b) kein Remote-PIN-KT zugeordnet, Fehlercode 4092</p> <p>(→6b) Card.TYP=eGK und Card.CtID ist nicht dem durch workplaceId bezeichneten Arbeitsplatz lokal zugeordnet: Fehlercode 4053</p> <p>(→7b) neue PIN zu kurz/lang: Fehlercode 4068</p> <p>(→7b) zweite neue PIN&lt;&gt; erste neue PIN: Fehlercode 4067</p> <p>(→7b) Timeout bei PIN-Eingabe: Fehlercode 4043.</p> <p>(→7b) Abbruch durch Nutzer: Fehlercode 4049.</p> <p>(→7b) Ist das Kartenterminal oder Teile davon (PIN-Pad, Display) durch einen anderen Vorgang reserviert: Fehlercode 4060</p> <p>(→7b) kein PIN-Pad am Kartenterminal verfügbar: Fehlercode 4066</p> <p>(→7b) Ungültige PIN-Referenz; Fehlercode 4072</p> <p>(→7b) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode</p> <p>&lt;Kartenfehlercode gemäß [gemSpec_COS]&gt;</p> <p>Zusätzliche Fehlerfälle ergeben sich aus der Remote-PIN-Eingabe gemäß (TIP1-A_5012)</p> |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | keine  |

3109

**Tabelle 65: TAB\_KON\_093 Fehlercodes TUC\_KON\_019 „PIN ändern“**

| Fehlercode  | ErrorType | Severity | Fehlertext                  |
|---|-----------|----------|-----------------------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |                             |
| 4043  | Technical | Warning  | Timeout bei der PIN-Eingabe |
| 4049  | Technical | Error    | Abbruch durch den Benutzer  |
| 4053  | Security  | Error    | Remote-PIN nicht möglich    |
| 4060  | Technical | Error    | Ressource belegt            |

|      |           |       |   |
|------|-----------|-------|---|
| 4063 | Security  | Error | PIN bereits blockiert (BLOCKED)                                     |
| 4066 | Technical | Error | PIN Pad nicht verfügbar   |
| 4067 | Security  | Error | neue PIN nicht identisch  |
| 4068 | Security  | Error | neue PIN zu kurz/zu lang  |
| 4071 | Technical | Error | keine Karte für C2C-Auth gesetzt                                    |
| 4072 | Technical | Error | ungültige PIN-Referenz <code>PinRef</code>                          |
| 4092 | Technical | Error | Remote-PIN-KT benötigt aber für diesen Arbeitsplatz nicht definiert |
| 4093 | Technical | Error | Karte wird in einer anderen Kartensitzung exklusiv verwendet        |
| 4094 | Technical | Error | Timeout beim Kartenzugriff aufgetreten                              |

3110

3111 [ $\leq$ ]

3112 4.1.5.4.4 TUC\_KON\_021 „PIN entsperren“

3113 **TIP1-A\_4569-02 - TUC\_KON\_021 „PIN entsperren“**3114 Der Konnektor MUSS den technischen Use Case „PIN entsperren“ gemäß TUC\_KON\_021  
3115 umsetzen.

3116

3117 **Tabelle 66: TAB\_KON\_236 – TUC\_KON\_021 „PIN entsperren“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_021 „PIN entsperren“   |
| Beschreibung   | Dieser Use Case setzt den Fehlbedienungs­zähler für diese PIN in der Karte auf seinen Anfangswert zurück und es wird optional eine neue PIN gesetzt.<br>Remote-PIN-Eingabe wird dabei automatisch unterstützt. |
| Auslöser       | <ul style="list-style-type: none"> <li>Aufruf der Operation UnblockPin des CardService (siehe 4.1.5.5.4) durch das Clientsystem.</li> </ul>  |
| Vorbedingungen | Karte unterstützt die übergebene pinRef  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>cardSession<br/>CardSession der Karte, deren PIN entsperret werden soll)</li> <li>workplaceId</li> </ul>  |

|                |   |
|----------------|---|
|                | <ul style="list-style-type: none"> <li>pinRef<br/>(Referenz auf die zu entsperrende PIN, gemäß der Objektsystem-Spezifikation des jeweiligen Kartentyps)</li> </ul> <p>setNewPin (true/false) - Angabe, ob eine neue PIN gesetzt oder die aktuelle weiterverwendet werden soll. Default = false</p> <p>sourceCardSession - <i>optional/wenn eGK G1+</i><br/>(CardSession der Karte, die für die Card-to-Card-Authentisierung bei Entsperrung der PIN einer eGK der Generation 1+ verwendet werden soll)</p>   |
| Komponenten    | Karte, Kartenterminal, Konnektor  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>result [PukResult])<br/>(Ergebnis der PIN-Entsperrung durch PUK-Eingabe)</li> <li>leftTries - <i>optional/verpflichtend, wenn pukStatus = REJECTED</i><br/>(verbleibende Versuche des PUKs)</li> </ul>   |
| Standardablauf | <ol style="list-style-type: none"> <li>Ermittle Card = CM_CARD_LIST(Target.CardHandle)</li> <li>Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>Wenn TUC_KON_022 „Liefere PIN-Status“ {<br/>cardSession;<br/>pinRef } &lt;&gt; ( „BLOCKED“ oder "TRANSPORT_PIN" )<br/>dann beende TUC erfolgreich.</li> <li>Wenn pinRef=PIN.AMTS_REP, dann <ol style="list-style-type: none"> <li>setNewPin = true</li> <li>rufe TUC_KON_012 „PIN verifizieren“ {<br/>cardSession;<br/>workplaceId;<br/>pinRef=PIN.CH;<br/>actionName= „“;<br/>mandatorisch}</li> </ol> </li> <li>Wenn Card.TYP=eGK UND Card.Version=Generation1+,<br/>dann Aufruf TUC_KON_005 „Card-to-Card authentisieren“ {<br/>sourceCardSession;<br/>targetCardSession=cardSession;<br/>AuthMode =einseitig }.<br/>Falls keine sourceCardSession angegeben ist, kann die CardSession der für den Mandanten verwalteten SMC-B verwendet werden.</li> <li>Ermittle PinInputKT: Wenn Card.ctId ein dem Arbeitsplatz(workplaceId) lokal zugeordnetes Kartenterminal ist (siehe Relation [6], Kapitel 4.1.1.1) <ol style="list-style-type: none"> <li>Setze PinInputKT = Card.CtID</li> </ol> </li> </ol> |

|  |   |
|--|---|
|  | <p>b. sonst „lokales Kartenterminal, das für die Remote-PIN-Eingabe zu verwenden ist“: PinInputKT = Arbeitsplatz(workplaceId).remote-PIN-KT(mandantId)</p> <p>7. Atomare Operation: PIN entsperren inkl. Eventing und Ergebnisvermerk</p> <p>a. Rufe TUC_KON_256 {<br/> topic = „CARD/PIN/CHANGE_STARTED“;<br/> eventType = Op;<br/> severity = Info;<br/> parameters = („CardHandle=\$, CardType=\$,<br/> ICCSN=\$, CtID=\$; SlotID=\$, PinRef=\$,<br/> PinInputCtID=\$PinInputKT“);<br/> doLog=false}</p> <p>b. PIN-Entsperrung mit Display Messages<br/> entsprechend TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal.<br/> Wenn PinInputKT=Card.CtID, dann PIN-Änderung direkt an Card.CtID, ansonsten Remote-PIN-Eingabe gemäß (TIP1-A_5012)</p> <ul style="list-style-type: none"> <li>Für pinRef == PIN.QES über „PERFORM VERIFICATION“ [SICCT] mit dem eingebetteten Kommando Reset Retry Counter in der Variante P1=1 (keine neue PIN setzen).</li> <li>Für pinRef&lt;&gt;PIN.QES<br/> wenn setNewPin = false,<br/> dann über PERFORM VERIFICATION“ [SICCT],<br/> sonst über „MODIFY VERIFICATION DATA“ [SICCT].<br/> Das mit dem SICCT-Kommando als Command-To-Perform mitgesandte „Reset Retry Counter“ wird entsprechend dem Wert von setNewPIN parametrisiert.</li> </ul> <p>c. Setze result in Abhängigkeit von Ergebnis Perform Verification bzw. Modify VerificationData:</p> <ul style="list-style-type: none"> <li>result = OK für erfolgreiche Entsperrung</li> <li>result = ERROR für Nutzer-Abbruch oder Bearbeitungsfehler (siehe Fehlerfälle)</li> <li>result = REJECTED für falsche PUK;</li> <li>result = BLOCKED für gesperrte PUK; (bei Kartenantwort '63 C0')</li> </ul> <p>d. Rufe TUC_KON_256 {<br/> topic=„CARD/PIN/CHANGE_FINISHED“;<br/> eventType=Op; severity=Info;<br/> parameters = („CardHandle=\$; CardType=\$;<br/> ICCSN=\$;CtID=\$; SlotID=\$; PinRef=\$;<br/> PinInputCtID=\$PinInputKT; Result=\$“);<br/> doLog=false }</p> <p>8. Liefere result und ggf. leftTries zurück</p> |
|--|---|

|                                |   |
|--------------------------------|---|
| Varianten/<br>Alternativen     | Schritt 4: Für eGK G2.0 gilt:<br>Wenn pinRef=PIN.AMTS_REP, dann<br>rufe TUC_KON_012 „PIN verifizieren“ {<br>cardSession;<br>workplaceId;<br>pinRef=MRPIN.AMTS;<br>actionName= „“;<br>mandatorisch}  |
| Fehlerfälle                    | Schritt 7 wird mit allen Teilschritten durchlaufen. Ein als Fehler ausgewiesener Zustand führt erst nach Schritt 7d zum Abbruch des TUCs.<br>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094<br>(→2) Karte wird in einer anderen Kartensitzung exklusiv verwendet, Fehlercode 4093<br>(→5) sourceCardSession benötigt aber leer, Fehlercode 4071<br>(→6b) kein Remote-PIN-KT zugeordnet, Fehlercode 4092<br>(→6b) Card.TYP=eGK und Card.CtID ist nicht dem durch workplaceId bezeichneten Arbeitsplatz lokal zugeordnet: Fehlercode 4053<br>(→7b) blockierte PUK: Fehlercode 4064<br>(→7b) neue PIN zu kurz/lang: Fehlercode 4068<br>(→7b) zweite neue PIN<> erste neue PIN: Fehlercode 4067<br>(→7b) Timeout bei PIN Eingabe: Fehlercode 4043.<br>(→7b) Abbruch durch Nutzer: Fehlercode 4049.<br>(→7b) Ist das Kartenterminal oder Teile davon (PIN-Pad, Display) durch einen anderen Vorgang reserviert: Fehlercode 4060<br>(→7b) Karte/Kartenterminal antwortet mit einer spezifischen Fehlermeldung, Fehlercode <gemäß [gemSpec_COS]/[SICCT]><br>(→7b) Ungültige PIN-Referenz; Fehlercode 4072.<br>Zusätzliche Fehlerfälle ergeben sich aus der Remote-PIN-Eingabe gemäß (TIP1-A_5012) |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

3118

**Tabelle 67: TAB\_KON\_193 Fehlercodes TUC\_KON\_021 „PIN entsperren“**

| Fehlercode  | ErrorType | Severity | Fehlertext |
|---|-----------|----------|------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |            |

|      |           |         |   |
|------|-----------|---------|---|
| 4043 | Technical | Warning | Timeout bei der PIN-Eingabe   |
| 4049 | Technical | Error   | Abbruch durch den Benutzer  |
| 4053 | Security  | Error   | Remote-PIN nicht möglich  |
| 4060 | Technical | Error   | Ressource belegt  |
| 4064 | Security  | Error   | alte PIN bereits blockiert (hier: PUK)                              |
| 4067 | Security  | Error   | neue PIN nicht identisch  |
| 4068 | Security  | Error   | neue PIN zu kurz/zu lang  |
| 4072 | Technical | Error   | ungültige PIN-Referenz <i>PinRef</i>                                |
| 4092 | Technical | Error   | Remote-PIN-KT benötigt aber für diesen Arbeitsplatz nicht definiert |
| 4093 | Technical | Error   | Karte wird in einer anderen Kartensitzung exklusiv verwendet        |
| 4094 | Technical | Error   | Timeout beim Kartenzugriff aufgetreten                              |

3119

3120

3121 [ $\leq$ ]

3122

3123 4.1.5.4.5 TUC\_KON\_022 „Liefere PIN-Status“

3124 **TIP1-A\_4570 - TUC\_KON\_022 „Liefere PIN-Status“**3125 Der Konnektor MUSS den technischen Use Case „Liefere PIN-Status“ gemäß  
3126 TUC\_KON\_022 umsetzen.3127 **Tabelle 68 TAB\_KON\_532 – TUC\_KON\_022 „Liefere PIN-Status“**

| Element | Beschreibung |
|---------|--------------|
|---------|--------------|

|                |  |
|----------------|--|
| Name           | TUC_KON_022 „Liefere PIN-Status“   |
| Beschreibung   | Dieser Use Case prüft den Zustand eines PIN-Objekts einer Karte im Kontext einer CardSession.  |
| Auslöser       | <ul style="list-style-type: none"> <li>• Aufruf des Use Case im Rahmen von technischen Use Cases der Basisdienste des Konnektors</li> <li>• Aufruf des Use Cases durch ein Fachmodul im Konnektor</li> <li>• Aufruf der Operation GetPinStatus des CardService (siehe 4.1.5.5.1) durch das Clientsystem.</li> </ul>  |
| Vorbedingungen | Karte unterstützt die übergebene pinRef  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• cardSession</li> <li>• pinRef<br/>(Pin-Referenz der angefragten PIN, gemäß der Objektsystem-Spezifikation des jeweiligen Kartentyps)</li> </ul>   |
| Komponenten    | Karte, Kartenterminal, Konnektor   |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>• pinStatus [PinStatus]</li> <li>• leftTries – <i>optional/verpflichtend, wenn pinStatus = VERIFYABLE</i><br/>(Anzahl der verbleibenden Versuche für die Verifikation der PIN)</li> </ul>   |
| Standardablauf | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>3. pinRef in CardSession.AUTHSTATE vorhanden: <ol style="list-style-type: none"> <li>a) Ja: Setze pinStatus = VERIFIED oder DISABLED (wie in AUTHSTATE)</li> <li>b) Nein: Aufruf der Kartenoperation „GET PIN STATUS“, Antwort der Karte wird ausgewertet: <ol style="list-style-type: none"> <li>a. '90 00' (NoError: Verifiziert): pinStatus = VERIFYABLE (da nicht in dieser CardSession verifiziert)</li> <li>b. '62 C1': pinStatus = TRANSPORT_PIN</li> <li>c. '62 C7': pinStatus = EMPTY_PIN (Leer-PIN)</li> <li>d. '63 Cx': pinStatus = VERIFYABLE (mit <math>1 \leq x \leq 3</math>);<br/>LeftTries=x</li> <li>e. '63 C0': pinStatus = BLOCKED; leftTries=0</li> <li>f. '62 D0': pinStatus = DISABLED (Verifikation nicht erforderlich, da PIN-Schutz ausgeschaltet); cardSession.AUTHSTATE aktualisieren</li> <li>g. Antwortet die Karte mit einer Fehlermeldung, bricht der TUC ab.</li> </ol> </li> </ol> </li> </ol> |

|                            |  |
|----------------------------|--|
|                            | Liefere leftTries nur in den Fällen d und e zurück.  |
| Varianten/<br>Alternativen |  |
| Fehlerfälle                | * Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden,<br>Fehlercode 4094<br>(→3b) pinRef nicht gefunden: Fehlercode 4072 |
| Zugehörige<br>Diagramme    | keine  |

3128 **Tabelle 69: TAB\_KON\_091 Fehlercodes TUC\_KON\_022 „Liefere PIN-Status“**

| Fehlercode  | ErrorType | Severity | Fehlertext                                 |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4072  | Technical | Error    | ungültige PIN-Referenz <code>PinRef</code> |
| 4094  | Technical | Error    | Timeout beim Kartenzugriff aufgetreten     |

3129  
3130 [**<=**]

3131 4.1.5.4.6 TUC\_KON\_027 „PIN-Schutz ein-/ausschalten“

3132 **TIP1-A\_5486 - TUC\_KON\_027 „PIN-Schutz ein-/ausschalten“**

3133 Der Konnektor MUSS den technischen Use Case TUC\_KON\_027 „PIN-Schutz ein-  
3134 /ausschalten“ umsetzen.

3135 **Tabelle 70: TAB\_KON\_240 - TUC\_KON\_027 „PIN-Schutz ein-/ausschalten“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_027 „PIN-Schutz ein-/ausschalten“  |
| Beschreibung   | Schaltet das Erfordernis, die PIN zu verifizieren, ein bzw. aus.<br>Diese Operation wird nur unterstützt für PINs der EGK G2 gemäß [gemSpec_eGK_ObjSys]; für sie können folgende Kommandos auf das Passwortobjekt angewendet werden: <ul style="list-style-type: none"> <li>DISABLE VERIFICATION REQUIREMENT</li> <li>ENABLE VERIFICATION REQUIREMENT</li> </ul> |
| Auslöser       | <ul style="list-style-type: none"> <li>Aufruf durch ein Fachmodul</li> <li>Aufruf der Operationen EnablePin und DisablePin des CardService durch das Clientsystem.</li> </ul>  |
| Vorbedingungen | Karte unterstützt die übergebene pinRef  |



|                |  |
|----------------|--|
| Eingangsdaten  | <ul style="list-style-type: none"> <li>cardSession<br/>(CardSession einer EGK G2)</li> <li>pinRef<br/>(PIN-Referenz der ab-/anzuschaltenden PIN, gemäß der Objektsystem-Spezifikation des jeweiligen Kartentyps)</li> <li>enable [Boolean]<br/>(enable = true: Erfordernis der Benutzerverifikation einschalten;<br/>enable = false: Erfordernis der Benutzerverifikation abschalten)</li> </ul>   |
| Komponenten    | Konnektor  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>pinResult [PinResult]<br/>(Ergebnis von PIN-Schutz ein-/ausschalten durch PIN-Eingabe)</li> <li>leftTries – <i>optional/verpflichtend nach fehlerhafter PIN</i><br/>(verbleibende Versuche)</li> </ul>  |
| Standardablauf | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer im Besitz des Karten-Locks ist.</li> <li>3. Prüfe Card.Type = EGK und Generation <math>\geq 2</math></li> <li>4. Prüfe pinRef = MRPIN.AMTS und Card.Type = EGK und Generation <math>&gt; 2.0</math></li> <li>5. Wenn enable <ol style="list-style-type: none"> <li>A: =true: <p>Atomare Operation: PIN bearbeiten inkl. Eventing und Ergebnisvermerk</p> <ol style="list-style-type: none"> <li>a. Rufe TUC_KON_256 {<br/>topic = „CARD/PIN/ENABLE_STARTED“;<br/>eventType = Op;<br/>severity = Info;<br/>parameters = („CardHandle=\$, CardType=\$,<br/>ICCSN=\$, CtID=\$, SlotID=\$, PinRef=\$,<br/>PinInputCtID=\$PinInputKT“);<br/>doLog = false }</li> <li>b. Aufruf des Kartenterminalkommandos „SICCT<br/>PERFORM<br/>VERIFICATION“ mit der Kartenoperation „ENABLE<br/>VERIFICATION REQUIREMENT“ als Command-To-Perform. Es ist der Parameter P1='00' (mit Benutzerverifikation) zu verwenden. Die Anzeige am KT erfolgt entsprechend TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal. Ersetze in displayMessage „ANW“<br/>entsprechend<br/>ANW(pinRef) gemäß Tabelle TAB_KON_838.</li> <li>c. Setze pinResult in Abhängigkeit von Ergebnis Perform Verification: <ul style="list-style-type: none"> <li>- pinResult = OK für erfolgreiche Änderung</li> <li>- pinResult = ERROR für Nutzer-Abbruch oder Bearbeitungsfehler (siehe Fehlerfälle)</li> <li>- pinResult = REJECTED für falsche PIN;<br/>leftTries = x (bei Kartenantwort '63 Cx', x &gt; 0)</li> </ul> </li> </ol> </li> </ol> </li> </ol> |

|  |   |
|--|---|
|  | <ul style="list-style-type: none"> <li>- pinResult = BLOCKED für gesperrte PIN (bei Kartenantwort '63 C0')</li> </ul> <p>d. Rufe TUC_KON_256 {<br/> topic = „CARD/PIN/ENABLE_FINISHED“;<br/> eventType = Op;<br/> severity = Info;<br/> parameters = („CardHandle=\$, CardType=\$, ICCSN=\$, CtID=\$, SlotID=\$, PinRef=\$, PinInputCtID=\$PinInputKT“);<br/> doLog = false }</p> <p>B: =false:<br/> Atomare Operation: PIN bearbeiten inkl. Eventing und Ergebnisvermerk</p> <p>a. Rufe TUC_KON_256 {<br/> topic = „CARD/PIN/DISABLE_STARTED“;<br/> eventType = Op;<br/> severity = Info;<br/> parameters = („CardHandle=\$, CardType=\$, ICCSN=\$, CtID=\$, SlotID=\$, PinRef=\$, PinInputCtID=\$PinInputKT“);<br/> doLog = false }</p> <p>b. Aufruf des Kartenterminalkommandos „SICCT PERFORM VERIFICATION“ mit der Kartenoperation „DISABLE VERIFICATION REQUIREMENT“ als Command-To-Perform. Es ist der Parameter P1='00' (mit Benutzerverifikation) zu verwenden. Die Anzeige am KT erfolgt entsprechend TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal. Ersetze in displayMessage „ANW“ entsprechend ANW(pinRef) gemäß Tabelle TAB_KON_838.</p> <p>c. Setze pinResult in Abhängigkeit von Ergebnis Perform Verification:</p> <ul style="list-style-type: none"> <li>- pinResult = OK für erfolgreiche Änderung</li> <li>- pinResult = ERROR für Nutzer-Abbruch oder Bearbeitungsfehler (siehe Fehlerfälle)</li> <li>- pinResult = REJECTED für falsche PIN;<br/> leftTries = x (bei Kartenantwort '63 Cx', x &gt; 0)</li> <li>- pinResult = BLOCKED für gesperrte PIN</li> </ul> <p>d. Rufe TUC_KON_256 {<br/> topic = „CARD/PIN/DISABLE_FINISHED“;<br/> eventType = Op;<br/> severity = Info;<br/> parameters = („CardHandle=\$, CardType=\$, ICCSN=\$, CtID=\$, SlotID=\$, PinRef=\$, PinInputCtID=\$PinInputKT“);<br/> doLog=false}</p> <p>6. Liefere pinResult und leftTries zurück</p> |
|--|---|

|                                |   |
|--------------------------------|---|
| Varianten/<br>Alternativen     | (->3) zur Optimierung kann vor Schritt 5 der PIN-Schutz geprüft werden:<br>a. pinStatus=TUC_KON_022 „Liefere PIN-Status“ { cardSession; pinRef }<br>b. Wenn pinStatus<>DISABLED und enable=true, dann pinResult=OK und -> weiter in Schritt 6<br>c. Wenn pinStatus=DISABLED und enable=false, dann pinResult=OK und -> weiter in Schritt 6  |
| Fehlerfälle                    | (→2) Karte ist fremd reserviert: Fehlercode 4093<br>(→3) Karte ist keine eGK der Generation 2 oder höher: Fehlercode 4209<br>(→4) PIN nicht gefunden; Karte ist eGK G2.0: Die Operation „PIN-Schutz ein-/ausschalten“ wird für MRPIN.AMTS nicht unterstützt: Fehlercode 4072<br>(→5) Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden: Fehlercode 4094<br>(→5) PIN nicht gefunden: Fehlercode 4072<br>(→5) PIN gesperrt: Fehlercode 4063<br>(→5) Zugriffsbedingung nicht erfüllt (PIN nicht abschaltbar): Fehlercode 4085 |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | keine   |

3136 **Tabelle 71: TAB\_KON\_838 Mapping von pinRef auf ANW**

| pinRef     | ANW (max. 16 Zeichen) |
|------------|-----------------------|
| MRPIN.NFD  | Notfalldaten          |
| MRPIN.DPE  | Pers.Erklärungen      |
| MRPIN.AMTS | Medikationsdaten      |
| MRPIN.GDD  | PIN•GDD               |

3137  
3138 Hinweis zu TAB\_KON\_838: Leerzeichen werden als "•" dargestellt.

3139 **Tabelle 72: TAB\_KON\_241 Fehlercodes TUC\_KON\_027 „PIN-Schutz ein/ausschalten“**

| Fehlercode  | ErrorType | Severity | Fehlertext |
|---|-----------|----------|------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |            |

|      |           |       |   |
|------|-----------|-------|---|
| 4063 | Security  | Error | PIN bereits blockiert (BLOCKED)   |
| 4072 | Technical | Error | ungültige PIN-Referenz <code>PinRef</code>                                      |
| 4085 | Security  | Error | Zugriffsbedingung nicht erfüllt   |
| 4093 | Technical | Error | Karte wird in einer anderen Kartensitzung exklusiv verwendet                    |
| 4094 | Technical | Error | Timeout beim Kartenzugriff aufgetreten  |
| 4209 | Technical | Error | Kartentyp <code>%CardType%</code> wird durch diese Operation nicht unterstützt. |

3140  
3141  
3142

[<=]

3143 4.1.5.4.7 TUC\_KON\_023 „Karte reservieren“

3144 **TIP1-A\_4571 - TUC\_KON\_023 „Karte reservieren“**

3145 Der Konnektor MUSS den technischen Use Case „Karte reservieren“ gemäß  
3146 TUC\_KON\_023 umsetzen.

3147

3148 **Tabelle 73: TAB\_KON\_533 - TUC\_KON\_023 „Karte reservieren“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | <p>TUC_KON_023 „Karte reservieren“</p> <p>Dem Aufrufer des TUC_KON_023 wird beim Reservieren (DoLock=Ja) der Karte zur ausschließlichen Nutzung ein Lock zugeordnet. Wird der TUC-KON_023 mit diesem Lock zum Freigeben der Reservierung (DoLock=Nein) aufgerufen, dann erlischt das Lock und die ausschließliche Nutzung wird beendet. Der Scope der Kartenreservierung wird vom Aufrufer des TUC_KON_023 gesteuert. Das Lock ist Konnektor-intern. Es darf nicht außerhalb des Konnektors referenzierbar sein. Zwei verschiedene Operationsaufrufe am Konnektor dürfen nie ein identisches Lock haben.</p> <p>Der Konnektor MUSS sicherstellen, dass auch im Fehlerfall die Reservierung zu einem Lock aufgehoben wird. Ein Lock darf nicht dauerhaft bestehen.</p> |
| Beschreibung   | Reservierung der Karte  |
| Auslöser       | <ul style="list-style-type: none"> <li>Aufruf des Use Case im Rahmen von technischen Use Cases der Basisdienste des Konnektors</li> <li>Aufruf des Use Cases durch ein Fachmodul im Konnektor</li> </ul>  |
| Vorbedingungen | Keine   |
| Eingangsdaten  | <ul style="list-style-type: none"> <li><code>cardSession</code></li> </ul>  |

|                                   |   |
|-----------------------------------|---|
|                                   | <ul style="list-style-type: none"> <li>doLock [Boolean]<br/>(Zielzustand der Karte; true = reserviert, false = freigegeben)</li> </ul>  |
| Komponenten                       | Konnektor   |
| Ausgangsdaten                     | Keine   |
| Standardablauf                    | 1. Ermittle Card = CM_CARD_LIST(cardSession)<br>2. Wenn doLock<br>A: = true:<br>i. Prüfe, dass der zur cardSession gehörenden Karte kein Lock zugeordnet ist<br>ii. Dem Aufrufer wird ein Lock auf die zur cardSession gehörende Karte zugeordnet. Es wird nicht explizit als Ausgangsdatum modelliert, sondern der Aufrufer hat das Lock durch die Zuordnung, muss es aber nicht verwalten.<br>B: = false:<br>i. Prüfe, dass der Aufrufer für die zur cardSession gehörende Karte ein Lock hat.<br>ii. Das der Karte zugeordnete Lock wird gelöscht. |
| Varianten/<br>Alternativen        | Keine   |
| Fehlerfälle                       | (→2Ai) Karte bereits reserviert, Fehlercode 4093<br>(→2Bi) Karte nicht durch Aufrufer reserviert, Fehlercode 4001   |
| Nichtfunktionale<br>Anforderungen | Keine   |
| Zugehörige<br>Diagramme           | Keine   |

3149 **Tabelle 74: TAB\_KON\_534 Fehlercodes TUC\_KON\_023 „Karte reservieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext               |
|---|-----------|----------|--------------------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |                          |
| 4001  | Technical | Error    | interner Fehler          |
| 4093  | Technical | Error    | Karte bereits reserviert |

3150  
3151 [**<=**]

#### 3152 4.1.5.4.8 TUC\_KON\_005 „Card-to-Card authentisieren“

3153 Die C2C-Authentisierung erfolgt konform zu den in [gemSpec\_COS#15] festgelegten  
3154 Authentisierungsprotokollen.

3155 **Definition Quellkarte/Zielkarte:**

3156 Bei einseitiger Card-to-Card-Authentisierung ohne Aushandlung eines Session Key ist die  
 3157 Quellkarte diejenige, die die Rolle des Karteninhabers bzw. der Organisation gemäß  
 3158 [gemSpec\_PKI\_TI#Tab\_PKI\_254] gegenüber der anderen Karte nachweist, z. B. der HBA  
 3159 bei der Freischaltung einer eGK.

3160 Bei gegenseitiger Card-to-Card-Authentisierung ohne Aushandlung eines Session Key  
 3161 erfolgen nach einander zwei einseitige Card-to-Card-Authentisierungen mit vertauschten  
 3162 Rollen. Quell- und Zielkarte habe daher für den Gesamtprozess keine nähere Bedeutung.

3163 Bei Card-to-Card-Authentisierung mit Aushandlung eines Session Key ist die Quellkarte  
 3164 diejenige, die die SM-APDUs produzieren kann, also die SMC (-KT oder -K).

3165 Die Zielkarte ist jeweils die Karte, die nicht die Quellkarte ist.

3166

#### 3167 **TIP1-A\_4572 - TUC\_KON\_005 „Card-to-Card authentisieren“**

3168 Der Konnektor MUSS den technischen Use Case „Card-to-Card authentisieren“ gemäß  
 3169 TUC\_KON\_005 umsetzen.

3170 Die Card-to-Card-Authentisierung zwischen zwei Karten, bei der eine Karte der  
 3171 Generation 1+ angehört MUSS das RSA-Verfahren verwenden.

3172 Die Card-to-Card-Authentisierung zwischen zwei Karten der Generation 2 MUSS das  
 3173 Verfahren der elliptischen Kurven verwenden.

3174

#### 3175 **Tabelle 75: TAB\_KON\_096 – TUC\_KON\_005 „Card-to-Card authentisieren“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_005 „Card-to-Card authentisieren“   |
| Beschreibung   | Durchführung einer Card-to-Card-Authentisierung   |
| Auslöser       | <ul style="list-style-type: none"> <li>• Aufruf des Use Case im Rahmen von technischen Use Cases der Basisdienste des Konnektors</li> <li>• Aufruf des Use Cases durch ein Fachmodul im Konnektor</li> </ul>  |
| Vorbedingungen | Wert von Source_CARDSESSION.AUTHSTATE: wenn Quellkarte<br>a) ein HBA ist: CHV; PIN.CH, verifiziert<br>b) eine SMC-B ist: CHV; PIN.SMC verifiziert   |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• sourceCardSession (Quellkarte)</li> <li>• targetCardSession (Zielkarte)</li> <li>• authMode (gemäß Tabelle TAB_KON_673)</li> </ul>   |
| Komponenten    | Karten, Konnektor, Kartenterminal   |
| Ausgangsdaten  | Keine   |
| Standardablauf | 1. Ermittle sCard = CM_CARD_LIST(sourceCardSession)<br>2. Ermittle tCard = CM_CARD_LIST(targetCardSession)<br>3. Prüfe, dass der <u>Quellkarte</u> entweder kein Lock zugeordnet ist oder der Aufrufer im Besitz auf das Lock der Quellkarte ist.<br>Prüfe, dass der <u>Zielkarte</u> entweder kein Lock zugeordnet |

|                            |  |
|----------------------------|--|
|                            | <p>ist oder der Aufrufer im Besitz auf das Lock der Zielkarte ist.</p> <ol style="list-style-type: none"> <li>4. Prüfe Aufrufparameter auf erlaubte Kombination gemäß Tabelle TAB_KON_674</li> <li>5. Wenn das zu verwendende CV-Zertifikat der Quellkarte ein CV-Zertifikat der Generation 2 oder höher ist, dann prüfe sein Ausstellungsdatum (CED) gegen die aktuelle Zeit</li> <li>6. Wenn Card.TYP=eGK UND Card.Version=Generation1+, dann prüfe, ob aktuelles System-Datum &lt; 01.01.2019 ist</li> <li>7. Wähle Key-Referenzen gemäß Tabelle TAB_KON_674</li> <li>8. Prüfe pinRef/keyRef in sCard.CARDSESSION.AUTHSTATE und tCard.CARDSESSION.AUTHSTATE für adressierte Schlüssel wie in Zugriffsbedingung der Karten definiert vorhanden</li> <li>9. Durchführung der Authentisierung gemäß Tabelle TAB_KON_673 mit Key-Referenzen gemäß Tabelle TAB_KON_674</li> <li>10. Ergänze targetCardSession.AUTHSTATE mit tKeyRef und Rolle aus sKeyRef (CHA bzw. CHAT aus dem EndEntity-CV-Zertifikat der Quellkarte)</li> </ol>  |
| Varianten/<br>Alternativen | <p>(→9) Wenn der für die CA-Zertifikatsprüfung zu selektierende CVC-Root-Key auf der Zielkarte nicht vorhanden ist (Returncode des Kartenkommandos „MANAGE SECURITY ENVIRONMENT“ ist '6A 88'), dann muss der Konnektor:</p> <ol style="list-style-type: none"> <li>a) das oder die passenden Cross-CV-Zertifikate aus dem Truststore auswählen</li> <li>b) mit dem Kartenkommando „PSO Verify Certificate“ jedes ausgewählte Cross-CV-Zertifikat durch die Zielkarte prüfen lassen.<br/>Dadurch wird der im Cross-CV-Zertifikat enthaltene öffentliche Schlüssel an die Zielkarte übertragen. Die Zielkarte speichert den darin enthaltenen neuen CVC-Root-Key.</li> <li>c) den neuen CVC-Root-Key auf der Zielkarte selektieren</li> <li>d) den Standardablauf der C2C-Authentisierung fortsetzen</li> </ol> <p>(→9) Wenn tCard.TYPE=EGK und AuthMode=gegenseitig, dann Echtheitsprüfung der eGK durch den Konnektor:</p> <ol style="list-style-type: none"> <li>a) Freischaltung der EGK durch den HBA/die SMC-B:<br/>Durchführen der Authentisierung gemäß Tabelle TAB_KON_673 mit Key-Referenzen gemäß Tabelle TAB_KON_674 aber mit AuthMode=einseitig</li> <li>b) Konnektor liest das CA-Zertifikat EF.C.CA_eGK.CS (G1+) bzw.<br/>C.CA_eGK.CS.E256 (G2)</li> <li>c) Konnektor liest das End-Entity-Zertifikat der EGK EF.C.eGK.AUT_CVC (G1+) bzw. EF.C.eGK.AUT_CVC.E256 (G2)</li> <li>d) Konnektor prüft das CVC-EE-Zertifikat mit TUC_KON_042</li> </ol> |

|                                   |  |
|-----------------------------------|--|
|                                   | <p>„CV-<br/>Zertifikat prüfen“ {<br/>    certificate = C.eGK.AUT_CVC/C.eGK.AUT_CVC.E256;<br/>    caCertificate = C.CA_eGK.CS/C.CA_eGK.CS.E256 }<br/>e) Konnektor erzeugt Zufallszahl<br/>f) Konnektor selektiert den PrK.eGK.AUT_CVC (G1+) bzw.<br/>PrK.eGK.AUT_CVC.E256 (G2) und stellt abhängig von der<br/>Version<br/>    der eGK den Algorithmus auf der eGK ein (MSE Set)<br/>g) Konnektor sendet Konkatenation aus Zufallszahl und<br/>CARD.ICCSN mit dem Befehl „INTERNAL AUTHENTICATE“ an<br/>die<br/>    eGK<br/>h) Konnektor wertet das von der Karte erhaltene Chifftrat aus</p>   |
| Fehlerfälle                       | <p>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD<br/>Sekunden, Fehlercode 4094<br/>(→3) Eine Karte ist fremd reserviert, Fehlercode 4093<br/>(→5) Zertifikat der Quellkarte fehlerhaft. Ausstellungsdatum liegt<br/>in der Zukunft; Fehlercode 4233<br/>(→6) eGK G1+ ausgealtert, Fehlercode 4192<br/>(→8) Nötige PIN, bzw. KeyRef ist nicht verifiziert, Fehlercode<br/>4085<br/>(→9) Je nachdem, welche Karte den Fehler verursachte, wird<br/>zum ursprünglichen Fehler (Fehlercode gemäß [gemSpec_COS])<br/>im Error-Trace (welcher an erster Stelle im Falle des HBA z. B.<br/>bereits ein Fehler bezüglich PIN-Verifikation enthalten kann) noch<br/>ein weiterer mit Code 4056 oder 4057 hinzugefügt. Kann der<br/>Fehler nicht eindeutig einer der beiden Karten zugeordnet so wird<br/>Error-Code 4048 verwendet.</p> |
| Nichtfunktionale<br>Anforderungen | keine  |
| Zugehörige<br>Diagramme           | Keine  |

3176 **Tabelle 76: TAB\_KON\_673 AuthMode für C2C**

| AuthMode       | Definition des Ablaufs   |
|----------------|--|
| einseitig      | Externe oder Interne Authentisierung<br>([gemSpec_COS#15.1] oder [gemSpec_COS#15.2], passend zu<br>den Zugriffsregeln der beteiligten CVC) |
| gegenseitig    | Card-2-Card-Authentisierung ohne Sessionkey-Aushandlung<br>([gemSpec_COS#15.3])  |
| gegenseitig+TC | Card-2-Card-Authentisierung mit Sessionkey-Aushandlung zur<br>Etablierung eines Trusted Channels ([gemSpec_COS#15.4])                      |



3177 **Tabelle 77: TAB\_KON\_674 Erlaubte Parameterkombinationen und resultierende CV-**  
 3178 **Zertifikate für C2C**

| Quellkarte          | Zielkarte | AuthMode           | sKeyRef  | tKeyRef                   | Fachlicher UseCase                                   |
|---------------------|-----------|--------------------|--|---------------------------|--|
| HBA<br>oder<br>SM-B | eGK G1+   | einseitig          | {HPC.AUTR_<br>CVC.R2048  <br>SMC.AUTR_<br>CVC.R2048} |                           | Freischaltung<br>eGK                                 |
| HBA<br>oder<br>SM-B | eGK G1+   | gegen<br>seitig    | {HPC.AUTR_<br>CVC.R2048  <br>SMC.AUTR_<br>CVC.R2048} | eGK.AUT_<br>CVC.R2048     | Freischaltung<br>eGK mit<br>Echtheits<br>prüfung eGK |
| HBA<br>oder<br>SM-B | eGK G2    | einseitig          | {HPC.AUTR_<br>CVC.E256  <br>SMC.AUTR_<br>CVC.E256}   |                           | Freischaltung<br>eGK                                 |
| HBA<br>oder<br>SM-B | eGK G2    | gegen<br>seitig    | {HPC.AUTR_<br>CVC.E256  <br>SMC.AUTR_<br>CVC.E256}   | eGK.AUT_<br>CVC.E256      | Freischaltung<br>eGK mit<br>Echtheits<br>prüfung eGK |
| SMC-K               | HBA       | gegen<br>seitig+TC | SAK.AUTD_<br>CVC.E256                                | HPC.AUTD_<br>SUK_CVC.E256 | DTBS-<br>Übertragung<br>bei QES                      |
| SMC-KT              | HBA       | gegen<br>seitig+TC | SMC.AUTD_<br>RPS_CVC.E256                            | HPC.AUTD_<br>SUK_CVC.E256 | Remote-PIN   |
| SMC-KT              | SM-B      | gegen<br>seitig+TC | SMC.AUTD_<br>RPS_CVC.E256                            | SMC.AUTD_<br>RPE_CVC.E256 | Remote-PIN   |

3179 **Tabelle 78: TAB\_KON\_535 Fehlercodes TUC\_KON\_005 „Card-to-Card authentisieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4048  | Technical | Error    | Fehler bei der C2C-Authentisierung                              |
| 4056  | Technical | Error    | Fehler bei der C2C-Authentisierung,<br>Quellkarte               |
| 4057  | Technical | Error    | Fehler bei der C2C-Authentisierung, Zielkarte                   |
| 4085  | Security  | Error    | Zugriffsbedingungen nicht erfüllt                               |
| 4093  | Technical | Error    | Karte wird in einer anderen Kartensitzung<br>exklusiv verwendet |
| 4094  | Technical | Error    | Timeout beim Kartenzugriff aufgetreten                          |
| 4192  | Security  | Error    | C2C mit eGK G1+ ab 01.01.2019 nicht mehr<br>gestattet           |
| 4233  | Security  | Error    | Ausstellungsdatum des Zertifikats liegt in der<br>Zukunft;      |

3180  
3181 [ $\leq$ ]

3182 4.1.5.4.9 TUC\_KON\_202 „LeseDatei“

3183 **TIP1-A\_4573 - TUC\_KON\_202 „LeseDatei“**

3184 Der Konnektor MUSS den technischen Use Case „LeseDatei“ gemäß TUC\_KON\_202  
3185 umsetzen.

3186

3187 **Tabelle 79: TAB\_KON\_218 – TUC\_KON\_202 „LeseDatei“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_202 „LeseDatei“   |
| Beschreibung   | Transparente Datei oder Teile davon lesen   |
| Auslöser       | <ul style="list-style-type: none"> <li>• Aufruf des Use Case im Rahmen von technischen Use Cases der Basisdienste des Konnektors</li> <li>• Aufruf des Use Cases durch ein Fachmodul im Konnektor</li> </ul>  |
| Vorbedingungen | Die Karte MUSS den nötigen Sicherheitszustand für den Zugriff besitzen  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• cardSession</li> <li>• fileIdentifier – <i>optional/verpflichtend, wenn kein sfid angegeben ist</i> (FID der zu lesenden Datei)</li> <li>• sfid – <i>optional/verpflichtend, wenn kein fileIdentifier angegeben ist</i> (Short File Identifier der zu lesenden Datei)</li> <li>• folder (Verzeichnis/Applikation auf der Karte, in dem sich die Datei befindet)</li> <li>• offset – <i>optional/nur verwendbar, wenn fileIdentifier angegeben ist</i> (Startposition innerhalb der Datei)</li> <li>• length – <i>optional</i> (Längenangabe, um den Zugriff auf Teile einer Datei einzuschränken)</li> </ul> |
| Komponenten    | Karte, Kartenterminal, Konnektor  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>• content (Gelesene Daten)</li> </ul>  |

|                                |   |
|--------------------------------|---|
| Standardablauf                 | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>3. Prüfe PinRef/KeyRef in CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden</li> <li>4. Selektiere Verzeichnis und Datei</li> <li>5. Lies Daten über Kartenkommando „READ BINARY“ unter Berücksichtigung von Offset- und Längenangaben</li> <li>6. Die gelesenen Daten werden an den Aufrufer zurückgegeben</li> </ol> |
| Varianten/<br>Alternativen     | Wenn Card.TYPE = KVK, sendet der Konnektor in diesem Fall ein "Read Binary" im Sinne von SICCT 1.2.1, 5.5.8.1 "Kommandos für synchrone Chipkarten".   |
| Fehlerfälle                    | <p>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094<br/>         (→2) Karte ist fremd reserviert, Fehlercode 4093<br/>         (→3) Nötige PIN ist nicht verifiziert, Fehlercode 4085<br/>         (→5) Verzeichnis deaktiviert, Fehlercode 4086<br/>         (→4-5) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;Kartenfehlercode gemäß [gemSpec_COS]&gt;</p>   |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

3188 **Tabelle 80: TAB\_KON\_536 Fehlercodes TUC\_KON\_202 „LeseDatei“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4085  | Security  | Error    | Zugriffsbedingungen nicht erfüllt                            |
| 4086  | Technical | Error    | Verzeichnis deaktiviert                                      |
| 4093  | Technical | Error    | Karte wird in einer anderen Kartensitzung exklusiv verwendet |
| 4094  | Technical | Error    | Timeout beim Kartenzugriff aufgetreten                       |

3189  
 3190  
 3191 [**<=**]

3192 4.1.5.4.10 TUC\_KON\_203 „SchreibeDatei“

3193 **TIP1-A\_4574 - TUC\_KON\_203 „SchreibeDatei,,**

3194 Der Konnektor MUSS den technischen Use Case „SchreibeDatei“ gemäß TUC\_KON\_203  
3195 umsetzen.

3196

3197 **Tabelle 81: TAB\_KON\_219 – TUC\_KON\_203 „SchreibeDatei“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_203 „SchreibeDatei“   |
| Beschreibung   | Daten in transparente Datei schreiben   |
| Auslöser       | <ul style="list-style-type: none"> <li>• Aufruf durch Fachmodul</li> </ul>  |
| Vorbedingungen | Die Karte MUSS den nötigen Sicherheitszustand für den Zugriff besitzen.   |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• cardSession</li> <li>• fileIdentifier – <i>optional/verpflichtend</i>, wenn kein sfid angegeben ist (FID der zu lesenden Datei)</li> <li>• sfid – <i>optional/verpflichtend</i>, wenn kein fileIdentifier angegeben ist (Short File Identifier der zu lesenden Datei)</li> <li>• folder (Verzeichnis/Applikation auf der Karte, in dem sich die Datei befindet)</li> <li>• offset – <i>optional</i> (Startposition innerhalb der Datei, default: 0)</li> <li>• length – <i>optional</i> (Längenangabe, um den Zugriff auf Teile einer Datei einzuschränken; default: alles ab offset)</li> <li>• dataToBeWritten (Zu schreibende Daten)</li> </ul> |
| Komponenten    | Karte, Kartenterminal, Konnektor  |
| Ausgangsdaten  | Keine   |

|                                   |   |
|-----------------------------------|---|
| Standardablauf                    | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>2. Prüfe Card.TYPE &lt;&gt; KVK</li> <li>3. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>4. Prüfe PinRef/KeyRef in CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden</li> <li>5. Selektiere Verzeichnis</li> <li>6. Selektiere Datei mittels SELECT mit P2='04' (Selektieren einer Datei, Antwortdaten mit FCP)</li> <li>7. Ermittle size (Größe der selektierten Datei in Byte) mit size = numberOfOctet aus FCP</li> <li>8. Wenn size - offset &gt;= Größe von dataToBeWritten in Byte,<br/>dann schreibe dataToBeWritten mittels Kartenkommando "UPDATE BINARY" unter Berücksichtigung von Offset- und Längenangaben</li> </ol> |
| Varianten/<br>Alternativen        | keine   |
| Fehlerfälle                       | <p>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094<br/>         (→2) Schreibzugriff auf KVK nicht gestattet, Fehlercode 4085<br/>         (→3) Karte ist fremd reserviert, Fehlercode 4093<br/>         (→4) Nötige PIN ist nicht verifiziert, Fehlercode 4085<br/>         (→5) Verzeichnis oder Datei existiert nicht, Fehlercode 4087<br/>         (→4-5) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;Kartenfehlercode gemäß [gemSpec_COS]&gt;<br/>         (→6) Ausgewählte Datei ist nicht transparent, Fehlercode 4089<br/>         (→6) Verzeichnis deaktiviert, Fehlercode 4086<br/>         (→8) dataToBeWritten sind größer als der zur Verfügung stehende Speicherplatz, Fehlercode 4247</p>  |
| Nichtfunktionale<br>Anforderungen | keine   |
| Zugehörige<br>Diagramme           | keine   |

3198

**Tabelle 82: TAB\_KON\_537 Fehlercodes TUC\_KON\_203 „Schreibe Datei“**

| Fehlercode  | ErrorType | Severity | Fehlertext                        |
|---|-----------|----------|-----------------------------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |                                   |
| 4085  | Security  | Error    | Zugriffsbedingungen nicht erfüllt |

|      |           |       |  |
|------|-----------|-------|--|
| 4086 | Technical | Error | Verzeichnis deaktiviert                                      |
| 4087 | Technical | Error | Datei nicht vorhanden  |
| 4089 | Technical | Error | Datei ist vom falschen Typ                                   |
| 4093 | Technical | Error | Karte wird in einer anderen Kartensitzung exklusiv verwendet |
| 4094 | Technical | Error | Timeout beim Kartenzugriff aufgetreten                       |
| 4247 | Technical | Error | Speicherplatz auf der Karte nicht ausreichend                |

3199

3200

3201 [ $\leq$ ]

3202 4.1.5.4.11 TUC\_KON\_204 „LöscheDateiInhalt“

3203 **TIP1-A\_5476 - TUC\_KON\_204 „LöscheDateiInhalt“**

3204 Der Konnektor MUSS den technischen Use Case „LöscheDateiInhalt“ gemäß  
 3205 TUC\_KON\_204 umsetzen.

3206

3207 **Tabelle 83: TAB\_KON\_204 – TUC\_KON\_204 „LöscheDateiInhalt“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_204 „LöscheDateiInhalt“  |
| Beschreibung   | Inhalt einer transparenten Datei löschen   |
| Auslöser       | <ul style="list-style-type: none"> <li>Aufruf des Use Cases durch ein Fachmodul im Konnektor</li> </ul>  |
| Vorbedingungen | Die Karte MUSS den nötigen Sicherheitszustand für den Zugriff besitzen   |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>cardSession</li> <li>fileIdentifier – <i>optional/verpflichtend, wenn kein sfid angegeben ist</i> (FID der zu bearbeitenden Datei)</li> <li>sfid – <i>optional/verpflichtend, wenn kein fileIdentifier angegeben ist</i> (Short File Identifier der zu bearbeitenden Datei)</li> <li>folder (Verzeichnis/Applikation auf der Karte, in dem sich die Datei befindet)</li> <li>offset – <i>optional</i> (Position, ab der der Inhalt gelöscht werden soll. Default: 0)</li> </ul> |
| Komponenten    | Karte, Kartenterminal, Konnektor   |

|                                   |  |
|-----------------------------------|--|
| Ausgangsdaten                     | keine  |
| Standardablauf                    | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>2. Prüfe Card.TYPE &lt;&gt; KVK</li> <li>3. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer im Besitz des Karten-Locks ist.</li> <li>4. Prüfe PinRef/KeyRef in CARDESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden</li> <li>5. Selektiere Verzeichnis und Datei</li> <li>6. Lösche Inhalt der selektierten Datei über Kartenkommando „ERASE BINARY“, ggf. ab angegebenem Offset, sonst ab Anfang</li> </ol>   |
| Varianten/<br>Alternativen        | keine  |
| Fehlerfälle                       | <p>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094<br/>         (→2) Karte ist keine eGK der Generation 2 oder höher: Fehlercode 4209 (→3) Karte ist fremd reserviert, Fehlercode 4093<br/>         (→4) Nötige PIN ist nicht verifiziert, Fehlercode 4085<br/>         (→5) Verzeichnis oder Datei existiert nicht, Fehlercode 4087<br/>         (→6) Ausgewählte Datei ist nicht transparent, Fehlercode 4089<br/>         (→6) Verzeichnis deaktiviert, Fehlercode 4086<br/>         (→5-6) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;Kartenfehlercode gemäß [gemSpec_COS]&gt;</p> |
| Nichtfunktionale<br>Anforderungen | Keine  |
| Zugehörige<br>Diagramme           | Keine  |

3208

**Tabelle 84: TAB\_KON\_785 Fehlercodes TUC\_KON\_204 „LöscheDateiInhalt“**

| Fehlercode  | ErrorType | Severity | Fehlertext                        |
|---|-----------|----------|-----------------------------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |                                   |
| 4085  | Security  | Error    | Zugriffsbedingungen nicht erfüllt |
| 4086  | Technical | Error    | Verzeichnis deaktiviert           |
| 4087  | Technical | Error    | Datei nicht vorhanden             |
| 4089  | Technical | Error    | Datei ist vom falschen Typ        |

|      |           |       |  |
|------|-----------|-------|--|
| 4093 | Technical | Error | Karte wird in einer anderen Kartensitzung exklusiv verwendet       |
| 4094 | Technical | Error | Timeout beim Kartenzugriff aufgetreten                             |
| 4209 | Technical | Error | Kartentyp %CardType% wird durch diese Operation nicht unterstützt. |

3209  
3210  
3211

[<=]

3212 4.1.5.4.12 TUC\_KON\_209 „LeseRecord“

3213 **TIP1-A\_4575 - TUC\_KON\_209 „LeseRecord“**

3214 Der Konnektor MUSS den technischen Use Case „LeseRecord“ gemäß TUC\_KON\_209  
3215 umsetzen.

3216

3217 **Tabelle 85: TAB\_KON\_538 – TUC\_KON\_209 „LeseRecord“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_209 „LeseRecord“  |
| Beschreibung   | Daten aus strukturierter Datei lesen  |
| Auslöser       | <ul style="list-style-type: none"> <li>Aufruf durch Fachmodul</li> </ul>  |
| Vorbedingungen | Die Karte MUSS den nötigen Sicherheitszustand für den Zugriff besitzen  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>cardSession</li> <li>fileIdentifier – <i>optional/verpflichtend, wenn kein sfid angegeben ist</i> (FID der zu lesenden Datei)</li> <li>sfid – <i>optional/verpflichtend, wenn kein fileIdentifier angegeben ist</i> (Short File Identifier der zu lesenden Datei)</li> <li>folder (Verzeichnis/Applikation auf der Karte, in dem sich die Datei befindet)</li> <li>recordNumber</li> </ul> |
| Komponenten    | Karte, Kartenterminal, Konnektor  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>content (Inhalt des Records)</li> </ul>  |



|                                |  |
|--------------------------------|--|
| Standardablauf                 | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt</li> <li>3. Prüfe PinRef/KeyRef in CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden</li> <li>4. Selektiere Verzeichnis und ggf. Datei</li> <li>5. Lies Daten über Kartenkommando „READ RECORD“ unter Berücksichtigung von recordNumber</li> <li>6. Rückgabe der Daten an den Aufrufer</li> </ol> |
| Varianten/<br>Alternativen     | keine  |
| Fehlerfälle                    | <p>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094<br/> (→2) Karte ist fremd reserviert, Fehlercode 4093<br/> (→3) Nötige PIN ist nicht verifiziert, Fehlercode 4085<br/> (→4) Verzeichnis oder Datei oder Record existiert nicht, Fehlercode 4087<br/> (→5) Wenn Karte WrongFileType liefert, Fehlercode 4089<br/> (→5) Verzeichnis deaktiviert, Fehlercode 4086<br/> (→4-5) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;Kartenfehlercode gemäß [gemSpec_COS]&gt;.</p>                   |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | keine  |

3218

**Tabelle 86: TAB\_KON\_539 Fehlercodes TUC\_KON\_209 „LeseRecord“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4085  | Security  | Error    | Zugriffsbedingungen nicht erfüllt                            |
| 4086  | Technical | Error    | Verzeichnis deaktiviert                                      |
| 4087  | Technical | Error    | Datei nicht vorhanden  |
| 4089  | Technical | Error    | Datei ist vom falschen Typ                                   |
| 4093  | Technical | Error    | Karte wird in einer anderen Kartensitzung exklusiv verwendet |
| 4094  | Technical | Error    | Timeout beim Kartenzugriff aufgetreten                       |

3219  
3220  
3221 [ $\leq$ ]

3222 4.1.5.4.13 TUC\_KON\_210 „SchreibeRecord“

3223 **TIP1-A\_4576 - TUC\_KON\_210 „SchreibeRecord“**

3224 Der Konnektor MUSS den technischen Use Case „SchreibeRecord“ gemäß TUC\_KON\_210  
3225 umsetzen.

3226

3227 **Tabelle 87: TAB\_KON\_224 – TUC\_KON\_210 „SchreibeRecord“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_210 „SchreibeRecord“  |
| Beschreibung   | Daten in lineare Datei schreiben  |
| Auslöser       | <ul style="list-style-type: none"> <li>Aufruf durch Fachmodul</li> </ul>  |
| Vorbedingungen | Die Karte MUSS den nötigen Sicherheitszustand für den Zugriff besitzen  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>cardSession</li> <li>fileIdentifier – <i>optional/verpflichtend, wenn kein sfid angegeben ist</i> (FID der zu bearbeitenden Datei)</li> <li>sfid – <i>optional/verpflichtend, wenn kein fileIdentifier angegeben ist</i> (Short File Identifier der zu bearbeitenden Datei)</li> <li>folder (Verzeichnis/Applikation auf der Karte, in dem sich die Datei befindet)</li> <li>recordNumber</li> <li>dataToBeWritten (Zu schreibende Daten)</li> </ul> |
| Komponenten    | Karte, Kartenterminal, Konnektor  |
| Ausgangsdaten  | keine   |

|                                |   |
|--------------------------------|---|
| Standardablauf                 | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>2. Prüfe Card.TYPE &lt;&gt; KVK</li> <li>3. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>4. Prüfe PinRef/KeyRef in CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden</li> <li>5. Selektiere Verzeichnis und ggf. Datei</li> <li>6. Schreibe Daten über Kartenkommando „UPDATE RECORD“ unter Berücksichtigung von recordNumber</li> </ol>                     |
| Varianten/<br>Alternativen     | keine   |
| Fehlerfälle                    | <p>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094<br/>         (→2) Schreibzugriff auf KVK nicht gestattet, Fehlercode 4085<br/>         (→3) Karte ist fremd reserviert, Fehlercode 4093<br/>         (→4) Nötige PIN ist nicht verifiziert, Fehlercode 4085<br/>         (→5) Verzeichnis, Datei existiert nicht, Fehlercode 4087<br/>         (→5-6) Verzeichnis deaktiviert, Fehlercode 4086<br/>         (→4-6) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;Kartenfehlercode gemäß [gemSpec_COS]&gt;</p> |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | keine   |

3228 **Tabelle 88: TAB\_KON\_540 Fehlercodes TUC\_KON\_210 „SchreibeRecord“**

| Fehlercode  | ErrorType | Severity | Fehlertext                        |
|---|-----------|----------|-----------------------------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |                                   |
| 4085  | Security  | Error    | Zugriffsbedingungen nicht erfüllt |
| 4086  | Technical | Error    | Verzeichnis deaktiviert           |
| 4087  | Technical | Error    | Datei nicht vorhanden             |
| 4088  | Technical | Error    | Datensatz zu groß                 |

|      |           |       |  |
|------|-----------|-------|--|
| 4093 | Technical | Error | Karte wird in einer anderen Kartensitzung exklusiv verwendet |
| 4094 | Technical | Error | Timeout beim Kartenzugriff aufgetreten                       |

3229

3230

3231 [ $\leq$ ]

3232 4.1.5.4.14 TUC\_KON\_211 „LöscheRecordInhalt“

3233 **TIP1-A\_5477 - TUC\_KON\_211 „LöscheRecordInhalt“**

3234 Der Konnektor MUSS den technischen Use Case „LöscheRecordInhalt“ gemäß  
3235 TUC\_KON\_211 umsetzen.

3236

3237 **Tabelle 89: TAB\_KON\_211 – TUC\_KON\_211 „LöscheRecordInhalt“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_211 „LöscheRecordInhalt“  |
| Beschreibung   | Inhalt eines Records einer strukturierten Datei löschen   |
| Auslöser       | <ul style="list-style-type: none"> <li>Aufruf des Use Case im Rahmen von technischen Use Cases der Basisdienste des Konnektors</li> <li>Aufruf des Use Cases durch ein Fachmodul im Konnektor</li> </ul>  |
| Vorbedingungen | Die Karte MUSS den nötigen Sicherheitszustand für den Zugriff besitzen  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>cardSession</li> <li>fileIdentifier – <i>optional/verpflichtend, wenn kein sfid angegeben ist</i> (FID der zu bearbeitenden Datei)</li> <li>sfid – <i>optional/verpflichtend, wenn kein fileIdentifier angegeben ist</i> (Short File Identifier der zu bearbeitenden Datei)</li> <li>folder (Verzeichnis/Applikation auf der Karte, in dem sich die Datei befindet)</li> <li>recordNumber</li> </ul> |
| Komponenten    | Karte, Kartenterminal, Konnektor  |
| Ausgangsdaten  | keine   |

|                                |  |
|--------------------------------|--|
| Standardablauf                 | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>2. Prüfe Card.TYPE &lt;&gt; KVK</li> <li>3. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer im Besitz des Karten-Locks ist.</li> <li>4. Prüfe PinRef/KeyRef in CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden</li> <li>5. Selektiere Verzeichnis und Datei</li> <li>6. Lösche Recordinhalt (identifiziert durch recordNumber) der selektierten Datei über Kartenkommando „ERASE RECORD“</li> </ol>  |
| Varianten/<br>Alternativen     | keine  |
| Fehlerfälle                    | <p>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094<br/>         (→2) Karte ist keine eGK der Generation 2 oder höher: Fehlercode 4209<br/>         (→3) Karte ist fremd reserviert, Fehlercode 4093<br/>         (→4) Nötige PIN ist nicht verifiziert, Fehlercode 4085<br/>         (→5) Verzeichnis, Datei oder Record existiert nicht, Fehlercode 4087<br/>         (→6) Verzeichnis deaktiviert, Fehlercode 4086<br/>         (→6) Record nicht vorhanden, Fehlercode 4091<br/>         (→5-6) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;Kartenfehlercode gemäß [gemSpec_COS]&gt;</p> |
| Nichtfunktionale Anforderungen | Keine  |
| Zugehörige Diagramme           | Keine  |

3238

**Tabelle 90: TAB\_KON\_786 Fehlercodes TUC\_KON\_211 „LöscheRecordInhalt“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4085  | Security  | Error    | Zugriffsbedingungen nicht erfüllt                            |
| 4086  | Technical | Error    | Verzeichnis deaktiviert                                      |
| 4087  | Technical | Error    | Datei nicht vorhanden  |
| 4091  | Technical | Error    | Record nicht vorhanden                                       |
| 4093  | Technical | Error    | Karte wird in einer anderen Kartensitzung exklusiv verwendet |
| 4094  | Technical | Error    | Timeout beim Kartenzugriff aufgetreten                       |

|      |           |       |  |
|------|-----------|-------|--|
| 4209 | Technical | Error | Kartentyp %CardType% wird durch diese Operation nicht unterstützt. |
|------|-----------|-------|--|

3239  
3240  
3241

[<=]

3242 4.1.5.4.15 TUC\_KON\_214 „FügeHinzuRecord“

3243 **TIP1-A\_4577 - TUC\_KON\_214 „FügeHinzuRecord“**

3244 Der Konnektor MUSS den technischen Use Case „FügeHinzuRecord“ gemäß  
3245 TUC\_KON\_214 umsetzen.

3246

3247 **Tabelle 91: TAB\_KON\_228 – TUC\_KON\_214 „FügeHinzuRecord“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_214 „FuegeHinzuRecord“  |
| Beschreibung   | Daten in lineare Datei anfügen  |
| Auslöser       | <ul style="list-style-type: none"> <li>• Aufruf durch Fachmodul</li> <li>• TUC_KON_006</li> </ul>   |
| Vorbedingungen | Die Karte MUSS den nötigen Sicherheitszustand für den Zugriff besitzen  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• cardSession</li> <li>• fileIdentifier – <i>optional/verpflichtend, wenn kein sfid angegeben ist</i><br/>(FID der zu bearbeitenden Datei)</li> <li>• sfid – <i>optional/verpflichtend, wenn kein fileIdentifier angegeben ist</i><br/>(Short File Identifier der zu bearbeitenden Datei)</li> <li>• folder<br/>(Verzeichnis/Applikation auf der Karte, in dem sich die Datei befindet)</li> <li>• dataToBeWritten<br/>(Zu schreibende Daten)</li> </ul> |
| Komponenten    | Karte, Kartenterminal, Konnektor  |
| Ausgangsdaten  | keine   |

|                                |   |
|--------------------------------|---|
| Standardablauf                 | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>2. Prüfe Card.TYPE &lt;&gt; KVK</li> <li>3. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>4. Prüfe PinRef/KeyRef in CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden</li> <li>5. Selektiere Verzeichnis und ggf. Datei</li> <li>6. Schreibe Daten über Kartenkommando „APPEND RECORD“</li> </ol>   |
| Varianten/<br>Alternativen     | keine   |
| Fehlerfälle                    | <p>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094<br/>         (→2) Schreibzugriff auf KVK nicht gestattet, Fehlercode 4085<br/>         (→3) Karte ist fremd reserviert, Fehlercode 4093<br/>         (→4) Nötige PIN ist nicht verifiziert, Fehlercode 4085<br/>         (→5-6) Verzeichnis, Datei existiert nicht, Fehlercode 4087<br/>         (→6) Verzeichnis deaktiviert, Fehlercode 4086<br/>         (→5-6) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;Kartenfehlercode gemäß [gemSpec_COS]&gt;</p> |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | keine   |

3248 **Tabelle 92: TAB\_KON\_541 Fehlercodes TUC\_KON\_214 „FügeHinzuRecord“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4085  | Security  | Error    | Zugriffsbedingungen nicht erfüllt                            |
| 4086  | Technical | Error    | Verzeichnis deaktiviert                                      |
| 4087  | Technical | Error    | Datei nicht vorhanden  |
| 4093  | Technical | Error    | Karte wird in einer anderen Kartensitzung exklusiv verwendet |
| 4094  | Technical | Error    | Timeout beim Kartenzugriff aufgetreten                       |

3249  
3250  
3251

[<=]

3252 4.1.5.4.16 TUC\_KON\_215 „SucheRecord“

3253 **TIP1-A\_4578 - TUC\_KON\_215 „SucheRecord“**

3254 Der Konnektor MUSS den technischen Use Case „SucheRecord“ gemäß TUC\_KON\_215  
3255 umsetzen.  
3256

3257 **Tabelle 93: TAB\_KON\_229 – TUC\_KON\_215 „SucheRecord“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_215 „SucheRecord“   |
| Beschreibung   | Daten in linearer Datei suchen  |
| Auslöser       | <ul style="list-style-type: none"> <li>Aufruf durch Fachmodul</li> </ul>  |
| Vorbedingungen | Die Karte MUSS den nötigen Sicherheitszustand für den Zugriff besitzen  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>cardSession</li> <li>fileIdentifier – <i>optional/verpflichtend</i>, wenn kein sfid angegeben ist (FID der zu bearbeitenden Datei)</li> <li>sfid – <i>optional/verpflichtend</i>, wenn kein fileIdentifier angegeben ist (Short File Identifier der zu bearbeitenden Datei)</li> <li>folder (Verzeichnis/Applikation auf der Karte, in dem sich die Datei befindet)</li> <li>pattern (SuchMuster)</li> <li>recordNumber – <i>optional; default = 1</i> (Recordnummer, bei der Suche beginnen soll) ()</li> </ul> |
| Komponenten    | Karte, Kartenterminal, Konnektor  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>numbersFound (Liste: Nummern der Records, die dem SuchMuster entsprechen)</li> </ul>   |



|                                |   |
|--------------------------------|---|
| Standardablauf                 | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>3. Prüfe PinRef/KeyRef in CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden</li> <li>4. Selektiere Verzeichnis und ggf. Datei</li> <li>5. Sende Kartenkommando „SEARCH RECORD“ mit SuchMuster <i>pattern</i> unter Berücksichtigung von recordNumber</li> <li>6. Liefere Antwort der Karte zurück</li> </ol> |
| Varianten/<br>Alternativen     | Keine   |
| Fehlerfälle                    | <p>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD-Sekunden, Fehlercode 4094<br/>         (→2) Karte ist fremd reserviert, Fehlercode 4093<br/>         (→3) Nötige PIN ist nicht verifiziert, Fehlercode 4085<br/>         (→4-5) Verzeichnis, Datei existiert nicht, Fehlercode 4087<br/>         (→5) Verzeichnis deaktiviert, Fehlercode 4086<br/>         (→4-5) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;Kartenfehlercode gemäß [gemSpec_COS]&gt;</p>   |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | keine   |

3258

**Tabelle 94: TAB\_KON\_542 Fehlercodes TUC\_KON\_215 „SucheRecord“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4085  | Security  | Error    | Zugriffsbedingungen nicht erfüllt                            |
| 4086  | Technical | Error    | Verzeichnis deaktiviert                                      |
| 4087  | Technical | Error    | Datei nicht vorhanden  |
| 4093  | Technical | Error    | Karte wird in einer anderen Kartensitzung exklusiv verwendet |
| 4094  | Technical | Error    | Timeout beim Kartenzugriff aufgetreten                       |

3259

3260

3261

[&lt;=]

3262 4.1.5.4.17 TUC\_KON\_018 „eGK-Sperrung prüfen“

3263 **TIP1-A\_4579 - TUC\_KON\_018 „eGK-Sperrung prüfen“**

3264 Der Konnektor MUSS den technischen Use Case „eGK-Sperrung prüfen“ gemäß  
3265 TUC\_KON\_018 umsetzen.

3266

3267 **Tabelle 95: TAB\_KON\_110 - TUC\_KON\_018 „eGK-Sperrung prüfen“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_018 „eGK-Sperrung prüfen“   |
| Beschreibung   | Es wird geprüft, dass DF.HCA (Health Care Application) der eGK nicht gesperrt ist und optional, dass das AUT-Zertifikat im DF.ESIGN gültig ist.<br>Für eine Karte ab der Generation G2.1 wird das AUT-Zertifikat (ECC) geprüft.<br>Für eine Karte der Generation G2.0 wird das AUT-Zertifikat (RSA) geprüft.  |
| Auslöser       | Aufruf durch Fachmodul im Konnektor   |
| Vorbedingungen | keine   |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>cardSession</li> <li>checkHcaOnly [Boolean] - <i>optional; default = false</i> (Prüfung auf die Frage beschränken, ob auf DF.HCA zugegriffen werden kann)</li> </ul>   |
| Komponenten    | Konnektor, Kartenterminal, eGK  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>Karte gesperrt: true   false</li> <li>Status – <i>optional/wenn checkHcaOnly = false</i> <ul style="list-style-type: none"> <li>DF.HCA gesperrt: true   false</li> <li>Ergebnis der Offline-Prüfung des C.CH.AUT-Zertifikats:<br/>gültig   ungültig</li> <li>Sperrstatus des C.CH.AUT-Zertifikats:<br/>gut   gesperrt   nicht ermittelbar</li> </ul> </li> </ul>   |
| Standardablauf | <ol style="list-style-type: none"> <li>Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>Selektiere DF.HCA : <ol style="list-style-type: none"> <li>Wenn die Karte '90 00' zurückmeldet, war das Selektieren möglich: DF.HCA gesperrt = false</li> <li>In allen anderen Fällen war das Selektieren nicht fehlerfrei möglich: DF.HCA gesperrt = true</li> </ol> </li> <li>Wenn checkHcaOnly = true<br/>Beende TUC, liefere Status.</li> <li>Ermittle Zertifikatsobjekt (fileIdentifier und folder) für C.AUT der Karte unter Berücksichtigung des kryptographischen Verfahrens crypt</li> </ol> |

|                                   |  |
|-----------------------------------|--|
|                                   | <p>gemäß TAB_KON_858.<br/>Für eine Karte ab der Generation G2.1 setze crypt=ECC.<br/>Für eine Karte der Generation G2.0 setze crypt=RSA.<br/>Rufe Cert = TUC_KON_216 „LeseZertifikat“<br/>{cardSession; fileIdentifier; folder}</p> <p>6. Bestimme per Aufruf von TUC_KON_037 „Zertifikat prüfen“</p> <p>a. das Ergebnis der Offline-Prüfung des C.CH.AUT-Zertifikats (gültig   ungültig) sowie</p> <p>b. den Sperrstatus des C.CH.AUT-Zertifikats (gut   gesperrt   nicht ermittelbar).</p> <p>7. Die Karte ist gesperrt = true, wenn</p> <p>a. DF.HCA gesperrt = true oder</p> <p>b. Ergebnis der Offline-Prüfung des C.CH.AUT-Zertifikats = ungültig oder</p> <p>c. Sperrstatus des C.CH.AUT-Zertifikats = gesperrt.</p> <p>In allen anderen Fällen ist die Karte gesperrt = false.</p> |
| Varianten/<br>Alternativen        | keine  |
| Fehlerfälle                       | (→2) Karte ist fremd reserviert, Fehlercode 4093   |
| Nichtfunktionale<br>Anforderungen | keine  |
| Zugehörige<br>Diagramme           | keine  |

3268 **Tabelle 96: TAB\_KON\_239 Fehlercodes TUC\_KON\_018 „eGK-Sperrung prüfen“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4093  | Technical | Error    | Karte wird in einer anderen Kartensitzung exklusiv verwendet |

3269

3270 [**<=**]

3271 **4.1.5.4.18 TUC\_KON\_006 „Datenzugriffsaudit eGK schreiben“**

3272 **TIP1-A\_4580 - TUC\_KON\_006 „Datenzugriffsaudit eGK schreiben“**

3273 Der Konnektor MUSS den technischen Use Case „Datenzugriffsaudit eGK schreiben“  
3274 gemäß TUC\_KON\_006 umsetzen.

3275

3276 **Tabelle 97: TAB\_KON\_108 - TUC\_KON\_006 „Datenzugriffsaudit eGK schreiben“**

| Element                    | Beschreibung   |
|----------------------------|--|
| Name                       | TUC_KON_006 „Datenzugriffsaudit eGK schreiben“   |
| Beschreibung               | Zugriff auf eGK in EF.Logging protokollieren.  |
| Auslöser                   | Aufruf durch ein Fachmodul   |
| Vorbedingungen             | Keine  |
| Eingangsdaten              | <ul style="list-style-type: none"> <li>• cardSession<br/>(CardSession einer eGK)</li> <li>• sourceCardSession<br/>(HBA/SMC-B, der/die für den eGK-Zugriff verwendet wird)</li> <li>• dataType<br/>(zugreifende Anwendung, siehe [gem_Spec_Karten_Fach_TIP#4.1 – Tabelle Tab_Karten_Fach_TIP_010 _StrukturEF.Logging – Struktur der Rekords der Datei EF.Logging])</li> <li>• accessType<br/>(Zugriffsart, siehe ebenda)</li> </ul>   |
| Komponenten                | eGK, HBA/SMC, Konnektor, Kartenterminal  |
| Ausgangsdaten              | Keine  |
| Standardablauf             | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>2. Prüfe Card.TYPE = EGK</li> <li>3. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>4. Wenn KeyRef in CARDESESSION.AUTHSTATE für DF.HCA.EF.LOGGING nicht mit passender Rolle vorhanden:<br/>Rufe TUC_CON_005 „Card-to-Card authentisieren“ {<br/>sourceCardSession;<br/>targetCardSession = cardSession;<br/>authMode = einseitig}</li> <li>5. Erzeuge Loggingdaten gemäß [gem_Spec_Karten_Fach_TIP#4.1 – Tabelle Tab_Karten_Fach_TIP_010 _StrukturEF.Logging – Struktur der Rekords der Datei EF.Logging]</li> <li>6. Rufe TUC_KON_214 „FügeHinzurecord“ {<br/>cardSession =\$cardSession;<br/>folder = MF;<br/>fileIdentifier = DF.HCA/EF.Logging;<br/>dataToBeWritten = Loggingdaten }</li> </ol> |
| Varianten/<br>Alternativen | Keine  |
| Fehlerfälle                | (→2) Protokoll nur für eGK gestattet, Fehlercode 4251<br>(→3) Karte ist fremd reserviert, Fehlercode 4093  |

|                                |       |
|--------------------------------|-------|
| Nichtfunktionale Anforderungen | keine |
| Zugehörige Diagramme           | keine |

3277 **Tabelle 98: TAB\_KON\_238 Fehlercodes TUC\_KON\_006 „Datenzugriffsaudit eGK**  
 3278 **schreiben“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4093  | Technical | Error    | Karte wird in einer anderen Kartensitzung exklusiv verwendet |
| 4251  | Technical | Error    | Protokoll nur für eGK gestattet                              |

3279  
 3280 [ $\leq$ ]

3281 4.1.5.4.19 TUC\_KON\_218 „Signiere“

3282 **TIP1-A\_4581 - TUC\_KON\_218 „Signiere“**

3283 Der Konnektor MUSS den technischen Use Case „Signiere“ gemäß TUC\_KON\_218  
 3284 umsetzen.

3285

3286 **Tabelle 99: TAB\_KON\_231 – TUC\_KON\_218 „Signiere“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_218 „Signiere“  |
| Beschreibung   | Dieser Use Case beschreibt das Anwenden eines privaten Schlüssels einer Karte zur Signatur oder Authentisierung.  |
| Auslöser       | <ul style="list-style-type: none"> <li>Aufruf einer der Operationen SignDocument des Signatordienstes oder ExternalAuthenticate des Authentifizierungsdienstes durch das Clientsystem.</li> <li>Aufruf durch Fachmodul</li> </ul> |
| Vorbedingungen | Zugriffsbedingung für referenzierten Schlüssel MUSS erfüllt sein  |

|                                |   |
|--------------------------------|---|
| Eingangsdaten                  | <ul style="list-style-type: none"> <li>• cardSession</li> <li>• pinRef<br/>(PIN-Referenz, gemäß der Objektsystem-Spezifikation des jeweiligen Kartentyps)</li> <li>• keyRef<br/>(Referenz auf den privaten Schlüssel, mit dem signiert werden soll, gemäß der Objektsystem-Spezifikation des jeweiligen Kartentyps)</li> <li>• algorithmusId<br/>(einer der laut Objektspezifikation für diesen Schlüssel zulässigen algorithmIdentifier)</li> <li>• dataToBeSigned<br/>(Zu signierende Daten, Hashwert)</li> </ul> |
| Komponenten                    | Karte, Kartenterminal, Konnektor  |
| Ausgangsdaten                  | <ul style="list-style-type: none"> <li>• chiffrat<br/>(Signatur)</li> </ul>   |
| Standardablauf                 | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>3. Prüfe pinRef in CARDESESSION.AUTHSTATE vorhanden:</li> <li>4. Setze keyRef und algorithmusId der Karte</li> <li>5. Sende „PSO: COMPUTE DS“ mit dataToBeSigned an Karte</li> <li>6. Gib chiffrat an den Aufrufer zurück</li> </ol>  |
| Varianten/<br>Alternativen     | Keine   |
| Fehlerfälle                    | <p>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094<br/>         (→2) Karte ist fremd reserviert, Fehlercode 4093<br/>         (→3) Nötige PIN ist nicht verifiziert, Fehlercode 4085<br/>         (→5) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;Kartenfehlercode gemäß [gemSpec_COS]&gt;</p>  |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

3287 **Tabelle 100: TAB\_KON\_543 Fehlercodes TUC\_KON\_218 „Signiere“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4085  | Security  | Error    | Zugriffsbedingungen nicht erfüllt                            |
| 4093  | Technical | Error    | Karte wird in einer anderen Kartensitzung exklusiv verwendet |
| 4094  | Technical | Error    | Timeout beim Kartenzugriff aufgetreten                       |

3288  
3289  
3290

[&lt;=]

3291 **4.1.5.4.20 TUC\_KON\_219 „Entschlüssele“**3292 **TIP1-A\_4582 - TUC\_KON\_219 „Entschlüssele“**

3293 Der Konnektor MUSS den technischen Use Case „Entschlüssele“ gemäß TUC\_KON\_219  
3294 umsetzen.

3295 **Tabelle 101: TAB\_KON\_232 – TUC\_KON\_219 „Entschlüssele“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_219 „Entschlüssele“   |
| Beschreibung   | Dieser Use Case beschreibt das Anwenden eines privaten Schlüssels einer Karte zur Entschlüsselung.  |
| Auslöser       | <ul style="list-style-type: none"> <li>Aufruf durch Fachmodul</li> </ul>  |
| Vorbedingungen | Zugriffsbedingung für referenzierten Schlüssel muss erfüllt sein  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>cardSession</li> <li>pinRef<br/>(Referenz auf die PIN, mit der der Entschlüsselungsschlüssel freigeschaltet werden kann, gemäß der Objektsystem-Spezifikation des jeweiligen Kartentyps)</li> <li>keyRef<br/>(Referenz auf den privaten Schlüssel, mit dem entschlüsselt werden soll, gemäß der Objektsystem-Spezifikation des jeweiligen Kartentyps.)</li> <li>algorithmusId<br/>(einer der für diesen Schlüssel zulässigen algorithmIdentifizier)</li> <li>encryptedData<br/>(Zu entschlüsselnde Daten, Chiffrat)</li> </ul> |
| Komponenten    | Karte(n), Kartenterminal, Konnektor   |

|                                   |   |
|-----------------------------------|---|
| Ausgangsdaten                     | <ul style="list-style-type: none"> <li>plainData<br/>(Entschlüsselte Daten)</li> </ul>  |
| Standardablauf                    | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>3. Prüfe pinRef in CARDESSION.AUTHSTATE vorhanden:</li> <li>4. Selektiere DF, in dem der private Schlüssel (keyRef) liegt, falls er noch nicht selektiert ist.</li> <li>5. Setze Schlüssel (keyRef) und algorithmusId.</li> <li>6. Sende encryptedData mittels Kommandos PSO: DECIPHER.</li> <li>7. gib plainData an den Aufrufer zurück</li> </ol> |
| Varianten/<br>Alternativen        | Keine   |
| Fehlerfälle                       | <p>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094<br/> (→2) Karte ist fremd reserviert, Fehlercode 4093<br/> (→3) Nötige PIN ist nicht verifiziert, Fehlercode 4085<br/> (→5) Schlüssel nicht vorhanden, Fehlercode 4079<br/> (→6) Fehler im Chifftrat: Fehlercode 4069<br/> (→4, 6) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;Kartenfehlercode gemäß [gemSpec_COS]&gt;</p>   |
| Varianten/<br>Alternativen        | Keine   |
| Nichtfunktionale<br>Anforderungen | Keine   |
| Zugehörige<br>Diagramme           | Keine   |

3296

**Tabelle 102: TAB\_KON\_210 Fehlercodes TUC\_KON\_219 „Entschlüssele“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4069  | Technical | Error    | korruptes Chifftrat bei asymmetrischer Entschlüsselung |
| 4079  | Technical | Error    | Schlüsseldaten fehlen                                  |
| 4085  | Security  | Error    | Zugriffsbedingungen nicht erfüllt                      |



|      |           |       |  |
|------|-----------|-------|--|
| 4093 | Technical | Error | Karte wird in einer anderen Kartensitzung exklusiv verwendet |
| 4094 | Technical | Error | Timeout beim Kartenzugriff aufgetreten                       |

3297  
3298  
3299

[<=]

3300 4.1.5.4.21 TUC\_KON\_200 „SendeAPDU“

3301 **TIP1-A\_4583 - TUC\_KON\_200 „SendeAPDU“**

3302 Der Konnektor MUSS den technischen Use Case „SendeAPDU“ gemäß TUC\_KON\_200  
3303 umsetzen.

3304

3305 **Tabelle 103: TAB\_KON\_215 TUC\_KON\_200 „SendeAPDU“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_200 „SendeAPDU“  |
| Beschreibung   | Dieser Use Case beschreibt das Senden einer APDU an eine Chipkarte bzw. an ein Kartenterminal und das Empfangen der Antwort.   |
| Auslöser       | <ul style="list-style-type: none"> <li>Aufruf durch Fachmodul</li> </ul>   |
| Vorbedingungen | Zugriffsbedingungen für das Kommando müssen in der Karte erfüllt sein und Karte muss für exklusiven Zugriff reserviert worden sein   |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>cardSession – <i>optional/verpflichtend</i>, wenn die APDU an die Karte gerichtet ist</li> <li>ctId – <i>optional/verpflichtend</i>, wenn die APDU an das Kartenterminal gerichtet ist<br/>(Kartenterminalidentifikator für Kommandos an das Kartenterminal)</li> <li>commandAPDU<br/>(versandfertige APDU (Bytefolge), in dem die Parameter {CLA, INS, P1,P2, Data (<i>optional</i>) Le(<i>optional</i>) } gesetzt sind.)</li> </ul> |
| Komponenten    | Karte(n), Kartenterminal, Konnektor  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>responseAPDU<br/>(Antwort der Chipkarte oder des Kartenterminals, Bytefolge)</li> </ul>   |
| Standardablauf | <p>A. cardSession ist gegeben</p> <ol style="list-style-type: none"> <li>Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>Prüfe, dass der Aufrufer für die zur cardSession gehörenden Karte ein Lock hat.</li> <li>commandAPDU wird über das Kartenterminal an die Zielkarte gesendet</li> </ol>   |

|                                |   |
|--------------------------------|---|
|                                | <p>4. die Antwort (responseAPDU) der Zielkarte wird an den Aufrufer zurückgegeben.</p> <p>B. ctId ist gegeben</p> <ol style="list-style-type: none"> <li>1. Sende commandAPDU an das Kartenterminal ctId</li> <li>2. gib die Antwort responseAPDU des Kartenterminals an den Aufrufer zurück</li> </ol>   |
| Varianten/Alternativen         | <ul style="list-style-type: none"> <li>• Soll Secure Messaging verwendet werden, MUSS vorher TUC_KON_023 „Karte reservieren“ aufgerufen werden</li> </ul>   |
| Fehlerfälle                    | <p>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094<br/>         (→2) Der Aufrufer ist nicht im Besitz des Karten-Locks, Fehlercode 4232<br/>         (→3) Kommunikationsfehler mit dem Kartenterminal: Fehlercode 4044.<br/>         (→3) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;Kartenfehlercode gemäß [gemSpec_COS]&gt;</p> |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | keine   |

**Tabelle 104: TAB\_KON\_216 Fehlercodes TUC\_KON\_200 „SendeAPDU“**

| Fehlercode  | ErrorType | Severity | Fehlertext                                 |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4044  | Technical | Error    | Fehler beim Zugriff auf das Kartenterminal |
| 4094  | Technical | Error    | Timeout beim Kartenzugriff aufgetreten     |
| 4232  | Technical | Error    | der Aufrufer besitzt nicht das Karten-Lock |

[<=]

#### 4.1.5.4.22 TUC\_KON\_024 „Karte zurücksetzen“

##### **TIP1-A\_4584 - TUC\_KON\_024 „Karte zurücksetzen“**

Der Konnektor MUSS den technischen Use Case „Karte zurücksetzen“ gemäß TUC\_KON\_024 umsetzen.

**Tabelle 105: TAB\_KON\_737 – TUC\_KON\_024 „Karte zurücksetzen“**

| Element      | Beschreibung   |
|--------------|--|
| Name         | TUC_KON_024 „Karte zurücksetzen“   |
| Beschreibung | Der technische Use Case setzt die gewählte Karte zurück (alle erreichten Sicherheitszustände werden auf der Karte und in der |

|                                |   |
|--------------------------------|---|
|                                | Verwaltung des Konnektors zurückgesetzt; auf der Karte wird MF selektiert). Ein eventuell laufendes C2C wird dabei abgebrochen.   |
| Auslöser                       | Fachmodul   |
| Vorbedingungen                 | keine   |
| Eingangsdaten                  | <ul style="list-style-type: none"> <li>ctId – <i>optional/verpflichtend, wenn keine cardSession angegeben ist</i> (Kartenterminalidentifikator)</li> <li>slotId – <i>optional/verpflichtend, wenn keine cardSession angegeben ist</i> (Nummer des Slots, in dem die Karte steckt)</li> <li>cardSession – <i>optional/verpflichtend, wenn ctId und slotId nicht angegeben sind</i> (Angabe der CardSession alternativ zur Angabe von ctId und slotId)</li> </ul>   |
| Komponenten                    | Karte, Kartenterminal, Konnektor  |
| Ausgangsdaten                  | Keine   |
| Standardablauf                 | <ol style="list-style-type: none"> <li>1. Wenn cardSession gegeben, dann ermittle ctId und slotId</li> <li>2. Der Konnektor prüft, dass entweder die Karte nicht reserviert ist oder der Aufrufer im Besitz des Karten-Locks ist.</li> <li>3. Brich eventuell parallel laufenden TUC_KON_005 ab</li> <li>4. Sende SICCT RESET ICC für slotId an das Kartenterminal CtID, um einen Warm Reset auszulösen</li> <li>5. Lösche alle Sicherheitszustände aus CARDSESSION.AUTHSTATE und den Inhalt von CARDSESSION.AUTHBY.</li> </ol> |
| Varianten/<br>Alternativen     | Keine   |
| Fehlerfälle                    | <p>* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094<br/>         (→2) Der Aufrufer ist nicht im Besitz des Karten-Locks, Fehlercode 4232<br/>         (→4) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode &lt;Kartenfehlercode gemäß [gemSpec_COS]&gt;</p>   |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

3316

**Tabelle 106: TAB\_KON\_544 Fehlercodes TUC\_KON\_024 „Karte zurücksetzen“**

| Fehlercode  | ErrorType | Severity | Fehlertext |
|---|-----------|----------|------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |            |

|      |           |       |   |
|------|-----------|-------|---|
| 4094 | Technical | Error | Timeout beim Kartenzugriff aufgetreten            |
| 4232 | Technical | Error | der Aufrufer ist nicht im Besitz des Karten-Locks |

3317

3318 [ $\leq$ ]

3319 4.1.5.4.23 TUC\_KON\_216 „LeseZertifikat“

3320 **TIP1-A\_4585 - TUC\_KON\_216 „LeseZertifikat“**3321 Der Konnektor MUSS den technischen Use Case „LeseZertifikat“ gemäß TUC\_KON\_216  
3322 umsetzen.  
33233324 **Tabelle 107: TAB\_KON\_230 – TUC\_KON\_216 „LeseZertifikat“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_216 „LeseZertifikat“   |
| Beschreibung   | Dieser Use Case beschreibt das Lesen eines Zertifikates von einer Karte  |
| Auslöser       | <ul style="list-style-type: none"> <li>• Aufruf der Operation ReadCardCertificate des Zertifikatsdienstes durch das Clientsystem.</li> <li>• Aufruf durch Fachmodul</li> <li>• Aufruf im Rahmen von technischen Use Cases der Basisdienste des Konnektors</li> </ul>   |
| Vorbedingungen | Keine  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• cardSession</li> <li>• fileIdentifier – <i>optional/verpflichtend, wenn kein sfid angegeben ist</i><br/>(FID der zu lesenden Datei)</li> <li>• sfid – <i>optional/verpflichtend, wenn kein fileIdentifier angegeben ist</i><br/>(Short File Identifier der zu lesenden Datei)</li> <li>• folder<br/>(Verzeichnis/Applikation auf der Karte, in dem sich das Zertifikat befindet)</li> </ul> |
| Komponenten    | Karte, Kartenterminal, Konnektor   |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>• certificate<br/>(gelesenes Zertifikat)</li> </ul>   |

|                                   |   |
|-----------------------------------|---|
| Standardablauf                    | <ol style="list-style-type: none"> <li>1. Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt.</li> <li>3. Prüfe CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden</li> <li>4. Rufe TUC_KON_202 „LeseDatei“ {<br/>cardSession;<br/>fileIdentifizier;<br/>folder }<br/>oder TUC_KON_202 „LeseDatei“ {<br/>cardSession;<br/>sfid;<br/>folder }</li> <li>5. Das Zertifikat wird an den Aufrufer zurückgegeben.</li> </ol> |
| Varianten/<br>Alternativen        | Keine   |
| Fehlerfälle                       | <p>(→2) Karte ist fremd reserviert, Fehlercode 4093</p> <p>(-&gt;4) Es wurde versucht, ein Zertifikat von der Karte zu lesen, welches auf der Karte nicht vorhanden ist (Fehlercode 4256). Hierbei kann es sich um ein fehlendes Zertifikatsobjekt (z.B. adressiertes ECC-Zertifikat auf HBA G2.0) oder ein leeres Zertifikatsobjekt (z.B. adressiertes ECC-Zertifikat auf gSMC-K G2.0, welches aber nicht personalisiert wurde) handeln.</p>   |
| Nichtfunktionale<br>Anforderungen | Keine   |
| Zugehörige<br>Diagramme           | Keine   |

3325 **Tabelle 108: TAB\_KON\_209 Fehlercodes TUC\_KON\_216 „LeseZertifikat“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4093  | Technical | Error    | Karte wird in einer anderen Kartensitzung exklusiv verwendet |
| 4256  | Technical | Warning  | Zertifikat auf Karte nicht vorhanden                         |

3326  
3327 **[<=]**

3328 4.1.5.4.24 TUC\_KON\_036 „LiefereFachlicheRolle“

3329 **TIP1-A\_5478 - TUC\_KON\_036 „LiefereFachlicheRolle“**

3330 Der Konnektor MUSS den technischen Use Case TUC\_KON\_036 „LiefereFachlicheRolle“  
3331 umsetzen.  
3332

3333 **Tabelle 109: TAB\_KON\_827 TUC\_KON\_036 „LiefereFachlicheRolle“**

| Element         | Beschreibung   |
|-----------------|--|
| Name            | TUC_KON_036 „LiefereFachlicheRolle“  |
| Beschreibung    | Dieser TUC liefert die fachliche Rolle, die der OID aus dem X.509Zertifikat der gesteckten Karte zugeordnet ist.<br>Es werden nur folgende Karten unterstützt:<br>HBAX, SM-B, EGK, KVK<br>Es werden nur die AUT-Zertifikate ausgelesen.<br>Für eine Karte ab der Generation G2.1 wird das AUT-Zertifikat (ECC) geprüft.<br>Für eine Karte der Generation G2.0 wird das AUT-Zertifikat (RSA) geprüft.   |
| Auslöser        | <ul style="list-style-type: none"> <li>Aufruf durch ein Fachmodul oder eine Basisanwendung des Konnektors</li> </ul>   |
| Vorbedingungen  | Keine  |
| Eingangsdaten   | <ul style="list-style-type: none"> <li>cardSession</li> </ul>  |
| Komponenten     | Konnektor, Karte   |
| Ausgangsdaten   | <ul style="list-style-type: none"> <li>role<br/>(fachliche Rolle gemäß [gemSpec_PKI#Tab_PKI_254 Zugriffsprofile für eine Rollenauthentisierung])</li> </ul>  |
| Nachbedingungen | Keine  |
| Standardablauf  | <ol style="list-style-type: none"> <li>Ermittle Card = CM_CARD_LIST(cardSession)</li> <li>Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer im Besitz des Karten-Locks ist.</li> <li>Wenn CARD.TYPE = KVK, dann setze fachliche Rolle = „Versicherter“ und springe zu Schritt 8.</li> <li>Ermittle <i>fileIdentifier</i> und folder des C.AUT-Zertifikates unter Berücksichtigung des kryptographischen Algorithmus crypt für die Karte, die durch die cardSession referenziert wird.<br/>Für eine Karte ab der Generation G2.1 setze crypt=ECC.<br/>Für eine Karte der Generation G2.0 setze crypt=RSA.<br/>Welches Zertifikat gelesen wird, ist in TAB_KON_858 beschrieben.</li> <li>Lies Zertifikat:<br/>Rufe TUC_KON_216 "LeseZertifikat" {<br/>    cardSession;<br/>    <i>fileIdentifier</i> = <i>fileIdentifier</i> (AUT-Zertifikat);<br/>    folder = folder(AUT-Zertifikat)}</li> </ol> |

|                                |  |
|--------------------------------|--|
|                                | 6. Ermittle ProfessionOIDs aus Extension Admission des Zertifikates: Rufe TUC_PKI_009 „Rollenermittlung“ {certificate}<br>7. Ermittle die fachliche Rolle, die den ProfessionOIDs entspricht, gemäß [gemSpec_PKI# Tab_PKI_254 Zugriffsprofile für eine Rollenauthentisierung].<br>8. Rückgabe \$role (fachliche Rolle) an den Aufrufer |
| Varianten/<br>Alternativen     | Keine  |
| Fehlerfälle                    | * Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094<br>(→2) Karte ist fremd reserviert, Fehlercode 4093<br>(→7) ProfessionOIDs mappen nicht auf die gleiche Rolle, Fehlercode 4210   |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | keine  |

**Tabelle 110: TAB\_KON\_829 Fehlercodes TUC\_KON\_036 „LiefereFachlicheRolle“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4093  | Technical | Error    | Karte wird in einer anderen Kartensitzung exklusiv verwendet |
| 4094  | Technical | Error    | Timeout beim Kartenzugriff aufgetreten                       |
| 4210  | Technical | Error    | ProfessionOIDs nicht eindeutig auf eine Rolle abbildbar      |

[&lt;=]

#### 4.1.5.5 Operationen an der Außenschnittstelle

##### **TIP1-A\_4586-03** **TIP1-A\_4586-02 - Basisanwendung Kartendienst**

Der Konnektor MUSS für Clients eine Basisanwendung Kartendienst mit den Operationen VerifyPin, ChangePin, UnblockPin, GetPinStatus an der Außenschnittstelle anbieten.

**Tabelle 111: TAB\_KON\_038 Basisanwendung Karten- und Kartenterminaldienst**

|             |             |
|-------------|-------------|
| <b>Name</b> | CardService |
|-------------|-------------|

|                          |   |  |
|--------------------------|---|--|
| <b>Version (KDV)</b>     | 8.1.0 (WSDL- und XSD-Version)<br>8.1.1 (WSDL- und XSD-Version)<br>8.1.2 (WSDL-Version) 8.1.3 (XSD-Version)<br><a href="#">Siehe Anhang D (WSDL-Version)</a> |  |
| <b>Namensraum</b>        | Siehe <a href="#">Anhang D GitHub</a>   |  |
| <b>Namensraum-Kürzel</b> | CARD für Schema und CARDW für WSDL  |  |
| <b>Operationen</b>       | <b>Name</b>   | <b>Kurzbeschreibung</b>                      |
|                          | VerifyPin   | PIN prüfen                                   |
|                          | ChangePin   | PIN ändern                                   |
|                          | UnblockPin  | PIN entsperren                               |
|                          | GetPinStatus  | PIN-Status ermitteln                         |
|                          | EnablePin   | Erfordernis der PIN-Verifikation einschalten |
|                          | DisablePin  | Erfordernis der PIN-Verifikation abschalten  |
| <b>WSDL</b>              | CardService.wsdl (WSDL-Version 8.1.0)<br>CardService_v8_1_1.wsdl<br>CardService_v8_1_2.wsdl   |  |
| <b>Schema</b>            | CardService.xsd (XSD-Version 8.1.0)<br>CardService_v8_1_1.xsd<br>CardService_v8_1_3.xsd   |  |

3345

3346 [**<=**]3347 4.1.5.5.1 *VerifyPin*3348 **TIP1-A\_4587 - Operation VerifyPin**

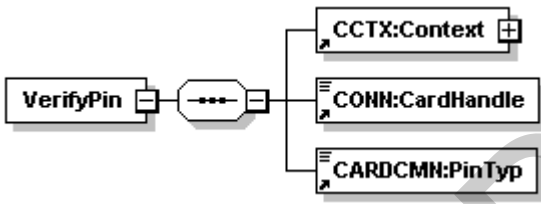
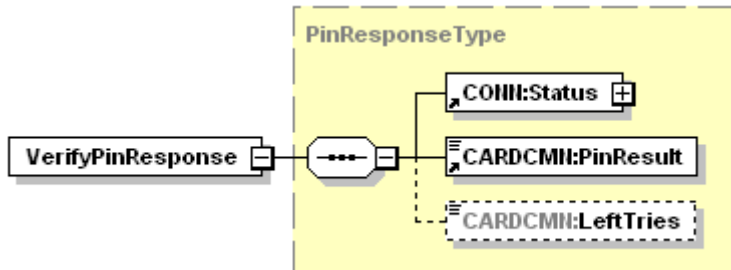
3349 Der Konnektor MUSS an der Außenschnittstelle eine Operation VerifyPin, wie in Tabelle  
 3350 TAB\_KON\_047 Operation VerifyPin beschrieben, anbieten.

3351

3352 **Tabelle 112: TAB\_KON\_047 Operation VerifyPin**

|                     |   |
|---------------------|---|
| <b>Name</b>         | <b>VerifyPin</b>  |
| <b>Beschreibung</b> | <p>Stößt die sichere Eingabe einer PIN am PIN-Pad des Kartenterminals für eine Karte an.</p> <p>Das Ergebnis der PIN-Prüfung gibt Auskunft darüber, ob die PIN richtig oder falsch war oder aufgrund zu vieler Fehlversuche blockiert ist.</p> <p>Eine Karte kann potentiell mehrere PINs haben. Insbesondere für die qualifizierte elektronische Signatur existiert eine separate PIN. Diese PIN darf nur über das PIN-Pad eingegeben werden.</p> <p>Die PIN-Verifikation und die damit verbundene Änderung des Sicherheitsstatus der Karte erfolgt nur für die durch den Aufrufkontext adressierte Kartensitzung. Falls die Karte in einem zentralen Kartenterminal steckt, auf das der Arbeitsplatz Zugriff hat, erfolgt eine Remote-PIN-Eingabe. Das Kartenterminal für</p> |



|                        |   |   |
|------------------------|---|---|
|                        | <p>die PIN-Eingabe ergibt sich dabei aus der im Aufrufkontext angegebenen Mandanten-ID und Arbeitsplatz-ID</p> <p>Diese Operation entspricht dem Aufruf von TUC_KON_012 „PIN verifizieren“. Dort sind auch die Display Messages definiert, die bei PIN-Eingabe am Kartenterminal anzuzeigen sind (TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal). Die beim Aufruf von TUC_KON_012 anzugebende PIN-Art lautet „mandatorisch“. Die PIN-Verifikation wird also unabhängig vom erreichten Sicherheitsstatus in der Karte immer durchgeführt.</p> |   |
| <b>Aufrufparameter</b> |   |   |
|                        | Name  | Beschreibung  |
|                        | Context   | MandantId, CsId, WorkplaceId verpflichtend; UserId verpflichtend für HBAX   |
|                        | CardHandle  | Adressiert die Karte, für die die PIN verifiziert werden soll.<br>Die Operation DARF die PIN-Verifikation mit der eGK NICHT unterstützen. Unterstützt werden die Kartentypen HBAX und SM-B. Wird die Operation mit einem nicht unterstützten Kartentypen aufgerufen, so MUSS der Konnektor die Bearbeitung mit dem Fehler 4209 abbrechen. |
|                        | PinTyp  | Gibt an, welche PIN der Karte verifiziert werden soll.<br>Erlaubte Belegung von PinTyp in Abhängigkeit der durch Cardhandle referenzierten Karte: <ul style="list-style-type: none"> <li>• HBAX: PIN.CH</li> <li>• SM-B: PIN.SMC</li> </ul>   |
| <b>Rückgabe</b>        |   |   |
|                        | Name  | Beschreibung  |

|               |           |  |   |
|---------------|-----------|--|---|
|               | Status    | Enthält den Ausführungsstatus der Operation (siehe 3.5.2)  |   |
|               | PinResult | Wert   | Bedeutung   |
|               |           | OK   | Prüfung war erfolgreich   |
|               |           | REJECTED   | PIN war falsch. Die Anzahl der verbleibenden Versuche ist im Element <code>LeftTries</code> |
|               |           | ERROR  | Ein Bearbeitungsfehler ist aufgetreten. Fehlerursache im Feld <code>Error</code>            |
|               |           | WASBLOCKED   | PIN war zum Aufrufzeitpunkt bereits gesperrt  |
|               |           | NOWBLOCKED   | PIN ist durch aktuellen Fehlversuch gesperrt  |
|               |           | TRANSPORT_PIN  | PIN ist mit Transportschutz versehen  |
|               | LeftTries | Im Falle von <code>Result=REJECTED</code> wird hier die Anzahl der verbleibenden möglichen Versuche zurückgegeben. |   |
| Vorbedingung  | Keine     |  |   |
| Nachbedingung | keine     |  |   |

3353 Der Ablauf der Operation VerifyPin ist in Tabelle TAB\_KON\_738 Ablauf VerifyPin  
 3354 beschrieben.  
 3355

3356 **Tabelle 113: TAB\_KON\_738 Ablauf VerifyPin**

| Nr. | Aufruf Technischer Use Case oder Interne Operation | Beschreibung   |
|-----|--|--|
| 1.  | checkArguments                                     | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.   |
| 2.  | TUC_KON_000 „Prüfe Zugriffs-berechtigung“          | Prüfung der Zugriffsberechtigung durch den Aufruf TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientSystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>userId = \$context.userId;<br>cardHandle }<br>Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab. |

|    |                                   |  |
|----|-----------------------------------|--|
| 3. | TUC_KON_026 „Liefere CardSession“ | Ermittle cardSession über TUC_KON_026 {<br>mandatId = \$context.mandantId;<br>clientsystemId = \$context.clientsystemId;<br>userId = \$context.userId;<br>cardHandle }   |
| 4. | TUC_KON_012 „PIN verifizieren“    | Verifiziere PIN über TUC_KON_012 {<br>cardSession;<br>workplaceId = \$context.workplaceId;<br>pinRef = PinRef(PinTyp);<br>appName = „“ (Leerstring);<br>verificationType = Mandatorisch }  |
| 5. | Verifikationsergebnis auswerten   | Wenn TUC_KON_012 mit Fehler 4065 abbricht, wird dies als erfolgreicher Operationsdurchlauf mit PinResult= TRANSPORT_PIN abgefangen.<br>Wenn TUC_KON_012 den Returncode BLOCKED liefert, wird dies als erfolgreicher Operationsdurchlauf mit PinResult= NOWBLOCKED zurückgegeben.<br>Wenn TUC_KON_012 mit Fehler 4063 abbricht, wird dies als erfolgreicher Operationsdurchlauf mit PinResult= WASBLOCKED zurückgegeben |

3357 **Tabelle 114: TAB\_KON\_545 Fehlercodes „VerifyPin“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4000  | Technical | Error    | Syntaxfehler   |
| 4078  | Security  | Error    | PIN-Eingabe über das Clientsystem ist nicht zugelassen             |
| 4209  | Technical | Error    | Kartentyp %CardType% wird durch diese Operation nicht unterstützt. |

3358  
3359  
3360 [**<=**]

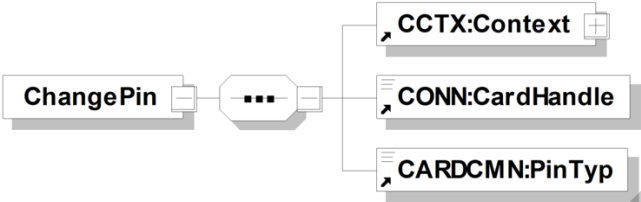
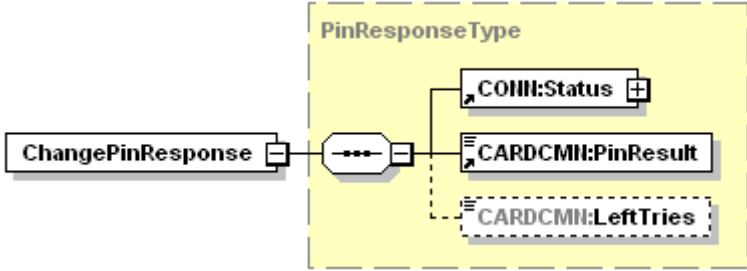
#### 3361 4.1.5.5.2 ChangePin

#### 3362 **TIP1-A\_4588 - Operation ChangePin**

3363 Der Konnektor MUSS an der Außenschnittstelle eine Operation ChangePin, wie in Tabelle  
3364 TAB\_KON\_049 Operation ChangePin beschrieben, anbieten.

3365 **Tabelle 115: TAB\_KON\_049 Operation ChangePin**

| Name                | ChangePin   |
|---------------------|---|
| <b>Beschreibung</b> | Ändert eine PIN einer Karte. Alte und neue PIN werden dabei am PIN-Pad des Kartenterminals eingegeben.<br>Falls die Karte in einem zentralen Kartenterminal steckt, auf das |

|                        |  |  |
|------------------------|--|--|
|                        | <p>der im Aufrufkontext angegebene Arbeitsplatz Zugriff hat, erfolgt eine Remote-PIN-Eingabe.<br/>Diese Operation entspricht dem Aufruf TUC_KON_019 „PIN ändern“ .</p> |  |
| <b>Aufrufparameter</b> |    |  |
|                        | Name   | Beschreibung   |
|                        | Context  | MandantId, CsId, WorkplaceId verpflichtend; UserId optional (verpflichtend beim HBA)   |
|                        | CardHandle   | Adressiert die Karte, für die die PIN geändert werden soll. Unterstützt werden die Kartentypen EGK, HBAX und SM-B. Wird die Operation mit einem nicht unterstützten Kartentypen aufgerufen, so MUSS der Konnektor die Bearbeitung mit dem Fehler 4209 abbrechen.   |
|                        | PinTyp   | <p>Gibt an, welche PIN der Karte geändert werden soll.<br/>Erlaubte Belegung von PinTyp in Abhängigkeit der durch CardHandle referenzierten Karte:</p> <ul style="list-style-type: none"> <li>eGK G1+: PIN.CH,</li> <li>eGK G2: PIN.CH, MRPIN.NFD, MRPIN.NFD_READ, MRPIN.DPE, MRPIN.GDD, MRPIN.OSE, MRPIN.AMTS, PIN.AMTS_REP</li> <li>zusätzlich eGK G2.0: MRPIN.DPE_READ</li> <li>HBAX: PIN.CH, PIN.QES</li> <li>SM-B: PIN.SMC</li> </ul> |
| <b>Rückgabe</b>        |    |  |
|                        | Name   | Beschreibung   |

|                      |           |  |  |
|----------------------|-----------|--|--|
|                      | LeftTries | Im Falle von REJECTED wird hier die Anzahl der verbleibenden möglichen Versuche zurückgegeben. |  |
|                      | Status    | Enthält den Ausführungsstatus der Operation, siehe 3.5.2                                       |  |
|                      |           |  |  |
|                      | PinResult | Wert   | Bedeutung  |
|                      |           | OK   | PIN-Änderung war erfolgreich   |
|                      |           | ERROR  | Ein Bearbeitungsfehler ist aufgetreten. Fehlerursache im Feld Error              |
|                      |           | REJECTED   | OldPIN war falsch Die Anzahl der verbleibenden Versuche ist im Element LeftTries |
|                      |           | WASBLOCKED   | PIN war zum Aufrufzeitpunkt bereits gesperrt                                     |
| NOWBLOCKED           |           | PIN ist durch aktuellen Fehlversuch gesperrt   |  |
| <b>Vorbedingung</b>  | Keine     |  |  |
| <b>Nachbedingung</b> | keine     |  |  |

3366 **Tabelle 116: TAB\_KON\_546 Ablauf ChangePin**

| Nr. | Aufruf<br>Technischer Use<br>Case oder Interne<br>Operation | Beschreibung  |
|-----|---|---|
| 1.  | checkArguments  | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.  |
| 2.  | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“            | Prüfung der Zugriffsberechtigung durch den Aufruf<br>TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientSystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>userId = \$context.userId;<br>cardHandle }<br>Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab. |

|    |   |  |
|----|---|--|
| 3. | TUC_KON_026<br>„Liefere<br>CardSession“ | Ermittle cardSession über TUC_KON_026 {<br>mandantId = \$context.mandantId;<br>clientsystemId = \$context.clientsystemId;<br>cardHandle;<br>userId = \$context.userId }  |
| 4. | TUC_KON_019 „PIN<br>ändern“             | Ändere PIN über TUC_KON_019 {<br>cardSession;<br>workplaceId = \$context.workplaceId;<br>pinRef = PinRef(PinTyp) }   |
| 5. | Verifikationsergebnis<br>auswerten      | Wenn TUC_KON_019 den Returncode BLOCKED liefert,<br>wird dies als erfolgreicher Operationsdurchlauf mit<br>PinResult= NOWBLOCKED zurückgegeben.<br>Wenn TUC_KON_019 mit Fehler 4063 abbricht, wird dies<br>als erfolgreicher Operationsdurchlauf mit PinResult=<br>WASBLOCKED zurückgegeben. |

3367 **Tabelle 117: TAB\_KON\_547 Fehlercodes „ChangePin“**

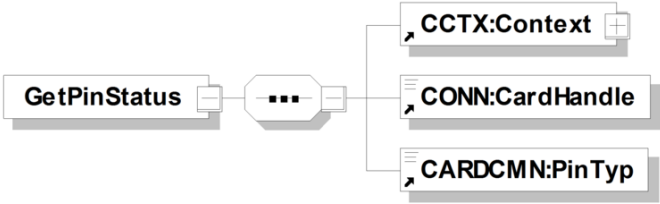
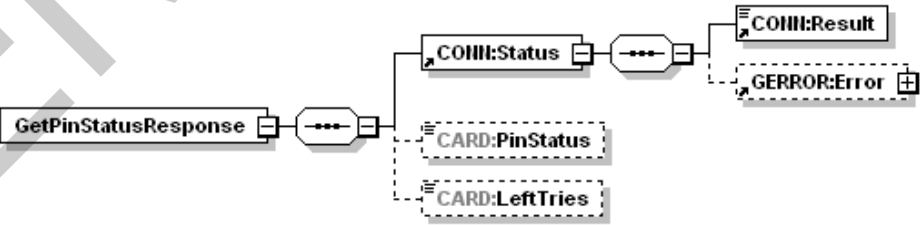
| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4000  | Technical | Error    | Syntaxfehler   |
| 4072  | Technical | Error    | Ungültige PIN-Referenz <code>PinRef</code>                         |
| 4209  | Technical | Error    | Kartentyp %CardType% wird durch diese Operation nicht unterstützt. |

3368  
3369  
3370 **[<=]**3371 **4.1.5.5.3 GetPinStatus**3372 **TIP1-A\_4589 - Operation GetPinStatus**

3373 Der Konnektor MUSS an der Außenschnittstelle eine Operation GetPinStatus, wie in  
3374 Tabelle TAB\_KON\_051 Operation GetPinStatus beschrieben, anbieten.

3376 **Tabelle 118: TAB\_KON\_051 Operation GetPinStatus**

| Name         | GetPinStatus   |
|--------------|--|
| Beschreibung | Diese Operation gibt Auskunft über den PIN-Zustand einer Karte.<br>Für transportgeschützte PIN gibt die Operation die Art des Transportschutzes an.<br>Für Echt-PINs kann mit dieser Operation die Anzahl der noch verbleibenden Versuche für PIN-Verifikationen ermittelt werden oder ob die PIN bereits verifiziert wurde. |

|                      |   |   |
|----------------------|---|---|
| Aufruf-<br>parameter |  <pre> sequenceDiagram     participant Caller     participant GetPinStatus     participant Context as CCTX:Context     participant CardHandle as CONN:CardHandle     participant PinType as CARDCMN:PinTyp     Caller-&gt;&gt;GetPinStatus     GetPinStatus-&gt;&gt;Context     GetPinStatus-&gt;&gt;CardHandle     GetPinStatus-&gt;&gt;PinType           </pre>   |   |
|                      | Name  | Beschreibung  |
|                      | Context   | MandantId, CsId, WorkplaceId; UserId  |
|                      | CardHandle  | Adressiert die Karte, für die der PIN-Status ermittelt werden soll. Unterstützt werden die Kartentypen EGK, HBax und SM-B. Eine KVK ist nicht zulässig. Wird die Operation mit einem nicht unterstützten Kartentypen aufgerufen, so MUSS der Konnektor die Bearbeitung mit dem Fehler 4209 abbrechen.   |
|                      | PinTyp  | Gibt an, für welche PIN der Karte der PIN-Status ermittelt werden soll.<br>Erlaubte Belegung von PinTyp in Abhängigkeit der durch Cardhandle referenzierten Karte: <ul style="list-style-type: none"> <li>eGK G1+: PIN.CH</li> <li>eGK G2: PIN.CH, MRPIN.NFD, MRPIN.NFD_READ, MRPIN.DPE, MRPIN.GDD, MRPIN.OSE, MRPIN.AMTS, PIN.AMTS_REP</li> <li>zusätzlich eGK G2.0: MRPIN.DPE_READ</li> <li>HBax: PIN.CH, PIN.QES</li> <li>SM-B: PIN.SMC</li> </ul> |
| Rückgabe             |  <pre> sequenceDiagram     participant Caller     participant GetPinStatusResponse     participant Status as CONN:Status     participant Result as CONN:Result     participant Error as GERROR&gt;Error     participant PinStatus as CARD:PinStatus     participant LeftTries as CARD:LeftTries     Caller-&gt;&gt;GetPinStatusResponse     GetPinStatusResponse-&gt;&gt;Status     GetPinStatusResponse-&gt;&gt;Result     GetPinStatusResponse-&gt;&gt;Error     GetPinStatusResponse-&gt;&gt;PinStatus     GetPinStatusResponse-&gt;&gt;LeftTries           </pre> |   |
|                      | Name  | Beschreibung  |
|                      | Status  | Enthält den Ausführungsstatus der Operation siehe 3.5.2   |
|                      | PinStatus   | Status der PIN. Die folgenden Werte sind verpflichtend:   |
|                      | Wert  | Bedeutung   |

|               |           |   |  |
|---------------|-----------|---|--|
|               |           | VERIFIED  | Bereits verifiziert (in CARDSESSION.AUTHSTATE vorhanden)   |
|               |           | TRANSPORT_PIN   | Transport-PIN  |
|               |           | EMPTY_PIN   | Leer-PIN   |
|               |           | BLOCKED   | gesperrt   |
|               |           | VERIFIABLE  | Echt-PIN, noch nicht verifiziert                           |
|               |           | DISABLED  | PIN-Schutz ausgeschaltet (Verifikation nicht erforderlich) |
|               | LeftTries | Bei einer Echt-PIN wird hier bei PinStatus = VERIFIABLE die Anzahl der verbleibenden möglichen Versuche für die Verifikation der PIN zurückgegeben, bei einer gesperrten PIN 0. |  |
| Vorbedingung  | keine     |   |  |
| Nachbedingung | keine     |   |  |

3377 **Tabelle 119: TAB\_KON\_548 Ablauf GetPinStatus**

| Nr . | Aufruf<br>Technischer<br>Use Case oder<br>Interne<br>Operation | Beschreibung   |
|------|--|--|
| 1.   | checkArguments   | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.   |
| 2.   | TUC_KON_000<br>„Prüfe Zugriffsberechtigung“                    | Prüfung der Zugriffsberechtigung durch den Aufruf TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientSystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>userId = \$context.userId; // falls angegeben<br>cardHandle }<br>Tritt bei der Prüfung ein Fehler auf, so bricht die Operation mit dem Fehlercode aus TUC_KON_000 ab. |
| 3.   | TUC_KON_026<br>„Liefere CardSession“                           | Ermittle cardSession über TUC_KON_026 {<br>mandantId = \$context.mandantId;<br>clientSystemId = \$context.clientsystemId;<br>userId = \$context.userId; // falls angegeben ;<br>cardHandle }   |
| 4.   | TUC_KON_022<br>„Liefere PIN-“                                  | Ermittle PinStatus über TUC_KON_022 {<br>cardSession;  |



|  |         |                           |
|--|---------|---------------------------|
|  | Status" | pinRef = PinRef(PinTyp) } |
|--|---------|---------------------------|

3378 **Tabelle 120: TAB\_KON\_549 Fehlercodes „GetPinStatus“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4000  | Technical | Error    | Syntaxfehler   |
| 4001  | Technical | Error    | interner Fehler  |
| 4072  | Technical | Error    | ungültige PIN-Referenz <i>PinRef</i>                               |
| 4209  | Technical | Error    | Kartentyp %CardType% wird durch diese Operation nicht unterstützt. |

3379

3380

3381 [**<=**]

3382 *4.1.5.5.4 UnblockPin*

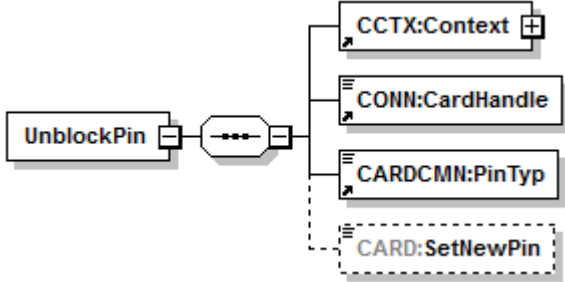
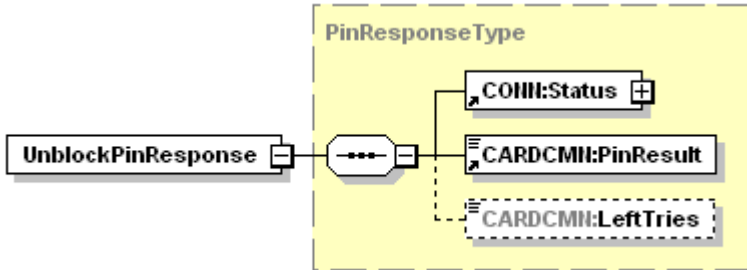
3383 **TIP1-A\_4590 - Operation UnblockPin**

3384 Der Konnektor MUSS an der Außenschnittstelle eine Operation UnblockPin, wie in Tabelle  
3385 TAB\_KON\_053 Operation UnblockPin beschrieben, anbieten.

3386

3387 **Tabelle 121: TAB\_KON\_053 Operation UnblockPin**

| Name         | UnblockPin  |
|--------------|---|
| Beschreibung | <p>Mit diesem Kommando kann eine blockierte PIN wieder freigeschaltet werden. Dabei wird der Wiederholungszähler für diese PIN in der Karte auf seinen Anfangswert zurückgesetzt und es KANN eine neue PIN gesetzt werden. Um diese Aktion durchführen zu können, muss eine PUK (auch als Resetting Code bezeichnet) präsentiert werden.</p> <p>PIN und PUK werden am PIN-Pad des Kartenterminals eingegeben. Falls die Karte in einem zentralen Kartenterminal steckt, auf das der im Aufrufkontext angegebene Arbeitsplatz Zugriff hat, erfolgt eine Remote-PIN-Eingabe. Das Kartenterminal für die PIN-Eingabe ergibt sich dabei aus der im Aufrufkontext angegebenen Mandanten-ID und Arbeitsplatz-ID.</p> <p>Diese Operation entspricht dem Aufruf von TUC_KON_021 „PIN entsperren“.</p> |

|                              |  |   |
|------------------------------|--|---|
| <b>Aufruf-<br/>parameter</b> |    |   |
|                              | Name   | Beschreibung  |
|                              | Context  | MandantId, CsId, WorkplaceId verpflichtend; UserId (optional, für HBA verpflichtend)  |
|                              | CardHandle   | Adressiert die Karte, für die die Blockierung der PIN aufgehoben werden soll. Unterstützt werden die Kartentypen EGK, HBAX und SM-B. Wird die Operation mit einem nicht unterstützten Kartentypen aufgerufen, so MUSS der Konnektor die Bearbeitung mit dem Fehler 4209 abbrechen.  |
|                              | PinTyp   | Gibt an, für welche PIN der Karte die Blockierung aufgehoben werden soll.<br>Erlaubte Belegung von PinTyp in Abhängigkeit der durch Cardhandle referenzierten Karte: <ul style="list-style-type: none"> <li>- eGK G1+: PIN.CH</li> <li>- eGK G2: PIN.CH, MRPIN.NFD, MRPIN.NFD_READ, MRPIN.DPE, MRPIN.GDD, MRPIN.OSE, MRPIN.AMTS, PIN.AMTS_REP</li> <li>- zusätzlich eGK G2.0: MRPIN.DPE_READ</li> <li>- HBAX: PIN.CH, PIN.QES</li> <li>- SM-B: PIN.SMC</li> </ul> |
|                              | SetNewPin  | Dieses Flag zeigt an, ob eine neue PIN gesetzt werden soll. Wird dieses Flag nicht angegeben, so wird FALSE angenommen.<br>Das Flag ist notwendig, um bei Eingabe am PIN-Pad eindeutig festzulegen, ob eine neue PIN gesetzt werden soll.   |
| <b>Rückgabe</b>              |  |   |

|                        |           |  |   |
|------------------------|-----------|--|---|
|                        | Name      | Beschreibung   |   |
|                        | LeftTries | Im Falle von REJECTED wird hier die Anzahl der verbleibenden möglichen Versuche für die Eingabe der PUK zurückgegeben. |   |
|                        | Status    | Enthält den Ausführungsstatus der Operation siehe 3.5.2  |   |
|                        | PinResult | Wert   | Bedeutung   |
|                        |           | OK   | Prüfung war erfolgreich.  |
|                        |           | ERROR  | Ein Bearbeitungsfehler ist aufgetreten. Fehlerursache im Feld Error.            |
|                        |           | REJECTED   | PUK war falsch. Die Anzahl der verbleibenden Versuche ist im Element LeftTries. |
|                        |           | WASBLOCKED   | PUK war zum Aufrufzeitpunkt bereits gesperrt                                    |
|                        |           | NOWBLOCKED   | PUK ist durch aktuellen Fehlversuch gesperrt                                    |
| <b>Vorbedingungen</b>  | keine     |  |   |
| <b>Nachbedingungen</b> | keine     |  |   |

3388 **Tabelle 122: TAB\_KON\_550 Ablauf UnblockPIN**

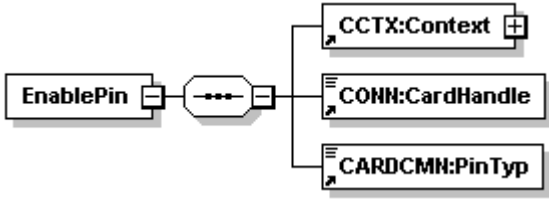
| Nr. | Aufruf Technischer Use Case oder Interne Operation | Beschreibung   |
|-----|--|--|
| 1.  | checkArguments                                     | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.   |
| 2.  | TUC_KON_000<br>„Prüfe Zugriffsberechtigung“        | Prüfung der Zugriffsberechtigung durch den Aufruf TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientSystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>userId = \$context.userId; // falls angegeben<br>cardHandle } |
| 3.  | TUC_KON_026<br>„Liefere CardSession“               | Ermittle cardSession über TUC_KON_026 {<br>mandantId = \$context.mandantId;<br>clientSystemId = \$context.clientsystemId;<br>userId = \$context.userId; // falls angegeben   |

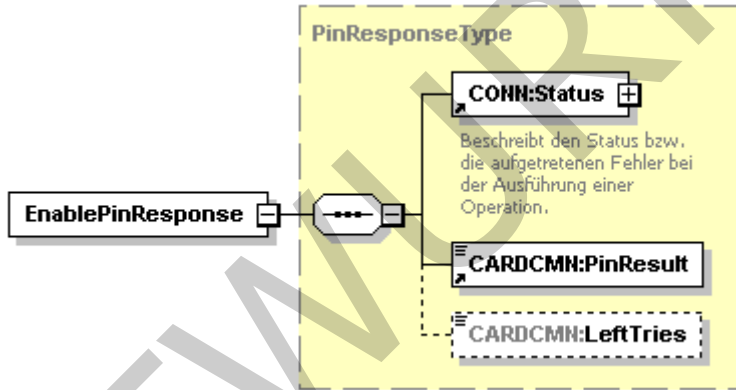
|    |                                 |  |
|----|---------------------------------|--|
|    |                                 | cardHandle }   |
| 4. | TUC_KON_021 „PIN entsperren“    | Rücksetzen des Fehlbedienungs Zählers über TUC_KON_021 {<br>cardSession;<br>workplaceId = \$context.workplaceId;<br>pinRef = pinRef(PinTyp);<br>setNewPIN = SetNewPIN }  |
| 5. | Verifikationsergebnis auswerten | Wenn TUC_KON_021 den Returncode BLOCKED liefert, wird dies als erfolgreicher Operationsdurchlauf mit PinResult= NOWBLOCKED zurückgegeben.<br>Wenn TUC_KON_021 mit dem Fehlercode 4064 abbricht, wird dies als erfolgreicher Operationsdurchlauf mit PinResult= WASBLOCKED zurückgegeben. |

3389 **Tabelle 123: TAB\_KON\_551 Fehlercodes „UnblockPin“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4000  | Technical | Error    | Syntaxfehler   |
| 4209  | Technical | Error    | Kartentyp %CardType% wird durch diese Operation nicht unterstützt. |

3390  
3391  
3392 **[<=]**3393 **4.1.5.5.5 EnablePin**3394 **TIP1-A\_5487 - Operation EnablePin**3395 Der Konnektor MUSS an der Außenschnittstelle eine Operation EnablePin, wie in Tabelle  
3396 TAB\_KON\_242 Operation EnablePin beschrieben, anbieten.3397 **Tabelle 124: TAB\_KON\_242 Operation EnablePin**

|                        |   |
|------------------------|---|
| <b>Name</b>            | EnablePin   |
| <b>Beschreibung</b>    | Schaltet für eine Multireferenz-PIN das Erfordernis, das Nutzergeheimnis zu verifizieren, <u>ein</u> , so dass der Sicherheitszustand nur durch eine erfolgreiche Benutzerverifikation gesetzt werden kann. |
| <b>Aufrufparameter</b> |   |

|                 |   |   |  |
|-----------------|---|---|--|
|                 | <b>Name</b>   | <b>Beschreibung</b>   |  |
|                 | Context   | MandantId, ClientSystemId, WorkplaceId verpflichtend;   |  |
|                 | CardHandle  | Adressiert die Karte, deren MRPIN bearbeitet werden soll. Es werden nur eGKs ab Generation 2 unterstützt.   |  |
|                 | PinTyp  | Gibt an, auf welche MRPIN der Karte die Operation angewendet werden soll.<br>Erlaubte Werte: <ul style="list-style-type: none"><li>eGK G2: MRPIN.NFD, MRPIN.DPE, MRPIN.GDD</li><li>zusätzlich ab eGK G2.1: MRPIN.AMTS</li></ul> |  |
| <b>Rückgabe</b> |  |   |  |
|                 | <b>Name</b>   | <b>Beschreibung</b>   |  |
|                 | Status  | Enthält den Ausführungsstatus der Operation, siehe 3.5.2  |  |
|                 | PinResult   | <b>Wert</b>   | <b>Bedeutung</b>   |
|                 |   | OK  | Aktivierung war erfolgreich  |
|                 |   | REJECTED  | PIN war falsch. Die Anzahl der verbleibenden Versuche ist im Element LeftTries |
|                 |   | ERROR   | Ein Bearbeitungsfehler ist aufgetreten. Fehlerursache im Feld Error            |
|                 |   | WASBLOCKED  | PIN war zum Aufrufzeitpunkt bereits gesperrt                                   |
|                 |   | NOWBLOCKED  | PIN ist durch aktuellen Fehlversuch gesperrt                                   |
|                 |   | TRANSPORT_PIN   | Dieser Wert wird nicht verwendet   |

|                      |   |   |
|----------------------|---|---|
|                      | LeftTries   | Im Falle von Result=REJECTED wird hier die Anzahl der verbleibenden möglichen Versuche zurückgegeben. |
| <b>Vorbedingung</b>  | keine   |   |
| <b>Nachbedingung</b> | Für das Erreichen des Sicherheitszustands der MRPIN ist eine Nutzereingabe erforderlich |   |

3398

3399

**Tabelle 125: TAB\_KON\_243 Ablauf EnablePin**

| Nr. | Aufruf<br>Technischer Use<br>Case oder Interne<br>Operation | Beschreibung   |
|-----|---|--|
| 1.  | checkArguments  | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.   |
| 2.  | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“            | Prüfung der Zugriffsberechtigung durch den Aufruf<br>TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientSystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>userId = \$context.userId;<br>cardHandle }<br>Tritt bei der Prüfung ein Fehler auf, so bricht die Operation mit dem Fehlercode aus TUC_KON_000 ab. |
| 3.  | TUC_KON_026<br>„Liefere<br>CardSession“                     | Ermittle cardSession über TUC_KON_026 {<br>mandantId = \$context.mandantId;<br>clientSystemId = \$context.clientsystemId;<br>userId = \$context.userId;<br>cardHandle }  |
| 4.  | TUC_KON_027 „PIN-<br>Schutz ein-<br>/ausschalten“           | Aktiviere das Erfordernis der Benutzerverifikation der MRPIN durch Aufruf des TUC_KON_027 „PIN-Schutz ein-/ausschalten“ {<br>cardSession;<br>pinRef = PinRef(PinType);<br>enable = true}   |
| 5.  | Verifikationsergebnis<br>auswerten                          | Als erfolgreicher Operationsdurchlauf wird nur PinResult=OK gewertet.<br>Alle anderen Resultate sind Fehlerfälle, und neben dem Status ist auch PinResult entsprechend zu setzen. Dabei gelten folgende Regeln:<br>Wenn TUC_KON_027 den PIN-Status BLOCKED liefert, wird auf PinResult=NOWBLOCKED abgebildet.                                      |

|  |  |   |
|--|--|---|
|  |  | Wenn TUC_KON_027 mit Fehler 4063 abbricht, wird dies auf PinResult=WASBLOCKED abgebildet. |
|--|--|---|

3400

3401 **Tabelle 126: TAB\_KON\_244 Fehlercodes „EnablePin“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4000  | Technical | Error    | Syntaxfehler  |
| 4072  | Technical | Error    | Ungültige PIN-Referenz <code>PinRef</code>                                      |
| 4209  | Technical | Error    | Kartentyp <code>%CardType%</code> wird durch diese Operation nicht unterstützt. |

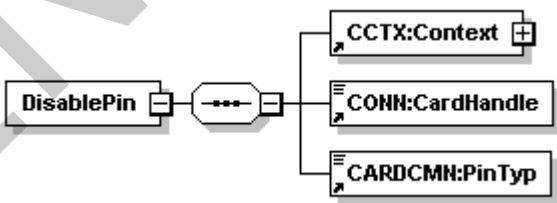
3402

3403

3404 [**<=**]3405 **4.1.5.5.6 DisablePin**3406 **TIP1-A\_5488 - Operation DisablePin**

3407 Der Konnektor MUSS an der Außenschnittstelle eine Operation DisablePin, wie in Tabelle  
 3408 TAB\_KON\_245 Operation DisablePin beschrieben, anbieten.

3409 **Tabelle 127: TAB\_KON\_245 Operation DisablePin**

|                        |   |   |  |
|------------------------|---|---|--|
| <b>Name</b>            | DisablePin  |   |  |
| <b>Beschreibung</b>    | Schaltet für eine Multireferenz-PIN das Erfordernis, das Nutzergeheimnis zu verifizieren, <u>ab</u> . Die MRPIN verhält sich danach bei allen Zugriffen auf die durch sie geschützten Objekte, als wäre sie freigeschaltet. |   |  |
| <b>Aufrufparameter</b> |   |   |  |
|                        | <b>Name</b>   | <b>Beschreibung</b>   |  |
|                        | Context   | MandantId, ClientSystemId, WorkplaceId verpflichtend;   |  |
|                        | CardHandle  | Adressiert die Karte, deren MRPIN bearbeitet werden soll. Es werden nur eGKS ab Generation 2 unterstützt. |  |
|                        | PinTyp  | Gibt an, auf welche MRPIN der Karte die Operation angewendet werden soll.<br>Erlaubte Werte:              |  |

|               |   |  |  |
|---------------|---|--|--|
|               |   | <ul style="list-style-type: none"><li>eGK G2: MRPIN.NFD, MRPIN.DPE, MRPIN.GDD</li><li>zusätzlich ab eGK G2.1: MRPIN.AMTS</li></ul> |  |
| Rückgabe      |   |  |  |
|               | Name  | Beschreibung   |  |
|               | Status  | Enthält den Ausführungsstatus der Operation<br>siehe 3.5.2   |  |
|               | PinResult   | Wert   | Bedeutung  |
|               |   | OK   | Aktivierung war erfolgreich  |
|               |   | REJECTED   | PIN war falsch. Die Anzahl der verbleibenden Versuche ist im Element LeftTries |
|               |   | ERROR  | Ein Bearbeitungsfehler ist aufgetreten. Fehlerursache im Feld Error            |
|               |   | WASBLOCKED   | PIN war zum Aufrufzeitpunkt bereits gesperrt                                   |
|               |   | NOWBLOCKED   | PIN ist durch aktuellen Fehlversuch gesperrt                                   |
|               |   | TRANSPORT_PIN  | Dieser Wert wird nicht verwendet   |
| LeftTries     | Im Falle von Result=REJECTED wird hier die Anzahl der verbleibenden möglichen Versuche zurückgegeben.   |  |  |
| Vorbedingung  | keine   |  |  |
| Nachbedingung | Der Sicherheitszustand der PIN ist dauerhaft (bis zur expliten Aktivierung mit EnablePin) gesetzt, ohne dass eine Nutzereingabe erforderlich wäre |  |  |



3410 **Tabelle 128: TAB\_KON\_246 Ablauf DisablePin**

| Nr. | Aufruf<br>Technischer<br>Use Case oder<br>Interne<br>Operation | Beschreibung   |
|-----|--|--|
| 1.  | checkArguments   | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.   |
| 2.  | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“               | Prüfung der Zugriffsberechtigung durch den Aufruf TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientSystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>userId = \$context.userId;<br>cardHandle }<br>Tritt bei der Prüfung ein Fehler auf, so bricht die Operation mit dem Fehlercode aus TUC_KON_000 ab.  |
| 3.  | TUC_KON_026<br>„Liefere<br>CardSession“                        | Ermittle cardSession über TUC_KON_026 {<br>mandantId = \$context.mandantId;<br>clientSystemId = \$context.clientSystemId;<br>userId = \$context.userId;<br>cardHandle }  |
| 4.  | TUC_KON_027<br>„PIN-Schutz ein-<br>/ausschalten“               | Deaktiviere das Erfordernis der Benutzerverifikation der MRPIN durch Aufruf des TUC_KON_027 „PIN-Schutz ein-/ausschalten“ {<br>cardSession;<br>pinRef = PinRef(PinType);<br>enable = false}  |
| 5.  | Verifikations-<br>ergebnis<br>auswerten                        | Als erfolgreicher Operationsdurchlauf wird nur PinResult=OK gewertet.<br>Alle anderen Resultate sind Fehlerfälle, und neben dem Status ist auch PinResult entsprechend zu setzen. Dabei gelten folgende Regeln:<br>Wenn TUC_KON_027 den PIN-Status BLOCKED liefert, wird auf PinResult=NOWBLOCKED abgebildet.<br>Wenn TUC_KON_027 mit Fehler 4063 abbricht, wird dies auf PinResult=WASBLOCKED abgebildet. |

3411

3412 **Tabelle 129: TAB\_KON\_247 Fehlercodes „DisablePin“**

| Fehlercode  | ErrorType | Severity | Fehlertext |
|---|-----------|----------|------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |            |

|      |           |       |   |
|------|-----------|-------|---|
| 4000 | Technical | Error | Syntaxfehler  |
| 4072 | Technical | Error | ungültige PIN-Referenz <code>PinRef</code>                                      |
| 4209 | Technical | Error | Kartentyp <code>%CardType%</code> wird durch diese Operation nicht unterstützt. |

3413  
3414  
3415

[<=]

#### 3416 4.1.5.6 Betriebsaspekte

##### 3417 TIP1-A\_4592 - Konfigurationswerte des Kartendienstes

3418 Der Konnektor MUSS es einem Administrator ermöglichen, Konfigurationsänderungen  
3419 gemäß Tabelle TAB\_KON\_554 vorzunehmen.

3420

##### 3421 Tabelle 130: TAB\_KON\_554 Konfiguration des Kartendienstes

| ReferenzID        | Belegung | Bedeutung   |
|-------------------|----------|---|
| CARD_TIMEOUT_CARD | Sekunden | Maximale Zeit, die ein Aufruf einer Kartenoperation dauern darf, bevor der Aufruf abgebrochen wird.<br>Der Konnektor MUSS sicherstellen, dass dieser Parameter einen Wert besitzt, mit dem ein reibungsloser Betrieb gewährleistet ist, und MUSS dem Administrator die Möglichkeit bieten, diesen Parameter zu konfigurieren. |

3422  
3423

[<=]

##### 3424 4.1.5.6.1 TUC\_KON\_025 "Initialisierung Kartendienst"

##### 3425 TIP1-A\_4593 - TUC\_KON\_025 „Initialisierung Kartendienst“

3426 Der Konnektor MUSS den technischen Use Case „Initialisierung Kartendienst“ gemäß  
3427 TUC\_KON\_025 umsetzen.

##### 3428 Tabelle 131: TAB\_KON\_555 - TUC\_KON\_025 „Initialisierung Kartendienst“

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_025 „Initialisierung Kartendienst“  |
| Beschreibung   | Nach dem Start des Kartendienstes MUSS der Konnektor für alle gesteckten Karten den TUC_KON_001 {ctId, slotId } aufrufen und CM_CARD_LIST befüllen. |
| Auslöser       | der Kartendienst wird gestartet   |
| Vorbedingungen | Kartenterminaldienst wurde gestartet  |
| Eingangsdaten  | CTM_CT_LIST   |
| Komponenten    | Karte, Kartenterminal, Konnektor  |
| Ausgangsdaten  | Aktuelle CM_CARD_LIST   |

|                                |  |
|--------------------------------|--|
| Standardablauf                 | <ol style="list-style-type: none"> <li>1. Rufe TUC_KON_001 „Karte öffnen“</li> <li>2. Wiederhole, bis für alle gesteckten Karten ein Eintrag in CM_CARD_LIST existiert.</li> </ol> |
| Varianten/Alternativen         | keine  |
| Fehlerfälle                    | keine  |
| Nichtfunktionale Anforderungen | Keine  |
| Zugehörige Diagramme           | Keine  |

3429

3430 [ $\leq$ ]

3431 4.1.5.6.2 Kartenübersicht und PIN-Management

3432 **TIP1-A\_5110 - Übersicht über alle verfügbaren Karten**3433 Die Administrationsoberfläche MUSS dem Administrator eine Übersichtsseite anbieten,  
3434 die alle in CM\_CARD\_LIST enthaltenen Karten listet.

3435 In dieser Übersichtsseite muss zu jeder Karte dargestellt werden:

- 3436 • CARD.CTID
- 3437 • CT(CARD.CTID).HOSTNAME
- 3438 • CARD.SLOTNO
- 3439 • CARD.TYPE
- 3440 • CARD.INSERTTIME
- 3441 • CARD.CARDHOLDERNAME

3442 Ferner MÜSSEN auf Verlangen des Administrators zu jeder Karte neben den obigen  
3443 Informationen auch folgende Details angezeigt werden:

- 3444 • CARD.ICCSN
- 3445 • CARD.CARDVERSION
- 3446 • CARD.CERTEXPIRATIONDATE

3447 [ $\leq$ ]3448 **TIP1-A\_5111 - PIN-Management der SM-Bs für den Administrator**3449 Über die Administrationsoberfläche MUSS der Administrator für jede in der  
3450 Übersichtsseite angezeigte Karte vom Typ SM-B die folgenden TUCs für die PIN.SMC  
3451 auslösen können.3452 Für diese MUSS er einen der gemäß Kapitel 4.1.1.6 [TIP1-A\_4526] definierten  
3453 Mandanten auswählen können:

- 3454 • TUC\_KON\_012 „PIN verifizieren“
- 3455 • TUC\_KON\_019 „PIN ändern“
- 3456 • TUC\_KON\_021 „PIN entsperren“
- 3457 • TUC\_KON\_022 „Liefere PIN-Status“

3458 Die Eingabe der PIN SOLL von jedem vom Informationsmodell her zulässigen  
3459 Kartenterminal aus möglich sein.  
3460 [ $\leq$ ]

3461 Der Konnektor kann den Administrator zur Laufzeit entscheiden lassen, an welchem  
3462 Kartenterminal die PIN eingegeben werden soll, indem er ihn wählen lässt, ob er die PIN  
3463 am Kartenterminal eingibt, in dem die betroffene SM-B steckt, oder ihn den Arbeitsplatz  
3464 wählen lässt, von dem aus er die Remote-PIN eingibt.

#### 3465 4.1.6 Systeminformationsdienst

3466 Der Systeminformationsdienst stellt Basisdiensten, Fachmodulen und Clientsystemen  
3467 sowohl aktiv (Push-Mechanismus) wie passiv (Pull-Mechanismus) Informationen zur  
3468 Verfügung. Dabei erhebt er selbst keine Daten, sondern dient nur als zentraler  
3469 Mechanismus, der von anderen Basisdiensten und Fachmodulen zur Verteilung und  
3470 Bereitstellung derer Informationen verwendet werden kann.

3471 Innerhalb des Systeminformationsdienstes werden folgende Präfixe für Bezeichner  
3472 verwendet:

- 3473 • Events (Topic Ebene 1): „EVT“
- 3474 • Konfigurationsparameter: „EVT\_“

#### 3475 Push-Mechanismus

3476 Der Push-Mechanismus des Systeminformationsdienstes hat die Aufgabe, Ereignisse von  
3477 internen Ereignisquellen im Konnektor (z. B. von anderen Basisdiensten wie  
3478 Kartendienst, Kartenterminaldienst oder Fachmodulen) an alle Basisdienste und  
3479 Fachmodule sowie an die bei ihm registrierten Ereignisempfänger (Clientsysteme)  
3480 weiterzuleiten.

3481 Die Namen der Ereignisse, die Topics, sind als Baumstruktur organisiert und werden  
3482 mittels „/“-getrennter Liste angegeben (z. B. „Auslöser/Ereigniskategorie1/.../Ereignis1“).  
3483 Die konkreten Topics werden innerhalb der einzelnen Funktionsmerkmale  
3484 kontextbezogen definiert und im Anhang in einer zentralen Liste übersichtlich dargestellt.

3485 Clientsysteme können sich für den Empfang bestimmter Ereigniskategorien (Topics)  
3486 anmelden. Der Systeminformationsdienst übernimmt dementsprechend bei der  
3487 Verteilung der Ereignisse auch eine Filterfunktion für die weiterzuleitenden Ereignisse.

3488 Die Zustellung der Ereignisse erfolgt unidirektional über eine Netzchnittstelle, deren  
3489 Kommunikationsendpunkt („Ereignissenke“) vom Clientsystem realisiert werden muss.  
3490 Zur Übertragung der Daten wird ein konnektoreigenes Protokoll cetp (Connector Event  
3491 Transport Protocol) verwendet.

#### 3492 Pull-Mechanismus

3493 Der Pull-Mechanismus des Systeminformationsdienstes hat die Aufgabe sowohl  
3494 Zustandswerte als auch statische Informationen des Konnektors selbst als auch von den  
3495 über ihn verwalteten Ressourcen durch Fachmodule und Clientsysteme abrufbar zu  
3496 machen. Dabei können entweder Listen von Ressourcen oder Details zu einzelnen  
3497 Ressourcen abgerufen werden.

3498 Die folgenden Unterkapitel regeln:

- 3499 • Das Kommunikationsprotokoll cetp
- 3500 • Die Struktur der Ereignisnachricht
- 3501 • Die TUCs für die Ereignisverteilung (PUSH)

- 3502 • Die TUCs und Operationen der Außenschnittstelle für den Abruf von
- 3503 Informationen (PULL)
- 3504 • Einstellungen, die der Administrator zur Steuerung des Verhaltens vornehmen
- 3505 kann.

#### 3506 **4.1.6.1 Funktionsmerkmalweite Aspekte**

##### 3507 **TIP1-A\_4594 - Richtung bei Verbindungsaufbau des**

##### 3508 **Systeminformationsdienstes**

3509 Der Konnektors MUSS zur Übertragung von Ereignissen eine cetp-Verbindung zu der

3510 Ereignissenke des Clientsystems aufbauen, die das Clientsystem beim Operationsaufruf

3511 Subscribe per `EventTo` angegeben hatte.

3512 [`<=`]

##### 3513 **TIP1-A\_5536 - Connector Event Transport Protocol über TCP**

3514 Der Konnektor MUSS das Anwendungsprotokoll cetp (Connector Event Transport

3515 Protocol) ausschließlich über das Transportprotokoll TCP (gegebenfalls TLS gesichert)

3516 anbieten.

3517 [`<=`]

##### 3518 **TIP1-A\_4595 - Gesicherte Übertragung von Ereignissen**

3519 Der Konnektor MUSS zur Übertragung der Ereignisse eine gesicherte Verbindung (TLS)

3520 verwenden, die vom Konnektor als TLS-Client initiiert wurde, wenn

3521 `ANCL_TLS_MANDATORY=Enabled`.

3522 Der Konnektor muss sich beim Aufbau der TLS-Sitzung gegenüber dem Clientsystem

3523 authentisieren, wenn dieses eine Authentisierung im Rahmen des TLS-Handshakes

3524 anfordert.

3525 Die Schalter `ANCL_CAUT_MODE` und `ANCL_CAUT_MANDATORY` wirken für die

3526 Übertragung der Ereignisse nicht.

3527 [`<=`]

3528 Für die Übermittlung der Ereignisse wurde ein leichtgewichtiges Protokoll gewählt, da

3529 vom Clientsystem keine Antwort auf Anwendungsebene erwartet wird.

##### 3530 **TIP1-A\_4596 - Nachrichtenaufbau und -kodierung des**

##### 3531 **Systeminformationsdienstes**

3532 Der Konnektors MUSS Ereignisse an Ereignissenken mittels Nachrichten verteilen, die

3533 gemäß `TAB_KON_030` „Ereignisnachricht“ aufgebaut sind. Der Konnektor MUSS die

3534 Nachrichten mit der Zeichenkette „CETP“ beginnen, gefolgt von der Länge der folgenden

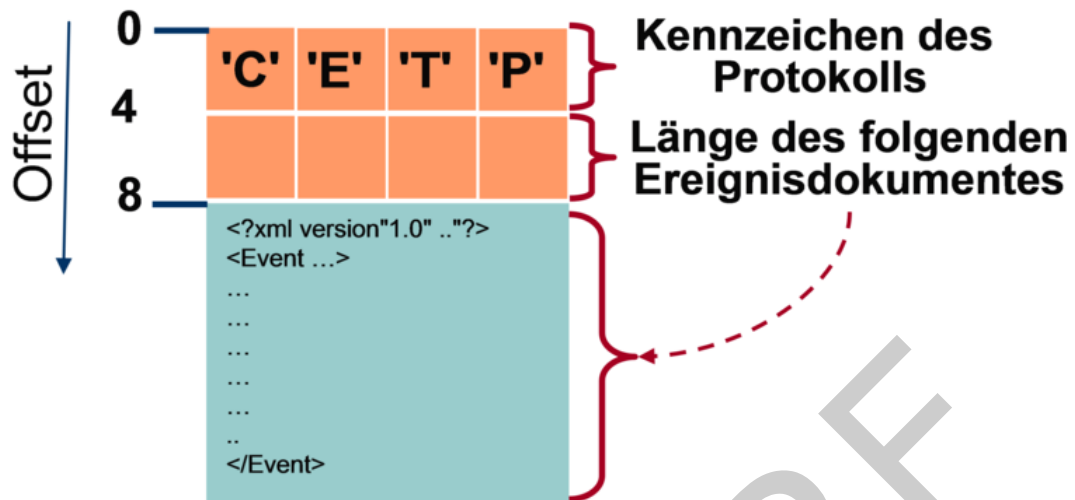
3535 Ereignisnachricht in Anzahl Bytes. Das vier Byte lange Längenfeld MUSS in der Byte-

3536 Reihenfolge Big-Endian codiert werden (das hochwertigste Byte wird als erstes

3537 übertragen).

3538

3539



3540

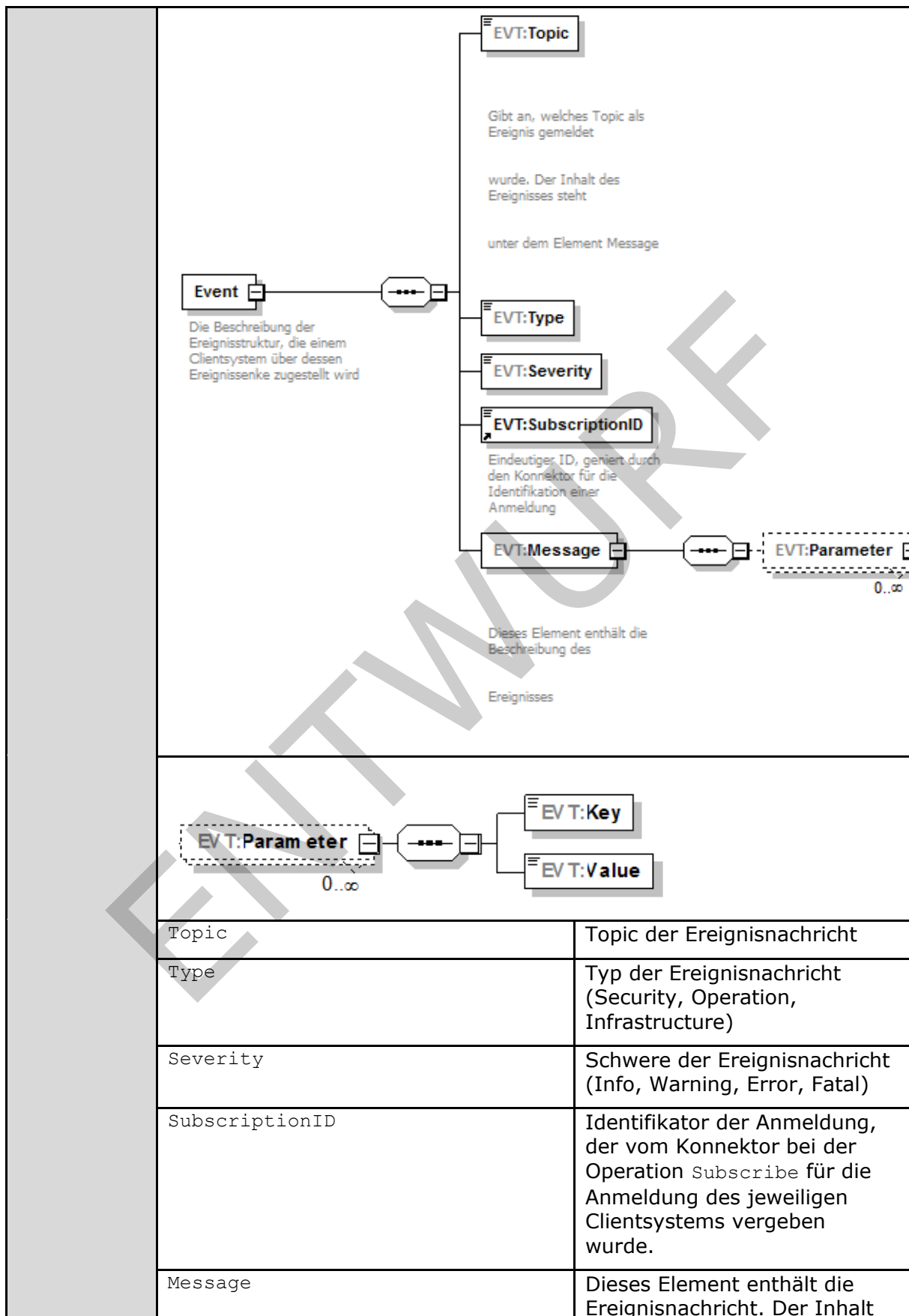
3541

3542

Abbildung 11: PIC\_KON\_022 Grundsätzlicher Aufbau der Ereignisnachricht

Tabelle 132: TAB\_KON\_030 Ereignisnachricht

|              |   |
|--------------|---|
| Beschreibung | Die Ereignisnachricht, die zur Ereignissenke gesendet wird, ist ein XML-Dokument. Die Ereignisnachricht wird in den „Umschlag“ Event gepackt. Wenn ein mandantenfähiges Clientsystem mehrere Anwendungskonnektoren verwendet, dann kann es die erhaltenen Ereignisbenachrichtigungen anhand der Subscription-ID einem Mandanten zuordnen. |
|--------------|---|



|          |   |  |
|----------|---|--|
|          |   | ist abhängig vom Topic und wird mittels „Key-Value“-Parametern übertragen. |
|          | Message/Parameter/Key                     | Name des Parameters (String), case sensitiv                                |
|          | Message/Parameter/Value                   | Wert des Parameters (String)   |
| Hinweise | Das XML-Dokument MUSS UTF-8-codiert sein. |  |

3543

3544

3545 [ $\leq$ ]3546 **4.1.6.2 Durch Ereignisse ausgelöste Reaktionen**

3547 Keine.

3548 **4.1.6.3 Interne TUCs, nicht durch Fachmodule nutzbar**

3549 Keine.

3550 **4.1.6.4 Interne TUCs, auch durch Fachmodule nutzbar**3551 *4.1.6.4.1 TUC\_KON\_256 „Systemereignis absetzen“*3552 **TIP1-A\_4598 - TUC\_KON\_256 „Systemereignis absetzen“**

3553 Der Konnektor MUSS für den PUSH-Mechanismus des Systeminformationsdienstes den technischen Use Case TUC\_KON\_256 „Systemereignis absetzen“ umsetzen.

3555

3556 **Tabelle 133: TAB\_KON\_556 - TUC\_KON\_256 „Systemereignis absetzen“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_256 „Systemereignis absetzen“   |
| Beschreibung   | Dieser TUC verteilt ein Ereignis im Konnektor intern (d.h. an Basisdienste und Fachmodule) sowie an Clientsysteme, die sich für den Empfang angemeldet haben (Operation <i>Subscribe</i> ). Zusätzlich wird, bei gesetztem Flag, das Ereignis durch den Protokollierungsdienst protokolliert.   |
| Auslöser       | Aufruf durch Basisdienst oder Fachmodul   |
| Vorbedingungen | Fall Topic = „BOOTUP/BOOTUP_COMPLETE“:<br>Zu allen URLs von clientseitigen Endpunkten, wie sie bei der <i>Subscribe</i> -Operation angegeben wurden, ist in der Subscription-Verwaltung des Konnektors eine TerminationTime gespeichert. Sie wird jeweils auf den Wert der TerminationTime der am längsten gültigen Subscription zu dem jeweiligen Endpunkt gesetzt. Die URLs von clientseitigen Endpunkten müssen bis zum Ablauf ihrer TerminationTime auch über Bootups hinweg gespeichert werden. Vor dem Versenden des BOOTUP_COMPLETE-Events werden sämtliche Subscriptions, jedoch nicht die URLs gelöscht. Bei |

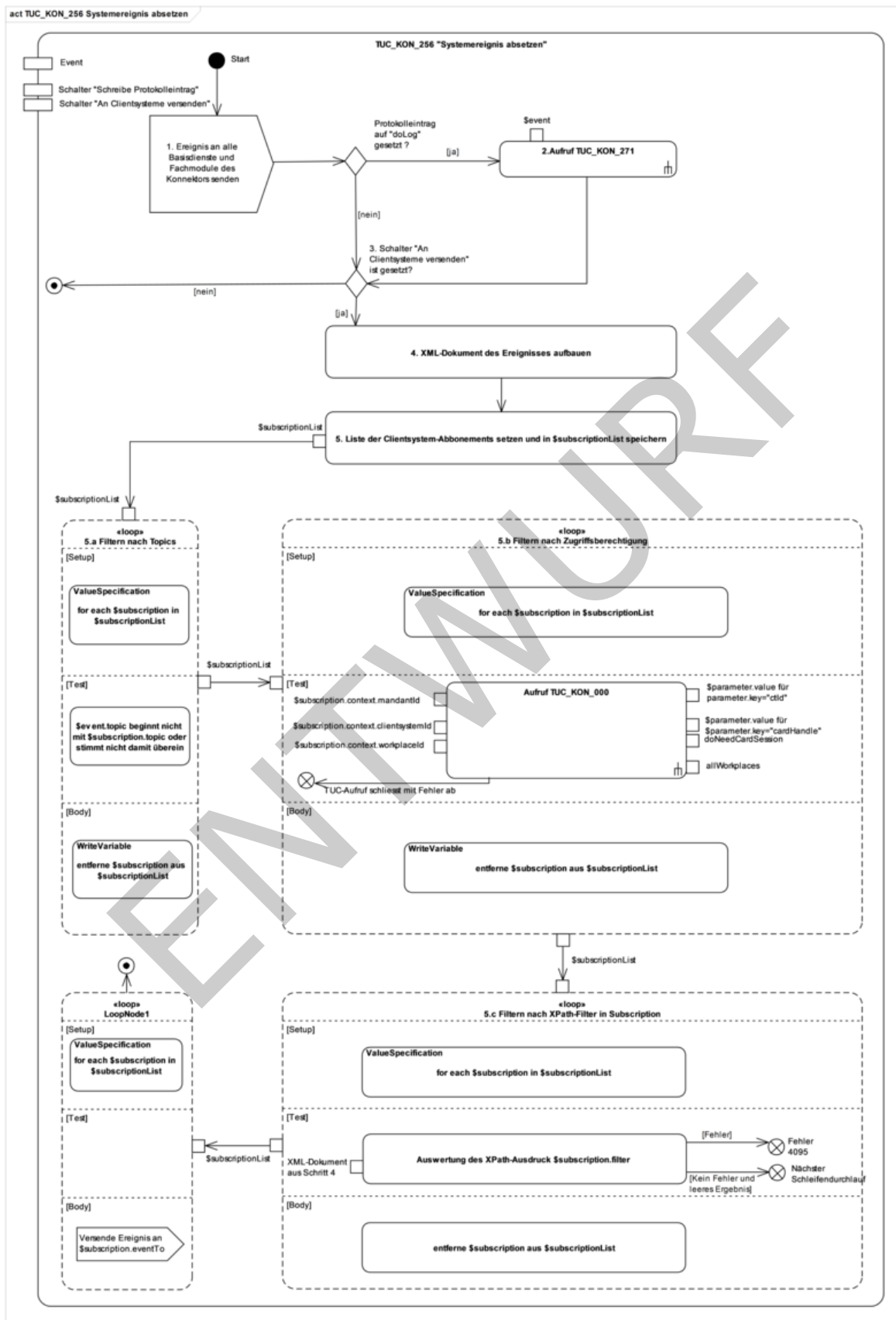


|                 |  |
|-----------------|--|
|                 | Ablauf ihrer TerminationTime werden nach dem Versenden des BOOTUP_COMPLETE-Events auch die URLs gelöscht.  |
| Eingangsdaten   | <p>Attribute des zu versendenden Ereignisses:</p> <ul style="list-style-type: none"> <li>• topic<br/>(Name des Ereignisses)</li> <li>• eventType [EventType]<br/>(Wenn statt eines EventType ein ErrorType übergeben wird, so wird der EventType daraus abgeleitet.<br/>Typ des Events: Op = Operation, Sec = Security, Infra = Infrastructure)</li> <li>• severity [EventSeverity]<br/>(Schwere des Ereignisses: Info = Information, Warn = Warning, Err = Error, Fatal)</li> <li>• parameters<br/>(weitere Parameter als key-value-Paare)</li> </ul> <p>Arbeitsanweisungen:</p> <ul style="list-style-type: none"> <li>• doLog [Boolean] – <i>optional; default = true</i><br/>(Schalter „Schreibe Protokolleintrag“)</li> <li>• doDisp [Boolean] – <i>optional; default = true</i><br/>(Schalter „An Clientsysteme versenden“)</li> </ul> <p>Die Bezeichnungen Op, Sec, Infra, Info, Warning, Err, Fatal werden nur in diesem Dokument verwendet und sind als Abkürzungen für die Werte Operation, Security, Infrastructure, Information, Warning, Error, Fatal in den jeweiligen Ereignisnachrichten gemäß Schema EventService.xsd zu verstehen.</p> |
| Komponenten     | Konnektor  |
| Ausgangsdaten   | Keine  |
| Nachbedingungen |  |
| Standardablauf  | <p>Für das übergebene Ereignis werden folgende Schritte durchgeführt:</p> <ol style="list-style-type: none"> <li>1. Das Ereignis wird an alle Basisdienste und Fachmodule des Konnektors gesendet.</li> <li>2. Wenn doLog = true, erfolgt der Aufruf von TUC_KON_271 {<br/> eventType = \$Event.eventType<br/> (mit eventType = „Op“, wenn \$Event.eventType in {„Op“, „Infra“}<br/> mit eventType = „Sec“, wenn \$Event.eventType gleich "Sec")<br/> severity=\$Event. severity;<br/> parameters= (\$Event.topic, \$Event.parameters) }<br/> Die Einschränkungen zur Protokollierung personenbezogener Daten<br/> gemäß TIP1-A_4710 müssen beim Aufruf von TUC_KON_271 beachtet werden.</li> </ol>  |

|  |  |
|--|--|
|  | <p>3. Falls doDisp = false ist, wird an dieser Stelle abgebrochen.</p> <p>4. Das für den Versand an Clientsysteme benötigte XML-Dokument des Ereignisses wird aufgebaut (Element „Event“ gemäß EventService.XSD).</p> <p>5. Setze \$subscriptionList = Liste der Clientsystem-Abonnements, die durch die Operationen Subscribe/Unsubscribe gepflegt werden und deren TerminationTime &gt; Systemzeit.</p> <p>Im Folgenden durchläuft diese Liste der Reihe nach drei Filter. Nach dem letzten Filterschritt enthält \$subscriptionList nur noch die Abonnements, an die das Ereignis versendet werden soll.</p> <p>a. Filtern nach Topics:<br/>für jede \$subscription in \$subscriptionList {<br/>wenn \$event.topic nicht mit \$subscription.topic beginnt oder übereinstimmt (case insensitive Vergleich),<br/>dann entferne \$subscription aus \$subscriptionList<br/>}</p> <p>b. Filtern nach Zugriffsberechtigung:<br/>für jede \$subscription in \$subscriptionList {<br/>wenn TUC_KON_000 mit einem Fehler abgeschlossen wird, dann entferne \$subscription aus \$subscriptionList.<br/>Wenn cardHandle in parameters übergeben wurde, dann<br/>TUC_KON_000 {<br/>mandantId = \$subscription.context.mandantId;<br/>clientId = \$subscription.context.clientsystemId;<br/>workplaceId = \$subscription.context.workplaceId;<br/>ctId = \$parameters.value<br/>für \$parameters.key = „ctId“<br/>cardHandle = \$parameters.value<br/>für \$parameters.key = „cardHandle“;<br/>needCardSession = false;<br/>allWorkplaces = false<br/>}<br/>oder im Fall nicht gegebenes cardHandle<br/>TUC_KON_000 {<br/>mandantId = \$subscription.context.mandantId;<br/>clientId = \$subscription.context.clientsystemId;<br/>workplaceId = \$subscription.context.workplaceId;<br/>ctId = \$parameters.value<br/>für \$parameters.key = „ctId“<br/>needCardSession = false;<br/>allWorkplaces = false<br/>} }<br/>c. Filtern nach XPath-Filter in Subscription ([XPATH]):<br/>für jede \$subscription in \$subscriptionList {<br/>wenn der XPath-Ausdruck \$subscription.filter angewandt auf das als XML-Dokument dargestellte Ereignis ein leeres Ergebnis liefert,</p> |
|--|--|

|                                |   |
|--------------------------------|---|
|                                | <p>dann entferne \$subscription aus \$subscriptionList</p> <pre>     } 6. Versenden:     für jede \$subscription in \$subscriptionList {         versende das Ereignis an \$subscription.eventTo     } </pre> <p>Für das versendete Ereignis wird keine Antwort durch das Clientsystem erwartet.</p>  |
| Varianten/<br>Alternativen     | <p>Fall Topic = „BOOTUP/BOOTUP_COMPLETE“:</p> <p>4. Das für den Versand an Clientsysteme benötigte XML-Dokument des Ereignisses wird aufgebaut (Element „Event“ gemäß EventService.XSD, SubscriptionID als leeres Element).</p> <p>5. Setze \$urlList = Liste der URLs von clientseitigen Endpunkten, wie sie bei der Subscribe-Operation angegeben wurden. Clientsysteme, deren Subscription-URL beim Einschalten des Konnektors noch nicht gelöscht waren, müssen benachrichtigt werden, auch wenn dann bereits deren TerminationTime &lt; Systemzeit ist.</p> <p>Versenden:</p> <pre> für jede \$url in \$urlList {     versende das Ereignis an \$url } </pre> <p>Für das versendete Ereignis wird keine Antwort durch das Clientsystem erwartet. Dadurch wird bei einer Nichtzustellung auch kein erneuter Versand des Ereignisses angestoßen, da der Konnektor keine Kenntnis über den Erfolg einer Ereignisübermittlung hat.</p> |
| Fehlerfälle                    | <p>Fehler in den folgenden Schritten des Standardablaufs führen zum Abbruch mit den ausgewiesenen Fehlercodes:</p> <p>(→5c) Fehler bei der Auswertung des XPath-Ausdrucks:<br/>Fehlercode: 4095, nur für die jeweilige Abonnement-Prüfung.</p>  |
| Fachliche Fehlermeldung        | Keine   |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Abbildung PIC_KON_112 Aktivitätsdiagramm zu „Systemereignis absetzen“   |

3557



3558

**Abbildung 12: PIC\_KON\_112 Aktivitätsdiagramm zu „Systemereignis absetzen“**

**Tabelle 134: TAB\_KON\_557 Fehlercodes TUC\_KON\_256 „Systemereignis absetzen“**

| Fehlercode  | ErrorType | Severity | Fehlertext                                     |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4095  | Technical | Error    | Fehler bei der Auswertung eines XPath-Ausdruck |

[<=]

**4.1.6.4.2 TUC\_KON\_252 „Liefere KT\_Liste“**

**TIP1-A\_4599 - TUC\_KON\_252 „Liefere KT\_Liste“**

Der Konnektor MUSS den technischen Use Case TUC\_KON\_252 „Liefere KT\_Liste“ umsetzen.

**Tabelle 135: TAB\_KON\_558 – TUC\_KON\_252 „Liefere KT\_Liste“**

| Element         | Beschreibung  |
|-----------------|---|
| Name            | TUC_KON_252 „Liefere KT_Liste“  |
| Beschreibung    | Dieser TUC liefert eine Liste der Kartenterminals, die unter Beachtung der Eingangsdaten verfügbar/erreichbar sind.   |
| Auslöser        | Aufruf durch ein Clientsystem (Operation <code>GetCardTerminals</code> ) oder ein Fachmodul   |
| Vorbedingungen  | Keine   |
| Eingangsdaten   | <ul style="list-style-type: none"> <li>workplaceId - <i>optional</i> (Arbeitsplatz ID)</li> <li>clientSystemId (Clientssystem ID)</li> <li>mandantId (Mandanten ID)</li> </ul>  |
| Komponenten     | Konnektor, Kartenterminal   |
| Ausgangsdaten   | <ul style="list-style-type: none"> <li>cardTerminals (Liste der Kartenterminals, die den angegebenen Arbeitsplätzen, Mandanten und Clientsystemen zugeordnet sind bzw. auf die diese zugreifen dürfen (siehe Zugriffsberechtigungsdienst), sowie deren aktuelle Verfügbarkeit. Verfügbarkeit bedeutet im Falle eines eHealth-Kartenterminals, dass der Konnektor eine Verbindung zum Kartenterminal aktuell hält.)</li> </ul> |
| Nachbedingungen | <ul style="list-style-type: none"> <li>Der Zustand der Kartenterminals bleibt unverändert</li> </ul>  |

|                                |   |
|--------------------------------|---|
| Standardablauf                 | <ol style="list-style-type: none"> <li>Erstellen der Liste aller Kartenterminals, auf die der angegebene Mandant und das angegebene Clientsystem zugreifen dürfen (siehe Zugriffsberechtigungsdienst) <ol style="list-style-type: none"> <li>Wurde der optionale Parameter workplaceId ID übergeben, so werden nur die Kartenterminals in die Liste aufgenommen, die diesem Arbeitsplatz zugeordnet sind (siehe Zugriffsberechtigungsdienst). Dazu zählen insbesondere nicht die als entfernte Kartenterminals bezeichneten KT.</li> <li>Fehlt dieser Parameter, so werden alle Kartenterminals in die Liste aufgenommen, die sowohl dem Clientsystem als auch dem Mandanten zugeordnet sind.</li> </ol> </li> <li>Rückgabe der Liste cardTerminals (der in Schritt 1 erstellten Liste) mit Angaben zu jedem Kartenterminal gemäß Schema „Eventservice.xsd“.</li> </ol> |
| Varianten/Alternativen         | Keine   |
| Fehlerfälle                    | Keine   |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

3570

3571 [ $\leq$ ]

3572 4.1.6.4.3 TUC\_KON\_253 „Liefere Karten\_Liste“

3573 **TIP1-A\_4600 - TUC\_KON\_253 „Liefere Karten\_Liste“**3574 Der Konnektor MUSS den technischen Use Case TUC\_KON\_253 „Liefere Karten\_Liste“  
3575 umsetzen.

3576

3577 **Tabelle 136: TAB\_KON\_559 – TUC\_KON\_253 „Liefere Karten\_Liste“**

| Element             | Beschreibung  |
|---------------------|---|
| Name                | TUC_KON_253 „Liefere Karten_Liste“  |
| Beschreibung        | Dieser TUC liefert eine Liste der gesteckten Karten, die unter Beachtung der Eingangsdaten verfügbar/erreichbar sind. |
| Auslöser            | Aufruf durch ein Clientsystem (Operation GetCards) oder ein Fachmodul   |
| Vorbedingungen      | Keine   |
| Eingangsanforderung | Keine   |
| Eingangsdaten       | <ul style="list-style-type: none"> <li>workplaceId – <i>optional</i> (Arbeitsplatz-ID)</li> </ul>                     |

|                 |  |
|-----------------|--|
|                 | <ul style="list-style-type: none"> <li>• clientSystemId<br/>(Clientssystem ID)</li> <li>• cardTerminalId - <i>optional</i>; <i>verpflichtend, wenn slotId übergeben wird</i><br/>(Kartenterminalidentifikator)</li> <li>• slotId - <i>optional</i><br/>(Nummer des Slots, beginnend bei 1)</li> <li>• mandantId<br/>(Mandanten ID)</li> <li>• cardType - <i>optional</i><br/>(Kartentyp gemäß Tabelle TAB_KON_500)</li> </ul>  |
| Komponenten     | Konnektor, Kartenterminal, Karte   |
| Ausgangsdaten   | <ul style="list-style-type: none"> <li>• cards<br/>(Liste der gesteckten Karten einschließlich der Informationen für CARD:card, auf die der Mandant und das Clientssystem von dem Arbeitsplatz aus zugreifen dürfen (siehe Zugriffsberechtigungsdienst)).<br/>Wird workplaceId nicht übergeben, so werden alle vom Clientssystem und dem Mandant erreichbaren Kartenterminals in die Liste aufgenommen. Die Eingangsdaten dienen als Filter, welche Karten in cards zurückgegeben werden.<br/>Beispiel: Falls cardTerminalId angegeben ist, werden nur Karten in die Liste aufgenommen, die im entsprechenden Kartenterminal stecken.)</li> </ul>  |
| Nachbedingungen | <ul style="list-style-type: none"> <li>• Der Zustand der Kartenterminals und der Karten bleibt unverändert</li> </ul>  |
| Standardablauf  | <ol style="list-style-type: none"> <li>1. Erstellen der Liste aller Karten, auf die der angegebene Mandant und das angegebene Clientssystem zugreifen dürfen (siehe Zugriffsberechtigungsdienst). <ol style="list-style-type: none"> <li>a. Wurde cardTerminalId übergeben, dann nur Karten berücksichtigen, die dem dadurch referenziertem Kartenterminal zugeordnet sind.</li> <li>b. Wurde außer cardTerminalId auch slotId übergeben, so ist nur die Karte zu berücksichtigen, die in dem angegebenen Slot steckt.</li> <li>c. Wurde workplaceId übergeben, so werden nur die Karten in die Liste aufgenommen, auf die von diesem Arbeitsplatz aus zugegriffen werden darf (siehe „Zugriffsberechtigung Ressourcen“).</li> <li>d. Wurde cardType übergeben, so werden nur die Karten in die Liste aufgenommen, die dem Kartentyp in CardType entsprechen.</li> </ol> </li> </ol> |

|                                   |   |
|-----------------------------------|---|
|                                   | 2. Rückgabe cards, der in Schritt 1 erstellten Liste mit Angaben zu jeder Karte gemäß Schema „Eventservice.xsd“.  |
| Varianten/<br>Alternativen        | Keine   |
| Fehlerfälle                       | Fehler in den folgenden Schritten des Standardablaufs führen zum Abbruch mit den ausgewiesenen Fehlercodes:<br>(→1 a) Ungültige Kartenterminal-ID: Fehlercode: 4007 |
| Nichtfunktionale<br>Anforderungen | Keine   |
| Zugehörige<br>Diagramme           | Keine   |

3578

3579

**Tabelle 137: TAB\_KON\_560 Fehlercodes TUC\_KON\_253 „Liefere Karten\_Liste“**

| Fehlercode  | ErrorType | Severity | Fehlertext                  |
|---|-----------|----------|-----------------------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |                             |
| 4007  | Technical | Error    | ungültige Kartenterminal-ID |

3580

3581 [ $\leq$ ]

3582 4.1.6.4.4 TUC\_KON\_254 „Liefere Ressourcendetails“

3583 **TIP1-A\_4602 - TUC\_KON\_254 „Liefere Ressourcendetails“**

3584 Der Konnektor MUSS den technischen Use Case TUC\_KON\_254 „Liefere  
 3585 Ressourcendetails“ umsetzen.

3586

3587 **Tabelle 138: TAB\_KON\_561 - TUC\_KON\_254 „Liefere Ressourcendetails“**

| Element             | Beschreibung  |
|---------------------|---|
| Name                | TUC_KON_254 „Liefere Ressourcendetails“   |
| Beschreibung        | Dieser TUC liefert Detailinformationen zu einer Ressource (KT, Karte) oder dem Konnektor  |
| Auslöser            | Aufruf durch ein Clientsystem (Operation <code>GetResourceInformation</code> ) oder ein Fachmodul   |
| Vorbedingungen      | Keine   |
| Eingangsanforderung | Keine   |
| Eingangsdaten       | <ul style="list-style-type: none"> <li>clientSystemId (Clientsystem ID)</li> <li>mandantId (Mandanten ID)</li> <li>workplaceId – <i>optional</i> (Arbeitsplatz ID)</li> </ul> |



|                            |  |
|----------------------------|--|
|                            | <ul style="list-style-type: none"> <li>• cardTerminalId – <i>optional</i><br/>(Kartenterminal ID)</li> <li>• slotId – <i>optional/zulässig nur, wenn auch cardTerminalId angegeben ist</i><br/>(Kartenslot-Nummer)</li> <li>• cardHandle – <i>optional</i></li> <li>• iccsn – <i>optional</i></li> </ul>   |
| Komponenten                | Konnektor, Kartenterminal, Karte, HSM  |
| Ausgangsdaten              | <ul style="list-style-type: none"> <li>• resource<br/>(Informationsobjekt einer Ressource<br/>(Kartenterminal, Karte, HSM))</li> </ul>   |
| Nachbedingungen            | <ul style="list-style-type: none"> <li>• Der Zustand der Kartenterminals, Karten und HSM bleibt unverändert</li> </ul>   |
| Standardablauf             | <ol style="list-style-type: none"> <li>1. Falls cardTerminalId und slotId übergeben wurde oder in den Eingangsparametern iccsn oder cardHandle enthalten ist, wird ein Informationsobjekt der Karte, die sich in dem angegebenen Slot befindet bzw. die über die Iccsn oder das CardHandle identifiziert werden kann, zurückgegeben.</li> <li>2. Falls cardTerminalId, aber keine slotId übergeben wurde, wird ein Informationsobjekt des Kartenterminals zurückgegeben.</li> <li>3. Wurde weder iccsn, cardHandle, cardTerminalId noch eine slotId übergeben, so wird ein Informationsobjekt des Konnektors zurückgegeben. Für das Element ErrorCondition ist aus der Tabelle TAB_KON_503 Betriebszustand_Fehlerzustandsliste der Text aus der Spalte ErrorCondition zu übernehmen, ggf. mit den in dieser Spalte angegebenen Parameterwerten. Vor der Rückgabe der Informationen über eine Ressource wird geprüft, ob der angegebene Mandant und das angegebene Clientsystem darauf zugreifen dürfen (siehe Zugriffsberechtigungsdiens). Wurde zusätzlich der optionale Parameter workplaceId übergeben, so wird auch geprüft, ob die Ressource diesem Arbeitsplatz zugeordnet ist. Die Rückgabe der Informationen erfolgt gemäß dem Schema „Eventservice.xsd“.</li> </ol> |
| Varianten/<br>Alternativen | Keine  |
| Fehlerfälle                | <p>Fehler in den folgenden Schritten des Standardablaufs führen zum Abbruch mit den ausgewiesenen Fehlercodes:</p> <p>(→1) Ungültige Kartenterminal-ID: Fehlercode: 4007</p> <p>(→1) Ungültige Kartenslot-ID: Fehlercode: 4097</p> <p>(→1) Keine Karte im angegebenen Slot: Fehlercode: 4098</p> <p>(→1) Keine Karte mit angegebener Iccsn gefunden: Fehlercode: 4099</p>  |

|                                |   |
|--------------------------------|---|
|                                | (→1) Karten-Handle ungültig: Fehlercode: 4101<br>(→2) Ungültige Kartenterminal-ID: Fehlercode: 4007 |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

3588 **Tabelle 139: TAB\_KON\_562 Fehlercodes TUC\_KON\_254 „Liefere Ressourcendetails“**

| Fehlercode  | ErrorType | Severity | Fehlertext                                 |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4007  | Technical | Error    | ungültige Kartenterminal-ID                |
| 4097  | Technical | Error    | ungültige Kartenslot-ID                    |
| 4098  | Technical | Error    | keine Karte im angegebenen Slot gefunden   |
| 4099  | Technical | Error    | keine Karte zur angegebenen Iccsn gefunden |
| 4101  | Technical | Error    | Karten-Handle ungültig                     |

3589  
3590 [ $\leq$ ]

#### 3591 4.1.6.5 Operationen an der Außenschnittstelle

##### 3592 **TIP1-A\_4603-02TIP1-A\_4603 - Basisanwendung Systeminformationsdienst**

3593 Der Konnektor MUSS für Clients eine Basisanwendung Systeminformationsdienst  
3594 anbieten.  
3595

3596 **Tabelle 140 TAB\_KON\_029 Basisanwendung Systeminformationsdienst**

| Name              | EventService   |  |
|-------------------|--|--|
| Version           | 7.2.0<br><a href="#">Siehe Anhang D</a> (WSDL-Version) <a href="#">7.2.1</a> (XSD-Version) |  |
| Namensraum        | Siehe <a href="#">Anhang D</a> <a href="#">GitHub</a>                                      |  |
| Namensraum-Kürzel | EVT für Schema und EVTW für WSDL   |  |
| Operationen       | Name   | Kurzbeschreibung                           |
|                   | GetCardTerminals   | Auflistung der verfügbaren Kartenterminals |
|                   | GetCards   | Auflistung der gesteckten Karten           |

|               |                        |   |
|---------------|------------------------|---|
|               | GetResourceInformation | Liefert Details zu einer Ressource (Kartenterminal, Karte, HSM) |
|               | Subscribe              | Anmeldung der Zustellung von Ereignissen                        |
|               | Unsubscribe            | Abmelden von der Zustellung von Ereignissen                     |
|               | RenewSubscriptions     | Gültigkeit bestehender Subscriptions verlängern                 |
|               | GetSubscriptions       | Abfrage der angemeldeten Zustellungen von Ereignissen           |
| <b>WSDL</b>   | EventService.wsdl      |   |
| <b>Schema</b> | EventService.xsd       |   |

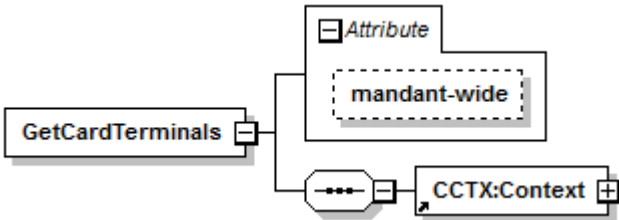
[&lt;=]

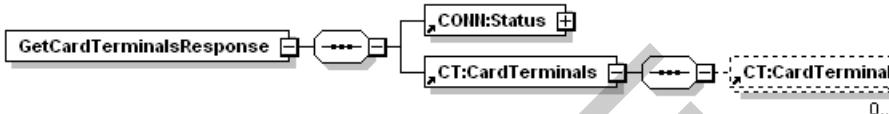
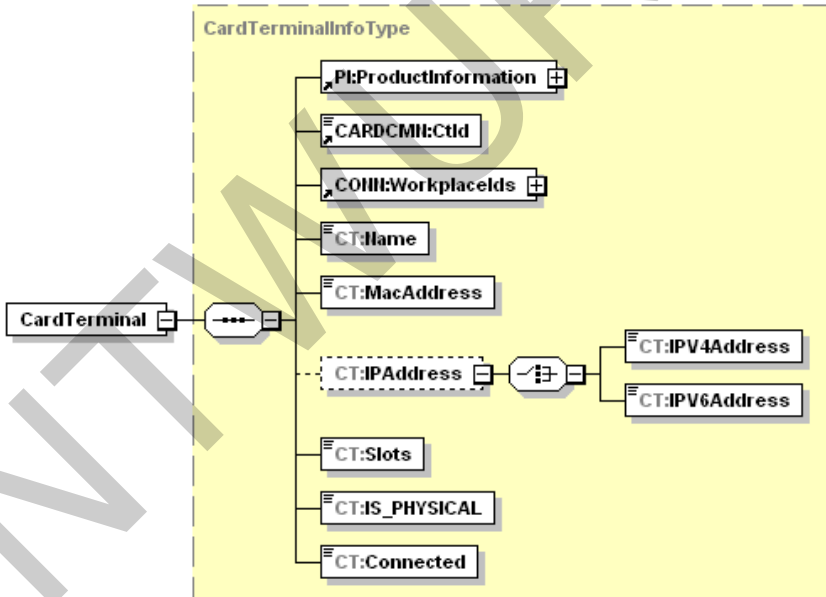
#### 4.1.6.5.1 GetCardTerminals

#### TIP1-A\_4604 - Operation GetCardTerminals

Der Konnektor MUSS an der Außenschnittstelle eine Operation GetCardTerminals, wie in Tabelle TAB\_KON\_563 „Operation GetCardTerminals“ beschrieben, anbieten.

**Tabelle 141: TAB\_KON\_563 Operation GetCardTerminals**

|                        |   |                     |
|------------------------|---|---------------------|
| <b>Name</b>            | GetCardTerminals  |                     |
| <b>Beschreibung</b>    | Liefert die Liste der Kartenterminals, auf die der aufrufende Mandant und das aufrufende Clientsystem zugreifen dürfen (siehe Zugriffsberechtigungsdienst) sowie deren aktuelle Verfügbarkeit. Verfügbarkeit bedeutet im Falle eines eHealth-Kartenterminals, dass der Konnektor eine Verbindung zum Kartenterminal aktuell hält. |                     |
| <b>Aufrufparameter</b> |   |                     |
|                        | <b>Name</b>   | <b>Beschreibung</b> |

|                 |   |  |
|-----------------|---|--|
|                 | @mandant-wide   | Wenn „true“, werden alle Kartenterminals zurückgegeben, auf die der Mandant und das aufrufende Clientsystem zugreifen dürfen. Wenn „false“ (Standardbelegung), werden nur Kartenterminals zurückgegeben, auf die von dem im Aufrufkontext spezifizierten Arbeitsplatz zugegriffen werden darf. |
|                 | Context   | Aufrufkontext  |
| <b>Rückgabe</b> |   |  |
|                 | <b>Name</b>   | <b>Beschreibung</b>  |
|                 | Status  | Ergebnis der Operation   |
|                 |  |  |
|                 | Die Liste der Kartenterminals   |  |
|                 | <b>Name</b>   | <b>Beschreibung</b>  |
|                 | Product Information   | Produktinformationen gemäß [gemSpec_OM] und dem Schema „ProductInformation.xsd“ zu formatieren.  |
|                 | CtId  | Eindeutige Identifikation des Kartenterminals  |
|                 | WorkplaceIds  | Die Liste der Arbeitsplätze, denen das Kartenterminal als lokales Kartenterminal zugeordnet ist. Insbesondere für Entfernte Kartenterminals kann diese Liste leer sein (siehe TUC_KON_252).  |
|                 | Name  | Sprechender Name des Kartenterminals   |

|                        |  |   |
|------------------------|--|---|
|                        | MacAddress   | MAC-Adresse des Kartenterminals   |
|                        | IPAddress  | IP-Adresse des Kartenterminals  |
|                        | Slots  | Anzahl der Slots des Kartenterminals  |
|                        | IS_PHYSICAL  | Attribut des Kartenterminals das anzeigt ob es sich um ein physisches oder logisches Kartenterminal handelt<br>(siehe auch TAB_KON_522 Parameterübersicht des Kartenterminaldienstes) |
|                        | Connected  | Zeigt an, ob dieses Kartenterminal aktuell verfügbar ist.<br>TRUE – ist verfügbar<br>FALSE – ist nicht verfügbar  |
| <b>Vorbedingungen</b>  | Keine  |   |
| <b>Nachbedingungen</b> | Der Zustand der Kartenterminals bleibt unverändert.  |   |
| <b>Hinweise</b>        | Der Aufruf DARF nur den im Konnektor verwalteten, aktuellen Zustand des Kartenterminals liefern und DARF NICHT Abfragen an die Kartenterminals absetzen. |   |

Der Ablauf der Operation GetCardTerminals ist in Tabelle TAB\_KON\_564 Ablauf GetCardTerminals beschrieben:

**Tabelle 142: TAB\_KON\_564 Ablauf GetCardTerminals**

| Nr. | Aufruf<br>Technischer Use<br>Case oder Interne<br>Operation | Beschreibung  |
|-----|---|---|
| 1.  | checkArguments  | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.  |
| 2.  | TUC_KON_000<br>„Prüfe Zugriffsberechtigung“                 | Prüfung der Zugriffsberechtigung durch den Aufruf TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientSystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>needCardSession = false;<br>allWorkplaces = @mandant-wide }<br>Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.   |
| 3.  | TUC_KON_252<br>„Liefere KT_Liste“                           | Die Liste der Kartenterminals wird erstellt und zurückgegeben. Tritt hierbei ein Fehler auf, so bricht die Operation mit dem Fehler des TUCs ab.<br>Wenn @mandant-wide=true dann ermittle die Liste der Kartenterminals für alle Arbeitsplätze des Mandanten für das angegebene Clientsystem durch den Aufruf von TUC_KON_252{<br>clientSystemId = \$context.ClientSystemId;<br>mandantId = \$context.mandantId } |

|  |  |   |
|--|--|---|
|  |  | Wenn @mandant-wide=false dann ermittle die Liste der Kartenterminals für den Arbeitsplatz des Mandanten für das angegebene Clientsystem entsprechend \$context durch den Aufruf von TUC_KON_252{<br>workplaceId = \$context.workplaceId;<br>clientSystemId = \$context.ClientSystemId;<br>mandantId = \$context.mandantId } |
|--|--|---|

3610

3611 **Tabelle 143: TAB\_KON\_823 Fehlercodes „GetCardTerminals“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |              |
| 4000  | Technical | Error    | Syntaxfehler |

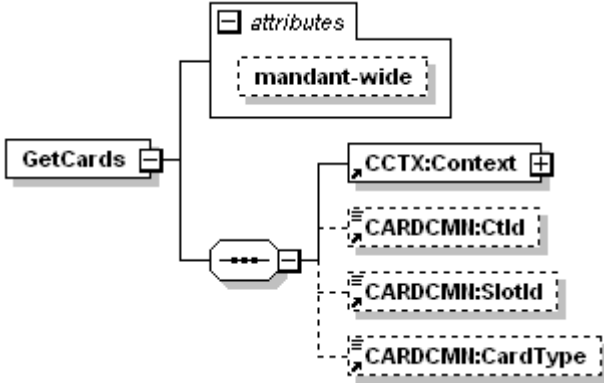
3612

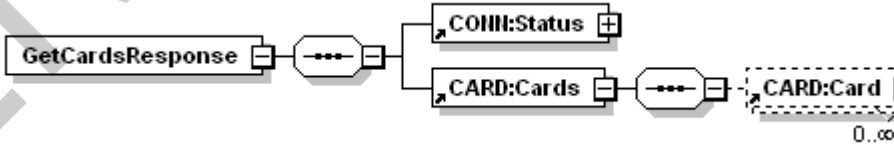
3613

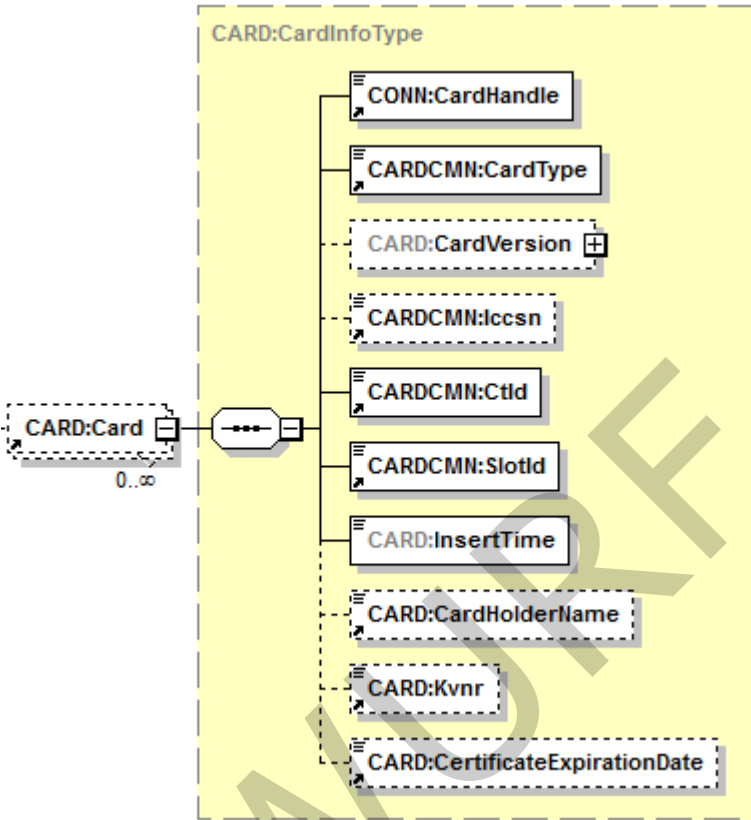
3614 [**<=**]3615 **4.1.6.5.2 GetCards**3616 **TIP1-A\_4605 - Operation GetCards**

3617 Der Konnektors MUSS an der Außenschnittstelle eine Operation GetCards, wie in Tabelle  
 3618 TAB\_KON\_565 „Operation GetCards“ beschrieben, anbieten und MUSS dabei Kartentypen  
 3619 aus Tabelle TAB\_KON\_500 Wertetabelle Kartentypen unterscheiden.

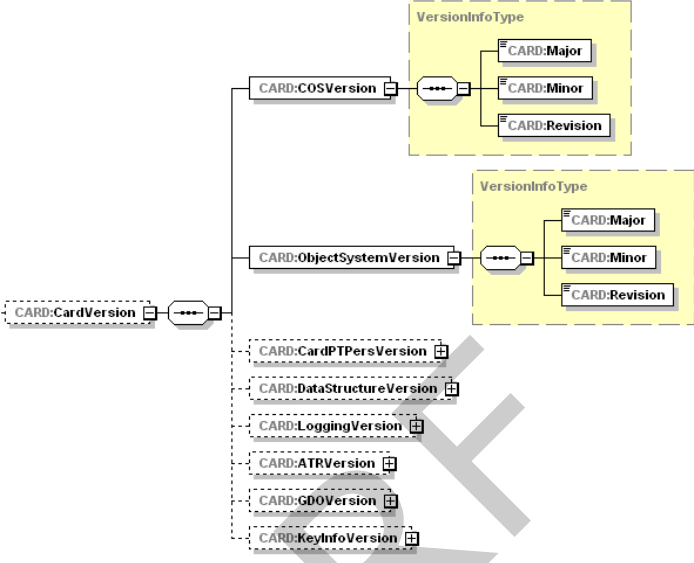
3620 **Tabelle 144: TAB\_KON\_565 Operation GetCards**

|                        |   |
|------------------------|---|
| <b>Name</b>            | GetCards  |
| <b>Beschreibung</b>    | Liefert Informationen zu den in den Kartenterminals verfügbaren Karten zurück, die in Kartenterminals stecken, auf die Mandant und Clientsystem zugreifen dürfen. Insbesondere umfasst die Information die sog. Karten-Handles. Die Karten-Handles können bei anderen Konnektoraufrufen zur Adressierung von Karten genutzt werden. |
| <b>Aufrufparameter</b> |   |

|                |   |  |
|----------------|---|--|
|                |   |  |
|                | <b>Name</b>   | <b>Beschreibung</b>  |
|                | @mandant-wide   | Wenn „true“, werden alle Karten zurückgegeben, auf die der Mandant und das aufrufende Clientsystem zugreifen dürfen. Wenn „false“ (Standardbelegung), werden nur Karten zurückgegeben, auf die von dem im Aufrufkontext spezifizierten Arbeitsplatz zugegriffen werden darf. |
|                | Context   | Aufrufkontext  |
|                | CtId  | Identifikation des Kartenterminals. Wenn angegeben, werden nur die Karten zurückgeliefert, die in diesem Kartenterminal verfügbar sind.  |
|                | SlotId  | Nummer des Slots, beginnend bei 1. Wird zusätzlich zur CtId auch SlotId übergeben, so wird die Karte zurückgegeben, die in dem angegebenen Slot des mit CtId adressierten Kartenterminals steckt.  |
|                | CardType  | Ein Kartentyp gemäß Tabelle TAB_KON_500 „Wertetabelle Kartentypen“ als optionaler Filter. Wenn angegeben, werden nur Karten vom spezifizierten Typ zurückgegeben. Unterstützt werden die Kartentypen EGK, KVK, HBAX, SM-B, SMC-KT und UNKNOWN.                               |
| <b>Antwort</b> |   |  |
|                | <b>Name</b>   | <b>Beschreibung</b>  |
|                | Status  | Ergebnis der Operation   |
|                | Im Element <code>Cards</code> wird die Liste der gesteckten Karten zurückgegeben. Für jede Karte wird dabei ein <code>Card</code> -Element angegeben. Leere Slots der Kartenterminals sind in dieser Liste nicht enthalten. |  |

|             |    |
|-------------|---|
| Name        | Beschreibung  |
| Card Handle | <p>Handle, mit dem die Karte in Folgeaufrufen adressiert werden kann. Der Konnektor garantiert, dass dieses Handle die gesteckte Karte eindeutig identifiziert und bei Entfernen der Karte aus dem Kartenterminal ungültig wird.</p> <p>Auch für nicht erkannte Karten (z. B. bei falscher Steckrichtung der Karte) SOLL der Konnektor gültige Handles liefern (sofern das Kartenterminal in diesem Fall in der Lage ist, das entsprechende Ereignis „Karte wurde gesteckt“ zu liefern), damit diese Karten z. B. zum Auswurf adressiert werden können.</p> |
| CardType    | <p>Erkannter Typ der Karte.</p> <p>Siehe Tabelle TAB_KON_500 Wertetabelle Kartentypen,</p>  |



|  |                 |  |
|--|-----------------|--|
|  | Card Version    |  <p>Der Konnektor MUSS in CardVersion bei eGK, HBA und SM-B/SMC-KT der Generation 2 die Versionsinformationen gemäß [gemSpec_Karten_Fach_TIP] übergeben, für G1+ aus /MF/EF.Version.<br/>Bei KVK, HBA-VK und unbekannten Karten MUSS das Element weggelassen werden.</p> |
|  | Iccsn           | Falls auslesbar, die ICC-Serial-Number der Karte. Im Fall der KVK wird das optionale Element Iccsn nicht zurückgegeben.  |
|  | CtId            | Identifikation des Kartenterminals, in dem die Karte steckt.   |
|  | SlotId          | Nummer des Slots (beginnend bei 1), in dem die Karte steckt.   |
|  | InsertTime      | Gibt den Zeitpunkt an, zu dem der Konnektor die Karte erkannt hat.<br>Die Zeit wird mit dem Datentyp <code>DateTime</code> in folgendem Format angegeben:<br><code>yyyy-mm-ddThh:mm:ss+hh:mm</code><br>Es ist also – gemäß ISO 8601 [ISO8601] – die lokale Zeit und die Differenz zur UTC anzugeben.   |
|  | CardHolder Name | Name des Karteninhabers bzw. der Institution/Organisation (subject.commonName).<br>Bei KVK und unbekannten Karten MUSS das Element weggelassen werden.   |

|                        |  |   |
|------------------------|--|---|
|                        | Kvnr   | KVNR (Unveränderbarer Teil) MUSS bei eGK belegt werden. Bei allen anderen Karten MUSS das Element weggelassen werden. |
|                        | Certificate Expiration Date  | Ablaufdatum des Zertifikates (AUT bzw. OSIG). Bei KVK und unbekannten Karten MUSS das Element weggelassen werden.     |
| <b>Vorbedingungen</b>  | Keine.   |   |
| <b>Nachbedingungen</b> | Der Zustand der Karten und der Kartenterminals bleibt unverändert.   |   |
| <b>Hinweise</b>        | Der Aufruf darf nur den im Konnektor verwalteten aktuellen Zustand der Karte liefern und keine Abfragen an die Kartenterminals absetzen. |   |

3621 Der Ablauf der Operation GetCards ist in Tabelle TAB\_KON\_566 Ablauf GetCards  
 3622 beschrieben:  
 3623

3624 **Tabelle 145: TAB\_KON\_566 Ablauf GetCards**

| Nr. | Aufruf Technischer Use Case oder Interne Operation | Beschreibung  |
|-----|--|---|
| 1.  | checkArguments                                     | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.  |
| 2.  | TUC_KON_000 „Prüfe Zugriffsberechtigung“           | Die Prüfung erfolgt durch den Aufruf<br>TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientSystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>needCardSession = false;<br>allWorkplaces = @mandant-wide}<br>Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.  |
| 3.  | TUC_KON_253 „Liefere Karten_Liste“                 | Die Liste der Karten wird erstellt und zurückgegeben. Tritt hierbei ein Fehler auf, so bricht die Operation mit dem Fehler des TUCs ab.<br>Wenn @mandant-wide=true dann ermittle die Liste der Karten für alle Arbeitsplätze des Mandanten für das angegebene Clientsystem durch den Aufruf<br>TUC_KON_253 „Liefere Karten_Liste“ {<br>clientSystemId = \$context.clientsystemId;<br>cardTerminalId = CtId;<br>slotId = SlotId; |

|  |  |   |
|--|--|---|
|  |  | <pre> mandantId = \$context.mandantId; cardType = CardType } Wenn @mandant-wide=false dann ermittle die Liste der Karten für den Arbeitsplatz des Mandanten für das angegebene Clientsystem entsprechend \$context durch den Aufruf TUC_KON_253 „Liefere Karten_Liste“ {   workplaceId= \$context.workplaceId;   clientSystemId = \$context.clientsystemId;   cardTerminalId = CtId;   slotId = SlotId;   mandantId = \$context.mandantId;   cardType = CardType } </pre> |
|--|--|---|

3625 Die Fehlerfälle der Operation GetCards sind in Tabelle TAB\_KON\_567 Fehlercodes  
 3626 „GetCards dargestellt:  
 3627

3628 **Tabelle 146: TAB\_KON\_567 Fehlercodes „GetCards“**

| Fehlercode   | ErrorType | Severity | Fehlertext   |
|--|-----------|----------|--------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weiteren Fehlercodes auftreten: |           |          |              |
| 4000   | Technical | Error    | Syntaxfehler |

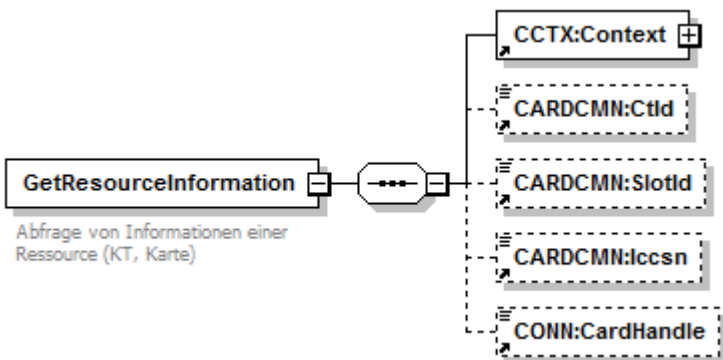
3629 [ $\leq$ ]

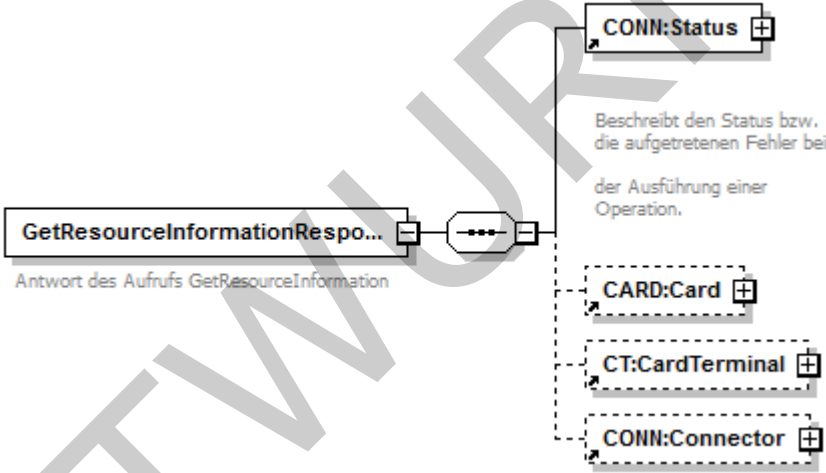
#### 3630 4.1.6.5.3 GetResourceInformation

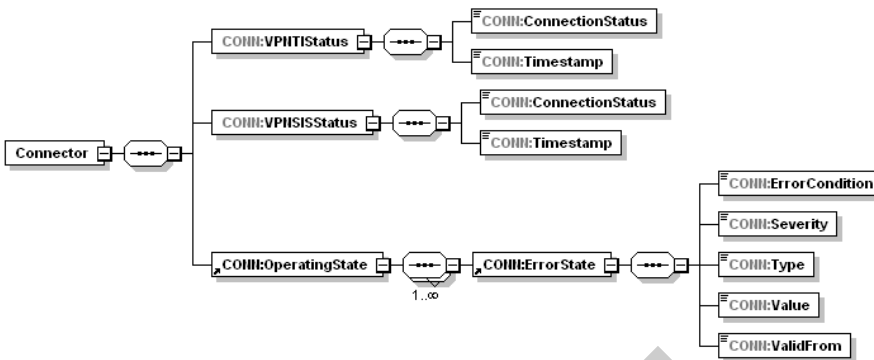
#### 3631 **TIP1-A\_4607 - Operation GetResourceInformation**

3632 Der Konnektors MUSS an der Außenschnittstelle eine Operation GetResourceInformation,  
 3633 wie in Tabelle TAB\_KON\_568 „Operation GetResourceInformation“ beschrieben, anbieten.  
 3634

3635 **Tabelle 147: TAB\_KON\_568 Operation GetResourceInformation**

|                        |  |                     |  |
|------------------------|--|---------------------|--|
| <b>Name</b>            | GetResourceInformation   |                     |  |
| <b>Beschreibung</b>    | Gibt Informationen zu einer Ressource (Karte, KT) oder dem Konnektor selbst zurück   |                     |  |
| <b>Aufrufparameter</b> |  |                     |  |
|                        | <b>Name</b>  | <b>Beschreibung</b> |  |

|                 |  |  |
|-----------------|--|--|
|                 | Context  | Aufrufkontext  |
|                 | CtId   | Identifikator eines Kartenterminals  |
|                 | SlotId   | Kartenslot-Nummer (in Kombination mit CtId)  |
|                 | Iccsn  | Iccsn einer Karte  |
|                 | CardHandle   | CardHandle einer Karte. Unterstützt werden die Kartentypen EGK, KVK, HBAX, SM-B, SMC-KT und UNKNOWN. |
|                 | Wurde keines der Elemente CtId, SlotId, Iccsn übergeben, so wird davon ausgegangen, dass der Aufrufer Informationen zum Konnektor selbst abfragen möchte.  |  |
| <b>Rückgabe</b> |  <p>Antwort des Aufrufs GetResourceInformation</p> <p>Beschreibt den Status bzw. die aufgetretenen Fehler bei der Ausführung einer Operation.</p> |  |
|                 | <b>Name</b>  | <b>Beschreibung</b>  |
|                 | Status   | Ergebnis der Operation   |
|                 | Card   | Informationen einer Karte (siehe GetCards)   |
|                 | CardTerminal   | Informationen eines Kartenterminals (siehe GetCardTerminals)   |
|                 | Connector  | Informationen zum Konnektor  |

|                      |  |  |
|----------------------|--|--|
|                      |  |  |
|                      | VPNTISStatus   |  |
|                      | VPNTISStatus/<br>ConnectionStatus  | Status der VPN-Verbindung zur TI<br>(Online oder Offline)  |
|                      | VPNTISStatus/<br>Timestamp   | Zeitstempel der letzten Statusänderung   |
|                      | VPNSISStatus   |  |
|                      | VPNSISStatus/<br>ConnectionStatus  | Status der VPN-Verbindung des SIS<br>(Online oder Offline)   |
|                      | VPNSISStatus/<br>Timestamp   | Zeitstempel der letzten Statusänderung   |
|                      | OperatingState   | Betriebszustand (siehe Kapitel 3.3)  |
|                      | OperatingState/<br>ErrorState  | Eine Zeile der Fehlerzustandsliste<br>gemäß Tabelle TAB_KON_503<br>Betriebszustand_Fehlerzustandsliste |
|                      | OperatingState/<br>ErrorState/<br>ErrorCondition                                   | ErrorCondition gemäß Tabelle<br>TAB_KON_503<br>Betriebszustand_Fehlerzustandsliste                     |
|                      | OperatingState/<br>ErrorState/Severity   | Schwere des Fehlerzustandes gemäß<br>Tabelle TAB_KON_503<br>Betriebszustand_Fehlerzustandsliste        |
|                      | OperatingState/<br>ErrorState/Type   | Fehlertyp gemäß Tabelle TAB_KON_503<br>Betriebszustand_Fehlerzustandsliste                             |
|                      | OperatingState/<br>ErrorState/Value  | Fehlerzustandswert   |
|                      | OperatingState/<br>ErrorState/ValidFrom  | Zeitstempel der letzten Änderung des<br>Fehlerzustands   |
| <b>Vorbedingung</b>  |  |  |
| <b>Nachbedingung</b> | Der Zustand der Ressource bleibt unverändert.                                      |  |

## Hinweise

3636 Der Ablauf der Operation GetResourceInformation ist in Tabelle TAB\_KON\_569 Ablauf  
 3637 GetResourceInformation beschrieben:  
 3638

3639 **Tabelle 148: TAB\_KON\_569 Ablauf GetResourceInformation**

| Nr. | Aufruf<br>Technischer Use<br>Case oder<br>Interne<br>Operation | Beschreibung  |
|-----|--|---|
| 1.  | checkArguments   | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Insbesondere wird geprüft, dass eine <code>SlotId</code> nur in Verbindung mit einer <code>CtId</code> übergeben werden kann (Abfrage einer Karte). Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.  |
| 2.  | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“               | <p>Die Prüfung erfolgt,</p> <p>falls die Ressource der Konnektor ist, durch den Aufruf</p> <pre>TUC_KON_000 {   mandantId = \$context.mandantId;   clientSystemId = \$context.clientsystemId;   workplaceId = \$context.workplaceId;   ctId = null;   cardHandle = null;   needCardSession = false;   allWorkplaces = true }</pre> <p>falls die Ressource ein Kartenterminal ist, durch den Aufruf</p> <pre>TUC_KON_000 {   mandantId = \$context.mandantId;   clientSystemId = \$context.clientsystemId;   workplaceId = \$context.workplaceId;   ctId = \$ctId;   cardHandle = null;   needCardSession = false;   allWorkplaces = true }</pre> <p>falls die Ressource eine Karte ist, durch den Aufruf</p> <pre>TUC_KON_000 {   mandantId = \$context.mandantId;   clientSystemId = \$context.clientsystemId;   workplaceId = \$context.workplaceId;   ctId = null;   cardHandle = \$cardHandle;   needCardSession = false;   allWorkplaces = false }</pre> |

|    |  |   |
|----|--|---|
|    |  | Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.   |
| 3. | TUC_KON_254<br>„Liefere Ressourcendetails“ | Die Informationen zu der Ressource werden zusammengetragen und zurückgegeben. Tritt hierbei ein Fehler auf, so bricht die Operation mit dem Fehler des TUCs ab. |

3640 Die Fehlerfälle der Operation GetResourceInformation sind in Tabelle TAB\_KON\_570  
 3641 Fehlercodes „GetResourceInformation dargestellt:  
 3642

3643 **Tabelle 149: TAB\_KON\_570 Fehlercodes „GetResourceInformation“**

| Fehlercode   | ErrorType | Severity | Fehlertext   |
|--|-----------|----------|--------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weiteren Fehlercodes auftreten: |           |          |              |
| 4000   | Technical | Error    | Syntaxfehler |

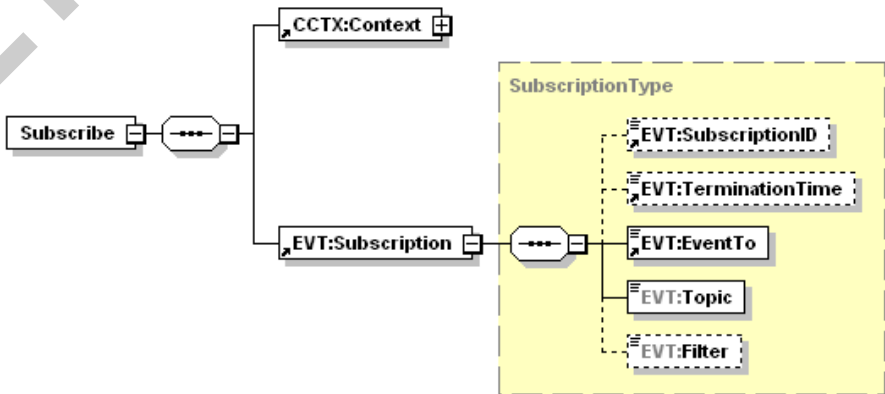
3644  
 3645  
 3646 [ <= ]

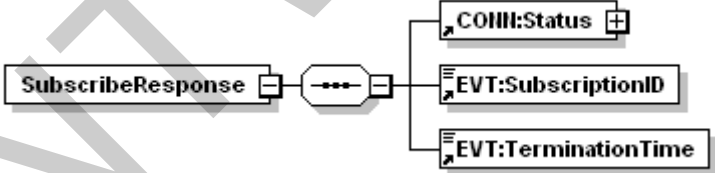
#### 3647 4.1.6.5.4 Subscribe

#### 3648 **TIP1-A\_4608 - Operation Subscribe**

3649 Der Konnektors MUSS an der Außenschnittstelle eine Operation Subscribe, wie in Tabelle  
 3650 TAB\_KON\_571 Operation Subscribe beschrieben, anbieten.  
 3651

3652 **Tabelle 150: TAB\_KON\_571 Operation Subscribe**

|                        |  |                     |
|------------------------|--|---------------------|
| <b>Name</b>            | Subscribe  |                     |
| <b>Beschreibung</b>    | Clientsysteme melden mit dieser Operation ihr Interesse an bestimmten Topics (Ereignissen) an. |                     |
| <b>Aufrufparameter</b> |            |                     |
|                        | <b>Name</b>  | <b>Beschreibung</b> |
|                        | Context  | Aufrufkontext       |

|                      |   |   |
|----------------------|---|---|
|                      | SubscriptionID  | Darf nicht verwendet werden   |
|                      | TerminationTime   | Darf nicht verwendet werden   |
|                      | EventTo   | URL des Endpunkts, wo die Ereignisse zugestellt werden sollen. Syntax:<br><i>ce</i> tp:// <i>host</i> : <i>port</i><br><i>host</i> : IP-Adresse oder FQDN des Clientsystems.<br><i>port</i> : Portnummer des Kommunikationsendpunkts, an dem das Clientsystem auf die Zustellung der Ereignisse wartet.   |
|                      | Topic   | Ein Topic, für das das Clientsystem sein Interesse anmeldet.  |
|                      | Filter  | Filter für die Ereignisnachricht (X-Path Ausdruck im Kontext mit Default Namespace gleich<br>"http://ws.gematik.de/conn/EventService/v7.2" ohne Verwendung eines Namespace-Präfixes sowie Namensraum mit Präfix EVT gleich<br>"http://ws.gematik.de/conn/EventService/v7.2", der beim Versand von Ereignissen in TUC_KON_256 ausgewertet wird. Ermöglicht die Filterung auf Basis der Elemente einer XML-Ereignisnachricht) |
| <b>Rückgabe</b>      |   |   |
|                      | <b>Name</b>   | <b>Beschreibung</b>   |
|                      | Status  | Ergebnis der Operation  |
|                      | SubscriptionID  | Ein Identifikator, der die Anmeldung für die Topics eindeutig identifiziert. Bei den Operationen Unsubscribe, GetSubscription und RenewSubscriptions MUSS dieser SubscriptionID angegeben werden.   |
|                      | TerminationTime   | Maximaler Gültigkeitszeitpunkt der Subscription. Sie MUSS auf Systemzeit + 25 h gesetzt werden.   |
| <b>Vorbedingung</b>  | Das Clientsystem muss die Ereignissenke realisieren.  |   |
| <b>Nachbedingung</b> | Nach erfolgreicher Anmeldung vermerkt der Konnektor das angemeldete Topic unter dem SubscriptionID. Der Konnektor muss die Anmeldungen so lange als gültig behandeln, bis entweder das Clientsystem diese explizit abmeldet |   |



|                 |   |
|-----------------|---|
|                 | <p>oder der Konnektor das Clientsystem als nicht mehr erreichbar erkennt (siehe nächsten Punkt) oder der Konnektor neu gestartet oder ausgeschaltet wird oder die TerminationTime kleiner als die Systemzeit ist.</p> <p>Der Konnektor muss die Anmeldung aus seiner Verwaltung entfernen („Auto-Unsubscribe“), wenn EVT_MAX_TRY Verbindungsaufbauversuche oder zählbare Zustellungsversuche (z.B. durch Zählung beim Absenden der Zustellversuche) in Folge fehlgeschlagen sind. Wenn die Ereignissenke Zustellungen grundsätzlich nicht beantwortet, so sind nur die Verbindungsaufbauversuche zu zählen.</p> |
| <b>Hinweise</b> |   |

3653 Der Ablauf der Operation Subscribe ist in Tabelle TAB\_KON\_572 Ablauf Subscribe  
 3654 beschrieben:  
 3655

3656 **Tabelle 151: TAB\_KON\_572 Ablauf Subscribe**

| Nr. | Aufruf<br>Technischer Use<br>Case oder<br>Interne<br>Operation | Beschreibung   |
|-----|--|--|
| 1.  | checkArguments   | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.   |
| 2.  | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“               | <p>Die Prüfung erfolgt durch den Aufruf TUC_KON_000_000 {</p> <pre> mandantId = \$context.mandantId; clientsystemId = \$context.clientsystemId; workplaceId = \$context.workplaceId; needCardSession = false; allWorkplaces = true </pre> <p>Tritt bei der Prüfung ein Fehler auf, so bricht die Operation mit dem Fehlercode aus TUC_KON_000 ab.</p>  |
| 3.  | saveSubscription   | <p>Die Anmeldung wird im Konnektor hinterlegt. Vorgehalten werden folgende Daten:</p> <ul style="list-style-type: none"> <li>• SubscriptionId (wird generiert)</li> <li>• TerminationTime (Systemzeit + 25h)</li> <li>• MandantId</li> <li>• ClientsystemId</li> <li>• WorkplaceId</li> <li>• Ereignissenke (Feld EventTo)</li> <li>• Abonnierter Topic (Feld Topic)</li> <li>• Filterausdruck (Feld Filter)</li> </ul> <p>Bei der Übernahme wird eine eindeutige SubscriptionId generiert, die dem aufrufenden System</p> |

|  |  |   |
|--|--|---|
|  |  | zurückgegeben wird, falls die Subscription noch nicht existiert. Existiert sie bereits, wird die bestehende SubscriptionID zurückgegeben. |
|--|--|---|

3657 Die Fehlerfälle der Operation Subscribe sind in Tabelle TAB\_KON\_573 Fehlercodes  
 3658 „Subscribe dargestellt:  
 3659

3660 **Tabelle 152 TAB\_KON\_573 Fehlercodes „Subscribe“**

| Fehlercode   | ErrorType | Severity | Fehlertext   |
|--|-----------|----------|--------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weiteren Fehlercodes auftreten: |           |          |              |
| 4000   | Technical | Error    | Syntaxfehler |

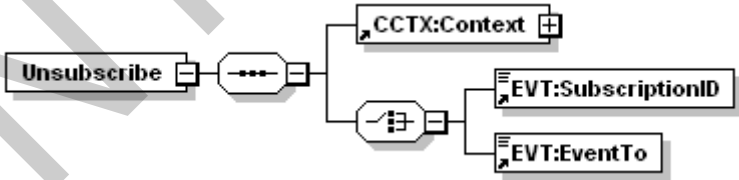
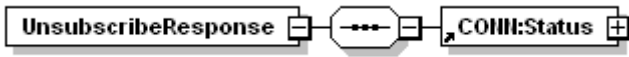
3661  
 3662  
 3663 [ $\leq$ ]

#### 3664 4.1.6.5.5 Unsubscribe

#### 3665 **TIP1-A\_4609 - Operation Unsubscribe**

3666 Der Konnektors MUSS an der Außenschnittstelle eine Operation Unsubscribe, wie in  
 3667 Tabelle TAB\_KON\_574 Operation Unsubscribe beschrieben, anbieten.  
 3668

3669 **Tabelle 153: TAB\_KON\_574 Operation Unsubscribe**

|                        |   |   |
|------------------------|---|---|
| <b>Name</b>            | Unsubscribe   |   |
| <b>Beschreibung</b>    | Löscht eine Anmeldung mit dem angegebenen SubscriptionID oder alle Anmeldungen zu einem Endpunkt EventTo. |   |
| <b>Aufrufparameter</b> |                       |   |
|                        | <b>Name</b>   | <b>Beschreibung</b>   |
|                        | Context   | Aufrufkontext   |
|                        | SubscriptionID  | Der Identifikator, der bei der Subscribe-Operation geliefert wurde.                   |
|                        | EventTo   | URL des clientseitigen Endpunkts, wie er bei der Subscribe-Operation angegeben wurde. |
| <b>Rückgabe</b>        |                       |   |
|                        | <b>Name</b>   | <b>Beschreibung</b>   |
|                        | Status  | Ergebnis der Operation  |

|                      |  |
|----------------------|--|
| <b>Vorbedingung</b>  | Die Anmeldung <code>Subscribe</code> muss vor dieser Operation aufgerufen worden sein.   |
| <b>Nachbedingung</b> | Der Konnektor entfernt aus seiner Verwaltung die Subscription zur <code>Subscription-ID</code> bzw. alle Subscriptions zur clientseitigen URL des Endpunkts <code>EventTo</code> . |
| <b>Hinweise</b>      | Keine  |

Der Ablauf der Operation `Unsubscribe` ist in Tabelle TAB\_KON\_575 Ablauf `Unsubscribe` beschrieben:

**Tabelle 154: TAB\_KON\_575 Ablauf `Unsubscribe`**

| Nr. | Aufruf<br>Technischer Use<br>Case oder<br>Interne<br>Operation | Beschreibung   |
|-----|--|--|
| 1.  | <code>checkArguments</code>                                    | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.   |
| 2.  | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“               | Die Prüfung erfolgt durch den Aufruf<br><pre>TUC_KON_000 {   mandantId = \$context.mandantId;   clientsystemId = \$context.clientsystemId;   workplaceId = \$context.workplaceId;   needCardSession = false;   allWorkplaces = true } </pre> Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab. |
| 3.  | <code>removeSubscription</code>                                | Entfernen der Subscriptions aus der Liste aller Subscriptions.   |

Die Fehlerfälle der Operation `Unsubscribe` sind in Tabelle TAB\_KON\_576 Fehlercodes „`Unsubscribe`“ dargestellt:

**Tabelle 155: TAB\_KON\_576 Fehlercodes „`Unsubscribe`“**

| Fehlercode   | ErrorType | Severity | Fehlertext                            |
|--|-----------|----------|---------------------------------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weiteren Fehlercodes auftreten: |           |          |                                       |
| 4000   | Technical | Error    | Syntaxfehler                          |
| 4102   | Technical | Error    | ungültige <code>SubscriptionId</code> |

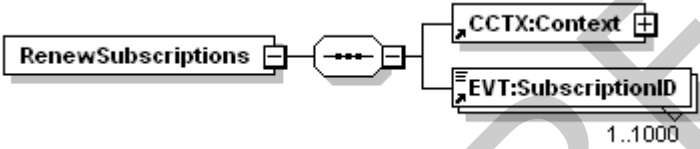
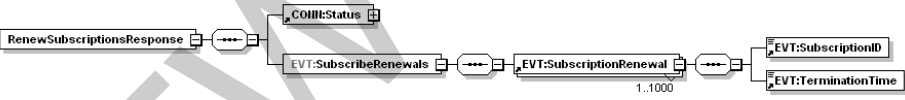
[<=]

## 3681 4.1.6.5.6 RenewSubscriptions

3682 **TIP1-A\_5112 - Operation RenewSubscriptions**

3683 Der Konnektors MUSS an der Außenschnittstelle eine Operation RenewSubscriptions, wie  
 3684 in Tabelle TAB\_KON\_792 Operation RenewSubscriptions beschrieben, anbieten.  
 3685

3686 **Tabelle 156: TAB\_KON\_792 Operation RenewSubscriptions**

|                        |   |  |
|------------------------|---|--|
| <b>Name</b>            | RenewSubscriptions  |  |
| <b>Beschreibung</b>    | Verlängert die Gültigkeit einer Liste von Anmeldungen, die jeweils per SubscriptionID identifiziert werden. |  |
| <b>Aufrufparameter</b> |                           |  |
|                        | <b>Name</b>   | <b>Beschreibung</b>  |
|                        | Context   | Aufrufkontext  |
|                        | Subscription ID   | Der Identifikator, der bei der Subscribe-Operation geliefert wurde.  |
| <b>Rückgabe</b>        |                         |  |
|                        | <b>Name</b>   | <b>Beschreibung</b>  |
|                        | Status  | Ergebnis der Operation   |
|                        | Subscription ID   | Ein Identifikator, der die Anmeldung für die Topics eindeutig identifiziert. Bei den Operationen Unsubscribe, GetSubscription und RenewSubscriptions MUSS diese SubscriptionID angegeben werden. |
|                        | Termination Time  | Maximaler Gültigkeitszeitpunkt der Subscription. Sie MUSS auf Systemzeit + 25 h gesetzt werden.  |
| <b>Vorbedingung</b>    |   |  |
| <b>Nachbedingung</b>   | Der Konnektor speichert jede neu vergebene TerminationTime in seiner Verwaltung der Subscriptions.          |  |
| <b>Hinweise</b>        | Keine   |  |

3687 Der Ablauf der Operation RenewSubscriptions ist in Tabelle TAB\_KON\_793 Ablauf  
 3688 RenewSubscriptions beschrieben:  
 3689

3690 **Tabelle 157: TAB\_KON\_793 Ablauf RenewSubscriptions**

| Nr. | Aufruf<br>Technischer Use<br>Case oder<br>Interne<br>Operation | Beschreibung  |
|-----|--|---|
| 1.  | checkArguments   | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.  |
| 2.  | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“               | Die Prüfung erfolgt durch den Aufruf<br>TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientsystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>needCardSession = false;<br>allWorkplaces = true }<br>Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.  |
| 3.  | renewSubscriptions   | Es wird eine neue <code>SubscribeRenewals</code> -Liste angelegt. Alle Subscriptions, deren <code>TerminationTime</code> kleiner als die Systemzeit sind, muss der Konnektor aus der Verwaltung entfernen.<br>Für jede <code>SubscriptionID</code> , die in der Verwaltung der Subscriptions existiert und deren <code>TerminationTime</code> größer als die Systemzeit ist, wird eine neue <code>TerminationTime = Systemzeit + 25h</code> bestimmt. Diese wird zusammen mit der <code>SubscriptionID</code> als <code>SubscribeRenewal</code> der <code>SubscribeRenewals</code> -Liste hinzugefügt.<br>Kommt es zu keiner Subscription-Verlängerung, weil nur ungültige SubscriptionIDs im Aufruf angegeben waren, wird der Fehler 4102 zurückgeliefert. Kommt es zu mindestens einer Subscription-Verlängerung, sind aber auch ungültige SubscriptionIDs im Aufruf, wird eine <code>RenewSubscriptionsResponse</code> zurückgeliefert, mit <code>CONN:Status/CONN:Result = "Warning"</code> , <code>GERROR:Trace</code> mit {Fehlercode: 4102, ErrorType: Technical, Severity: Error, Fehlertext: "Ungültige SubscriptionId"}, und der Information, welche SubscriptionsIDs ungültig waren. |

3691 Die Fehlerfälle der Operation `RenewSubscriptions` sind in Tabelle TAB\_KON\_794  
 3692 Fehlercodes „RenewSubscriptions dargestellt:

3693

3694 **Tabelle 158: TAB\_KON\_794 Fehlercodes „RenewSubscriptions“**

| Fehlercode   | ErrorType | Severity | Fehlertext               |
|--|-----------|----------|--------------------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weiteren Fehlercodes auftreten: |           |          |                          |
| 4000   | Technical | Error    | Syntaxfehler             |
| 4102   | Technical | Error    | Ungültige SubscriptionId |

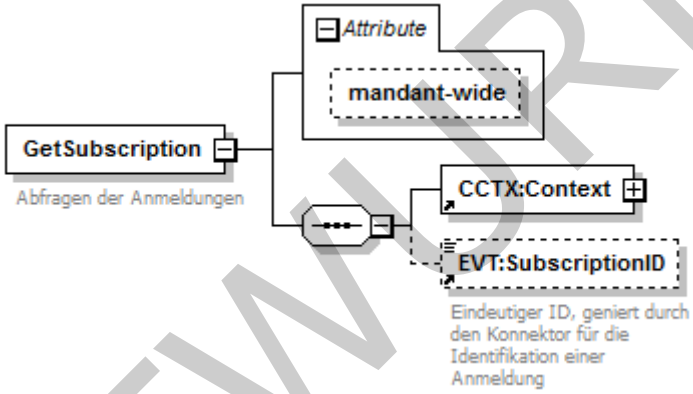
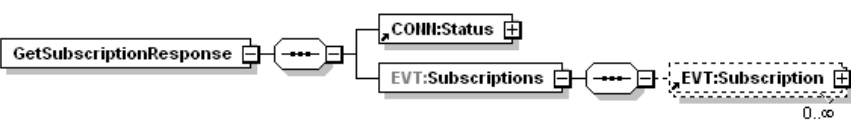
3695  
3696  
3697 [ $\leq$ ]

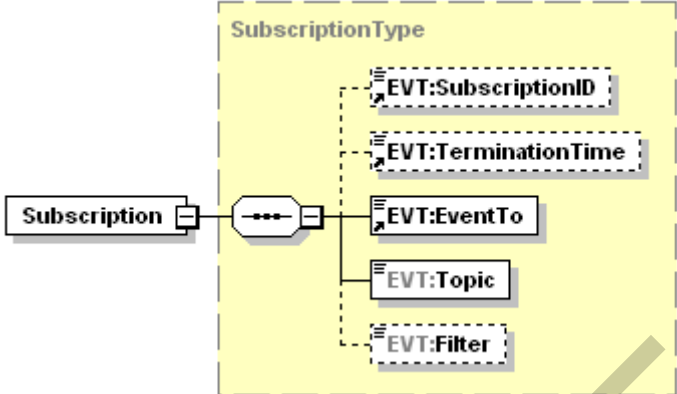
#### 3698 4.1.6.5.7 GetSubscription

#### 3699 TIP1-A\_4610 - Operation GetSubscription

3700 Der Konnektor MUSS an der Außenschnittstelle eine Operation GetSubscription, wie in  
3701 Tabelle TAB\_KON\_577 Operation GetSubscription beschrieben, anbieten.  
3702

3703 **Tabelle 159: TAB\_KON\_577 Operation GetSubscription**

|                        |  |  |
|------------------------|--|--|
| <b>Name</b>            | GetSubscription  |  |
| <b>Beschreibung</b>    | Gibt die Liste der angemeldeten Topics zurück  |  |
| <b>Aufrufparameter</b> |   |  |
|                        | <b>Name</b>  | <b>Beschreibung</b>  |
|                        | @mandant-wide  | Wenn „true“, werden alle Subscriptions zurückgegeben, die Mandant und Clientsystem zugeordnet sind.<br>Wenn „false“ (Standardbelegung) werden alle Subscriptions zurückgegeben, die dem im Aufrufkontext spezifizierten Tripel aus Clientsystem, Mandanten und Arbeitsplatz zugeordnet sind. |
|                        | Context  | Aufrufkontext  |
|                        | SubscriptionID   | Der Identifikator, der bei der Subscribe-Operation geliefert wurde.  |
| <b>Rückgabe</b>        |  |  |
|                        | <b>Name</b>  | <b>Beschreibung</b>  |
|                        | Status   | Ergebnis der Operation   |
|                        | Subscriptions  | Die Liste Subscriptions (vgl. Operation Subscribe)   |

|                      |  |   |
|----------------------|--|---|
|                      |  |   |
|                      | Subscription   | Angefordertes Subscription-Element  |
|                      | Subscription/<br>SubscriptionID  | Identifikator der Subscription  |
|                      | Subscription/<br>TerminationTime   | Maximaler Gültigkeitszeitpunkt der Subscription.                              |
|                      | Subscription/<br>EventTo   | URL des Endpunkts, wo die Ereignisse zugestellt werden sollen (Ereignissenke) |
|                      | Subscription/<br>Topic   | Angemeldeter Topic  |
|                      | Subscription/<br>Filter  | Filterausdruck (falls vorhanden)  |
| <b>Vorbedingung</b>  | Keine  |   |
| <b>Nachbedingung</b> | Die Liste der Subscriptions bleibt unverändert                                     |   |
| <b>Hinweise</b>      | Keine  |   |

3704 Der Ablauf der Operation GetSubscription ist in Tabelle TAB\_KON\_578 Ablauf  
 3705 GetSubscription beschrieben:  
 3706

3707 **Tabelle 160: TAB\_KON\_578 Ablauf GetSubscription**

| Nr. | Aufruf<br>Technischer Use<br>Case oder<br>Interne<br>Operation | Beschreibung  |
|-----|--|---|
| 1.  | checkArguments   | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.  |
| 2.  | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“               | Die Prüfung erfolgt durch den Aufruf<br>TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientsystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>needCardSession = false;<br>allWorkplaces = @mandant-wide }<br>} |

|    |                  |   |
|----|------------------|---|
|    |                  | Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.   |
| 3. | getSubscriptions | Rückgabe der Subscription, die durch <code>SubscriptionId</code> identifiziert wird.<br>Wurde keine <code>SubscriptionId</code> angegeben und <code>@mandant-wide="true"</code> , werden alle Subscriptions zurückgegeben, die dem angegebenen Clientsystem und Mandanten zugeordnet werden können.<br>Wurde keine <code>SubscriptionId</code> angegeben und <code>@mandant-wide="false"</code> , werden alle Subscriptions zurückgegeben, die dem angegebenen Clientsystem, Mandanten und Arbeitsplatz zugeordnet werden können. |

Die Fehlerfälle der Operation GetSubscription sind in Tabelle TAB\_KON\_579 Fehlercodes „GetSubscription dargestellt:

**Tabelle 161: TAB\_KON\_579 Fehlercodes „GetSubscription“**

| Fehlercode   | ErrorType | Severity | Fehlertext               |
|--|-----------|----------|--------------------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weiteren Fehlercodes auftreten: |           |          |                          |
| 4000   | Technical | Error    | Syntaxfehler             |
| 4102   | Technical | Error    | ungültige SubscriptionId |

[<=]

GetLeCards

#### 4.1.6.6 Betriebsaspekte

##### **TIP1-A\_4611 - Konfigurationswerte des Systeminformationsdienstes**

Die Managementschnittstelle MUSS es einem Administrator ermöglichen Konfigurationsänderungen gemäß Tabelle TAB\_KON\_580 vorzunehmen:

**Tabelle 162: TAB\_KON\_580 Konfigurationswerte des Systeminformationsdienstes (Administrator)**

| ReferenzID  | Belegung | Bedeutung und Administrator-Interaktion  |
|-------------|----------|--|
| EVT_MAX_TRY | Nummer   | Der Administrator MUSS über diesen Konfigurationsparameter die Anzahl der Fehlversuche bzgl. Verbindungsversuche bzw. Ereigniszustellungen festlegen können.<br>Ist diese maximal zulässige Anzahl der Fehlversuche überschritten, muss der Konnektor automatisch ein „Auto-Unsubscribe“ (analog Operation „Unsubscribe“ mit „EventTo gleich der URL des clientseitigen Endpunkts“) durchführen. |



3723  
3724

[<=]

3725 **TIP1-A\_4612 - Maximale Anzahl an Subscriptions**

3726 Der Konnektor MUSS eine Mindestmenge von 999 Subscriptions insgesamt unterstützen.  
3727 Der Konnektorhersteller kann jedoch die Anzahl der maximal möglichen Subscriptions  
3728 (insgesamt und/oder pro Ziel) festlegen.  
3729 [<=]

3730 **TIP1-A\_4613 - Initialisierung Subscriptions-Liste beim Bootup**

3731 Der Konnektor MUSS beim Bootup mit einer leeren Liste an Subscriptions starten.  
3732 [<=]

3733 **4.1.7 Verschlüsselungsdienst**

3734 Der Verschlüsselungsdienst bietet Schnittstellen zum hybriden und symmetrischen Ver-  
3735 und Entschlüsseln von Dokumenten an.

3736 Der Verschlüsselungsdienst bietet für alle `Alle_DocFormate` die hybride und  
3737 symmetrische Ver- und Entschlüsselung nach dem Cryptographic Message Syntax (CMS)  
3738 Standard an [RFC5652].

3739 Darüber hinaus werden folgende formaterhaltende Ver-/Entschlüsselungsmechanismen  
3740 unterstützt:

- 3741
  - hybride Ver-/Entschlüsselung von XML-Dokumenten nach der W3C  
3742 Recommendation „XML Encryption Syntax and Processing“ [XMLEnc]
  - hybride Ver-/Entschlüsselung von MIME-Dokumenten nach dem S/MIME-Standard  
3743 [S/MIME]  
3744

3745 Der Konnektor muss bezüglich der zur Ver- und Entschlüsselung von Dokumenten  
3746 verwendeten Verfahren und Algorithmen die Vorgaben in [gemSpec\_Krypt#3.1.4] sowie  
3747 in [gemSpec\_Krypt#3.1.5] und hinsichtlich ECC-Migration die Vorgaben aus  
3748 [gemSpec\_Krypt#5] erfüllen.

3749 **4.1.7.1 Funktionsmerkmalweite Aspekte**

3750 **TIP1-A\_4614 - Missbrauchserkennung Verschlüsselungsdienst**

3751 Der Konnektors MUSS zur Unterstützung von Missbrauchserkennungen die in Tabelle  
3752 TAB\_KON\_581 gelisteten Operationen als Einträge in EVT\_MONITOR\_OPERATIONS  
3753 berücksichtigen.  
3754

3755 **Tabelle 163: TAB\_KON\_581 Verschlüsselungsdienst-Operationen für**  
3756 **EVT\_MONITOR\_OPERATIONS**

| Operationsname  | OK_Val | NOK_Val | Alarmwert (Default-Grenzwert 10 Minuten-Σ) |
|-----------------|--------|---------|--|
| EncryptDocument | 1      | 5       | 101  |
| DecryptDocument | 1      | 5       | 101  |

3757  
3758

[<=]

**TIP1-A\_5434 - Verschlüsselung/Entschlüsselung eines XML Dokuments ergibt unverändertes XML-Dokument**

Der Konnektor MUSS das Operationspaar Verschlüsselung/Entschlüsselung so implementieren, dass Dokumente vom Typ XML unverändert bleiben, wobei zwei XML-Dokumente als identisch zu betrachten sind, wenn sie gemäß Canonical XML 1.1 gleich sind [CanonXML1.1].[<=]

**A\_17746 - Einsatzbereich und Vorgaben für Ver- und Entschlüsselung (ECC-Migration)**

Der Konnektor MUSS für die kartenbasierte Ver- und Entschlüsselung die Zertifikate und Schlüssel in Abhängigkeit vom kryptographischen Verfahren unter Berücksichtigung des Einsatzbereiches aus TAB\_KON\_747 ermitteln.[<=]

**Tabelle 164: TAB\_KON\_747 KeyReference für Encrypt-/DecryptDocument**

| Karte  | KeyReference | Crypt          | Zertifikat (Encrypt)<br>...in DF.ESIGN  | Schlüssel (Decrypt)<br>...in DF.ESIGN | Einsatzbereich      |                         |
|--------|--------------|----------------|---|---------------------------------------|---------------------|-------------------------|
|        |              |                |   |                                       | Außen-schnittstelle | Fachmodul-schnittstelle |
| HBA    | C.ENC        | RSA_ECC        | EF.C.HP.ENC.R2048<br>EF.C.HP.ENC.E256   | PrK.HP.ENC.R2048<br>PrK.HP.ENC.E256   | Ja                  | Ja                      |
|        |              | ECC            | EF.C.HP.ENC.E256                        | PrK.HP.ENC.E256                       | Ja                  | Ja                      |
|        |              | RSA            | EF.C.HP.ENC.R2048                       | PrK.HP.ENC.R2048                      | Ja                  | Ja                      |
| SM-B   | C.ENC        | RSA_ECC        | EF.C.HCI.ENC.R2048<br>EF.C.HCI.ENC.E256 | PrK.HCI.ENC.R2048<br>PrK.HP.ENC.E256  | Ja                  | Ja                      |
|        |              | ECC            | EF.C.HCI.ENC.E256                       | PrK.HP.ENC.E256                       | Ja                  | Ja                      |
|        |              | RSA            | EF.C.HCI.ENC.R2048                      | PrK.HCI.ENC.R2048                     | Ja                  | Ja                      |
| HBA-VK | C.ENC        | RSA_ECC<br>RSA | EF.C.HP.ENC                             | PrK.HP.ENC                            | Ja                  | Ja                      |

|     |       |     |                |                  |      |    |
|-----|-------|-----|----------------|------------------|------|----|
| eGK | C.ENC | ECC | C.CH.ENC.E256  | PrK.CH.ENC.E256  | Nein | Ja |
|     | C.ENC | RSA | C.CH.ENC.R2048 | PrK.CH.ENC.R2048 | Nein | Ja |

3773

3774

3775

**Tabelle 165: TAB\_KON\_859 Werteliste und Defaultwert des Parameters crypt bei hybrider Verschlüsselung**

| Typname   | Werteliste            | Defaultwert | Bedeutung   |
|-----------|-----------------------|-------------|---|
| ENC_CRYPT | RSA<br>ECC<br>RSA_ECC | RSA         | Werteliste des Parameters crypt bei der hybriden Verschlüsselung<br>RSA: Es wird RSA-2048 basiert verschlüsselt.<br>ECC: Es wird ECC-256 basiert verschlüsselt.<br>RSA_ECC: Es wird dual RSA-2048 basiert und ECC-256 basiert verschlüsselt. Es wird als Fehlerfall gewertet, wenn weder RSA- noch ECC-Zertifikat von der Karte geladen werden konnten, und als Warnung, wenn nur ein Zertifikat geladen werden konnte. |

3776

#### 4.1.7.2 Durch Ereignisse ausgelöste Reaktionen

3777

Keine.

3778

#### 4.1.7.3 Interne TUCs, nicht durch Fachmodule nutzbar

3779

Keine.

3780

#### 4.1.7.4 Interne TUCs, auch durch Fachmodule nutzbar

3781

3782

3783

3784

3785

Die in diesem Kapitel beschriebenen TUCs zur hybriden Ver- und Entschlüsselung werden den Fachmodulen und Außenoperationen angeboten. Die TUCs zur symmetrischen Ver-/Entschlüsselung werden den Fachmodulen angeboten. Es gibt keine Aufrufhierarchie innerhalb der hier beschriebenen TUCs zur hybriden und symmetrischen Ver-/Entschlüsselung.

3786

##### 4.1.7.4.1 TUC\_KON\_070 „Daten hybrid verschlüsseln“

3787

#### TIP1-A\_4616-02 - TUC\_KON\_070 „Daten hybrid verschlüsseln“

3788

3789

3790

Der Konnektor MUSS den technischen Use Case TUC\_KON\_070 „Daten hybrid verschlüsseln“ umsetzen.

3791

**Tabelle 166: TAB\_KON\_739 - TUC\_KON\_070 „Daten hybrid verschlüsseln“**

| Element | Beschreibung |
|---------|--------------|
|         |              |

|                |  |
|----------------|--|
| Name           | TUC_KON_070 „Daten hybrid verschlüsseln“   |
| Beschreibung   | <p>Dieser TUC verschlüsselt ein Dokument oder Teile eines Dokumentes. Die Verschlüsselung erfolgt zweistufig, d. h. die Daten werden symmetrisch mit einem generierten Schlüssel verschlüsselt und anschließend wird dieser Schlüssel mit einem asymmetrischen Verfahren verschlüsselt.</p> <p>Die asymmetrische Verschlüsselung des symmetrischen Schlüssels kann für mehrere Identitäten, repräsentiert durch X.509-Zertifikate oder öffentliche Schlüssel, erfolgen. Das Ergebnis sind entsprechend viele Verschlüsselungen desselben symmetrischen Schlüssels.</p> <p>Es werden die folgenden formaterhaltenden Verschlüsselungsverfahren für die genannten Dokumententypen unterstützt:</p> <ul style="list-style-type: none"> <li>• XML mit [XMLEnc]</li> <li>• MIME mit [S/MIME]</li> </ul> <p>Des Weiteren ist für alle unterstützten Dokumentformate (Alle_DocFormate) die Verschlüsselung mit CMS [RFC5652] möglich.</p>     |
| Auslöser       | Aufruf durch einen Fachmodul-TUC oder durch die Operation EncryptDocument des Verschlüsselungsbasisdienstes  |
| Vorbedingungen | Falls mit einem öffentlichen Schlüssel auf einer Karte verschlüsselt werden soll, muss die Karte gesteckt sein.  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• documentToBeEncrypted<br/>(Zu verschlüsselndes Dokument )</li> <li>• encryptionCertificates – <i>optional/entfällt, wenn encryptionKeys übergeben wird</i><br/>(X.509v3-Zertifikate)</li> <li>• encryptionKeys – <i>optional/entfällt, wenn encryptionCertificates übergeben wird</i><br/>(öffentliche Schlüssel; unterstützte Karten sind SM-B, HBAX und eGK)</li> <li>• encryptionType [EncryptionType]<br/>(Angaben zum einzusetzenden Verschlüsselungsverfahren (CMS, XMLEnc oder S/MIME)).</li> <li>• cardSession – <i>optional/verpflichtend, wenn ein Zertifikat von einer Karte gelesen werden soll</i><br/>(Kartensitzung; unterstützte Karten sind SM-B, HBAX und eGK.)</li> <li>• certificateReference – <i>optional/verpflichtend, wenn ein Zertifikat von einer Karte gelesen werden soll</i><br/>(Zertifikatsreferenz; unterstützte Karten sind SM-B, HBAX und eGK).</li> </ul> |

|                |  |
|----------------|--|
|                | <ul style="list-style-type: none"> <li>• <code>crypt</code> [ENC_CRYPT] - <i>optional</i>;<br/>default und Wertebereich siehe TAB_KON_859<br/>(Wenn das Verschlüsselungszertifikat von einer Karte kommt, steuert <code>crypt</code>, mit welchen kryptographischen Verfahren die Verschlüsselung der Hybridschlüssel erfolgt.)</li> <li>• <code>xmlElements</code> – <i>optional/verpflichtend</i>, wenn <code>encryptionType = XMLEnc</code><br/>(Festlegung der zu verschlüsselnden Teile des Dokumentes durch Spezifikation eines Xpath-Ausdruckes (XML-Elements).</li> <li>• <code>keyInfoMode</code> [embedded   separate] – <i>optional/verpflichtend</i>, wenn <code>encryptionType = XMLEnc</code><br/>(Angabe, ob die KeyInfo in das XML-Dokument eingebettet oder separat an den Aufrufer zurückgegeben werden soll)</li> </ul>   |
| Komponenten    | Konnektor, Kartenterminal, Karte   |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>• <code>encryptedDocument</code><br/>(Verschlüsseltes Dokument)</li> <li>• <code>encryptedKeys</code> – <i>optional/verpflichtend</i>, wenn diese nicht im verschlüsselten Dokument enthalten sind<br/>(Verschlüsselte symmetrische Schlüssel)</li> <li>• <code>keyInfo</code> – <i>optional/verpflichtend</i>, wenn <code>encryptionType = XMLEnc</code> und <code>keyInfoMode = separate</code><br/>(KeyInfo, falls nicht ins Dokument eingebettet)</li> </ul>  |
| Standardablauf | <ol style="list-style-type: none"> <li>1. Das Verschlüsselungsverfahren wird anhand des Eingangsparameters <code>EncryptionType</code> gewählt.</li> <li>2. <u>Nur für XMLEnc:</u><br/>Die zu verschlüsselnden XML-Elemente werden lokalisiert. Falls kein zu verschlüsselndes XML-Element gefunden wurde, wird Fehler 4103 gemeldet. Die zu verschlüsselnden XML-Elemente dürfen nicht ineinander verschachtelt sein. Sind die zu verschlüsselnden XML-Elemente ineinander verschachtelt, so wird Fehler 4104 gemeldet.</li> <li>3. Für jedes von der Karte zu lesende Zertifikat, wird TUC_KON_216 „Lese Zertifikat“ aufgerufen. Welches Zertifikat von der Karte gelesen werden soll, wird durch den Parameter <code>crypt</code> über Tabelle TAB_KON_747 gesteuert.<br/>In den Fällen <code>crypt = RSA</code> und <code>crypt = ECC</code> bricht der TUC ab, wenn dabei ein Fehler auftritt.<br/>Im Fall <code>crypt = RSA_ECC</code> bricht der TUC im Fehlerfall dann ab, wenn weder RSA- noch ECC-Zertifikat geladen werden konnte, und läuft mit einer Warnung durch, wenn nur ein Zertifikat geladen werden konnte.</li> </ol> |

|  |   |
|--|---|
|  | <p>4. Falls Zertifikate übergeben oder von der Karte gelesen wurden, werden diese durch Aufruf von TUC_KON_037 „Zertifikat prüfen“ geprüft.<br/>Als Parameter des TUC-Aufrufs gilt für Zertifikate, die mit Zertifikaten aus CERT_IMPORTED_CA_LIST geprüft werden:</p> <pre>TUC_KON_037 „Zertifikat prüfen“ {     certificate = Zertifikat;     qualifiedCheck = not_required;     offlineAllowNoCheck = true;     intendedKeyUsage= intendedKeyUsage(Zertifikate aus CERT_IMPORTED_CA_LIST);     validationMode = NONE }</pre> <p>Für alle anderen Zertifikate gilt: {</p> <pre>    certificate = [C.CH.ENC];     qualifiedCheck=not_required;     offlineAllowNoCheck=false;     policyList =[ oid_egk_enc];     intendedKeyUsage= intendedKeyUsage(C.CH.ENC);     validationMode=OCSP }</pre> <p>oder</p> <pre>{     certificate = [C.CH.ENCV];     qualifiedCheck=not_required;     offlineAllowNoCheck=false;     policyList =[ oid_egk_encv ];     intendedKeyUsage= intendedKeyUsage(C.CH.ENCV);     validationMode=OCSP }</pre> <p>oder</p> <pre>{     certificate = [C.HCI.ENC];     qualifiedCheck=not_required;     offlineAllowNoCheck=false;     policyList =[ oid_smc_b_enc ];     intendedKeyUsage= intendedKeyUsage(C.HCI.ENC);     validationMode=OCSP }</pre> <p>oder</p> <pre>{     certificate = [C.HP.ENC];     qualifiedCheck=not_required;     offlineAllowNoCheck=false;     policyList =[ oid_hba_enc, oid_vk_pt_enc, oid_vk_eaa_enc ];     intendedKeyUsage= intendedKeyUsage(C.HP.ENC);     validationMode=OCSP }</pre> <p>5. Die öffentlichen Schlüssel werden aus den Zertifikaten extrahiert, falls sie nicht direkt übergeben wurden.<br/>Falls ein Schlüssel keinen der zugelassenen Verschlüsselungsalgorithmen gemäß [gemSpec_Krypt#3.5.2] bzw. [gemSpec_Krypt#5.8] erlaubt, wird Fehler 4200 gemeldet.</p> |
|--|---|

|  |  |
|--|--|
|  | <p>6. Der Konnektor generiert einen symmetrischen Schlüssel. Dabei muss der symmetrische Schlüssel den Kriterien aus [gemSpec_Krypt#2.4] entsprechen.</p> <p>7. Der Konnektor verschlüsselt das Dokument oder Teile des Dokuments mit dem generierten symmetrischen Schlüssel.</p> <p>a. <u>CMS:</u><br/>Es MÜSSEN die Vorgaben aus [gemSpec_Krypt#3.5.1] beachtet werden.</p> <p>b. <u>XMLEnc:</u><br/>Alle zu verschlüsselnden XML-Elemente werden mit demselben symmetrischen Schlüssel verschlüsselt. Dabei MÜSSEN die Vorgaben aus [gemSpec_Krypt#3.1.4] beachtet werden.</p> <p>8. Der symmetrische Schlüssel wird asymmetrisch für die einzelnen Identitäten verschlüsselt. Dabei müssen die Vorgaben aus [gemSpec_Krypt#3.1.5; 3.5.2; 5.8] beachtet werden.</p> <p>9. Das Zieldokument wird erstellt.<br/><u>XMLEnc</u><br/>Format und Inhalt des verschlüsselten Dokuments SOLLEN dem XML Encryption Format in [COMMON_PKI#Part 8] folgen. Zum Format des verschlüsselten XML-Dokumentes siehe auch Tabelle TAB_KON_073 Vorgaben zum Format verschlüsselter XML-Dokumente.<br/>Die verschlüsselten Datenelemente (EncryptedData) werden erstellt.<br/>EncryptedData ersetzt jeweils das zu verschlüsselnde Element des XML-Dokuments. In [COMMON_PKI] wird die Verwendung des Attributs Type in EncryptedData ausgeschlossen; diese Spezifikation sieht jedoch dessen Verwendung für verschlüsselte XML-Bestandteile (element, content) wie in [XMLEnc] beschrieben vor. Der Namespace von EncryptedData ist als <a href="http://www.w3.org/2001/04/xmlenc#">http://www.w3.org/2001/04/xmlenc#</a> anzugeben.</p> <p>Für das Element EncryptedData wird das Sub-Element EncryptionMethod mit Angaben zum Verschlüsselungsalgorithmus als obligatorisch vorgegeben, ebenso die Elemente KeyInfo und CipherData.</p> <p>Das Element EncryptedData/KeyInfo hat den Namespace "<a href="http://www.w3.org/2000/09/xmldsig#">http://www.w3.org/2000/09/xmldsig#</a>". Es muss pro Hybridschlüssel ein Element EncryptedKey enthalten. In jedem EncryptedKey-Element wird neben dem eigentlichen Hybridschlüssel ein Element zur EncryptionMethod der asymmetrischen Verschlüsselung und ein KeyInfo-Element mit dem Zertifikat angelegt, das für die Verschlüsselung des symmetrischen Schlüssels verwendete wurde. Das Zertifikat wird jeweils im Element EncryptedKey/KeyInfo/X509Data/X509Certificate</p> |
|--|--|



|                            |   |
|----------------------------|---|
|                            | <p>base64-kodiert und darin DER-kodiert abgelegt.</p> <p>Hybridschlüssel (RSA):<br/>Das Element EncryptedData/KeyInfo/EncryptedKey muss die Verschlüsselungsmethode im Element EncryptionMethod angeben, den hybridSchlüssel im Element CipherData speichern und das Zertifikat, mit dem der symmetrische Schlüssel zum Hybridschlüssel verschlüsselt wurde, im Element EncryptedKey/KeyInfo/X509Data/X509Certificate base64-kodiert und darin DER-kodiert ablegen.</p> <p>Hybridschlüssel (ECC): Es gelten die Vorgaben aus [gemSpec_Krypt#5.8]</p> <p><u>CMS:</u><br/>Es ist CMS mit Authenticated-Enveloped-Data Content Type gemäß [RFC-5083] und der AES-GCM-Encryption gemäß [RFC-5084] zu verwenden. Bei der Verschlüsselung des „content-encryption key“ wird die Technik „key transport“ eingesetzt. Pro Empfänger wird eine Instanz vom Typ KeyTransRecipientInfo erzeugt. Dabei ist für RecipientIdentifier die Option IssuerAndSerialNumber zu wählen.<br/>ContentType = OID {... authEnvelopedData}<br/>                  = 1.2.840.113549.1.9.16.1.23</p> <p>Im Fall ECC sind die Vorgaben aus [gemSpec_Krypt#5.8] zur Erzeugung des Hybridschlüssels zu beachten.<br/>Im Fall RSA sind die Vorgaben aus [gemSpec_Krypt#3.5.2] zur Erzeugung des Hybridschlüssels zu beachten.</p> <p>10. Der symmetrische Schlüssel wird aktiv gelöscht (überschrieben).</p> |
| Varianten/<br>Alternativen | <p><u>Zur Rückgabe der Hybridschlüssel</u> MUSS auch die Variante vorgesehen werden, dass die Hybridschlüssel („KeyInfo“) nicht eingebettet im Zieldokument zurückgegeben werden, sondern separat.</p> <p>Im Fall des Verschlüsselungsverfahrens S/MIME wird der Standardablauf des CMS Verschlüsselungsverfahrens durch einen vorgelagerten S/MIME-Vorbereitungsschritt und einen nachgelagerten S/MIME-Nachbereitungsschritt ergänzt. Das S/MIME-Verfahren MUSS konform [S/MIME] und SOLL konform [COMMON_PKI#Part 3] erfolgen.</p> <p>Der S/MIME-Vorbereitungsschritt bereitet das übergebene MIME-Dokument gemäß [S/MIME#3.1] auf die nachfolgende CMS-Verschlüsselung durch eine Kanonisierung für Text [S/MIME#3.1.1] vor. Eine weitere Kanonisierung oder eine Anpassung des Transfer Encodings [S/MIME#3.1.2] erfolgt nicht.</p> <p>Im S/MIME-Nachbereitungsschritt wird das im Standardablauf erzeugt CMS-Objekt in eine MIME-Nachricht vom Typ „application/pkcs7-mime“ eingebettet.</p> <p>Sämtliche Header-Felder der Nachricht MÜSSEN in die Header-Felder der S/MIME-Nachricht übernommen werden. Die im Folgenden explizit zu setzenden Header-Felder überschreiben die betroffenen Header-Felder.</p>   |



|             |   |
|-------------|---|
|             | <p>Es MUSS ein neues message-id Element für den S/MIME-Header generiert werden.</p> <p>"MIME-Version: 1.0" MUSS definiert sein.</p> <p>Das Feld "Subject" MUSS mit "Subject: Verschlüsselte Nachricht" überschrieben werden.</p> <p>Die Codierung des verschlüsselten Inhalts der Nachricht MUSS in "base64" erfolgen. Entsprechend ist das zugehörige Header-Feld zu füllen: "Content-Transfer-Encoding: base64".</p> <p>Das Feld "Content-Type:" ist als "application/pkcs7-mime" zu definieren. Die weiteren Attribute dieses Feldes sind:</p> <ul style="list-style-type: none"> <li>• "smime-type=enveloped-data;"</li> <li>• "name=\$dateiname", wobei \$dateiname auf ".p7m" endet.</li> </ul> <p>Das Feld "Content-Disposition" definiert den Inhalt der Nachricht als Dateianhang: "Content-Disposition: attachment; filename=\$dateiname"</p> <p><u>Zu Schritten 5 und 8 für TI-fremde X.509-Zertifikate</u></p> <p>Der Konnektor MUSS beim asymmetrischen Anteil der hybriden Verschlüsselung auch TI-fremde X.509-Zertifikate unterstützen, wenn diese von einem CA-Zertifikat aus CERT_IMPORTED_CA_LIST ausgestellt wurden und die kryptographischen Vorgaben aus Tabelle [gemSpec_Krypt#Tab_KRYPT_002] erfüllen.</p> <p>Der Konnektor MUSS Anfragen zur Hybridverschlüsselung mit einer Fehlermeldung (Fehler 4200) abweisen, wenn hierfür TI-fremde X509-Zertifikate vorgegeben werden, die nicht die kryptographischen Vorgaben aus Tabelle [gemSpec_Krypt#Tab_KRYPT_002] oder [gemSpec_Krypt#Tab_KRYPT_002a] erfüllen.</p> |
| Fehlerfälle | <p>Siehe Tabelle TAB_KON_740 Fehlercodes TUC_KON_070 „Daten hybrid verschlüsseln“. Wenn im Ablauf des TUCs ein anderer Fehler als die in Tabelle TAB_KON_740 beschriebenen Fehler auftritt, wird Fehler 4105 gemeldet.</p> <p>(-&gt;4) Schritt 4 – Zertifikatsprüfung „für alle anderen Zertifikate“</p> <p>Für MGM_LU_ONLINE=Enabled gilt:</p> <p>Liefert die Zertifikatsprüfung (OCSP-Abfrage) mdt. eine der folgenden Warnungen gemäß [gemSpec_PKI#Tab_PKI_274]</p> <ul style="list-style-type: none"> <li>• CERT_REVOKED</li> <li>• CERT_UNKNOWN</li> </ul> <p>dann wird der TUC mit Fehler 4105 abgebrochen,</p> <p>Ausnahme: Falls im Falle crypt=RSA_ECC der Hybridschlüssel nur für eines der beiden Zertifikate erzeugt werden konnte, dann wird die Warnung 4259 mit &lt;Zertifikat&gt; gemäß TAB_KON_747 in der Response zurückgegeben.</p>  |

|                                |   |
|--------------------------------|---|
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | Abbildung PIC_KON_058 Aktivitätsdiagramm „Daten hybrid verschlüsseln“<br>Das Diagramm dient nur der Veranschaulichung und ist nicht vollständig. Beispielsweise enthält es nicht die Steuerung durch den Parameter crypt. |

ENTWURF

3792

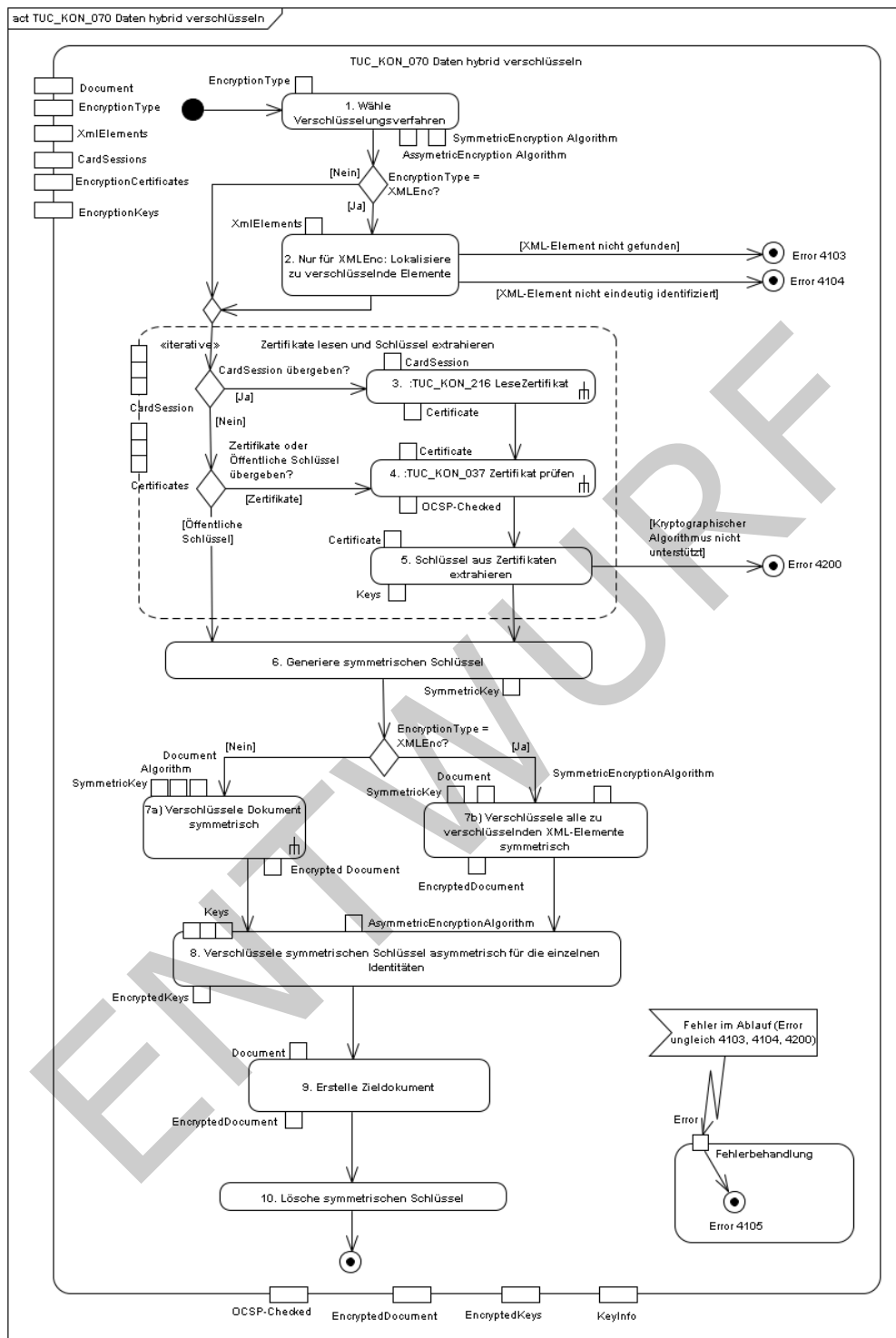


Abbildung 13: PIC\_KON\_058 Aktivitätsdiagramm „Daten hybrid verschlüsseln“

Tabelle 167: TAB\_KON\_073 Vorgaben zum Format verschlüsselter XML-Dokumente

| # | Beschreibung |
|---|--------------|
|---|--------------|

|  |   |
|--|---|
|  | xenc:EncryptedData MUSS ein ds:KeyInfo Element enthalten, welches wiederum ein xenc:EncryptedKey Element enthält.   |
|  | Der xenc:EncryptedKey MUSS [XMLEnc] konform sein.   |
|  | Die xenc:EncryptionMethod für den Schlüssel MUSS gemäß [gemSpec_Krypt#3.1.5] gewählt werden   |
|  | Der xenc:EncryptedKey MUSS ein ds:KeyInfo Element mit ds:X509Data und ds:X509Certificate Subelement enthalten, in dem das X.509-Zertifikat hinterlegt wird. |

3797

3798

**Tabelle 168: TAB\_KON\_740 Fehlercodes TUC\_KON\_070 „Daten hybrid verschlüsseln“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4103  | Technical | Error    | XML-Element nicht gefunden  |
| 4104  | Technical | Error    | XML-Element nicht eindeutig identifiziert. (Überschneidung)       |
| 4105  | Technical | Error    | hybride Verschlüsselung konnte nicht durchgeführt werden          |
| 4200  | Security  | Error    | Schlüssel erlaubt keinen zugelassenen Verschlüsselungsalgorithmus |
| 4259  | Technical | Warning  | Verschlüsselung für Zertifikat <Zertifikat> nicht möglich         |

3799

3800 [**<=**]

3801

3802 **4.1.7.4.2 TUC\_KON\_071 „Daten hybrid entschlüsseln“**3803 **TIP1-A\_4617-02 - TUC\_KON\_071 „Daten hybrid entschlüsseln“**3804 Der Konnektor MUSS den technischen Use Case TUC\_KON\_071 „Daten hybrid  
3805 entschlüsseln“ umsetzen.

3806

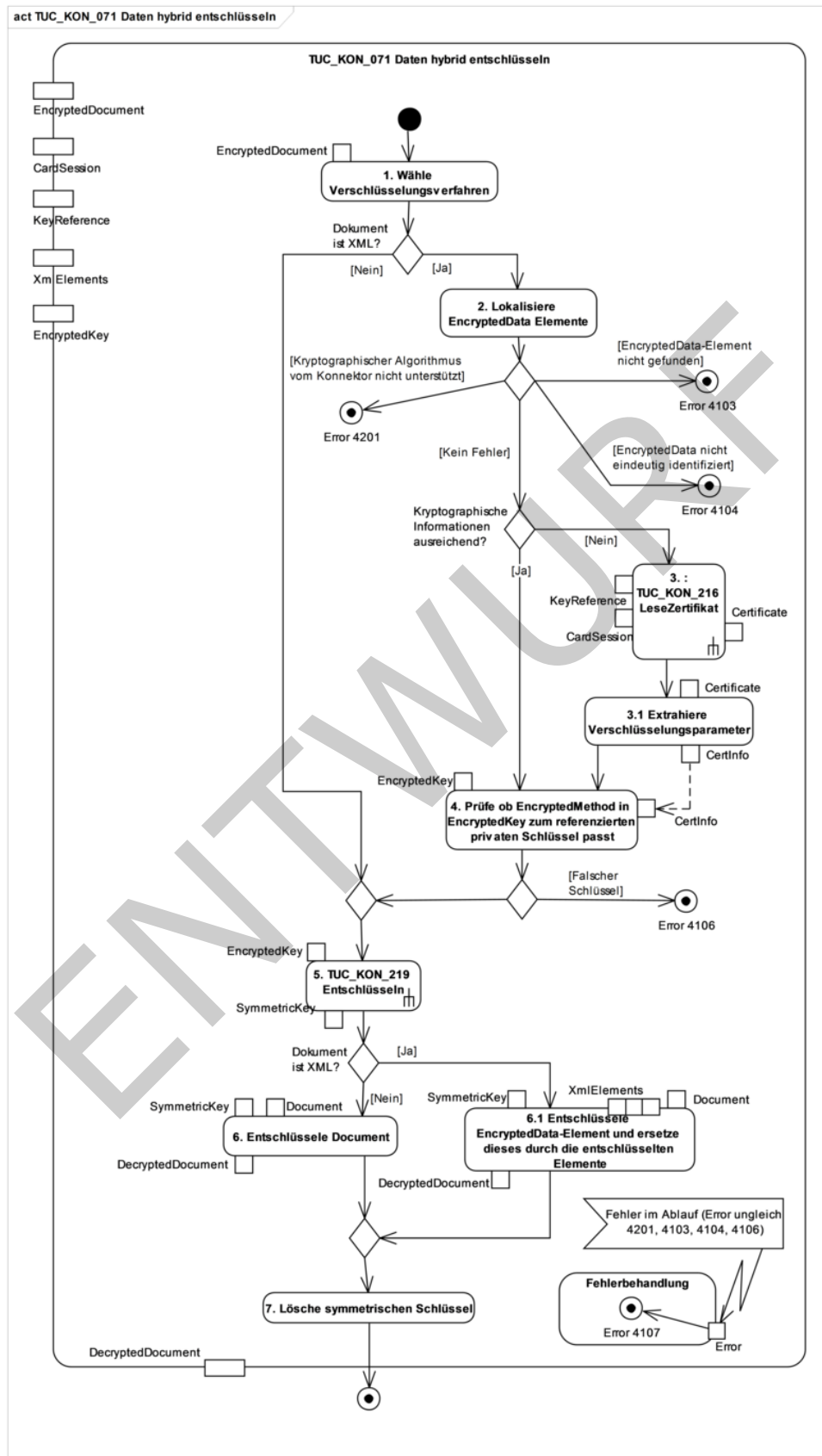
3807 **Tabelle 169: TAB\_KON\_140 – TUC\_KON\_071 „Daten hybrid entschlüsseln“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_071 „Daten hybrid entschlüsseln“   |
| Beschreibung   | Ein hybrid verschlüsseltes Dokument, das konform zu TUC_KON_070 erstellt wurde, wird entschlüsselt.<br>Es muss eine asymmetrische Verschlüsselung vorliegen, zu der der Schlüssel auf einer Karte vorliegt.  |
| Auslöser       | Aufruf in einem fachlichen Use Case oder des Verschlüsselungsbasisdienstes.  |
| Vorbedingungen | Die Karte mit dem privaten Schlüssel muss gesteckt sein und der Sicherheitszustand zur Nutzung des privaten Schlüssels muss gesetzt sein.<br>Ein konform zu TUC_KON_070 hybrid verschlüsseltes Dokument liegt vor.<br>Bei XML-Dokumenten: Das Dokument enthält EncryptedData Elemente. Falls mehrere Elemente des Dokumentes zu entschlüsseln sind, müssen diese alle mit demselben symmetrischen Schlüssel verschlüsselt sein.  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>encryptedDocument (Zu entschlüsselndes Dokument)</li> <li>cardSession (Kartensitzung; unterstützt werden SM-B, HBAX und eGK mit der jeweiligen C.ENC-KeyReference).</li> <li>privateKeyReference (Referenz auf den privaten Schlüssel; unterstützt werden SM-B, HBAX und eGK mit der jeweiligen C.ENC-KeyReference).</li> <li>encryptionCertificate – <i>optional</i> (Verschlüsselungszertifikat passend zur Schlüsselreferenz).</li> <li>encryptionCertificateReference – <i>optional</i> (Referenz auf das Zertifikat auf obiger Karte passend zur Schlüsselreferenz).</li> <li>encryptedKey – <i>optional, falls nicht in encryptedDocument enthalten</i> (asymmetrisch verschlüsselter symmetrischer Schlüssel)<br/>Darüber hinaus werden die folgenden, vom Dokumentformat und dem Verschlüsselungsverfahren abhängigen Eingangsdaten benötigt:<br/>Bei XML-Dokumenten:</li> <li>xmlElements – <i>optional/verpflichtend, wenn encryptionType = XMLEnc</i></li> </ul> |

|                        |   |
|------------------------|---|
|                        | (bei XML-Dokumenten Angabe der zu entschlüsselnden Teile des XML-Dokuments)   |
| Komponenten            | Konnektor, Kartenterminal, Karte  |
| Ausgangsdaten          | <ul style="list-style-type: none"> <li>plainDocument<br/>(Unverschlüsseltes Dokument. Bei XML-Dokumenten: Das EncryptedData-Element ist durch das entschlüsselte ersetzt.)</li> </ul>   |
| Standardablauf         | <ol style="list-style-type: none"> <li>1. Das Verfahren zum Entschlüsseln wird entsprechend dem Format des übergebenen zu entschlüsselnden Dokuments (EncryptedDocument) gewählt.<br/>Der Konnektor MUSS beim asymmetrischen Anteil der Entschlüsselung hybrid verschlüsselter Dokumente die in [gemSpec_Krypt] beschriebenen Verfahren unterstützen.</li> <li>2. XMLEnc:<br/>Das EncryptedData Element (oder mehrere Elemente) werden im Dokument lokalisiert. Falls sie nicht oder nicht eindeutig gefunden werden können wird Fehler 4103 bzw. 4104 gemeldet.<br/>Ist in einem EncryptedData Element ein vom Konnektor nicht unterstützter Mechanismus spezifiziert, wird Fehler 4201 gemeldet.</li> <li>3. Falls erforderlich, wird TUC_KON_216 „Lese Zertifikat“ aufgerufen, um das Zertifikat von der Karte zu lesen.<br/>3.1 Die Kenntnis des Zertifikats kann erforderlich sein, um im Zertifikat kodierte Verschlüsselungsparameter auszulesen. (Zur Zeit der Erstellung dieser Spezifikation werden zur Laufzeit keine zusätzlichen Parameter aus dem Zertifikat benötigt, da alle nötigen Informationen aus den PKI- und Kartenspezifikationen abgeleitet werden können.)</li> <li>4. XMLEnc:<br/>Es wird geprüft, ob die Verschlüsselungsparameter (EncryptionMethod in EncryptedKey) zum referenzierten privaten Schlüssel auf der Karte passen. Ist dies nicht der Fall, bricht der Use Case mit Fehler 4106 ab.</li> <li>5. Es wird TUC_KON_219 „Entschlüssele“ aufgerufen, um den symmetrischen Schlüssel mit Hilfe des angegebenen privaten Schlüssels zu entschlüsseln.</li> <li>6. Mit dem symmetrischen Schlüssel wird der unverschlüsselte Dateninhalt wiederhergestellt.<br/>6.1 XMLEnc: Das EncryptedData Element wird durch die entschlüsselten Daten ersetzt.</li> <li>7. Der symmetrische Schlüssel wird aktiv gelöscht (überschrieben).</li> </ol> |
| Varianten/Alternativen | Zu 6.: Zur Unterstützung von Bestandssystemen werden, neben den für den symmetrischen Teil der hybriden Verschlüsselung vorgeschriebenen kryptographischen Algorithmen, für den   |

|                                |  |
|--------------------------------|--|
|                                | <p>symmetrischen Teil der hybriden Entschlüsselung auch folgende Algorithmen unterstützt (siehe [gemSpec_Krypt#3.5.1]):</p> <ul style="list-style-type: none"> <li>• AES-128 GCM</li> <li>• AES-192 GCM</li> </ul> <p>RSA- und ECC-basierter Hybridschlüssel:<br/>Wenn sowohl ein RSA- als auch ein ECC-basierter Hybridschlüssel vorliegen, muss zuerst die Entschlüsselung des ECC-basierten Hybridschlüssels erfolgen. Falls dabei ein Fehler auftritt, muss der Fehler protokolliert werden, und dann - ohne Abbruch - mit der Entschlüsselung des RSA-basierten Hybridschlüssels fortgefahren werden.</p> |
| Fehlerfälle                    | <p>Siehe Tabelle TAB_KON_142 Fehlercodes TUC_KON_071 „Daten hybrid entschlüsseln“.</p> <p>Wenn im Ablauf des TUCs ein anderer Fehler als die in Tabelle TAB_KON_142 Fehlercodes TUC_KON_071 „Daten hybrid entschlüsseln“ beschriebenen Fehler auftritt, wird Fehler 4107 gemeldet.</p>   |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | Abbildung PIC_KON_059 Aktivitätsdiagramm „Daten hybrid entschlüsseln“  |

3808



3809



**Abbildung 14: PIC\_KON\_059 Aktivitätsdiagramm „Daten hybrid entschlüsseln“**

**Tabelle 170: TAB\_KON\_142 Fehlercodes TUC\_KON\_071 „Daten hybrid entschlüsseln“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4106  | Technical | Error    | falscher Schlüssel  |
| 4107  | Technical | Error    | hybride Entschlüsselung konnte nicht durchgeführt werden      |
| 4103  | Technical | Error    | XML-Element nicht gefunden                                    |
| 4104  | Technical | Error    | XML-Element nicht eindeutig identifiziert                     |
| 4201  | Technical | Error    | kryptographischer Algorithmus vom Konnektor nicht unterstützt |

[<=]

**4.1.7.4.3 TUC\_KON\_072 „Daten symmetrisch verschlüsseln“**

**TIP1-A\_4618 - TUC\_KON\_072 „Daten symmetrisch verschlüsseln“**

Der Konnektor MUSS den technischen Use Case TUC\_KON\_072 „Daten symmetrisch verschlüsseln“ umsetzen.

**Tabelle 171: TAB\_KON\_741 – TUC\_KON\_072 „Daten symmetrisch verschlüsseln“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_072 „Daten symmetrisch verschlüsseln“  |
| Beschreibung   | Es wird ein Dokument symmetrisch verschlüsselt. Dabei kann der zu verwendende symmetrische Schlüssel optional übergeben werden.  |
| Auslöser       | Aufruf durch ein Fachmodul in einem fachlichen Use Case  |
| Vorbedingungen | keine  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>documentToBeEncrypted (zu verschlüsselndes Dokument.)</li> <li>symmetricKey – <i>optional</i> (zu verwendender symmetrischer Schlüssel)</li> </ul>  |
| Komponenten    | Konnektor  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>encryptedDocument (Verschlüsseltes Dokument)</li> <li>symmetricKey – <i>optional/verpflichtend, wenn Schlüssel durch den TUC erzeugt wurde</i> (erzeugter symmetrischer Schlüssel)</li> </ul> |
| Standardablauf | 1. Wurde kein symmetrischer Schlüssel übergeben, so wird ein Schlüssel erzeugt. Die Qualität des   |

|                                |  |
|--------------------------------|--|
|                                | <p>Schlüssels muss den Vorgaben in [gemSpec_Krypt#2.2] genügen.</p> <p>2. Das Dokument wird mit dem erzeugten oder übergebenen symmetrischen Schlüssel verschlüsselt. Als Verfahren zum Verschlüsseln wird CMS gewählt ([RFC5652]). Die Content Type Option „Encrypted-data Content Type“ ist zu verwenden.<br/> Content Type = OID{... pkcs-7 encryptedData}<br/> = 1.2.840.113549.1.7.6<br/> Die symmetrische Verschlüsselung binärer Daten erfolgt nach den Vorgaben gemäß [gemSpec_Krypt#<a href="#">GS-A 5016</a>]. Falls die Verschlüsselung fehlschlägt, wird Fehler 4108 gemeldet.</p> <p>3. Das verschlüsselte Dokument und der symmetrische Schlüssel (falls dieser erzeugt wurde) werden zurückgeliefert.</p> |
| Varianten/Alternativen         | keine  |
| Fehlerfälle                    | Siehe Standardablauf.  |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | keine  |

3820

3821

3822

**Tabelle 172: TAB\_KON\_742 Fehlercodes TUC\_KON\_072 „Daten symmetrisch verschlüsseln“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4108  | Technical | Error    | Symmetrische Verschlüsselung konnte nicht durchgeführt werden |

3823

3824

[&lt;=]

3825

#### 4.1.7.4.4 TUC\_KON\_073 „Daten symmetrisch entschlüsseln“

3826

#### **TIP1-A\_4619 - TUC\_KON\_073 „Daten symmetrisch entschlüsseln“**

3827

Der Konnektor MUSS den technischen Use Case TUC\_KON\_073 „Daten symmetrisch entschlüsseln“ umsetzen.

3828

3829

3830

**Tabelle 173: TAB\_KON\_743 - TUC\_KON\_073 „Daten symmetrisch entschlüsseln“**

| Element | Beschreibung                                  |
|---------|---|
| Name    | TUC_KON_073 „Daten symmetrisch entschlüsseln“ |

|                                |   |
|--------------------------------|---|
| Beschreibung                   | Es wird ein Dokument symmetrisch entschlüsselt. Der zu verwendende symmetrische Schlüssel wird übergeben.   |
| Auslöser                       | Aufruf durch ein Fachmodul in einem fachlichen Use Case   |
| Vorbedingungen                 | keine   |
| Eingangsdaten                  | <ul style="list-style-type: none"> <li>encryptedDocument (Verschlüsseltes Dokument)</li> <li>symmetricKey (zu verwendender symmetrischer Schlüssel)</li> </ul>                                  |
| Komponenten                    | Konnektor   |
| Ausgangsdaten                  | <ul style="list-style-type: none"> <li>plainDocument (Entschlüsseltes Dokument)</li> </ul>  |
| Standardablauf                 | Das verschlüsselte Dokument wird mit dem symmetrischen Schlüssel entschlüsselt. Als Verfahren zum Entschlüsseln wird CMS gewählt ([RFC5652]). Das entschlüsselte Dokument wird zurückgeliefert. |
| Varianten/Alternativen         | keine   |
| Fehlerfälle                    | Bei Auftreten eines Fehlers im Standardablauf wird Fehlercode 4109 gemeldet.  |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | keine   |

3831

3832

3833

**Tabelle 174: TAB\_KON\_744 Fehlercodes TUC\_KON\_073 „Daten symmetrisch entschlüsseln“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4109  | Technical | Error    | symmetrische Entschlüsselung konnte nicht durchgeführt werden |

3834

3835

[&lt;=]

3836

4.1.7.4.5 TUC\_KON\_075 „Symmetrisch verschlüsseln“

3837

**A\_18001 - TUC\_KON\_075 „Symmetrisch verschlüsseln“**

3838

Der Konnektor MUSS den technischen Use Case TUC\_KON\_075 „Symmetrisch verschlüsseln“ umsetzen.

3839

3840

3841

**Tabelle 175: TAB\_KON\_860 – TUC\_KON\_075 „Symmetrisch verschlüsseln“**

| Element | Beschreibung |
|---------|--------------|
|---------|--------------|

|                        |  |
|------------------------|--|
| Name                   | TUC_KON_075 „Symmetrisch verschlüsseln“  |
| Beschreibung           | Es werden binäre Daten symmetrisch verschlüsselt. Optional können der zu verwendende symmetrische Schlüssel und AssociatedData für Authenticated Encryption with Associated Data (AEAD) übergeben werden.  |
| Auslöser               | Aufruf durch ein Fachmodul in einem fachlichen Use Case  |
| Vorbedingungen         | keine  |
| Eingangsdaten          | <ul style="list-style-type: none"> <li>• dataToBeEncrypted (zu verschlüsselnde Daten)</li> <li>• symmetricKey – optional (zu verwendender symmetrischer Schlüssel)</li> <li>• associatedData – optional (Parameter für den Verschlüsselungsalgorithmus)</li> </ul>   |
| Komponenten            | Konnektor  |
| Ausgangsdaten          | <ul style="list-style-type: none"> <li>• encryptedData (Verschlüsselte Daten mit der Struktur gemäß Punkt 2 aus <a href="#">A_18004</a>)</li> <li>• symmetricKey – optional/verpflichtend, wenn Schlüssel durch den TUC erzeugt wurde (erzeugter symmetrischer Schlüssel)</li> </ul>   |
| Standardablauf         | <ol style="list-style-type: none"> <li>1. Wurde kein symmetrischer Schlüssel übergeben, so wird ein Schlüssel erzeugt. Die Qualität des Schlüssels muss den Vorgaben in <a href="#">GS-A_4367</a> genügen.</li> <li>2. dataToBeEncrypted wird mit dem erzeugten oder übergebenen symmetrischen Schlüssel unter Berücksichtigung der optional übergebenen associatedData verschlüsselt. Die symmetrische Verschlüsselung binärer Daten erfolgt nach den Vorgaben gemäß <a href="#">A_17872</a>.</li> <li>3. encryptedData wird erzeugt mit der Struktur gemäß Punkt 2 aus <a href="#">A_18004</a>.</li> <li>4. Das verschlüsselte Dokument und der symmetrische Schlüssel (falls dieser erzeugt wurde) werden zurückgeliefert.</li> </ol> |
| Varianten/Alternativen | keine  |
| Fehlerfälle            | -> 2: Falls die Verschlüsselung fehlschlägt, wird Fehler 4108 gemäß TAB_KON_742 gemeldet.  |

|                                |       |
|--------------------------------|-------|
| Nichtfunktionale Anforderungen | keine |
| Zugehörige Diagramme           | keine |

3842 [ $\leq$ ]

3843 4.1.7.4.6 TUC\_KON\_076 „Symmetrisch entschlüsseln“

3844 **A\_18002 - TUC\_KON\_076 „Symmetrisch entschlüsseln“**3845 Der Konnektor MUSS den technischen Use Case TUC\_KON\_076 „Symmetrisch  
3846 entschlüsseln“ umsetzen.

3847

3848 **Tabelle 176: TAB\_KON\_861 - TUC\_KON\_076 „Symmetrisch entschlüsseln“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_076 „Symmetrisch entschlüsseln“  |
| Beschreibung   | Es werden verschlüsselte Daten symmetrisch entschlüsselt. Für Authenticated Encryption with Associated Data (AEAD) kann AssociatedData optional übergeben werden. Der zu verwendende symmetrische Schlüssel wird übergeben.  |
| Auslöser       | Aufruf durch ein Fachmodul in einem fachlichen Use Case  |
| Vorbedingungen | keine  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• encryptedData (Verschlüsselte Daten mit der Struktur gemäß Punkt 2 aus <a href="#">A_18004</a>)</li> <li>• symmetricKey (zu verwendender symmetrischer Schlüssel)</li> <li>• associatedData - optional (Parameter für den Verschlüsselungsalgorithmus)</li> </ul> |
| Komponenten    | Konnektor  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>• plainData (Entschlüsselte Daten)</li> </ul>   |
| Standardablauf | Das verschlüsselte Dokument wird mit dem symmetrischen Schlüssel und associatedData unter Verwendung der kryptographischen Verfahren aus <a href="#">A_17872</a> entschlüsselt. Die entschlüsselten Daten werden zurückgeliefert.  |

|                                |  |
|--------------------------------|--|
| Varianten/Alternativen         | keine  |
| Fehlerfälle                    | Bei Auftreten eines Fehlers im Standardablauf wird Fehlercode 4109 gemäß TAB_KON_744 gemeldet. |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | keine  |

[&lt;=]

#### 4.1.7.5 Operationen an der Außenschnittstelle

##### **TIP1-A\_4620-03** **TIP1-A\_4620-02 - Basisdienst Verschlüsselungsdienst**

Der Konnektor MUSS für Clients einen Basisdienst Verschlüsselungsdienst anbieten.

**Tabelle 177: TAB\_KON\_745 Basisdienst Verschlüsselungsdienst**

|                          |  |                               |
|--------------------------|--|-------------------------------|
| <b>Name</b>              | EncryptionService  |                               |
| <b>Version (KDV)</b>     | 6.1.0 (WSDL-Version), 6.1.1 (XSD-Version)<br>6.1.1 (WSDL-Version), 6.1.2 (XSD-Version) |                               |
| <b>Namensraum</b>        | Siehe <a href="#">Anhang D</a> <a href="#">GitHub</a>                                  |                               |
| <b>Namensraum-Kürzel</b> | CRYPT für Schema und CRYPTW für WSDL   |                               |
| <b>Operationen</b>       | <b>Name</b>  | <b>Kurzbeschreibung</b>       |
|                          | EncryptDocument  | Dokument hybrid verschlüsseln |
|                          | DecryptDocument  | Dokument hybrid entschlüsseln |
| <b>WSDL</b>              | EncryptionService.wsdl (WSDL-Version 6.1.0)<br>EncryptionService_v6_1_1.wsdl           |                               |
| <b>Schema</b>            | EncryptionService.xsd (XSD-Version 6.1.1)<br>EncryptionService_v6_1_2.xsd              |                               |

[&lt;=]

#### 4.1.7.5.1 EncryptDocument

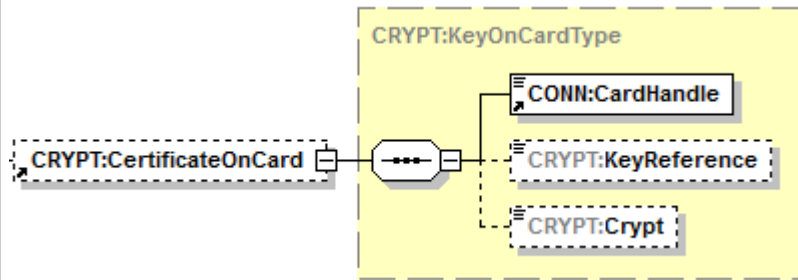
##### **TIP1-A\_4621-02 - Operation EncryptDocument**

Der Basisdienst Verschlüsselungsdienst des Konnektors MUSS an der Clientschnittstelle eine Operation EncryptDocument anbieten.

3863

Tabelle 178: TAB\_KON\_071 Operation EncryptDocument

| Name  | EncryptDocument  |      |              |         |  |
|---|--|------|--------------|---------|--|
| <b>Beschreibung</b>   | <p>Diese Operation verschlüsselt ein übergebenes Dokument hybrid. Es werden die Dokumententypen <code>Alle_DocFormate</code> unterstützt. Für die hybride Verschlüsselung wird ein asymmetrischer Schlüssel aus einem X.509v3-Zertifikat genutzt. Dieses Zertifikat kann von einer Karte kommen oder als Parameter übergeben werden. Pro Operationsaufruf können mehrere Hybridschlüssel erzeugt werden. Übergibt der Aufrufer die Zertifikate beim Aufruf, steuert er durch die Wahl der Zertifikate, ob RSA-basierte oder ECC-basierte Hybridschlüssel erzeugt werden. Wenn das Verschlüsselungszertifikat von einer Karte kommt, kann der Aufrufer durch Angabe des Kryptoverfahrens <code>crypt</code> steuern, ob Hybridschlüssel für RSA oder für ECC oder beide erzeugt werden. Das Defaultverhalten ist die Hybridschlüsselerzeugung für RSA und entspricht dem Verhalten aus der Version 6.1.0 der Schnittstelle. Es werden die folgenden Karten unterstützt: HBAX und SM-B. Die Operation EncryptDocument DARF das Verschlüsseln mit der eGK NICHT unterstützen. Für alle Dokumententypen wird immer das gesamte Dokument verschlüsselt.</p> |      |              |         |  |
| <div data-bbox="429 1077 1107 1312"> </div> <table border="1" data-bbox="405 1384 1401 1686"> <thead> <tr> <th>Name</th><th>Beschreibung</th></tr> </thead> <tbody> <tr> <td>Context</td><td>           Aufrufkontext:           <ul style="list-style-type: none"> <li>MandantID, ClientSystemID, WorkplaceId verpflichtend</li> <li>UserID verpflichtend bei HBAX, bei SM-B nicht ausgewertet</li> </ul> </td></tr> </tbody> </table> <div data-bbox="405 1733 1107 1861"> </div> |  | Name | Beschreibung | Context | Aufrufkontext: <ul style="list-style-type: none"> <li>MandantID, ClientSystemID, WorkplaceId verpflichtend</li> <li>UserID verpflichtend bei HBAX, bei SM-B nicht ausgewertet</li> </ul> |
| Name  | Beschreibung   |      |              |         |  |
| Context   | Aufrufkontext: <ul style="list-style-type: none"> <li>MandantID, ClientSystemID, WorkplaceId verpflichtend</li> <li>UserID verpflichtend bei HBAX, bei SM-B nicht ausgewertet</li> </ul>   |      |              |         |  |



Das RecipientKeys-Element identifiziert die Empfänger der zu verschlüsselnden Nachricht über X.509-Zertifikate (öffentliche Schlüssel). Quelle für die Zertifikate kann eine gesteckte Karte sein, die per CertificateOnCard-Element referenziert wird, oder der Aufrufer, der X.509-Zertifikate im Certificate-Element übergibt.

|             |   |
|-------------|---|
| Card Handle | Identifiziert die zu verwendende Karte mit dem (öffentlichen) Schlüssel.<br>Ist das Element nicht vorhanden, so werden nur Zertifikate per Element Certificate übergeben. |
|-------------|---|

|              |   |
|--------------|---|
| KeyReference | Der Wert dieses Parameters ist in Tabelle TAB_KON_747 KeyReference für Encrypt-/DecryptDocument spezifiziert. Ist der Parameter nicht angegeben, gilt der Default-Wert C.ENC. |
|--------------|---|

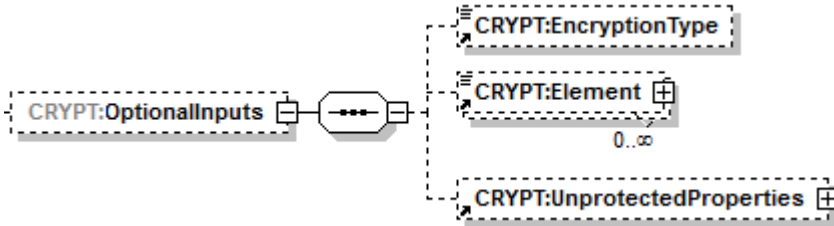

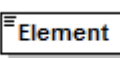

|       |  |
|-------|--|
| Crypt | Optional;<br>Default: siehe TAB_KON_859<br>Wertebereich: [ENC_CRYPT]<br>Gibt den Typ von Zertifikaten vor, die von der per CardHandle referenzierten Karte für die Erzeugung der Hybridschlüssel gemäß Tabelle TAB_KON_747 verwendet werden. |
|-------|--|

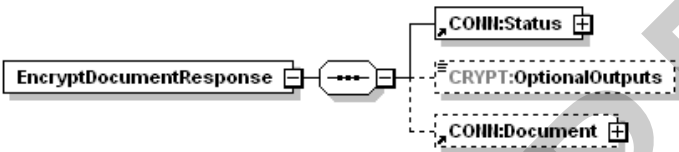
|             |  |
|-------------|--|
| Certificate | Certificate ist ein Base64-kodiertes XML-Element, in dem das Zertifikat, das den asymmetrischen Schlüssel enthält (öffentlicher Schlüssel), DER-kodiert übergeben wird.<br>Es kann eine Liste von Zertifikaten übergeben werden. Kommt das Zertifikat ausschließlich von einer Karte, dann kann dieser Parameter weggelassen werden. |
|-------------|--|

**Document** 

|                |  |
|----------------|--|
| CONN: Document | Dieses entsprechend [OASIS-DSS] Section 2.4.2 spezifizierende Element enthält das zu verschlüsselnde Dokument, wobei die Kindelemente CONN:Base64XML und dss:Base64Data verwendet werden. Im Fall dss:Base64Data wird ein etwaig übergebenes MIME-Type-Attribut nicht ausgewertet. |
|----------------|--|



|   |   |
|---|---|
|   |   |
| CRYPT:OptionalInputs  | Enthält eine Auswahl der folgenden unten näher erläuterten (optionalen) Eingabeparameter:   |
|    |   |
| EncryptionType  | <p>Zu wählendes Verschlüsselungsverfahren, wobei folgende URI vorgesehen sind:</p> <ul style="list-style-type: none"> <li>• XMLEnc: „http://www.w3.org/TR/xmlenc-core/“</li> <li>• CMS: „urn:ietf:rfc:5652“</li> <li>• S/MIME: „urn:ietf:rfc:5751“</li> </ul> <p>Im Fall XMLEnc wird ein Base64-codiertes XML-Dokument im Element <code>CONN:Document/CONN:Base64XML</code> übergeben. In den Fällen CMS und S/MIME wird ein Base64-codiertes Binär-Dokument im Element <code>CONN:Document/dss:Base64Data</code> übergeben.</p> <p>Ist der Parameter EncryptionType nicht gesetzt, dann gilt folgendes Default-Verhalten: Für ein im Element <code>CONN:Document/CONN:Base64XML</code> übergebenes XML-Dokument wird als Verschlüsselungsverfahren [XMLEnc] angewandt, und für ein im Element <code>CONN:Document/dss:Base64Data</code> übergebenes Dokument wird das Verschlüsselungsverfahren CMS angewandt.</p> <p>XML-Dokumente werden nach <code>Type=http://www.w3.org/2001/04/xmlenc#Element</code> verschlüsselt. Im Fall S/MIME ist das in <code>CONN:Document/dss:Base64Data</code> übergebene Dokument eine MIME-Nachricht.</p> |
|  |   |
| Element   | Der Parameter wird nicht ausgewertet.   |
|  |   |

|                        |   |   |
|------------------------|---|---|
|                        | CRYPT:UnprotectedProperties   | <p>Dieses optionale Element wird im CMS-Fall (EncryptionType = urn:ietf:rfc:5652) ausgewertet. Die Elemente <code>./UnprotectedProperties/Property/Value/CMSAttribute</code> müssen base64/DER-kodiert ein vollständiges ASN.1-Attribute enthalten, definiert in [CMS# 9.1.AuthenticatedData Type]. Es muss bei der Erstellung des CMS-Containers unter "unauthAttrs" aufgenommen werden. Das zugehörige Element <code>./UnprotectedProperties/Property/Identifier</code> wird nicht ausgewertet.</p>   |
| <b>Rückgabe</b>        |                   |   |
|                        | Status  | Enthält den Ausführungsstatus der Operation.  |
|                        | CRYPT:OptionalOutputs   | Kann – in zukünftigen Versionen der Spezifikation – optionale Ausgabeparameter enthalten.   |
|                        | CONN:Document   | <p>Enthält das verschlüsselte Dokument in base64-codierter Form, wenn die Verschlüsselung erfolgreich durchgeführt wurde.</p> <p>Im Fall XMLEnc wird das Base64-codierte verschlüsselte XML-Dokument im Element <code>CONN:Document/CONN:Base64XML</code> zurückgegeben.</p> <p>Im Fall CMS wird das Base64-codierte Binär-Dokument im Element <code>CONN:Document/dss:Base64Data</code> zurückgegeben.</p> <p>Im Fall S/MIME wird die Base64-codierte S/MIME-Nachricht im Element <code>CONN:Document/dss:Base64Data</code> zurückgegeben. Das Attribut <code>CONN:Document/dss:Base64Data/@MimeType</code> wird auf „application/pkcs7-mime“ gesetzt. Die S/MIME-Nachricht hat Content-Transfer-Encoding: base64.</p> |
| <b>Fehler</b>          | Bei Auftreten eines Fehlers im Standardablauf werden Fehlercodes entsprechend TAB_KON_141 gemeldet. |   |
| <b>Vorbedingungen</b>  | Keine   |   |
| <b>Nachbedingungen</b> | Keine   |   |

3865 Der Ablauf der Operation EncryptDocument ist in Tabelle TAB\_KON\_746 Ablauf  
 3866 EncryptDocument beschrieben:

3867

3868 **Tabelle 179: TAB\_KON\_746 Ablauf EncryptDocument**

| Nr. | Aufruf<br>Technischer Use<br>Case oder<br>Interne<br>Operation | Beschreibung  |
|-----|--|---|
| 1.  | checkArguments   | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab. Wird die Operation für einen nicht unterstützten Kartentypen aufgerufen, so bricht die Operation mit Fehler 4058 ab.   |
| 2.  | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“               | Die Prüfung erfolgt durch den Aufruf<br>TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientsystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>userId = \$context.userId;<br>cardHandle = \$cardHandle }<br>Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab. |
| 3.  | TUC_KON_026<br>„Liefere<br>CardSession“                        | Ermittle CardSession über TUC_KON_026 {<br>mandatId = \$context..mandantId;<br>clientSystemId = \$context.clientSystemId;<br>cardHandle = \$context..cardHandle;<br>userId = \$context.userId }   |
| 4.  | TUC_KON_070<br>„Daten hybrid<br>verschlüsseln“                 | Die hybride Verschlüsselung wird durchgeführt. Tritt hierbei ein Fehler auf, bricht die Operation ab.<br>Die KeyInfo, d.h. die Liste der Hybridschlüssel inklusive des bei ihrer Erzeugung verwendeten Zertifikates, sind dabei in das Dokument einzubetten.  |

3869 **Tabelle 180: TAB\_KON\_141 Fehlercodes „EncryptDocument“**

| Fehlercode  | ErrorType | Severity | Fehlertext      |
|---|-----------|----------|-----------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |                 |
| 4000  | Technical | Error    | Syntaxfehler    |
| 4001  | Security  | Error    | Interner Fehler |

|      |          |       |                       |
|------|----------|-------|-----------------------|
| 4058 | Security | Error | Aufruf nicht zulässig |
|------|----------|-------|-----------------------|

3870

3871 [ $\leq$ ]

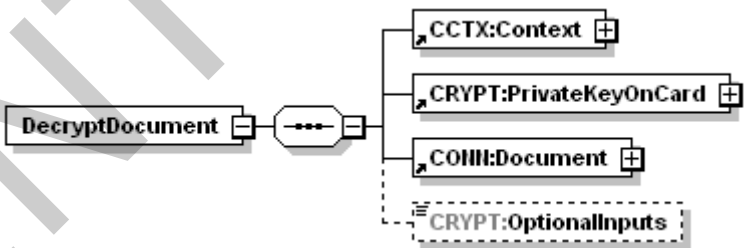
3872 4.1.7.5.2 DecryptDocument

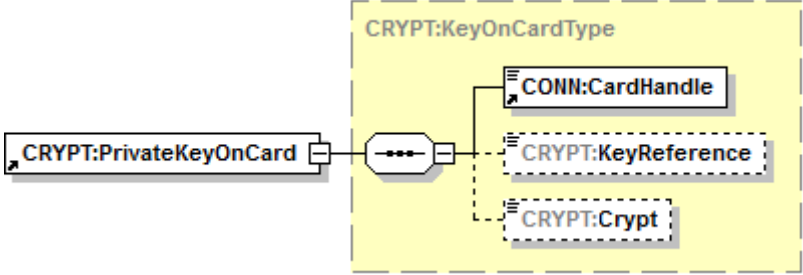
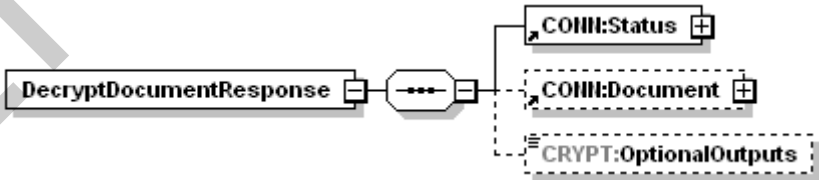
3873 **TIP1-A\_4622-02 - Operation DecryptDocument**

3874 Der Basisdienst Verschlüsselungsdienst des Konnektors MUSS an der Clientschnittstelle  
 3875 eine Operation DecryptDocument anbieten.

3876

3877 **Tabelle 181: TAB\_KON\_075 Operation DecryptDocument**

|                        |   |   |
|------------------------|---|---|
| <b>Name</b>            | DecryptDocument   |   |
| <b>Beschreibung</b>    | <p>Die Operation entschlüsselt alle hybrid verschlüsselten Dokumente, die mit der Operation EncryptDocument erzeugt wurden. Es werden die Dokumententypen <code>Alle_DocFormate</code> unterstützt.</p> <p>Für die Entschlüsselung wird ein asymmetrischer Schlüssel zu einem X.509v3-Zertifikat genutzt. Dieses Zertifikat und der Schlüssel müssen von einer Karte kommen.</p> <p>Das bei der Entschlüsselung verwendete Kryptoverfahren (RSA oder ECC) wird durch den Hybridschlüssel bestimmt, der durch die Karte entschlüsselt werden soll. Sind sowohl RSA- als auch ECC-Hybridschlüssel für die referenzierte Karte vorhanden, versucht der Konnektor die Entschlüsselung des ECC-Hybridschlüssels, und wenn das nicht erfolgreich war, die Entschlüsselung des RSA-Hybridschlüssels.</p> |   |
| <b>Aufrufparameter</b> |   |   |
|                        | <b>Name</b>   | <b>Beschreibung</b>   |
|                        | Context   | <p>Aufrufkontext:</p> <ul style="list-style-type: none"> <li>• MandantId, ClientSystemId, WorkplaceId verpflichtend</li> <li>• UserId verpflichtend bei HBAX, bei SM-B nicht ausgewertet</li> </ul> |

|                 |   |   |
|-----------------|---|---|
|                 |                   |   |
|                 | PrivateKeyOnCard  | Identifiziert die zu verwendende Karte mit dem (privaten) Schlüssel.<br>Es werden die folgenden Karten unterstützt: HBAX und SM-B. Die Operation DecryptDocument DARF das Entschlüsseln mit der eGK NICHT unterstützen. |
|                 | CardHandle  | Identifiziert die gesteckte Karte.  |
|                 | KeyReference  | Der Wert dieses Parameters ist in der Tabelle TAB_KON_747 KeyReference für Encrypt-/DecryptDocument spezifiziert. Ist der Parameter nicht angegeben, gilt der Default-Wert C.ENC.                                       |
|                 | Crypt   | Ist nicht enthalten.  |
|                 | CONN:Document   | Enthält das base64-codierte Dokument, das entschlüsselt werden soll.  |
|                 | CRYPT:OptionalInputs  | Kann – in zukünftigen Versionen der Spezifikation – optionale Aufrufparameter enthalten.  |
| <b>Rückgabe</b> |                 |   |
|                 | Status  | Enthält den Ausführungsstatus der Operation.  |
|                 | CRYPT:OptionalOutputs   | Kann – in zukünftigen Versionen der Spezifikation – optionale Ausgabeparameter enthalten.   |
|                 | CONN:Document   | Enthält das entschlüsselte Dokument in base64-codierter Form  |
| <b>Fehler</b>   | Bei Auftreten eines Fehlers im Standardablauf werden Fehlercodes entsprechend TAB_KON_145 gemeldet. |   |

|                        |       |
|------------------------|-------|
| <b>Vorbedingungen</b>  | Keine |
| <b>Nachbedingungen</b> | Keine |

Der Ablauf der Operation DecryptDocument ist in Tabelle TAB\_KON\_076 Ablauf DecryptDocument beschrieben:

**Tabelle 182: TAB\_KON\_076 Ablauf DecryptDocument**

| Nr.   | Aufruf<br>Technischer Use<br>Case oder<br>Interne<br>Operation | Beschreibung  |
|-------|--|---|
| 1. 2. | checkArguments   | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab. Wird die Operation für einen nicht unterstützten Kartentypen aufgerufen, so bricht die Operation mit Fehler 4058 ab.   |
| 2. 1. | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“               | Die Prüfung erfolgt durch den Aufruf<br>TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientsystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>userId = \$context.userId;<br>cardHandle = \$cardHandle }<br>Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab. |
| 3.    | TUC_KON_026<br>„Liefere<br>CardSession“                        | Ermittle CardSession über 026 {<br>mandatId = \$context.mandantId;<br>clientsystemId = \$context.clientsystemId;<br>cardHandle = \$context.cardHandle;<br>userId = \$context.userId }   |
| 4. 4. | TUC_KON_071<br>Daten hybrid<br>entschlüsseln                   | Die Entschlüsselung wird durchgeführt.<br>Im Fall eines XML-Dokuments mit mehreren verschlüsselten Elementen sind alle mit dem angegebenen Schlüssel entschlüsselbaren Elemente zu entschlüsseln.   |

**Tabelle 183: TAB\_KON\_145 Fehlercodes „DecryptDocument“**

| Fehlercode  | ErrorType | Severity | Fehlertext            |
|---|-----------|----------|-----------------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |                       |
| 4000  | Technical | Error    | Syntaxfehler          |
| 4001  | Security  | Error    | interner Fehler       |
| 4058  | Security  | Error    | Aufruf nicht zulässig |

3883 [ $\leq$ ]

#### 3884 4.1.7.6 Betriebsaspekte

3885 keine

#### 3886 4.1.8 Signaturdienst

3887 Der Signaturdienst bietet Clientsystemen und Fachmodulen eine Schnittstelle zum  
3888 Signieren von Dokumenten und Prüfen von Dokumentensignaturen

3889 Innerhalb des Signaturdienstes werden folgende Präfixe für Bezeichner verwendet:

- 3890 • Events (Topic Ebene 1): keine Events vorhanden
- 3891 • Konfigurationsparameter: „SAK\_“

#### 3892 4.1.8.1 Funktionsmerkmalweite Aspekte

##### 3893 4.1.8.1.1 Dokumentensignatur

3894 Der Signaturdienst umfasst die Funktionalität der nicht-qualifizierten elektronischen  
3895 Signatur (nonQES) mit der SM-B, sowie die qualifizierte elektronische Signatur (QES) mit  
3896 dem HBA und den HBA-Vorläuferkarten HBA-qSig und ZOD\_2.0 (=HBAX).

3897 In der Abbildung fachlicher Abläufe kann es nötig sein, ein Dokument mehrfach parallel  
3898 zu signieren, oder existierende Signaturen gegenzusignieren. Der Konnektor unterstützt  
3899 **parallele Signaturen** (QES und nonQES). Ebenso unterstützt er Gegensignaturen (QES  
3900 und nonQES), die jeweils alle bestehenden Signaturen gegensignieren. Die angebotene  
3901 Möglichkeit des Gegensignierens bezieht sich dabei auf das Signieren aller vorhandenen  
3902 parallelen Signaturen, während ein Gegensignieren von Gegensignaturen nicht  
3903 angeboten wird. Der Konnektor unterstützt ausschließlich  
3904 eine **dokumentexkludierende Gegensignatur**, bei der alle Signaturen gegensigniert  
3905 werden, aber nicht der fachliche Inhalt des Dokumentes selbst.

##### 3906 TIP1-A\_4623 - Unterstützte Signaturverfahren nonQES

3907 Der Signaturdienst MUSS für die Erstellung und Prüfung von nicht-qualifizierten  
3908 elektronischen Signaturen (nonQES) für die nonQES\_DocFormate die Signaturverfahren  
3909 entsprechend Tabelle TAB\_KON\_582 – Signaturverfahren unterstützen.

3910 [ $\leq$ ]

##### 3911 TIP1-A\_4627 - Unterstützte Signaturverfahren QES

3912 Der Signaturdienst MUSS für die Erstellung und Prüfung von qualifizierten elektronischen  
3913 Signaturen (QES) für die QES\_DocFormate die Signaturverfahren entsprechend Tabelle  
3914 TAB\_KON\_582 – Signaturverfahren unterstützen.

3915 [ $\leq$ ]

3916

#### 3917 Tabelle 184: TAB\_KON\_582 – Signaturverfahren Dokumentensignatur

| Signaturformat             | Standard               | Dokumentformate | QES/<br>nonQES | Bemerkung  |
|----------------------------|------------------------|-----------------|----------------|--|
| <b>XMLDSig<br/>(XAdES)</b> | [RFC3275]<br>[XMLDSig] | XML             | QES,<br>nonQES | Hierdurch können<br>abgesetzte<br>(detached),<br>umschließende |

|                      |                      |                                     |                |   |
|----------------------|----------------------|-------------------------------------|----------------|---|
|                      | [XAdES]<br>[RFC6931] |                                     |                | (enveloping) und eingebettete (enveloped) Signaturen erzeugt werden.  |
| <b>CMS (CAdES)</b>   | [RFC5652]<br>[CAdES] | QES_DocFormate<br>nonQES_DocFormate | QES,<br>nonQES | Hierdurch können abgesetzte (detached) und umschließende (enveloping) Signaturen erzeugt werden.                          |
| <b>PDF/A (PAdES)</b> | [PAdES-3]            | PDF/A                               | QES,<br>nonQES | Hierdurch können CMS-basierte Signaturen in PDF/A-Dokumente eingefügt und dadurch eingebettete Signaturen erzeugt werden. |
| <b>S/MIME</b>        | [RFC5751]            | nonQES_DocFormate                   | nonQES         | Es werden MIME-Nachrichten signiert.  |

3918 Zu den Begriffen detached, enveloping und enveloped Signaturen siehe beispielsweise  
 3919 auch [HüKo06#Abs. 4.3.3. und 4.3.1.5].

3920

#### 3921 **TIP1-A\_5447 - Einsatzbereich der Signaturvarianten**

3922 Der Signatordienst MUSS für die Erstellung und Prüfung von nicht-qualifizierten  
 3923 elektronischen Signaturen (nonQES) und qualifizierten elektronischen Signaturen (QES)  
 3924 die Vorgaben zum Einsatzbereich gemäß Tabelle TAB\_KON\_778 umsetzen.

3925 **Tabelle 185: TAB\_KON\_778 – Einsatzbereich der Signaturvarianten für XAdES, CAdES**  
 3926 **und PAdES**

| Signaturvarianten |                  |                             |   | Einsatzbereich |                         |                               |
|-------------------|------------------|-----------------------------|---|----------------|-------------------------|-------------------------------|
| Signaturverfahren | Signaturvariante | WAS wird signiert?          | WO wird die Signatur abgelegt?              | nonQES         | QES Außen-schnittstelle | QES Fachmodul - schnittstelle |
| XAdES             | detached         | beliebiges (Binär)-Dokument | außerhalb des Dokuments in der SignResponse | Nein           | Nein                    | Nein                          |



|       |            |  |   |      |         |         |
|-------|------------|--|---|------|---------|---------|
| XAdES | detached   | gesamtes Input XML-Dokument (= Root-Element mit Subelementen)          | außerhalb des Dokuments in der SignResponse                     | Nein | Nein    | Nein    |
| XAdES | detached   | ausgewähltes nicht Root-Element mit Subelementen im Input XML-Dokument | außerhalb des Dokuments in der SignResponse                     | Nein | Nein    | Nein    |
| XAdES | detached   | ausgewähltes nicht Root-Element mit Subelementen im Input XML-Dokument | Innerhalb des Dokuments, aber außerhalb des signierten Subbaums | Nein | Bedingt | Bedingt |
| XAdES | enveloped  | gesamtes Input XML-Dokument (= Root-Element mit Subelementen)          | Als direktes Child des Root-Elements                            | Ja   | Bedingt | Bedingt |
| XAdES | enveloped  | ausgewähltes nicht Root-Element mit Subelementen im Input XML-Dokument | Als direktes Child des ausgewählten Elements                    | Nein | Nein    | Bedingt |
| XAdES | enveloping | gesamtes Input XML-Dokument (= Root-Element mit Subelementen)          | Im Dokument, das Root-Element umschließend                      | Ja   | Bedingt | Bedingt |
| XAdES | enveloping | ausgewähltes nicht Root-Element mit Subelementen im Input XML-Dokument | Im Dokument, das ausgewählte Element umschließend               | Nein | Nein    | Nein    |

|       |            |                         |   |    |    |    |
|-------|------------|-------------------------|---|----|----|----|
| CAdES | detached   | gesamtes Binärdokument  | außerhalb des Dokuments in der SignResponse | Ja | Ja | Ja |
| CAdES | enveloping | gesamtes Binär-Dokument | innerhalb des CMS-Dokuments                 | Ja | Ja | Ja |
| PAdES | -          | gesamtes PDF-Dokument   | Im PDF-Dokument                             | Ja | Ja | Ja |

**Legende:**

Ja: Die Signaturvariante ist für den Einsatzbereich erlaubt.

Nein: Die Signaturvariante ist für den Einsatzbereich nicht erlaubt.

Bedingt: Die Signaturvariante ist für den Einsatzbereich nicht erlaubt, es sei denn es wird durch eine im Konnektor integrierte Signaturreichtlinie explizit gefordert.

Die Spalten mit gelber Kopfzeile definieren die Signaturvarianten, die mit grauer, den Einsatzbereich. Beim Einsatzbereich wird zwischen nonQES und QES unterschieden und im Fall QES nach der Bereitstellung an der Außenschnittstelle oder intern für Fachmodule.

Die benötigten Signaturvarianten werden für XAdES über die Aufrufparameter

IncludeObject und SignaturePlacement gemäß [OASIS-DSS] gesteuert.

Für CAdES erfolgt die Steuerung welche Signaturvariante gewählt wird, über den Aufrufparameter IncludeEContent.

[<=]

**A\_18756 - Optionalität von nonQES-XAdES Signatur**

Der Konnektor KANN alle Aufrufe zu Signaturerstellung einer nonQES-XAdES Signatur mit Fehler 4111 und alle Aufrufe zur Signaturprüfung einer nonQES-XAdES Signatur mit Fehler 4112 beantworten. Die Signaturvarianten aus TAB\_KON\_778 werden damit weiter eingeschränkt. Wird die nonQES-XAdES Signatur umgesetzt, so ist diese in der Sicherheitszertifizierung zu betrachten.[<=]

**TIP1-A\_5402 - Baseline-Profilierung der AdES-EPES-Profile**

Der Konnektor MUSS von den AdES-Profilen die AdES-EPES-Profile umsetzen, ergänzt um

- RevocationValues gemäß AdES-X-L,
- SignatureTimeStamp (für Signaturprüfung, nicht für Signaturerstellung) gemäß AdES-T

Dabei MUSS der Konnektor die Baseline-Profilierung gemäß Kapitel 6 in [XAdES Baseline Profile] für XAdES, Kapitel 6 in [CAdES Baseline Profile] für CAdES und Kapitel 6 in [PAdES Baseline Profile] für PAdES umsetzen.

[<=]

Durch die Baseline-Profilierung der AdES-BES-Profile wird festgelegt, wie der Signaturzeitpunkt, gemessen als Systemzeit des Konnektors, in die Signatur eingebracht wird.

**TIP1-A\_5403 - Common PKI konforme Profile**

Der Konnektor SOLL die signierten Dokumente konform zu [COMMON\_PKI#Part 3] und [COMMON\_PKI#Part 8] erstellen.

[<=]

**TIP1-A\_4624 - Default-Signaturverfahren nonQES**

Bei fehlender expliziter Angabe durch den Aufrufer MUSS der Signatordienst bei der Erstellung von nicht-qualifizierten elektronischen Signaturen (nonQES) die Default-Signaturverfahren entsprechend TAB\_KON\_583 Default-Signaturverfahren wählen.  
[<=]

**TIP1-A\_4628 - Default-Signaturverfahren QES**

Bei fehlender expliziter Angabe durch den Aufrufer MUSS der Signatordienst bei der Erstellung von qualifizierten elektronischen Signaturen (QES) die Default-Signaturverfahren entsprechend TAB\_KON\_583 – Default-Signaturverfahren wählen.  
[<=]

**Tabelle 186: TAB\_KON\_583 – Default-Signaturverfahren**

| Dokument-Format | Signaturverfahren (und -variante) |                  |   |   |
|-----------------|-----------------------------------|------------------|---|---|
|                 | Signaturverfahren                 | Signaturvariante | WAS wird signiert?  | WO wird die Signatur abgelegt?              |
| XML             | XAdES                             | enveloped        | gesamtes Input XML-Dokument (= Root-Element mit Subelementen) | als direktes Child des Root-Elements        |
| PDF/A           | PAdES                             | -                | gesamtes PDF-Dokument   | im PDF-Dokument                             |
| alle anderen    | CADES                             | detached         | gesamtes Binärdokument  | außerhalb des Dokuments in der SignResponse |

**TIP1-A\_5387 - Erweiterte Nutzung der AdES-Profile**

Der Konnektor MUSS auf eine vollständige Nutzung der AdES-Profile erweiterbar sein.  
[<=]

**TIP1-A\_5033 - Missbrauchserkennung Signatordienst (nonQES)**

Der Konnektor MUSS zur Unterstützung von Missbrauchserkennungen die in Tabelle TAB\_KON\_584 gelisteten Operationen als Einträge in EVT\_MONITOR\_OPERATIONS berücksichtigen.

**Tabelle 187: TAB\_KON\_584 nonQES-Operationen für EVT\_MONITOR\_OPERATIONS**

| Operationsname          | OK_Val | NOK_Val | Alarmwert (Default-Grenzwert 10 Minuten-Σ) |
|-------------------------|--------|---------|--|
| SignDocument (nonQES)   | 1      | 5       | 41   |
| VerifyDocument (nonQES) | 1      | 5       | 61   |

[&lt;=]

3986 **TIP1-A\_4629 - Unterstützte Karten QES-Erstellung**  
 3987 Der Signaturdienst MUSS für die QES-Erstellung die Kartentypen HBA, HBA-qSig und  
 3988 ZOD\_2.0 unterstützen.  
 3989 [ $\leq$ ]

3990 **TIP1-A\_5436 - XML Dokument nach Entfernen der Signatur unverändert**  
 3991 Der Konnektor MUSS die Operation SignDocument für XML-Dokumente so  
 3992 implementieren, dass das Dokument nach Entfernen der Signatur, insbesondere auch  
 3993 einer Teilsignatur, als Ganzes unverändert ist, wobei zwei XML-Dokumente als identisch  
 3994 zu betrachten sind, wenn sie gemäß Canonical XML 1.1 gleich sind [CanonXML1.1].  
 3995 [ $\leq$ ]

3996 **TIP1-A\_5682 - XML Nicht geeignete Algorithmen im VerificationReport**  
 3997 Der Konnektor MUSS im VerificationReport einer QES-Signaturprüfung ausweisen, wenn  
 3998 die für die Signatur verwendeten Algorithmen nach dem Algorithmenkatalog [ALGCAT]  
 3999 als nicht geeignet eingestuft werden.  
 4000 [ $\leq$ ]

4001 **A\_17768 - Zertifikate und Schlüssel für Signaturerstellung und Signaturprüfung**  
 4002 **(QES und nonQES)**  
 4003 Der Konnektor MUSS bei der Signaturerstellung und Signaturprüfung (QES und nonQES) die  
 4004 Zertifikate und Schlüssel gemäß den Vorgaben in TAB\_KON\_900 ermitteln.

4005 **Tabelle 188: TAB\_KON\_900 Zertifikate und private Schlüssel für Signaturerstellung und**  
 4006 **Signaturprüfung (QES und nonQES)**

| Karte  | Crypt   | Zertifikat (Verify)   | Schlüssel (Sign)  | Einsatzbereich      |                         |
|--------|---------|---|---|---------------------|-------------------------|
|        |         |   |   | Außen-schnittstelle | Fachmodul-schnittstelle |
| QES    |         | ...in DF.QES  |   |                     |                         |
| HBA    | RSA     | EF.C.HP.QES.R2048   | PrK.HP.QES.R2048  | ja                  | ja                      |
|        | ECC     | EF.C.HP.QES.E256  | PrK.HP.QES.E256   | ja                  | ja                      |
|        | RSA_ECC | [ab <b>G2.1</b> ]: EF.C.HP.QES.E256<br>[ <b>G2.0</b> ]: EF.C.HP.QES.R2048 | [ab <b>G2.1</b> ]: PrK.HP.QES.E256<br>[ <b>G2.0</b> ]: PrK.HP.QES.R2048 | ja                  | ja                      |
| HBA-VK | RSA     | EF.C.HP.QES   | PrK.HP.QES  | ja                  | ja                      |
| nonQES |         | ...in DF.ESIGN  |   |                     |                         |
| SM-B   | RSA     | EF.C.HCI.OSIG.R2048   | PrK.HCI.OSIG.R2048  | ja                  | ja                      |
|        | ECC     | EF.C.HCI.OSIG.E256  | PrK.HCI.OSIG.E256   | ja                  | ja                      |

|     |             |   |   |      |    |
|-----|-------------|---|---|------|----|
|     | RSA_E<br>CC | [ab <b>G2.1</b> ]: EF.C.HCI.OSI<br>G.E256<br>[ <b>G2.0</b> ]: EF.C.HCI.OSIG.R<br>2048 | [ab <b>G2.1</b> ]: PrK.HCI.OSI<br>G.E256<br>[ <b>G2.0</b> ]: PrK.HCI.OSIG.R<br>2048 | ja   | ja |
| eGK | RSA         | EF.C.CH.AUT.R2048   | PrK.CH.AUT.R2048  | nein | ja |
|     | ECC         | EF.C.CH.AUT.E256  | PrK.CH.AUT.E256   | nein | ja |
|     | RSA_E<br>CC | [ab <b>G2.1</b> ]: EF.C.CH.AUT.<br>E256<br>[ <b>G2.0</b> ]: EF.C.CH.AUT.R2<br>048     | [ab <b>G2.1</b> ]: PrK.CH.AUT.<br>E256<br>[ <b>G2.0</b> ]: PrK.CH.AUT.R20<br>48     | nein | ja |

[&lt;=]

**Tabelle 189: TAB\_KON\_862 Werteliste und Defaultwert des Parameters crypt bei QES-Erzeugung**

| Typname       | Werteliste            | Defaultwert | Bedeutung   |
|---------------|-----------------------|-------------|---|
| SIG_CRYPT_QES | RSA<br>ECC<br>RSA_ECC | RSA_ECC     | Werteliste des Parameters crypt bei der bei der Erzeugung einer QES-Signatur<br>RSA: Es wird eine RSA-2048 Signatur erzeugt.<br>ECC: Es wird eine ECC-256 Signatur erzeugt.<br>RSA_ECC: In Abhängigkeit von der Kartengeneration wird eine RSA-2048 bzw. eine ECC-256 Signatur erzeugt (siehe TAB_KON_900). |

**Tabelle 190: TAB\_KON\_863 Werteliste und Defaultwert des Parameters crypt bei nonQES-Erzeugung**

| Typname          | Werteliste            | Defaultwert | Bedeutung  |
|------------------|-----------------------|-------------|--|
| SIG_CRYPT_nonQES | RSA<br>ECC<br>RSA_ECC | RSA         | Werteliste des Parameters crypt bei der bei der Erzeugung einer nonQES-Signatur<br>RSA: Es wird eine RSA-2048 Signatur erzeugt.<br>ECC: Es wird eine ECC-256 Signatur erzeugt.<br>RSA_ECC: In Abhängigkeit von der Kartengeneration wird eine RSA-2048 bzw. eine ECC-256 Signatur erzeugt (siehe TAB_KON_900). |

## 4015 4.1.8.1.2 Signaturreichtlinien

4016 Signaturreichtlinien dienen der Profilierung von Signaturerstellung und -prüfung. Beim  
4017 Aufruf der Operation SignDocument kann eine URI übergeben werden, die eine im  
4018 Konnektor hinterlegte Signaturreichtlinie referenziert. Die Plattform des Konnektors stellt  
4019 selbst keine Signaturreichtlinien bereit. Fachanwendungen, die Signaturreichtlinien  
4020 erfordern, definieren diese im Fachmodul des Konnektors. Für XML-Dokumentenformate  
4021 aus der Menge von QES\_DocFormate können die nachfolgenden Aspekte über eine  
4022 Signaturreichtlinie gekapselt festgelegt werden:

- 4023 • XML-Schemas für die Typkonformitätsprüfung (im Konnektor zu hinterlegen)
- 4024 • Constraints für den Aufruf der Schnittstelle SignDocument und VerifyDocument,  
4025 die zur Profilierung der Schnittstelle dienen.

4026 **TIP1-A\_5538 - Signaturreichtlinien bei QES für XML-Dokumentenformate**

4027 Der Konnektor MUSS Signaturreichtlinien für XML-Dokumentenformate aus der Menge von  
4028 QES\_DocFormate bei der Signaturerstellung und -prüfung umsetzen.

4029 Der Konnektor MUSS den für jede Signaturreichtlinie definierten Bezeichner (URI) bei der  
4030 Signatur als SigPolicyId im Feld SignaturePolicyIdentifier einbetten. Bei der  
4031 Signaturprüfung MUSS der Konnektor über eine etwaig vorhandene SigPolicyId die  
4032 Signaturreichtlinie identifizieren.

4033 Die gemäß AdES erforderliche Hash-Referenz über die Policy (SigPolicyHash) MUSS  
4034 Schema-konform leer gelassen werden. Bei der Signaturprüfung DARF die Hash-Referenz  
4035 über die Policy NICHT geprüft werden.

4036 [**<=**]

## 4037 4.1.8.1.3 Signaturzeitpunkt

4038 Bezogen auf den vom Konnektor für die Signaturprüfung anzunehmenden  
4039 Signaturstellungszeitpunkt werden in dieser Spezifikation die Bezeichner  
4040 Ermittelter\_Signaturzeitpunkt und Benutzerdefinierter\_Zeitpunkt verwendet.

4041 **Ermittelter\_Signaturzeitpunkt:** Vom Konnektor ermittelter Zeitpunkt, zu dem eine  
4042 Signatur geprüft wird. Es werden folgende Signaturzeitpunkte ermittelt:

- 4043 1. Ermittelter\_Signaturzeitpunkt\_Eingebettet:  
4044 in der Signatur eingebetteter Zeitpunkt (falls vorhanden)
- 4045 2. Ermittelter\_Signaturzeitpunkt\_System:  
4046 Systemzeit des Konnektors bei Signaturprüfung

4047 Anmerkung: Bei vom Konnektor selbst erstellten Signaturen ist immer ein in der Signatur  
4048 eingebetteter Zeitpunkt vorhanden, jedoch kein qualifizierter Zeitstempel, da in der TI  
4049 keine qualifizierten Zeitstempel ausgestellt werden. Sollte ein Dokument mit einem  
4050 qualifizierten Zeitstempel versehen sein, so wird dieser nicht für die Ermittlung des  
4051 Signaturzeitpunktes herangezogen.

4052 **Benutzerdefinierter\_Zeitpunkt:** Vom Benutzer beim Aufruf der Signaturprüfoperation  
4053 als Parameter an den Konnektor übergebener Zeitpunkt, zu dem eine Signatur geprüft  
4054 werden soll.

## 4055 4.1.8.1.4 Jobnummer

4056 Da die eHealth-Kartenterminals dezentral über eine Netzwerkschnittstelle am Konnektor  
4057 betrieben werden, fehlt die Möglichkeit zur direkten physischen und vom Anwender  
4058 kontrollierbaren Zuordnung eines solchen Terminals zu einem Arbeitsplatz, auf dem sich  
4059 das Clientsystem befindet.

4060 Daher ist es bei einer fehlerhaften Zuordnung eines eHealth-Kartenterminals zu einem  
4061 Arbeitsplatz möglich, dass die PIN-Eingabeaufforderung – beispielsweise zu einem  
4062 Signaturauftrag – an ein entferntes Kartenterminal weitergeleitet wird. Diese fehlerhafte  
4063 Zuordnung kann durch einen Fehler des Clientsystems oder den Versuch eines Angriffes  
4064 hervorgerufen werden.

4065 Die Jobnummern werden vom Konnektor erzeugt und können durch Clientsystem oder  
4066 Signaturproxy abgerufen werden. Der Konnektor stellt jedoch keine Verbindung zwischen  
4067 erzeugten und verwendeten Jobnummern her. Es wird also nicht geprüft, ob nur  
4068 Jobnummern verwendet werden, die vorher vom Konnektor erzeugt wurden, oder ob alle  
4069 Jobnummern verwendet werden, die vom Konnektor erzeugt wurden.

#### 4070 **TIP1-A\_4639 - Generierung von Jobnummern für PIN-Eingaben**

4071 Um Fehler- und Angriffsmöglichkeiten auszuschließen, MUSS der Konnektor bei  
4072 bestimmten PIN-Verifikationen vor der Aufforderung zur PIN-Eingabe an einem eHealth-  
4073 Kartenterminal eine hinreichend eindeutige Nummer – die Jobnummer – generieren,  
4074 welche den Auftrag kennzeichnet, für dessen Verarbeitung die PIN-Eingabe erfolgen soll.  
4075 Bei welchen PIN-Verifikationen dies der Fall ist, kann den PIN-Prompts in TAB\_KON\_090  
4076 Terminalanzeigen beim Eingeben der PIN am Kartenterminal entnommen werden.

4077 [ $\leq$ ]

#### 4078 **TIP1-A\_4640 - Anzeige der Jobnummern für PIN-Eingaben**

4079 Diese Jobnummer MUSS vom Konnektor im Display des eHealth-Kartenterminals neben  
4080 der PIN-Eingabeaufforderung angezeigt werden.

4081 [ $\leq$ ]

#### 4082 **TIP1-A\_4992 - Guidance zur Jobnummer**

4083 Das Handbuch des Konnektors MUSS den Benutzer über den korrekten Gebrauch der  
4084 Jobnummer informieren. Es MUSS ihm verdeutlichen, dass er seine PIN über die Tastatur  
4085 des eHealth-Kartenterminals nur eingeben darf, wenn am Signaturproxy bzw.  
4086 Primärsystem und am Display des Kartenterminals die gleiche Jobnummer angezeigt  
4087 wird. Stimmen die beiden Nummern nicht überein, so soll der Benutzer seine PIN nicht  
4088 eingeben und stattdessen weitergehende Schritte zur Klärung des aufgetretenen  
4089 Fehlverhaltens einleiten.

4090 [ $\leq$ ]

#### 4091 **TIP1-A\_4642 - Ableitung der Jobnummer von einem Zufallswert**

4092 Zur hinreichend eindeutigen Kennzeichnung des Vorganges MUSS eine Jobnummer von  
4093 einem Zufallswert abgeleitet sein, wobei die Vorgaben an einen solchen Zufallswert  
4094 beachtet werden MÜSSEN [gemSpec\_Krypt#2.2].

4095 [ $\leq$ ]

#### 4096 **TIP1-A\_4643 - Beschaffenheit der Jobnummer**

4097 Zur Wahrung der Benutzerfreundlichkeit MUSS eine Reduzierung der Jobnummer auf eine  
4098 Länge von sechs Zeichen erfolgen. Diese sechs Zeichen MÜSSEN in zwei Zeichengruppen  
4099 mit je drei Zeichen, getrennt durch einen Bindestrich (0x2D), dargestellt werden. Die  
4100 erste Zeichengruppe MUSS ausschließlich die Zeichen "A-Z" beinhalten, die zweite  
4101 Zeichengruppe MUSS aus Ziffern "0-9" bestehen. Die Länge der resultierenden,  
4102 reduzierten Jobnummer ist sieben und wird durch den Umfang der darstellbaren Zeichen  
4103 auf dem Display des eHealth-Kartenterminals beschränkt.

4104 [ $\leq$ ]

#### 4105 **TIP1-A\_4644 - Jobnummer über 1.000 Vorgänge eindeutig**

4106 Der Konnektor MUSS die Eindeutigkeit einer Jobnummer sicherstellen:

- 4107 • Bei Aufruf der Operation GetJobnumber MUSS der Konnektor innerhalb von 1000  
4108 Aufrufen eine eindeutige Jobnummer generieren. Die Zählung der Aufrufe erfolgt  
4109 dabei unabhängig vom Aufrufkontext.



- Wird die Operation SignDocument mit einer Jobnummer aufgerufen, die innerhalb der vorangegangenen 1.000 Vorgänge verwendet wurde, so MUSS der Konnektor die Bearbeitung mit dem Fehler 4252 abbrechen. Die Zählung der Aufrufe erfolgt dabei unabhängig vom Aufrufkontext.

4114 [**<=**]

## 4115 **TIP1-A\_4645 - Zeichen der Jobnummer**

4116 Die einzelnen Zeichen der Jobnummer MÜSSEN für die Anzeige am Kartenterminal gemäß dem Zeichensatz ISO 646DE/DIN66003, bzw. ISO 646 US codiert werden. Aus diesem Zeichensatz dürfen nur die Zeichen „A-Z“ (0x41 bis 0x5A) und die Ziffern „0-9“ (0x30 bis 0x39) für die Anzeige der Jobnummer verwendet werden.

4120 [**<=**]

4121 Beispiele für eine Jobnummer sind ABC-475 und HZF-696.

4122 Die Einbettung der Jobnummer in den Nachrichtentext für den Bildschirm des Kartenlesers wird in TAB\_KON\_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal beschrieben.

## 4125 *4.1.8.1.5 Komfortsignatur*

4126 Für die QES unterstützt der Konnektor die Komfortsignaturfunktion. In diesem Modus können für ein- und denselben HBA mehrere vom Clientsystem initiierte Signaturaufträge (Einzel- oder Stapelsignatur) abgearbeitet werden, ohne dass der Inhaber des HBA für jeden einzelnen dieser Signaturaufträge die PIN.QES am Kartenterminal eingeben muss.

4131 Im Auslieferungszustand ist die Komfortsignaturfunktion ausgeschaltet (SAK\_COMFORT\_SIGNATURE = Disabled), d. h. mit dem Konnektor können zunächst keine Komfortsignaturen durchgeführt werden. Die Komfortsignaturfunktion kann vom Administrator eingeschaltet werden. Dies ist nur möglich, wenn an der Clientsystemschnittstelle des Konnektors verpflichtend TLS mit Clientauthentisierung (Konfigurationsvariante SOAP1 und SOAP2 in TAB\_KON\_852) konfiguriert ist. Das Einschalten der Komfortsignaturfunktion im Konnektor hat zur Folge, dass alle Operationen an der Clientsystemschnittstelle nur über TLS mit Clientauthentisierung angesprochen werden können (außer ggf. Dienstverzeichnisdienst).

4140 Bei eingeschalteter Komfortsignaturfunktion können potentiell alle HBAs in der Umgebung, in der der Konnektor eingesetzt ist, Komfortsignaturen durchführen. Die eigentliche Aktivierung der Komfortsignatur muss separat für jeden einzelnen HBA erfolgen.

4144 Durch Aufruf der Operation ActivateComfortSignature des Konnektors durch das Primärsystem wird die Nutzung der Komfortsignatur für einen HBA (Komfortsignaturmodus) aktiviert. Dazu muss der HBA-Inhaber die PIN.QES eingeben.

4147 Der Konnektor merkt sich für die Cardsession des HBA, dass die Komfortsignatur aktiviert wurde. Bei den folgenden Aufrufen von SignDocument werden dann Komfortsignaturen ausgeführt, solange bis eines der folgenden Abbruchkriterien eintritt:

- Die vom HBA (entsprechend Personalisierung) oder die vom Konnektor (entsprechend Konfiguration SAK\_COMFORT\_SIGNATURE\_MAX) durchgesetzte maximale Anzahl von Signaturen wurde erreicht.
- Das konfigurierte Zeitintervall für die Komfortsignatur (entsprechend Konfiguration SAK\_COMFORT\_SIGNATURE\_TIMER) ist für die Cardsession abgelaufen.
- Der Komfortsignaturmodus wurde für die betroffene Cardsession deaktiviert.



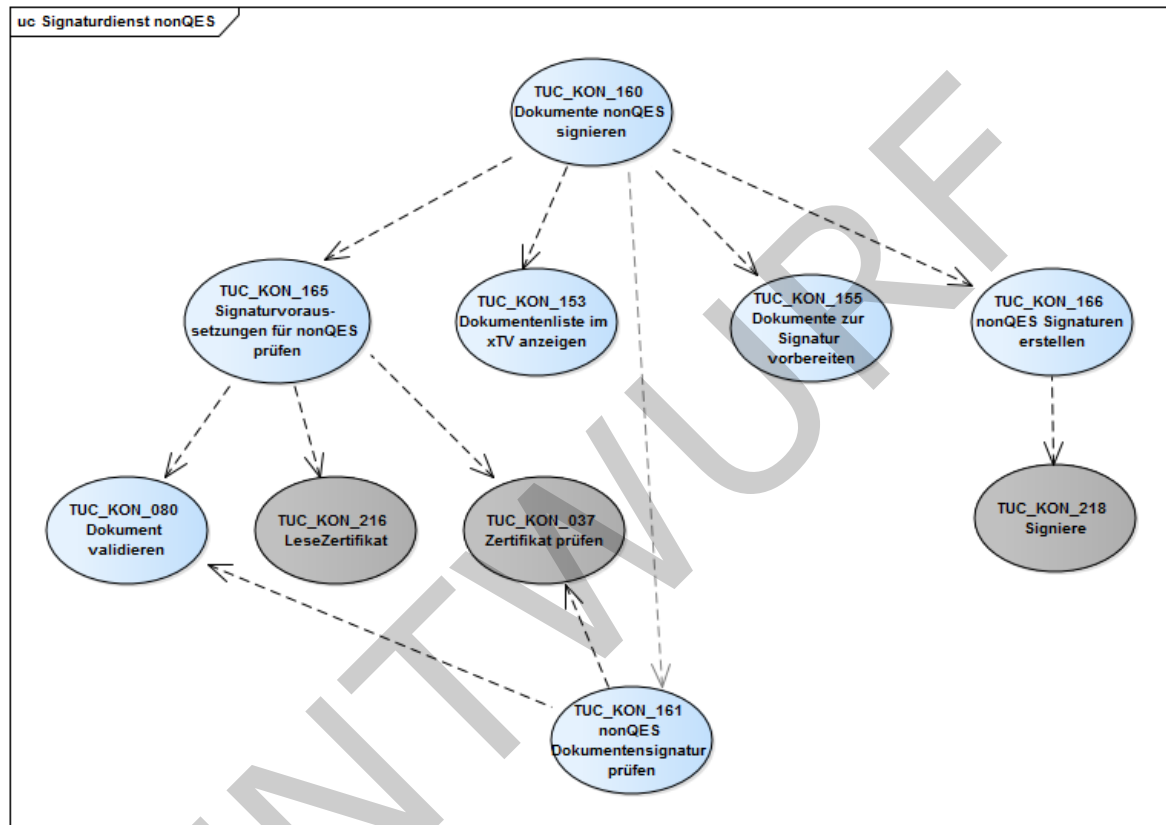
- 4156 • Der HBA wurde gezogen.
- 4157 • Der Sicherheitszustand des HBA wurde zurückgesetzt.
- 4158 • Die Komfortsignaturfunktion wurde für den Konnektor durch den Administrator deaktiviert.
- 4159 **A\_19945 - Unterstützte Signaturvarianten bei Komfortsignatur**
- 4160 Der Signaturdienst MUSS bei der Komfortsignatur die Signaturvarianten für die QES
- 4161 gemäß TAB\_KON\_778 unterstützen. [ <= ]
- 4162 **A\_18597 - Sicherheitszustand der PIN.QES bei Komfortsignatur**
- 4163 Bei der Komfortsignatur DARF der Konnektor den Sicherheitszustand der PIN.QES NICHT
- 4164 selbsttätig zurücksetzen, außer wenn dies explizit spezifikatorisch gefordert wird. [ <= ]
- 4165 A\_18597 kann z. B. umgesetzt werden, indem
- 4166 • ein dedizierter logischer Kanal des HBA für die Komfortsignatur verwendet
- 4167 wird und
- 4168 • im dedizierten logischen Kanal des HBA die Selektion von DF.QES solange
- 4169 beibehalten wird, bis ein Verlassen von DF.QES durch die Spezifikation explizit
- 4170 gefordert wird.
- 4171 **A\_18686 - Komfortsignatur-Timer**
- 4172 Der Konnektor MUSS für jede HBA-Kartensitzung mit eingeschalteter Komfortsignatur
- 4173 einen Komfortsignatur-Timer gemäß konfiguriertem Zeitintervall
- 4174 SAK\_COMFORT\_SIGNATURE\_TIMER einrichten und nach Erreichen des Maximalwerts den
- 4175 Sicherheitszustand des HBA zurücksetzen. [ <= ]
- 4176 **A\_19100 - Komfortsignatur-Zähler**
- 4177 Der Konnektor MUSS für jeden gesteckten HBA mit eingeschalteter Komfortsignatur die an
- 4178 die Karte gesendeten Signaturaufträge zählen und nach Erreichen des Maximalwerts den
- 4179 Sicherheitszustand des HBA zurücksetzen. [ <= ]
- 4180 **A\_19258 - Secure Messaging bei Komfortsignatur**
- 4181 Bei der Komfortsignatur MUSS der Signaturdienst die zu signierenden Daten (DTBS) über
- 4182 Secure Messaging vom Konnektor zum HBA übertragen. Dieser Secure Messaging-Kanal
- 4183 MUSS über die gSMC-K zum HBA mittels C.SAK.AUTD\_CVC aufgebaut werden. [ <= ]
- 4184 **A\_20073 - Prüfung der Länge der UserId**
- 4185 Der Konnektor MUSS die beim Aktivieren des Komfortsignaturmodus vom PS übermittelte
- 4186 UserId für die Kartensitzung des HBA, für den der Modus aktiviert wird, auf die
- 4187 ausreichende Länge von 128 Bit prüfen und die Aktivierung mit Fehler 4272 ablehnen,
- 4188 wenn die UserId nicht ausreichend lang ist. [ <= ]
- 4189 **A\_20074 - UserId über 1.000 Vorgänge eindeutig**
- 4190 Der Konnektor MUSS die Eindeutigkeit der UserId sicherstellen. Wird die Operation
- 4191 ActivateComfortSignature mit einer UserId im Aufrufkontext aufgerufen, die innerhalb
- 4192 der vorangegangenen 1.000 Vorgänge bereits verwendet wurde, so MUSS der Konnektor
- 4193 die Bearbeitung mit dem Fehler 4270 abbrechen. Die Zählung der Aufrufe erfolgt dabei
- 4194 unabhängig vom Aufrufkontext. [ <= ]
- 4195 **A\_19101 - Handbuch-Hinweis zu Nutzerauthentisierung am Clientsystem bei**
- 4196 **Komfortsignatur**
- 4197 Das Handbuch des Konnektors MUSS einen Hinweis enthalten, dass die Authentifizierung
- 4198 des HBA-Inhabers für die Komfortsignatur vom Clientsystem vorgenommen wird und
- 4199 dass die Authentifizierung des Nutzers am Clientsystem einen unverzichtbaren Beitrag
- 4200 zur Sicherheit der Lösung leistet. [ <= ]
- 4201

#### 4.1.8.2 Durch Ereignisse ausgelöste Reaktionen

keine

#### 4.1.8.3 Interne TUCs, nicht durch Fachmodule nutzbar

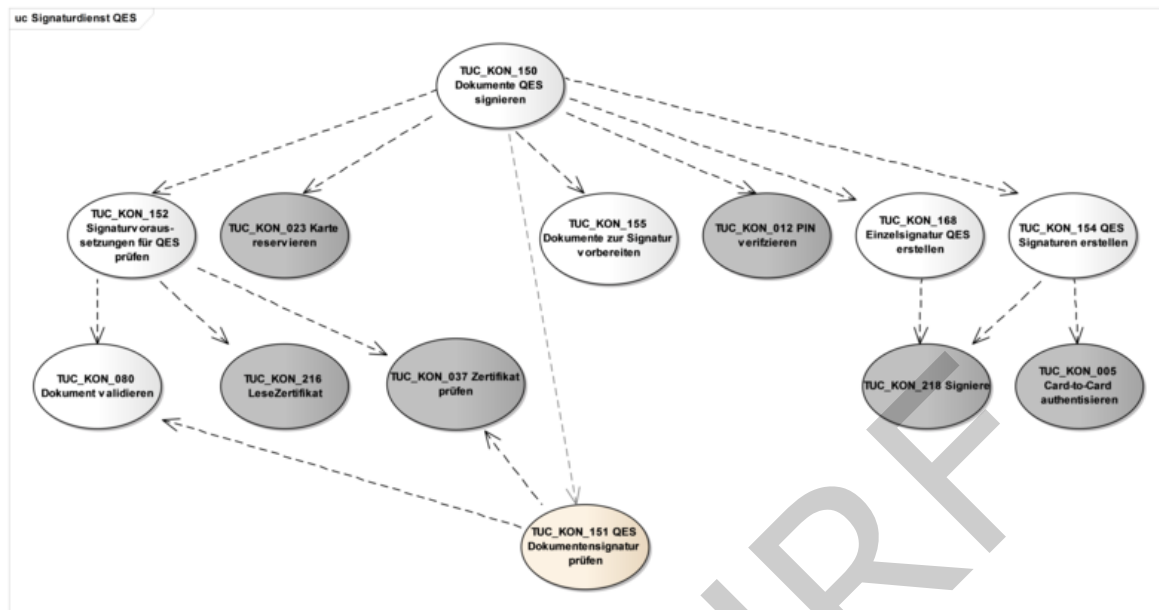
Abbildung PIC\_KON\_103 Use Case Diagramm Signaturdienst (nonQES) beschreibt die Aufrufbeziehungen der nonQES-TUCs des Signaturdienstes. Die TUCs des Signaturdienstes sind weiß dargestellt. Genutzte TUCs anderer Basisdienste sind grau hinterlegt.



**Abbildung 15: PIC\_KON\_103 Use Case Diagramm Signaturdienst (nonQES)**

Abbildung PIC\_KON\_104 Use Case Diagramm Signaturdienst (QES) beschreibt die Aufrufbeziehungen der QES-TUCs des Signaturdienstes.

4213

4214  
4215

4216

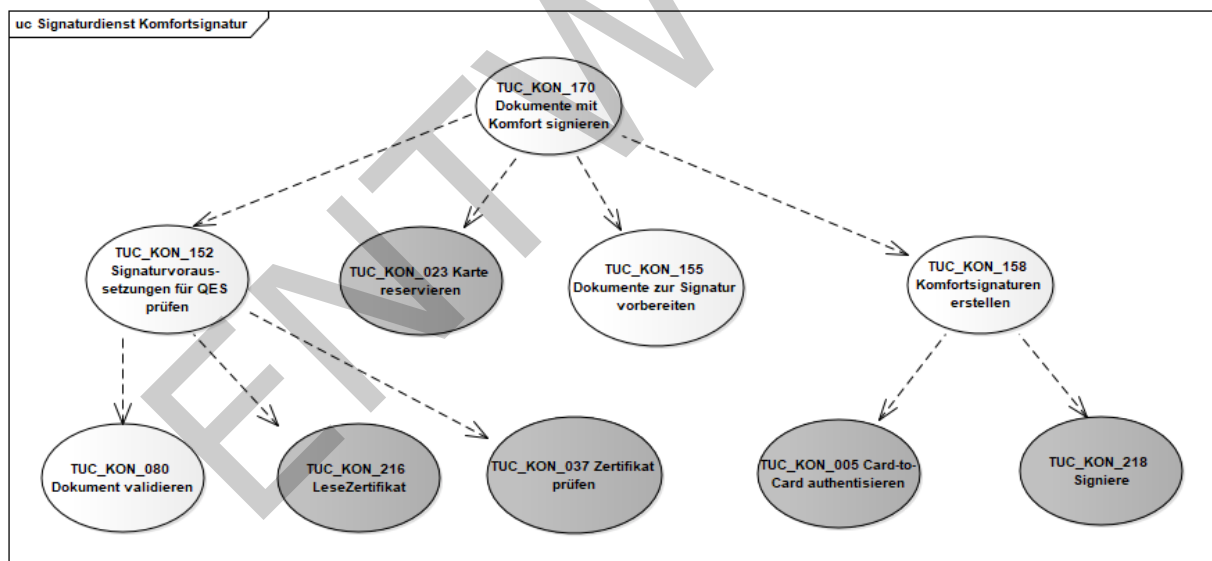
Abbildung 16: PIC\_KON\_104 Use Case Diagramm Signaturdienst (QES)

4217

Abbildung PIC\_KON\_102 Use Case Diagramm Signaturdienst (Komfortsignatur)

4218

beschreibt die Aufrufbeziehungen der TUCs des Signaturdienstes für die Komfortsignatur.



4219

4220

Abbildung 17: PIC\_KON\_102 Use Case Diagramm Signaturdienst (Komfortsignatur)

4221

4.1.8.3.1 TUC\_KON\_155 „Dokumente zur Signatur vorbereiten“

4222

**TIP1-A\_4646-02 - ab PTV4: TUC\_KON\_155 „Dokumente zur Signatur vorbereiten“**

4223

Der Konnektor MUSS den technischen Use Case TUC\_KON\_155 „Dokumente zur Signatur vorbereiten“ umsetzen.

4225

4226

4227 **Tabelle 191: TAB\_KON\_748 - TUC\_KON\_155 „Dokumente zur Signatur vorbereiten“**

| Element          | Beschreibung   |
|------------------|--|
| Name             | TUC_KON_155 „Dokumente zur Signatur vorbereiten“   |
| Beschreibung     | Die zu signierenden Dokumente werden entsprechend den Erfordernissen der Signaturverfahren für die QES oder nonQES vorbereitet.  |
| Anwendungsumfeld | Erstellung von qualifizierten elektronischen Signaturen (QES) und nicht-qualifizierten elektronischen Signaturen (nonQES)  |
| Auslöser         | Aufruf durch TUC_KON_150 „Dokumente QES signieren“ oder TUC_KON_160 „Dokumente nonQES signieren“   |
| Vorbedingungen   | keine  |
| Eingangsdaten    | <ul style="list-style-type: none"> <li>- signatureMode (Signaturart: QES   nonQES)</li> <li>- documentsToBeSigned (Zu signierendes Dokument bzw. zu signierende Dokumente) und pro Dokument:</li> <li>- documentFormat (Formatangabe für das zu signierende Dokument)</li> <li>- optionalInputs (weitere optionale Eingabeparameter zur Steuerung der Details bei der zu erstellenden Signatur (siehe Operation SignDocument, Parameter dss:OptionalInputs), darin u.a.</li> <li>- signatureType (URI für den Signatortyp XML-, CMS-, S/MIME- oder PDF-Signatur)</li> <li>- certificate (Signaturzertifikat)</li> <li>- ocspsResponses – <i>optional</i> (OCSP-Response des EE-Zertifikats, das bei der Signaturerstellung in die Signatur eingebettet wird.)</li> </ul> |
| Komponenten      | Konnektor  |
| Ausgangsdaten    | <ul style="list-style-type: none"> <li>• preProcessedDocuments (Aufbereitetes zu signierendes Dokument bzw. aufbereitete zu signierende Dokumente)</li> </ul>  |
| Standardablauf   | signatureType = XMLDSig (XAdES)<br>Entsprechend den Regeln für die QES und die nonQES werden zunächst weitere Signatureigenschaften zum jeweiligen Dokument in Form von <i>QualifyingProperties</i> (siehe [XAdES]) hinzugefügt. Die Systemzeit des Anwendungskonnektors muss in das XML-Element   |

|  |  |
|--|--|
|  | <p>SigningTime (siehe [XAdES]) eingetragen werden. Die Signatur wird anschließend entsprechend [XMLDSig] vorbereitet. D. h., es wird je Dokument nach Erzeugung der Reference Elemente das SignedInfo Element aufgebaut. Dessen Inhalt ergibt dann nach erfolgter XML-Kanonisierung und Hashing die DTBS (Data To Be Signed), die später zur Karte gesendet werden.</p> <p>certificate wird im Element <code>ds:KeyInfo/ds:X509Data</code> gespeichert.</p> <p>Im Fall <code>signatureMode = QES</code> können neben den reinen Nutzdaten auch alle weiteren Elemente in die Signatur einbezogen werden, die für die Rekonstruktion der ursprünglich dargestellten Daten in der sicheren Anzeige erforderlich sind. Für XML-Dokumente sind das, falls vorhanden, das/die XML-Schema(ta). Für diese werden Referenzen (Hash + URI) in die Signatur eingebettet.</p> <p>Die URI ist im Fall übergebener XML-Schemata der übergebene <i>signatureType</i> - Parameter. Die URI ist im Fall der im Konnektor im Rahmen einer Signaturreichtlinie hinterlegten XML-Schemata/XSL-Stylesheets die URI der Signaturreichtlinie, ergänzt um den Dateinamen mit Pfad, wie in der Signaturreichtlinie festgelegt.</p> <p>(Beispiel: URI für Schemadatei <code>NFD_Document.xsd</code> der Signaturreichtlinie <code>SR_DF_NFDM_NOTFALLDATEN</code> lautet: <code>urn:gematik:fa:sak:nfdm:r1:v1:NFD_Document.xsd</code>) Das Einbetten der Referenzen erfolgt über das XML-Element <code>ds:object/ds:manifest (XMLDSig)</code> mit eingebetteten XML-Elementen <code>ds:Reference</code>, die eine URI (RefURI) als Identifier für die jeweilige Datei und einen Hash über die jeweilige Resource enthalten. Der ShortTextClientsystem muss in die Signatur in das <code>DataObjectFormat/Description-Element</code> gemäß [XAdES] (Abschnitt 7.2.5) eingebettet werden.</p> <p>Falls durch den Aufrufparameter <code>SIG:IncludeRevocationInfo</code> angefordert, wird die für die Offline-Prüfung notwendige OCSP-Antwort im Sinne vom ES-X-L vom Konnektor in die Signatur eingebettet: Die base-64 kodierte OCSP-Response wird im Feld</p> |
|--|--|

|                            |   |
|----------------------------|---|
|                            | <p>QualifyingProperties/UnsignedProperties/UnsignedSignatureProperties/RevocationValues/OCSPValues/EncapsulatedOCSPValue (selbst DER-kodiert) gespeichert.</p> <p>signatureType = CMS (CAvES)<br/>         Etwaig einzubettende XML-Schemata werden zunächst wie für XAdES definiert in ein ds:manifest-Element eingebettet. Die so erzeugte Zeichenkette wird als genau ein ASN.1 Character String vom Typ UTF8String verpackt. Dieser wird als contentDescription in einen Content-Hints Attributwert vom Typ ContentHints verpackt, wobei der contentType=id-data gemäß [CAvES]. Der ShortTextClientsystem muss in die Signatur in das content-hints.ContentDescription-Attribut gemäß [CAvES] (Abschnitt 5.10.3) eingebettet werden.</p> <p>Ist die Einbettung von OCSP-Responses gefordert, wird die für die Offline-Prüfung notwendige OCSP-Antwort des EE-Zertifikats im Attribut SignedData.crls.other abgelegt.</p> <p>signatureType = PDF/A (PAvES)<br/>         Der ShortTextClientsystem muss bei einer PDF-Signatur in das Reason-Feld eingebettet werden.</p> <p>OCSP-Responses werden bei PAvES nicht eingebettet.</p> <p>Es sind die Vorgaben zum Signaturprofil gemäß Tabelle TAB_KON_779 „Profilierung der Signaturformate“ zu erfüllen.</p> <p>Die aufbereiteten zu signierenden Dokumente werden an den Aufrufer zurückgegeben.</p> |
| Varianten/<br>Alternativen | keine   |
| Fehlerfälle                | <p>Das Verhalten des TUCs bei einem Fehlerfall ist in TAB_KON_586 Fehlercodes TUC_KON_155 „Dokumente zur Signatur vorbereiten“ „PDF/A (PAvES)“</p> <p>Es ist nicht genügend Speicherplatz im PDF-Dokument verfügbar: 4205</p>   |

|                                |       |
|--------------------------------|-------|
| Nichtfunktionale Anforderungen | keine |
| Zugehörige Diagramme           | keine |

4228

4229 **Tabelle 192: TAB\_KON\_586 Fehlercodes TUC\_KON\_155 „Dokumente zur Signatur vorbereiten“**

4230

| Fehlercode | ErrorType | Severity | Fehlertext   |
|------------|-----------|----------|--|
| 4205       | Technical | Error    | Es ist nicht genügend Speicherplatz im PDF-Dokument verfügbar. |

4231

4232 [ $\leq$ ]

4233

4234 4.1.8.3.2 TUC\_KON\_165 „Signaturvoraussetzungen für nonQES prüfen“

4235 **TIP1-A\_4647-02 - TUC\_KON 165 „Signaturvoraussetzungen für nonQES prüfen“**

4236 Der Konnektor MUSS den technischen Use Case „Signaturvoraussetzungen für nonQES prüfen“ umsetzen.

4237

4238

4239 **Tabelle 193: TAB\_KON\_749 – TUC\_KON\_165 „Signaturvoraussetzungen für nonQES prüfen“**

4240

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_165 „Signaturvoraussetzungen für nonQES prüfen“  |
| Beschreibung   | Es werden die Voraussetzungen an die zu signierenden Dokumente und das Signaturzertifikat geprüft. Es werden die nonQES_DocFormate unterstützt.  |
| Auslöser       | TUC_KON_160 „Dokumente nonQES signieren“   |
| Vorbedingungen | keine  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• Zu signierende Dokumente</li> <li>• optionale Eingabeparameter zur Steuerung der Details der Signaturerstellung</li> <li>• cardSession Signaturkarte</li> <li>• zu verwendende Identität (Zertifikatsreferenz)</li> </ul> |
| Komponenten    | Konnektor, Kartenterminal, Signaturkarte   |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>• Prüfergebnis</li> <li>• Signaturzertifikat</li> </ul>   |
| Standardablauf | 1. Abhängig von seinem Typ werden für jedes Dokument die typabhängigen Validierungsschritte (ohne Prüfung auf sichere Anzeigbarkeit) durchgeführt. Dies geschieht durch Aufruf von TUC_KON_080 „Dokument validieren“.  |

|                                |  |
|--------------------------------|--|
|                                | <p>Wird der Aufruf von TUC_KON_080 mit einem Fehler beendet, wird die Prüfung im laufenden TUC mit diesem Fehler abgebrochen.</p> <p>2. Durch Aufruf von TUC_KON_216 „Lese Zertifikat“ wird das Signaturzertifikat von der Signaturkarte gelesen.</p> <p>3. Das Signaturzertifikat wird durch Aufruf von TUC_KON_037 „Zertifikat prüfen“{<br/> certificate = Zertifikatsreferenz;<br/> qualifiedCheck = not_required;<br/> offlineAllowNoCheck = true;<br/> validationMode = OCSP}<br/> geprüft.</p> |
| Varianten/Alternativen         | Keine  |
| Fehlerfälle                    | Keine  |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | keine  |

**Tabelle 194: TAB\_KON\_587 Fehlercodes TUC\_KON\_165 „Signaturvoraussetzungen für nonQES prüfen“**

| Fehlercode  | ErrorType | Severity | Fehlertext |
|---|-----------|----------|------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können keine weiteren Fehlercodes auftreten. |           |          |            |

[<=]

#### 4.1.8.3.3 TUC\_KON\_166 „nonQES Signaturen erstellen“

##### **TIP1-A\_4648 - TUC\_KON\_166 „nonQES Signaturen erstellen“**

Der Konnektor MUSS den technischen Use Case TUC\_KON\_166 „nonQES Signaturen erstellen“ umsetzen.

**Tabelle 195: TAB\_KON\_750 – TUC\_KON\_166 „nonQES Signaturen erstellen“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_166 „nonQES Signaturen erstellen“  |
| Beschreibung   | Die Data To Be Signed (DTBS) werden erzeugt, an die Signaturkarte gesendet und dort signiert.  |
| Auslöser       | TUC_KON_160 „Dokumente nonQES signieren“   |
| Vorbedingungen | keine  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>Liste der zu signierenden Dokumente</li> <li>cardSession Signaturkarte</li> <li>zu verwendende Identität (Zertifikatsreferenz)</li> </ul> |



|                                |   |
|--------------------------------|---|
|                                | <ul style="list-style-type: none"> <li>crypt [SIG_CRYPT_nonQES]: <i>optional</i>; <i>default</i> und Wertebereich: SIG_CRYPT_DEFAULT siehe TAB_KON_863 (Dieser Parameter steuert, ob RSA-basierte oder ECC-basierte Signaturen erzeugt werden.)</li> </ul>  |
| Komponenten                    | Konnektor, Kartenterminal, Signaturkarte  |
| Ausgangsdaten                  | <ul style="list-style-type: none"> <li>Signierte Dokumente</li> </ul>   |
| Standardablauf                 | <p>Die folgenden Schritte werden für jedes Dokument der Liste durchgeführt.</p> <ol style="list-style-type: none"> <li>Das zu signierende Dokument wird, soweit noch nicht erfolgt, für die XML-Signatur vorbereitet. Ein Ergebnis dieser Vorbereitung sind die Data To Be Signed (DTBS): der Hash-Wert (Digest des SignedInfo-Elementes), der zur Signatur an die Karte gesendet werden soll. Für XML-Signaturen müssen die Vorgaben aus [gemSpec_Krypt#3.1.1] beachtet werden.</li> <li>Für das zu signierende Dokument werden die DTBS zur Signatur an die Signaturkarte übermittelt (Aufruf von TUC_KON_218 „Signiere“). Dabei werden der Schlüssel und der Algorithmusidentifizierer über die Tabelle TAB_KON_900 bestimmt.</li> <li>Die erstellte Signatur wird mathematisch geprüft.</li> <li>Der ermittelte Signaturwert wird in die zuvor vorbereitete XML-Signatur eingefügt.</li> <li>Der Konnektor löst TUC_KON_256 {"SIG/SIGNDOC/NEXT_SUCCESSFUL"; Op; Info; „\$Jobnummer“; noLog; \$doInformClients } aus.</li> </ol> |
| Varianten/Alternativen         | keine   |
| Fehlerfälle                    | Fehler in den folgenden Schritten des Standardablaufs führen zum Abbruch mit den ausgewiesenen Fehlercodes: (→3) Fehlgeschlagene mathematische Prüfung der Signatur: 4120   |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | keine   |

4250

**Tabelle 196: TAB\_KON\_120 Fehlercodes TUC\_KON\_166 „nonQES Signaturen erstellen“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |              |
| 4120  | Security  | Error    | Kartenfehler |

4251  
4252 [ $\leq$ ]

4253 4.1.8.3.4 TUC\_KON\_152 "Signaturvoraussetzungen für QES prüfen"

4254 **TIP1-A\_4649 - TUC\_KON\_152 „Signaturvoraussetzungen für QES prüfen“**

4255 Der Konnektor MUSS den technischen Use Case TUC\_KON\_152

4256 „Signaturvoraussetzungen für QES prüfen“ umsetzen.

4257

4258 **Tabelle 197: TAB\_KON\_751 – TUC\_KON\_152 „Signaturvoraussetzungen für QES prüfen“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_152 „Signaturvoraussetzungen für QES prüfen“   |
| Beschreibung   | Es werden die Voraussetzungen an die zu signierenden Dokumente und das Signaturzertifikat geprüft. Es werden die <code>QES_DocFormate</code> unterstützt.  |
| Auslöser       | TUC_KON_150 „Dokumente QES signieren“  |
| Vorbedingungen | keine  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• Zu signierende Dokumente</li> <li>• optionale Eingabeparameter zur Steuerung der Details der Signaturerstellung</li> <li>• <code>cardSession</code> Signaturkarte</li> <li>• zu verwendende Identität (Zertifikatsreferenz)</li> <li>• <code>includeRevocationInfo</code> [Boolean] - optional; Default: <code>true</code><br/>(Dieser Parameter steuert die Einbettung von OCSP-Responses in die Signatur.<br/><code>true</code>: Die Sperrinformationen werden in <code>ocspResponses</code> zurückgegeben.)</li> </ul> |
| Komponenten    | Konnektor, Kartenterminal, Signaturkarte   |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>• Prüfergebnis</li> <li>• Signaturzertifikat</li> <li>• <code>ocspResponses</code> - optional/nur wenn <code>includeRevocationInfo = true</code><br/>(OCSP-Response des EE-Zertifikats, die beim Aufruf von TUC_KON_037 „Zertifikat prüfen“ zurückgegeben wird)</li> </ul>  |
| Standardablauf | <ol style="list-style-type: none"> <li>1. Abhängig von seinem Typ werden für jedes Dokument die typabhängigen Dokumentvalidierungsschritte durchgeführt (Aufruf TUC_KON_080 „Dokument validieren“). Wird der Aufruf von TUC_KON_080 mit einem Fehler beendet, wird die Prüfung im laufenden TUC mit diesem Fehler abgebrochen.</li> </ol>  |

|                                |  |
|--------------------------------|--|
|                                | <p>2. Durch Aufruf von TUC_KON_216 „Lese Zertifikat“ wird das Signaturzertifikat von der Signaturkarte gelesen.</p> <p>3. Das Signaturzertifikat wird durch Aufruf von TUC_KON_037 „Zertifikat prüfen“ {<br/> certificate = Zertifikatsreferenz;<br/> qualifiedCheck = required;<br/> offlineAllowNoCheck = true;<br/> validationMode = OCSP;<br/> getOCSPResponses = includeRevocationInfo}<br/> geprüft.</p> |
| Varianten/Alternativen         | keine  |
| Fehlerfälle                    | (->3) Für MGM_LU_ONLINE=Enabled gilt:<br>Liefert die Zertifikatsprüfung (OCSP-Abfrage) die Warnung CERT_REVOKED oder CERT_UNKNOWN gemäß [gemSpec_PKI#Tab_PKI_274], dann wird der TUC mit Fehler 4123 abgebrochen.  |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | keine  |

**Tabelle 198: TAB\_KON\_588 Fehlercodes TUC\_KON\_152 „Signaturvoraussetzungen für QES prüfen“**

| Fehlercode  | ErrorType | Severity | Fehlertext |
|---|-----------|----------|------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können keine weiteren Fehlercodes auftreten. |           |          |            |

[<=]

#### 4.1.8.3.5 TUC\_KON\_154 "QES Signaturen erstellen"

Der TUC\_KON\_154 stellt den Standardsignaturfall in der TI, die Stapelsignatur dar (auch für Stapel der Größe 1). Da die Stapelsignatur auf der Zielkarte passende CVC voraussetzt, die auf den HBA-Vorläuferkarten nicht vorhanden sind, kann dieser TUC nur den HBA unterstützen. Für HBA-Vorläuferkarten kann TUC\_KON\_168 verwendet werden.

#### **TIP1-A\_4651 - TUC\_KON\_154 „QES Signaturen erstellen“**

Der Konnektor MUSS den technischen Use Case TUC\_KON\_154 „QES Signaturen erstellen“ umsetzen.

**Tabelle 199: TAB\_KON\_752 – TUC\_KON\_154 „QES Signaturen erstellen“**

| Element | Beschreibung                           |
|---------|--|
| Name    | TUC_KON_154 „QES Signaturen erstellen“ |

|                |  |
|----------------|--|
| Beschreibung   | Die Data To Be Signed (DTBS) werden erzeugt, an die Signaturkarte gesendet und dort signiert.  |
| Auslöser       | TUC_KON_150 „Dokumente QES signieren“  |
| Vorbedingungen | Die Ressourcen Signaturkarte und Kartenterminal sind für den Vorgang reserviert.<br>DF.QES ist selektiert.<br>PIN.QES ist initial verifiziert  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• Zu signierendes Dokument bzw. zu signierende Dokumente</li> <li>• cardSession (nur HBA erlaubt)</li> <li>• zu verwendende Identität (Zertifikatsreferenz)</li> <li>• crypt [SIG_CRYPT_QES] - <i>optional</i>; <i>default und Wertebereich</i>: siehe TAB_KON_862 (Dieser Parameter steuert, ob RSA-basierte oder ECC-basierte Signaturen erzeugt werden.)</li> <li>• WorkplaceId</li> </ul>   |
| Komponenten    | Konnektor, Kartenterminal, Signaturkarte (HBA)   |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>• Signierte Dokumente</li> </ul>  |
| Standardablauf | <p>Basierend auf SAK.AUTD_CVC und HPC.AUTD_SUK_CVC und den zugehörigen privaten Schlüsseln wird ein sicherer Kanal zwischen der gSMC-K des Konnektors und dem HBA aufgebaut mittels Aufruf TUC_KON_005 „Card-to-Card authentisieren“ {<br/> sourceCardSession = gSMC-K;<br/> targetCardSession = CardSession;<br/> authMode = „gegenseitig+TC“}</p> <p>Die folgenden Schritte werden für jedes Dokument des Stapels durchgeführt.</p> <ol style="list-style-type: none"> <li>1. Das zu signierende Dokument wird, soweit noch nicht erfolgt, für die Signatur gemäß des entsprechenden Formats vorbereitet. Ein Ergebnis dieser Vorbereitung sind die Data To Be Signed (DTBS): der Hash-Wert (Digest des SignedInfo-Elementes), der zur Signatur an die Karte gesendet werden soll.</li> <li>2. Für das zu signierende Dokument werden die DTBS zur Signatur im sicheren Kanal an den HBA übermittelt (Aufruf TUC_KON_218 „Signiere“). Dabei werden der Schlüssel und der Algorithmusidentifizier über die Tabelle TAB_KON_900 bestimmt.</li> <li>3. Falls Schritt 3 fehlgeschlagen ist, weil der PIN.QES-Nutzungszähler abgelaufen ist (erkennbar z. B. daran, dass die Karte einen Autorisierungsfehler zurückmeldet), wird die PIN.QES verifiziert (Aufruf TUC_KON_012 „PIN</li> </ol> |

|                                |  |
|--------------------------------|--|
|                                | <p>verifizieren", nachdem der im Konnektor verwaltete Sicherheitszustand (CARDSESSION.AUTHSTATE) aktualisiert wurde). Am Display des Kartenterminals wird dabei die Jobnummer für den Signaturvorgang angezeigt. Aus der WorkplaceId geht hervor, ob es sich um eine Remote-PIN-Eingabe handelt. Nach der PIN-Verifikation wird erneut die zuvor fehlgeschlagene Signatur in Schritt 3 ausgeführt.</p> <ol style="list-style-type: none"> <li>4. Die erstellte Signatur wird mathematisch geprüft.</li> <li>5. Der ermittelte Signaturwert wird in den zuvor vorbereiteten Signaturprototypen eingefügt.</li> <li>6. Der Konnektor löst TUC_KON_256 { "SIG/SIGNDOC/NEXT_SUCCESSFUL"; Op; Info; „\$Jobnummer"; noLog; \$doInformClients } aus.</li> </ol> |
| Varianten/<br>Alternativen     | <p>Alternativ zum Standardablauf kann zu Beginn die maximal erlaubte Stapelgröße SSEC durch Auslesen von EF.SSEC ermittelt werden. Der zu signierende Dokumentenstapel wird in Teilstapel von maximaler Größe SSEC zerlegt. Für jeden Teilstapel wird die PIN.QES verifiziert. Die Dokumente des Teilstapels werden wie im Standardablauf beschrieben signiert. Der Nutzer kann den Vorgang der PIN-Eingabe abbrechen.</p>   |
| Fehlerfälle                    | <p>(-&gt;2) Fehler im Signaturvorgang führen zum Abbruch des gesamten Signaturvorgangs, Fehlercode 4123<br/>         (-&gt;3) Fehler bei der PIN-Eingabe führen zum Abbruch des Signaturvorgangs<br/>         (-&gt;4) Fehler in mathematischer Prüfung der Signatur führen zum Abbruch des Signaturvorgangs, Fehlercode 4120<br/>         Das Verhalten des TUCs bei einem Fehlerfall, einem Timeout der PIN-Eingabe oder beim Abbruch durch den Benutzer ist in Tabelle TAB_KON_192 Verhalten des Konnektors beim Abbruch einer Stapelsignatur beschrieben.</p>  |
| Sicherheitsanforderungen       | <p>Zum Aufbau des sicheren Kanals bzw. zur Aushandlung des symmetrischen Schlüssels DARF DF.QES NICHT verlassen werden. Benötigte CVCs des HBA MÜSSEN also bereits vor dem Signaturvorgang eingelesen und gecacht werden. Dies KANN bereits beim Stecken des HBA geschehen.<br/>         Die in [gemSpec_Krypt#3.1.2] angegebenen Festlegungen der zu unterstützenden Algorithmen MÜSSEN berücksichtigt werden.</p>  |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | <p>Abbildung PIC_KON_113 Aktivitätsdiagramm zu „QES Signaturen erstellen“<br/>         Das Diagramm dient nur der Veranschaulichung und ist</p>  |

nicht vollständig. Beispielsweise enthält es nicht die Steuerung durch den Parameter crypt.

act TUC\_KON\_154 QES Signaturen erstellen

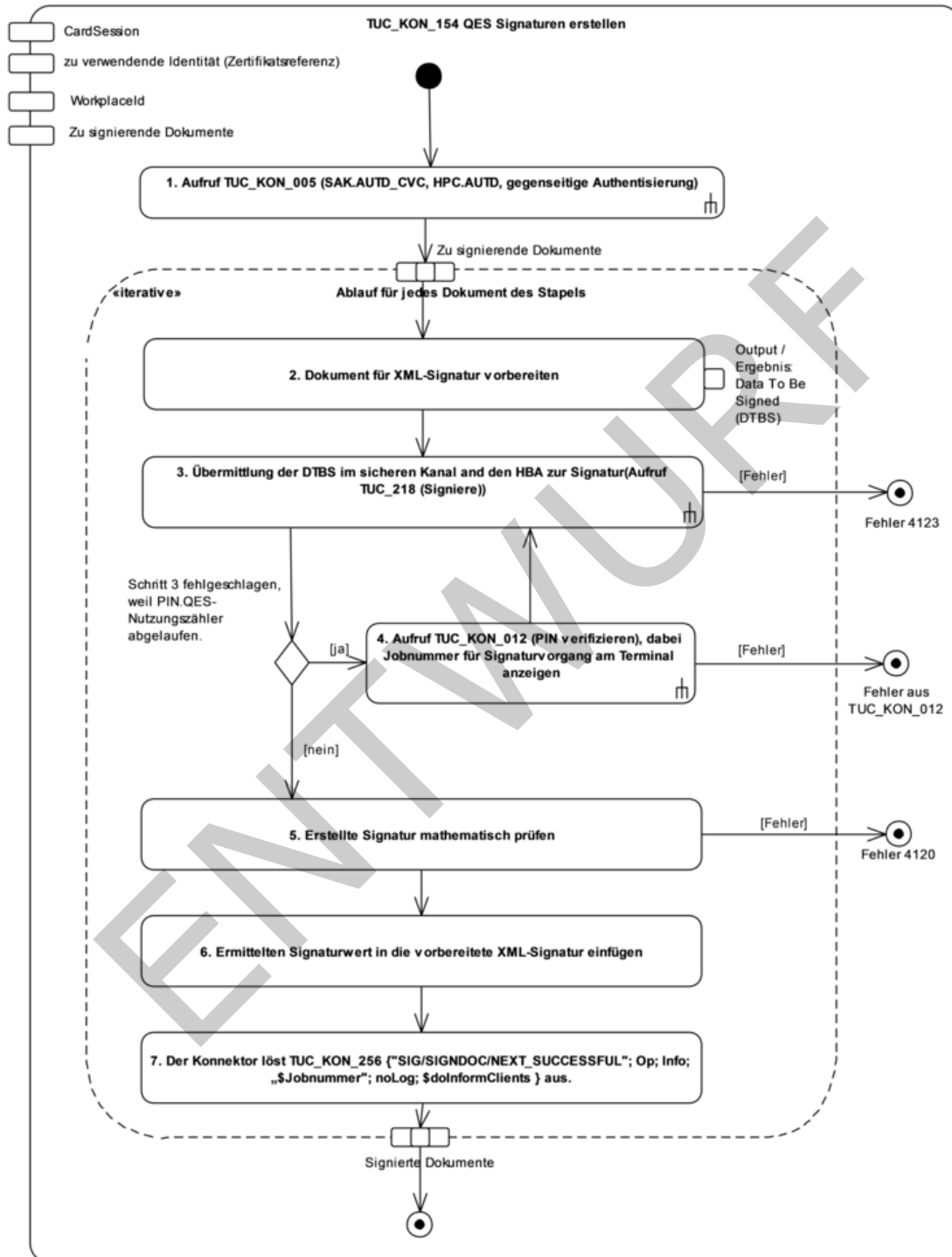


Abbildung 18: PIC\_KON\_113 Aktivitätsdiagramm zu „QES Signaturen erstellen“

4278 **Tabelle 200: TAB\_KON\_126 Fehlercodes TUC\_KON\_154 „QES Signaturen erstellen“**

| Fehlercode  | ErrorType | Severity | Fehlertext                    |
|---|-----------|----------|-------------------------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |                               |
| 4120  | Security  | Error    | Kartenfehler                  |
| 4123  | Security  | Error    | Fehler bei Signaturerstellung |

4279 [ $\leq$ ]

4280 4.1.8.3.6 TUC\_KON\_168 „Einzelsignatur QES erstellen“

4281 **TIP1-A\_4652 - TUC\_KON\_168 „Einzelsignatur QES erstellen“**

4282 Der Konnektor MUSS den technischen Use Case TUC\_KON\_168 „Einzelsignatur QES  
4283 erstellen“ umsetzen.

4284

4285 **Tabelle 201: TAB\_KON\_293 - TUC\_KON\_168 „Einzelsignatur QES erstellen“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_168 "Einzelsignatur QES erstellen"  |
| Beschreibung   | Es wird ein Dokument technisch mit einer Signatur versehen. Im Gegensatz zum TUC_KON_154 „QES Signaturen erstellen“ wird hier nur eine einzelne Signatur ohne vorhergehendes C2C erstellt. Die Übertragung der DTBS erfolgt ohne Secure Messaging.  |
| Auslöser       | TUC_KON_150 Dokumente QES signieren   |
| Vorbedingungen | Die Ressourcen Signaturkarte und Kartenterminal sind für den Vorgang reserviert.<br>DF.QES ist selektiert.  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• zu signierendes Dokument</li> <li>• CardSession (HBAX)</li> <li>• zu verwendende Identität (Zertifikatsreferenz)</li> <li>• crypt: [SIG_CRYPT_QES] - <i>optional</i>;<br/><i>default und Wertebereich</i>: siehe TAB_KON_862<br/>Dieser Parameter steuert, ob RSA-basierte oder ECC-basierte Signaturen erzeugt werden.</li> <li>• WorkplaceId</li> </ul>                        |
| Komponenten    | Konnektor, Kartenterminal, Signaturkarte (HBAX)   |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>• Signiertes Dokument</li> </ul>   |
| Standardablauf | <ol style="list-style-type: none"> <li>1. Das zu signierende Dokument wird, soweit noch nicht erfolgt, für die Signatur vorbereitet. Ein Ergebnis dieser Vorbereitung sind die DTBS: der Hash-Wert (Digest des SignedInfo-Elementes), der zur Signatur an die Karte gesendet werden soll.</li> <li>2. Für das zu signierende Dokument werden die DTBS zur Signatur an den HBAX übermittelt (Aufruf TUC_KON_218</li> </ol> |

|                                   |   |
|-----------------------------------|---|
|                                   | <p>„Signiere“). Dabei werden der Schlüssel und der Algorithmusidentifizier über die Tabelle TAB_KON_900 bestimmt.<br/>Jeder Fehler führt zum Abbruch des Signaturvorgangs</p> <p>3. Die erstellte Signatur wird mathematisch geprüft. Der ermittelte Signaturwert wird in den zuvor gemäß des entsprechenden Signaturformates vorbereiteten Signaturprototypen eingefügt.</p> |
| Varianten/<br>Alternativen        | keine   |
| Fehlerfälle                       | <p>Das Verhalten des TUCs bei einem Fehlerfall, einem Timeout der PIN-Eingabe oder beim Abbruch durch den Benutzer ist in Tabelle TAB_KON_192 Verhalten des Konnektors beim Abbruch einer Stapelsignatur beschrieben.<br/>(→3) Fehler in mathematischer Prüfung der Signatur: Abbruch mit 4120</p>  |
| Nichtfunktionale<br>Anforderungen | keine   |
| Zugehörige<br>Diagramme           | keine   |

4286 **Tabelle 202: TAB\_KON\_590 Fehlercodes TUC\_KON\_168 „Einzelsignatur QES erstellen“**

| Fehlercode  | ErrorType | Severit<br>y | Fehlertext   |
|---|-----------|--------------|--------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten. |           |              |              |
| 4120  | Security  | Error        | Kartenfehler |

4287  
4288 **[<=]**

4289 **4.1.8.3.7 TUC\_KON\_158 "Komfortsignaturen erstellen"**

4290 Der TUC\_KON\_158 führt die Komfortsignatur für ein Dokument oder mehrere Dokumente  
4291 eines Stapels aus. Da die Komfortsignatur auf der Zielkarte passende CVC voraussetzt,  
4292 die auf den HBA-Vorläuferkarten nicht vorhanden sind, unterstützt dieser TUC nur den  
4293 HBA.

4294 **A\_19102 - TUC\_KON\_158 „Komfortsignaturen erstellen“**

4295 Der Konnektor MUSS den technischen Use Case TUC\_KON\_158 „Komfortsignaturen  
4296 erstellen“ umsetzen.

4297

4298 **Tabelle 203: TAB\_KON\_870 – TUC\_KON\_158 „Komfortsignaturen erstellen“**

| Element      | Beschreibung  |
|--------------|---|
| Name         | TUC_KON_158 „Komfortsignaturen erstellen“   |
| Beschreibung | Die Data To Be Signed (DTBS) werden erzeugt, an die Signaturkarte gesendet und dort signiert. |



|                |  |
|----------------|--|
|                | Die Übertragung der DTBS erfolgt mit Secure Messaging.<br>Die Abarbeitung der Signatur erfolgt im SE#2.  |
| Auslöser       | TUC_KON_170 „Dokumente mit Komfort signieren“  |
| Vorbedingungen | Die Ressource Signaturkarte ist für den Vorgang reserviert.<br>DF.QES ist selektiert.<br>PIN.QES ist initial verifiziert   |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• Zu signierendes Dokument bzw. zu signierende Dokumente</li> <li>• cardSession (nur HBA erlaubt)</li> <li>• zu verwendende Identität (Zertifikatsreferenz)</li> <li>• crypt [SIG_CRYPT_QES] - <i>optional</i>; <i>default und Wertebereich</i>: siehe TAB_KON_862 (Dieser Parameter steuert, ob RSA-basierte oder ECC-basierte Signaturen erzeugt werden.)</li> <li>• WorkplaceId</li> </ul>   |
| Komponenten    | Konnektor, Kartenterminal, Signaturkarte (HBA)   |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>• Signierte Dokumente</li> </ul>  |
| Standardablauf | <ol style="list-style-type: none"> <li>1. Wenn noch nicht erfolgt, wird basierend auf SAK.AUTD_CVC und HPC.AUTD_SUK_CVC und den zugehörigen privaten Schlüsseln ein sicherer Kanal zwischen der gSMC-K des Konnektors und dem HBA aufgebaut mittels Aufruf TUC_KON_005 „Card-to-Card authentisieren“ {<br/>sourceCardSession = gSMC-K;<br/>targetCardSession = CardSession;<br/>authMode = „gegenseitig+TC“}</li> <li>Die folgenden Schritte werden für jedes Dokument des Stapels durchgeführt.</li> <li>2. Das zu signierende Dokument wird, soweit noch nicht erfolgt, für die Signatur gemäß des entsprechenden Formats vorbereitet. Ein Ergebnis dieser Vorbereitung sind die Data To Be Signed (DTBS): der Hash-Wert (Digest des SignedInfo-Elementes), der zur Signatur an die Karte gesendet werden soll.</li> <li>3. Es wird geprüft, ob der Komfortsignatur-Zähler der cardSession den Wert SAK_COMFORT_SIGNATURE_MAX überschritten hat .</li> <li>4. Es wird geprüft, ob der Komfortsignatur-Timer der cardSession (SAK_COMFORT_SIGNATURE_TIMER) abgelaufen ist.</li> <li>5. Für das zu signierende Dokument werden die DTBS zur Signatur im sicheren Kanal an den HBA übermittelt (Aufruf TUC_KON_218 „Signiere“). Dabei werden der</li> </ol> |

|                            |   |
|----------------------------|---|
|                            | <p>Schlüssel und der Algorithmusidentifizierer über die Tabelle TAB_KON_900 bestimmt.</p> <p>6. Der Komfortsignatur-Zähler der cardSession wird um 1 erhöht.</p> <p>7. Die erstellte Signatur wird mathematisch geprüft.</p> <p>8. Der ermittelte Signaturwert wird in den zuvor vorbereiteten Signaturprototypen eingefügt.</p> <p>9. Der Konnektor löst TUC_KON_256 {"SIG/SIGNDOK/NEXT_SUCCESSFUL"; Op; Info; „\$Jobnummer“; noLog; \$doInformClients } aus.</p>  |
| Varianten/<br>Alternativen | Keine   |
| Fehlerfälle                | <p>In den Fehlerfällen, die zum Abbruch des Komfortsignaturmodus mit Fehlercode 4271 führen, wird vor dem Abbruch TUC_KON_172 für das cardHandle des HBA ausgeführt.</p> <p>(-&gt;3) Der Komfortsignatur-Zähler der cardSession hat den Maximalwert überschritten: Fehlercode 4271</p> <p>(-&gt;4) Der Komfortsignatur-Timer der cardSession ist abgelaufen: Fehlercode 4271</p> <p>(-&gt;5) Der PIN.QES-Nutzungszähler der Karte ist abgelaufen (erkennbar z. B. daran, dass die Karte einen Autorisierungsfehler zurückmeldet): Fehlercode 4271</p> <p>(-&gt;5) Fehler im Signaturvorgang führen zum Abbruch des gesamten Signaturvorgangs: Fehlercode 4123</p> <p>(-&gt;7) Fehler in mathematischer Prüfung der Signatur führen zum Abbruch des Signaturvorgangs: Fehlercode 4120</p> <p>Das weitere Verhalten des TUCs bei einem Fehlerfall oder beim Abbruch durch den Benutzer ist in TAB_KON_192, Verhalten des Konnektors beim Abbruch einer Stapelsignatur, beschrieben.</p> |
| Sicherheitsanforderungen   | <p>Zum Aufbau des sicheren Kanals bzw. zur Aushandlung des symmetrischen Schlüssels DARF DF.QES NICHT verlassen werden. Benötigte CVCs des HBA MÜSSEN also bereits vor dem Signaturvorgang eingelesen und gecached werden. Dies KANN bereits beim Stecken des HBA geschehen.</p> <p>Komfortsignaturen MÜSSEN im SE#2 abgearbeitet werden.</p> <p>Die in [gemSpec_Krypt] angegebenen Festlegungen der zu unterstützenden Algorithmen MÜSSEN berücksichtigt werden.</p>   |

4299 **Tabelle 204: TAB\_KON\_873 Fehlercodes TUC\_KON\_158 „Komfortsignaturen erstellen“**

| Fehlercode  | ErrorType | Severity | Fehlertext                       |
|---|-----------|----------|----------------------------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |                                  |
| 4120  | Security  | Error    | Kartenfehler                     |
| 4123  | Security  | Error    | Fehler bei Signaturerstellung    |
| 4271  | Technical | Error    | Komfortsignaturmodus abgebrochen |

4300 [ $\leq$ ]

4301

#### 4302 **4.1.8.4 Interne TUCs, auch durch Fachmodule nutzbar**

##### 4303 **A\_20478 - Zusätzliche Dokumentformate für nonQES-Signatur**

4304 Der Konnektor KANN für die nonQES-Signaturerstellung an der Schnittstelle zu  
4305 Fachmodulen zusätzliche Dokumentformate unterstützen. [ $\leq$ ]

4306 Die in der obigen Anforderung benannten Signaturen von Dokumentenformaten  
4307 umfassen beispielsweise die Signatur von Token nach SAML2 für das Fachmodul ePA  
4308 entsprechend [gemSpec\_FM\_ePA#A\_14927].

##### 4309 **4.1.8.4.1 TUC\_KON\_160 „Dokumente nonQES signieren“**

##### 4310 **TIP1-A\_4653 - TUC\_KON\_160 „Dokumente nonQES signieren“**

4311 Der Konnektor MUSS den technischen Use Case TUC\_KON\_160 „Dokumente nonQES  
4312 signieren“ umsetzen.

4313

4314 **Tabelle 205: TAB\_KON\_753 – TUC\_KON\_160 „Dokumente nonQES signieren“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_160 „Dokumente nonQES signieren“   |
| Beschreibung   | Im Rahmen von Fachanwendungen werden ein oder mehrere Dokumente mit einer nicht-qualifizierten elektronischen Signatur (nonQES) versehen. Es werden die nonQES_DocFormate unterstützt. |
| Auslöser       | Aufruf durch ein Clientsystem (Operation SignDocument) oder ein Fachmodul.   |
| Vorbedingungen | Die Signaturkarte muss gesteckt sein.  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>cardSession (Kartensitzung; zulässig sind SM-B, oder bei Aufruf durch Fachmodul auch zusätzlich eGK)</li> </ul>                                 |

|                |   |
|----------------|---|
|                | <ul style="list-style-type: none"> <li>signRequests<br/>(Liste von Signaturaufträgen.)<br/>Jeder Signaturauftrag (SignRequest) kapselt: <ul style="list-style-type: none"> <li>documentsToBeSigned<br/>(Zu signierendes Dokument bzw. zu signierende Dokumente);<br/>darin u.a.<br/>documentFormat (Formatangabe für das zu signierende Dokument)</li> <li>optionalInputs<br/>(weitere optionale Eingabeparameter zur Steuerung der Details bei der zu erstellenden Signatur, siehe Operation SignDocument, Parameter dss:OptionalInputs); darin u.a. signatureType (URI für den Signaturtyp XML-, CMS-, S/MIME-, PDF-Signatur)</li> <li>includeRevocationInfo: – <i>optional; default: true</i><br/>(Dieser optionale Parameter steuert die Einbettung von OCSP-Antworten in die Signatur: nur wirksam bei der Prüfung von enthaltenen Parallelsignaturen, wenn eine Gegensignatur erstellt werden soll. Die OCSP-Antworten werden in die jeweils geprüfte Parallelsignatur eingebettet.)</li> <li>workplaceId<br/>(Identifikator des Arbeitsplatzes)</li> </ul> </li> </ul> |
| Komponenten    | Konnektor, Kartenterminal, Signaturkarte bzw. HSM-B   |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>signedDocuments<br/>(Liste der signierten Dokumente)</li> </ul>  |
| Standardablauf | <p>Der Konnektor KANN die Schritte 1 bis 4 in einer beliebigen Reihenfolge durchführen.</p> <ol style="list-style-type: none"> <li>Der signatureType und die Signaturvariante werden für jedes Dokument der Liste entsprechend SignatureType und SignatureVariant festgelegt. Wenn signatureType oder SignatureVariant nicht übergeben wurden (als Element von optionalInputs), wird das dem dokumentformat entsprechende Default-Verfahren gewählt (siehe TAB_KON_583 – Default-Signaturverfahren).</li> <li>Für alle Dokumente des Stapels wird die Zulässigkeit des Kartentyps geprüft. Das für die Signatur zu nutzende Zertifikat wird anhand des Kartentyps implizit ausgewählt.</li> <li>Es werden die Voraussetzungen für die Signatur geprüft. Dies erfolgt durch Aufruf von TUC_KON_165 „Signaturvoraussetzungen für nonQES prüfen“.</li> </ol>   |

|                                |  |
|--------------------------------|--|
|                                | <ol style="list-style-type: none"> <li>4. Zum Vorbereiten der Dokumente für die Signatur wird TUC_KON_155 „Dokumente zur Signatur vorbereiten“ mit ocspsResponses aufgerufen.</li> <li>5. Die Signaturen werden durch den Aufruf von TUC_KON_166 erstellt.</li> <li>6. Die signierten Dokumente werden an den Aufrufer zurückgegeben.</li> </ol>   |
| Varianten/<br>Alternativen     | <p><u>Im Fall signatureType=S/MIME-Signatur</u> wird der Standardablauf des CMS Signaturverfahrens durch einen vorgelagerten S/MIME-Vorbereitungsschritt und einen nachgelagerten S/MIME-Nachbereitungsschritt ergänzt. Das S/MIME-Verfahren MUSS konform [S/MIME] und SOLL konform [COMMON_PKI], Part 3, erfolgen.</p> <p>Der S/MIME-Vorbereitungsschritt bereitet das übergebene MIME-Dokument gemäß [S/MIME], Kapitel 3.1, auf die nachfolgende CMS-Signatur durch eine Kanonisierung für Text [S/MIME], Kapitel 3.1.1, vor. Eine weitere Kanonisierung oder eine Anpassung des Transfer Encodings [S/MIME], Kapitel 3.1.2, erfolgt nicht.</p> <p>Im S/MIME-Nachbereitungsschritt wird das im Standardablauf erzeugte CMS-Objekt in eine MIME-Nachricht vom Typ „application/pkcs7-mime“ eingebettet.</p> <p>Sämtliche Header-Felder der Nachricht MÜSSEN in die Header-Felder der S/MIME-Nachricht übernommen werden.</p> <p>"MIME-Version: 1.0" MUSS definiert sein.</p> <p>Das Feld "Content-Type:" ist als "application/pkcs7-mime" zu definieren. Die weiteren Attribute dieses Feldes sind:</p> <ul style="list-style-type: none"> <li>• "smime-type=signed-data;"</li> <li>• "name=\$dateiname", wobei \$dateiname auf ".p7m" endet.</li> </ul> <p>Die Codierung des signierten Inhalts der Nachricht MUSS in "base64" erfolgen. Entsprechend ist das zugehörige Header-Feld zu füllen: "Content-Transfer-Encoding: base64".</p> <p>Das Feld "Content-Disposition" definiert den Inhalt der Nachricht als Dateianhang: "Content-Disposition: attachment; filename=\$dateiname"</p> |
| Fehlerfälle                    | <p>Fehler in den folgenden Schritten des Standardablaufs führen zum Abbruch mit den ausgewiesenen Fehlercodes:</p> <p>(→2) Ungültige Angabe des Signaturverfahrens: Fehlercode 4111</p> <p>Übergabe eines für die nonQES nicht unterstützten Dokumentformats: Fehlercode 4110</p> <p>(→3) Kartentyp nicht zulässig für Signatur: Fehlercode 4126</p>   |
| Nichtfunktionale Anforderungen | keine  |

|                      |       |
|----------------------|-------|
| Zugehörige Diagramme | keine |
|----------------------|-------|

4315 **Tabelle 206: TAB\_KON\_127 Fehlercodes TUC\_KON\_160 „Dokumente nonQES signieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4110  | Technical | Error    | ungültiges Dokumentformat (%Format%)<br>Der Parameter Format enthält das übergebene Dokumentformat. |
| 4111  | Technical | Error    | ungültiger Signaturtyp oder Signaturvariante  |
| 4126  | Security  | Error    | Kartentyp nicht zulässig für Signatur   |

4316  
4317 [ $\leq$ ]

4318 **TIP1-A\_4653-02 - ab PTV4: TUC\_KON\_160 „Dokumente nonQES signieren“**  
4319 Der Konnektor MUSS den technischen Use Case TUC\_KON\_160 „Dokumente nonQES  
4320 signieren“ umsetzen.  
4321

4322 **Tabelle 207: TAB\_KON\_753 – TUC\_KON\_160 „Dokumente nonQES signieren“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_160 „Dokumente nonQES signieren“  |
| Beschreibung   | Im Rahmen von Fachanwendungen werden ein oder mehrere Dokumente mit einer nicht-qualifizierten elektronischen Signatur (nonQES) versehen. Es werden die nonQES_DocFormate unterstützt.  |
| Auslöser       | Aufruf durch ein Clientsystem (Operation SignDocument) oder ein Fachmodul.  |
| Vorbedingungen | Die Signaturkarte muss gesteckt sein.   |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>cardSession<br/>(Kartensitzung; zulässig sind SM-B, oder bei Aufruf durch Fachmodul auch zusätzlich eGK)</li> <li>crypt [SIG_CRYPT_nonQES] - <i>optional</i>;<br/>default und Wertebereich: siehe TAB_KON_863<br/>Dieser Parameter steuert, ob RSA-basierte oder ECC-basierte Signaturen erzeugt werden.</li> <li>signRequests<br/>(Liste von Signaturaufträgen. Jeder Signaturauftrag (SignRequest) kapselt:</li> </ul> |

|                |  |
|----------------|--|
|                | <ul style="list-style-type: none"> <li>documentsToBeSigned<br/>(Zu signierendes Dokument bzw. zu signierende Dokumente);<br/>darin u.a.<br/>documentFormat (Formatangabe für das zu signierende Dokument)</li> <li>optionalInputs<br/>(weitere optionale Eingabeparameter zur Steuerung der Details bei der zu erstellenden Signatur, siehe Operation SignDocument, Parameter dss:OptionalInputs); darin u.a. signatureType (URI für den Signatortyp XML-, CMS-, S/MIME-, PDF-Signatur)</li> <li>includeRevocationInfo: – <i>optional; default: true</i><br/>(Dieser optionale Parameter steuert die Einbettung von OCSP-Antworten in die Signatur: nur wirksam bei der Prüfung von enthaltenen Parallelsignaturen, wenn eine Gegensignatur erstellt werden soll. Die OCSP-Antworten werden in die jeweils geprüfte Parallelsignatur eingebettet.)</li> <li>workplaceId<br/>(Identifikator des Arbeitsplatzes)</li> </ul>  |
| Komponenten    | Konnektor, Kartenterminal, Signaturkarte bzw. HSM-B  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>signedDocuments<br/>(Liste der signierten Dokumente)</li> </ul>   |
| Standardablauf | <p>Der Konnektor KANN die Schritte 1 bis 4 in einer beliebigen Reihenfolge durchführen.</p> <ol style="list-style-type: none"> <li>Der signatureType und die Signaturvariante werden für jedes Dokument der Liste entsprechend SignatureType und SignatureVariant festgelegt. Wenn signatureType oder SignatureVariant nicht übergeben wurden (als Element von optionalInputs), wird das dem dokumentformat entsprechende Default-Verfahren gewählt (siehe TAB_KON_583 – Default-Signaturverfahren).</li> <li>Für alle Dokumente des Stapels wird die Zulässigkeit des Kartentyps geprüft. Das für die Signatur zu nutzende Zertifikat wird anhand des Kartentyps und des Parameters crypt ausgewählt.</li> <li>Es werden die Voraussetzungen für die Signatur geprüft. Dies erfolgt durch Aufruf von TUC_KON_165 „Signaturvoraussetzungen für nonQES prüfen“.</li> <li>Zum Vorbereiten der Dokumente für die Signatur wird TUC_KON_155 „Dokumente zur Signatur vorbereiten“ mit ocsResponses aufgerufen.</li> </ol> |

|                                |   |
|--------------------------------|---|
|                                | <p>5. Die Signaturen werden durch den Aufruf von TUC_KON_166 erstellt.</p> <p>6. Die signierten Dokumente werden an den Aufrufer zurückgegeben.</p>   |
| Varianten/<br>Alternativen     | <p>Im Fall signatureType=S/MIME-Signatur wird der Standardablauf des CMS Signaturverfahrens durch einen vorgelagerten S/MIME-Vorbereitungsschritt und einen nachgelagerten S/MIME-Nachbereitungsschritt ergänzt. Das S/MIME-Verfahren MUSS konform [S/MIME] und SOLL konform [COMMON_PKI], Part 3, erfolgen.</p> <p>Der S/MIME-Vorbereitungsschritt bereitet das übergebene MIME-Dokument gemäß [S/MIME], Kapitel 3.1, auf die nachfolgende CMS-Signatur durch eine Kanonisierung für Text [S/MIME], Kapitel 3.1.1, vor. Eine weitere Kanonisierung oder eine Anpassung des Transfer Encodings [S/MIME], Kapitel 3.1.2, erfolgt nicht.</p> <p>Im S/MIME-Nachbereitungsschritt wird das im Standardablauf erzeugte CMS-Objekt in eine MIME-Nachricht vom Typ „application/pkcs7-mime“ eingebettet.</p> <p>Sämtliche Header-Felder der Nachricht MÜSSEN in die Header-Felder der S/MIME-Nachricht übernommen werden.</p> <p>"MIME-Version: 1.0" MUSS definiert sein.</p> <p>Das Feld "Content-Type:" ist als "application/pkcs7-mime" zu definieren. Die weiteren Attribute dieses Feldes sind:</p> <ul style="list-style-type: none"> <li>• "smime-type=signed-data;"</li> <li>• "name=\$dateiname", wobei \$dateiname auf ".p7m" endet.</li> </ul> <p>Die Codierung des signierten Inhalts der Nachricht MUSS in "base64" erfolgen. Entsprechend ist das zugehörige Header-Feld zu füllen: "Content-Transfer-Encoding: base64".</p> <p>Das Feld "Content-Disposition" definiert den Inhalt der Nachricht als Dateianhang: "Content-Disposition: attachment; filename=\$dateiname"</p> |
| Fehlerfälle                    | <p>Fehler in den folgenden Schritten des Standardablaufs führen zum Abbruch mit den ausgewiesenen Fehlercodes:</p> <p>(→2) Ungültige Angabe des Signaturverfahrens: Fehlercode 4111</p> <p>Übergabe eines für die nonQES nicht unterstützten Dokumentformats: Fehlercode 4110</p> <p>(→3) Kartentyp nicht zulässig für Signatur: Fehlercode 4126</p>  |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | keine   |



4323 **Tabelle 208: TAB\_KON\_127 Fehlercodes TUC\_KON\_160 „Dokumente nonQES signieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4110  | Technical | Error    | ungültiges Dokumentformat (%Format%)<br>Der Parameter Format enthält das übergebene Dokumentformat. |
| 4111  | Technical | Error    | ungültiger Signaturtyp oder Signaturvariante  |
| 4126  | Security  | Error    | Kartentyp nicht zulässig für Signatur   |

4324 Die zulässigen Zertifikate und Schlüssel sind in TAB\_KON\_900 aufgelistet. [ <= ]

4326

4327 4.1.8.4.2 TUC\_KON\_161 „nonQES Dokumentsignatur prüfen“

4328 **TIP1-A\_4654-03 - TUC\_KON\_161 „nonQES Dokumentsignatur prüfen“**

4329 Der Konnektor MUSS den technischen Use Case TUC\_KON\_161 „nonQES

4330 Dokumentsignatur prüfen“ umsetzen.

4331

4332 **Tabelle 209: TAB\_KON\_121 - TUC\_KON\_161 „nonQES Dokumentsignatur prüfen“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_161 „nonQES Dokumentsignatur prüfen“  |
| Beschreibung   | Es wird die nicht-qualifizierte elektronische Signatur (nonQES) eines Dokuments geprüft. Dabei werden die Signaturverfahren laut Tabelle TAB_KON_582 – Signaturverfahren unterstützt. Sind mehrere Signaturen vorhanden, so werden alle geprüft. Auch Parallel- und Gegensignaturen MÜSSEN unterstützt werden.  |
| Auslöser       | Aufruf durch ein Clientsystem (Operation VerifyDocument) oder ein Fachmodul   |
| Vorbedingungen | keine   |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>signedDocument<br/>(Signiertes Document vom Typ nonQES_DocFormate)</li> <li>signature – <i>optional/falls detached Signatur</i><br/>(Signatur. Es werden Parallel- und Gegensignaturen unterstützt.)</li> <li>optionalInputs<br/>(optionale Eingabeparameter, siehe Operation VerifyDocument, Parameter SIG:OptionalInputs)</li> <li>certificate – <i>optional/verpflichtend, wenn das Zertifikat nicht im signierten Dokument enthalten ist</i><br/>(X.509-Zertifikat, gegen das die Signatur geprüft werden soll)</li> </ul> |

|                |  |
|----------------|--|
|                | <p>ocspGracePeriod<br/>(OCSP-Grace Period: maximal zulässiger Zeitraum, den die letzte<br/>OCSP-Antwort aus dem Cache bezüglich des Referenzzeitpunkts zurückliegen darf)</p> <ul style="list-style-type: none"> <li>• xmlSchemas – <i>optional/nur für XML-Dokumente</i> (XMLSchema und ggf. weitere vom Hauptschema benutzte Schemata)</li> <li>• includeRevocationInfo: – <i>optional; Default = false</i> (Dieser optionale Parameter steuert die Einbettung von OCSP Antworten in die Signatur)</li> </ul>  |
| Komponenten    | Konnektor  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>• verificationResult [VerificationResult] (Ergebnis der Signaturprüfung)</li> <li>• optionalOutput – <i>optional</i> (weitere Ausgabedaten gemäß SIG:OptionalOutput)</li> </ul>   |
| Standardablauf | <ol style="list-style-type: none"> <li>1. <b>„DocumentValidation“:</b><br/>Falls die Signatur im Dokument eingebettet ist, wird das signierte Dokument validiert durch Aufruf TUC_KON_080 „Dokument validieren“ { CheckDisplayability=false; ... } Treten dabei Fehler bei Validierung der Typkonformität auf, wird die Prüfung mit einem Fehler abgebrochen.</li> <li>2. <b>„CoreValidation“:</b><br/>Es erfolgt die mathematische Prüfung der Signatur bestehend aus der Prüfung der Hash-Kette bis zum signierten Hashwert und der Prüfung der kryptographischen Signatur unter Verwendung des öffentlichen Schlüssels, des Signaturwertes und des signierten Hashwertes.<br/><br/>XML-Signatur:<br/>Die Core Validation erfolgt entsprechend [XMLDSig]<br/>Kapitel 3.2 Core Validation.<br/>CMS-Signatur:<br/>Die Core Validation erfolgt entsprechend Cryptographic<br/>Message Syntax (CMS) Kapitel 5.6 Signature Verification<br/>Process [RFC5652].<br/>PDF-Signatur:<br/>Die Core Validation erfolgt entsprechend [PAdES-3]<br/>Kapitel 4.6 Signature Validation aus PAdES-BES Part 3.<br/><br/>Auch wenn die Validierung fehlschlägt, werden die folgenden Prüfschritte durchgeführt, so dass ein vollständiges Prüfprotokoll erstellt werden kann.</li> </ol> |

|  |  |
|--|--|
|  | <p><b>3. „CheckSignatureCertificate“:</b></p> <p><b>Teil 1: Signaturzertifikat ermitteln</b></p> <p>XML-Signatur:<br/>Das Signaturzertifikat ist im XMLDSig Element <code>ds:KeyInfo/ds:X509Data</code> gespeichert [XMLDSig] oder wird als Eingangsparameter übergeben.</p> <p>CMS-Signatur:<br/>Das Signaturzertifikat für CADES ist im Feld <code>certificates</code> im <code>SignedData</code> Container gespeichert [CADES] oder wird als Eingangsparameter übergeben.</p> <p>PDF-Signatur:<br/>Das PDF Signaturzertifikat für PAdES ist im Feld <code>SignedData.certificates</code> entsprechend Kapitel 6.1.1 „Placements of the signing certificate“ [PAdES Baseline Profile] gespeichert oder wird als Eingangsparameter übergeben.</p> <p><b>Teil 2: Signaturzeitpunkt bestimmen</b></p> <p>Der Signaturzeitpunkt <code>Ermittelter_Signaturzeitpunkt_Eingebettet</code> wird wie folgt selektiert:</p> <p>XML-Signatur:<br/>Das XML element <code>SigningTime</code> spezifiziert den Signaturzeitpunkt entsprechend Kapitel 7.2.1 XAdES [XAdES].</p> <p>CMS-Signatur:<br/>Das Attribut <code>SigningTime</code> spezifiziert den Signaturzeitpunkt entsprechend Kapitel 11.3 CMS [CMS].</p> <p>PDF-Signatur:<br/>Der Signaturzeitpunkt kann dem M Eintrag des Signature Dictionary entnommen werden [PAdES Baseline Profile] Kapitel 6.2.1 Signing time.</p> <p>Der Signaturzeitpunkt <code>Benutzerdefinierter_Zeitpunkt</code> liegt gegebenenfalls als Aufrufparameter vor. Der Signaturzeitpunkt <code>Ermittelter_Signaturzeitpunkt_System</code> wird ermittelt.</p> <p><b>Teil 3: Signaturzertifikatsprüfung:</b></p> <p>Bei der folgenden Signaturzertifikatsprüfung sind die Signaturzeitpunkte gemäß [TIP1-A_5545] zu berücksichtigen.</p> |
|--|--|

|  |   |
|--|---|
|  | <p>Die Signaturzertifikatsprüfung erfolgt durch Aufruf von TUC_KON_037 „Zertifikat prüfen“, und zwar:<br/>Wenn es sich um das X.509-Zertifikat einer eGK handelt (PolicyList = oid_egk_aut bzw. oid_egk_autn), dann:</p> <pre>TUC_KON_037 „Zertifikat prüfen“ {     certificate;     qualifiedCheck = not_required;     baseTime = Signaturzeitpunkt;     offlineAllowNoCheck = true;     policyList = [oid_egk_aut   oid_egk_autn];     intendedKeyUsage= intendedKeyUsage(C.CH.AUT C.CH.AUTN);      intendedExtendedKeyUsage = id-kp-clientAuth;     ocsResponses = OCS-Response;     gracePeriod = ocsGracePeriod;     validationMode = OCS;     getOCSResponses = includeRevocationInfo }</pre> <p>Wenn es ein X.509-Zertifikat der SM-B ist (PolicyList = oid_smc_b_osig), dann:</p> <pre>TUC_KON_037 „Zertifikat prüfen“ {     certificate;     qualifiedCheck = not_required;     baseTime = Signaturzeitpunkt;     offlineAllowNoCheck = true;     policyList = oid_smc_b_osig;     intendedKeyUsage = intendedKeyUsage(C.HCI.OSIG);     ocsResponses = OCS-Response;     gracePeriod = ocsGracePeriod;     validationMode = OCS ;     getOCSResponses = includeRevocationInfo }</pre> <p>Sind OCS-Responses in der Signatur eingebettet, ist die jüngste OCS-Response, die für die Zertifikatsprüfung notwendig ist, beim Aufruf von TUC_KON_037 zu übergeben.<br/>Sofern der Aufruf von TUC_KON_037 ocsResponsesRenewed zurückgibt, wird die Liste der OCS-Responses in die Signatur eingebettet.<br/>Auch wenn die Zertifikatsprüfung fehlschlägt, werden die folgenden Prüfungen durchgeführt.</p> <p><b>4. "CheckPolicyConstraints"</b></p> <p>In diesem Schritt wird das signierte Dokument entsprechend der Profilierung der Signaturformate (siehe Anhang B.2) geprüft.<br/>Es sind die Vorgaben für die Prüfung von Signaturen aus den Standards für AdES [XAdES], [XAdES Baseline], [CAdES], [CAdES Baseline], [PAdES-3] und [PAdES Baseline] umzusetzen. Dabei sind die Vorgaben</p> |
|--|---|

|                                |  |
|--------------------------------|--|
|                                | <p>aus Tabelle TAB_KON_779 „Profilierung der Signaturformate“ und Tabelle TAB_KON_778 „Einsatzbereich der Signaturvarianten“ zu erfüllen.</p> <p>Auch wenn nicht alle Anforderungen an das Format des signierten Dokuments erfüllt werden, wird die Prüfung mit den folgenden Schritten fortgesetzt, um ein vollständiges Prüfungsprotokoll zu erhalten.</p> <p>5. Das <b>Prüfergebnis</b> (verificationResult, optionalOutput wird an den Aufrufer zurückgegeben (siehe TAB_KON_754 Übersicht Status für Prüfung einer Dokumentensignatur).</p>   |
| Varianten/<br>Alternativen     | <p>Im Fall, dass die Online-Prüfung des Sperrzustands des Signaturzertifikats nicht möglich ist und eine möglicherweise gecachte OCSP-Response nicht vorhanden ist oder nicht mehr verwendet werden darf, wird das Prüfergebnis mit der entsprechenden Warnung zurückgegeben.</p> <p>Im Fall einer PKCS#1-Signatur ist das verwendete Signaturverfahren, RSASSA-PSS bzw. RSASSA-PKCS1-v1_5, aus der Signatur zu bestimmen.</p>   |
| Fehlerfälle                    | <p>Das Verhalten des TUCs bei einem Fehlerfall ist in TAB_KON_124 Fehlercodes TUC_KON_161 „nonQES Dokumentensignatur prüfen“ beschrieben.</p> <p>(-&gt;1) keine Signatur in signedDocument und signature vorhanden: 4253<br/>(→2 „<b>CoreValidation</b>“)<br/>Interner Fehler: 4001, Signatur des Dokument ungültig: 4115.<br/>Signatur umfasst nicht das gesamte Dokument: 4262.<br/>(→3 „<b>CheckSignatureCertificate</b>“)<br/>Interner Fehler: 4001, Signaturzertifikat ermitteln fehlgeschlagen: 4206.<br/>(→4 „<b>CheckPolicyConstraints</b>“)<br/>Interner Fehler: 4001, Dokument nicht konform zu Regeln für nonQES: 4112.</p> |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | keine  |

**Tabelle 210: TAB\_KON\_124 Fehlercodes TUC\_KON\_161 „nonQES Dokumentensignatur prüfen“**

| Fehlercode   | ErrorType | Severity | Fehlertext                                      |
|--|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weiteren Fehlercodes auftreten. |           |          |   |
| 4001   | Technical | Error    | Interner Fehler                                 |
| 4206   | Technical | Error    | Signaturzertifikat ermitteln ist fehlgeschlagen |

|      |           |         |  |
|------|-----------|---------|--|
| 4112 | Technical | Error   | Dokument nicht konform zu Regeln für nonQES  |
| 4115 | Security  | Error   | Signatur des Dokuments ungültig.<br>Der SignatureValue des Dokuments ist falsch oder für mindestens eine Reference ist der DigestValue falsch. |
| 4253 | Technical | Error   | Keine Signatur im Aufruf   |
| 4262 | Technical | Error   | Signatur umfasst nicht das gesamte Dokument  |
| 4264 | Technical | Warning | Ein oder mehrere Zertifikate ignoriert   |

4335 Das Gesamtergebnis (VerificationResult) für die Prüfung einer Dokumentensignatur fasst  
 4336 die Ergebnisse aller Prüfungsschritte in einem einzelnen Statuswert zusammen.  
 4337

4338 **Tabelle 211: TAB\_KON\_754 Übersicht Status für Prüfung einer Dokumentensignatur**

| VerificationResult für gesamtes Dokument (VerificationResult/HighLevelResult) |  |
|---|--|
| Wert  | Bedeutung  |
| VALID   | Wenn VerificationResult für alle Signaturen zum Dokument VALID   |
| INVALID   | Wenn VerificationResult für eine Signatur zum Dokument INVALID   |
| INCONCLUSIVE  | in allen anderen Fällen  |
| VerificationResult pro Signatur (VerificationReport/IndividualReport/Result)  |  |
| Wert  | Bedeutung<br>mögliche Ausprägungen im VerificationReport   |
| VALID   | <p>Die Signatur wurde gemäß den Regeln für die nonQES geprüft und für gültig befunden.</p> <p>ResultMajor = urn:oasis:names:tc:dss:1.0:resultmajor:Success<br/>           ResultMinor = urn:oasis:names:tc:dss:1.0:resultminor:valid:signature:OnAllDocuments</p> <p>ResultMajor = urn:oasis:names:tc:dss:1.0:resultmajor:Success<br/>           ResultMinor = urn:oasis:names:tc:dss:1.0:resultminor:valid:signature:HasManifestResults</p> |
| INVALID   | Die Signatur ist ungültig oder aufgrund eines Fehlers konnte die Signaturprüfung nicht durchgeführt werden.  |

|              |   |
|--------------|---|
|              | ResultMajor = urn:oasis:names:tc:dss:1.0:resultmajor:Success<br>ResultMinor =<br>urn:oasis:names:tc:dss:1.0:resultminor:invalid:IncorrectSignature  |
|              | ResultMajor = urn:oasis:names:tc:dss:1.0:resultmajor:Success<br>ResultMinor =<br>urn:oasis:names:tc:dss:1.0:resultminor:valid:signature:InvalidSignatureTimestamp   |
|              | ResultMajor = urn:oasis:names:tc:dss:1.0:resultmajor:RequesterError   |
|              | ResultMajor = urn:oasis:names:tc:dss:1.0:resultmajor:ResponderError   |
|              | ResultMajor =<br>urn:oasis:names:tc:dss:1.0:resultmajor:InsufficientInformation<br>ResultMinor =<br>urn:oasis:names:tc:dss:1.0:resultminor:invalid:IncorrectSignature   |
|              | ResultMajor =<br>urn:oasis:names:tc:dss:1.0:resultmajor:InsufficientInformation<br>ResultMinor =<br>urn:oasis:names:tc:dss:1.0:resultminor:CertificateChainNotComplete  |
| INCONCLUSIVE | Die Signatur wurde gemäß den Regeln für die nonQES geprüft. Allerdings konnten eine oder mehrere Prüfungen nicht vollständig durchgeführt werden. Einzelheiten finden sich in Result-Detail. Die Prüfungen, die durchgeführt werden konnten, waren erfolgreich.   |
|              | ResultMajor =<br>urn:oasis:names:tc:dss:1.0:resultmajor:InsufficientInformation<br>ResultMinor =<br>urn:oasis:names:tc:dss:1.0:resultminor:OcsfNotAvailable<br><br>Hinweis: Das Erreichen dieses Zustandes hängt davon ab, ob eine OCSP-Abfrage nicht durchgeführt werden konnte, unabhängig davon, ob die Ursache dafür die Offlineschaltung des Konnektors (MGM_LU_ONLINE = Disabled) oder die Nichterreichbarkeit des OCSP-Responders im Online-Betrieb (MGM_LU_ONLINE = Enabled) ist. |

4339 [ $\leq$ ]4340 **TIP1-A\_5545 - nonQES-Signaturprüfergebnis bezogen auf Signaturzeitpunkt**4341 Der Konnektor MUSS zur nonQES-Signaturprüfung ein Prüfergebnis das sich auf genau  
4342 einen angenommenen Signaturzeitpunkt bezieht, an den Aufrufer zurückgeben.4343 Die Auswahl des angenommenen Signaturzeitpunkts, auf den sich das Signaturergebnis  
4344 bezieht, erfolgt hierarchisch:

- 4345 • Benutzerdefinierter\_Zeitpunkt
- 4346     falls vorhanden, sonst
- 4347 • Ermittelter\_Signaturzeitpunkt\_Eingebettet
- 4348     falls vorhanden, sonst
- 4349 • Ermittelter\_Signaturzeitpunkt\_System

4350 [ $\leq$ ]

4351 4.1.8.4.3 TUC\_KON\_162 „Kryptographische Prüfung der XML-Dokumentensignatur“

4352 **TIP1-A\_5505 - TUC\_KON\_162 „Kryptographische Prüfung der XML-**  
 4353 **Dokumentensignatur“**

4354 Der Konnektor MUSS den technischen Use Case TUC\_KON\_162 „Kryptographische  
 4355 Prüfung der XML-Dokumentensignatur“ umsetzen.

4356

4357 **Tabelle 212: TAB\_KON\_430 – TUC\_KON\_162 „Kryptographische Prüfung der XML-**  
 4358 **Dokumentensignatur“**

| Element                | Beschreibung   |
|------------------------|--|
| Name                   | TUC_KON_162 „Kryptographische Prüfung der XML-Dokumentensignatur“  |
| Beschreibung           | Es wird die mathematische Korrektheit der elektronischen Signatur eines XML-Dokuments geprüft. Sind mehrere Signaturen vorhanden, so werden alle geprüft.  |
| Auslöser               | Aufruf durch ein Fachmodul   |
| Vorbedingungen         | <ul style="list-style-type: none"> <li>signedDocument ist ein XML-Dokument</li> <li>signedDocument hat TUC_KON_080 erfolgreich durchlaufen</li> </ul>  |
| Eingangsdaten          | <ul style="list-style-type: none"> <li>signedDocument – <i>optional</i> (QES-signiertes XML-Dokument -&gt; siehe Definition in Operation VerifyDocument mit SIG:Document)</li> <li>signatureObject – <i>optional</i> ( -&gt; siehe Definition in Operation VerifyDocument mit dss:SignatureObject)</li> </ul>  |
| Komponenten            | Konnektor  |
| Ausgangsdaten          | <ul style="list-style-type: none"> <li>result (Ergebnis der Signaturprüfung)</li> </ul>  |
| Standardablauf         | <p><b>„CoreValidation“:</b><br/>         Es erfolgt die mathematische Prüfung der Signatur, bestehend aus der Prüfung der Hash-Kette bis zum signierten Hashwert und der Prüfung der kryptographischen Signatur unter Verwendung des öffentlichen Schlüssels aus dem Zertifikat, des Signaturwertes und des signierten Hashwertes.</p> <p><u>XML-Signatur:</u><br/>         Die Core Validation erfolgt entsprechend [XMLDSig] Kapitel 3.2<br/>         Core Validation.<br/>         a) CoreValidation erfolgreich -&gt; result = true<br/>         b) CoreValidation fehlerhaft -&gt; result = false</p> |
| Varianten/Alternativen | keine  |



|                                |   |
|--------------------------------|---|
| Fehlerfälle                    | Fehler in den folgenden Schritten des Standardablaufs führen zu den ausgewiesenen Fehlercodes:<br>Interner Fehler: 4001 |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | keine   |

4359

4360

4361

**Tabelle 213: TAB\_KON\_431 Fehlercodes TUC\_KON\_162 „Kryptographische Prüfung der XML-Dokumentensignatur“**

| Fehlercode   | ErrorType | Severit<br>y | Fehlertext      |
|--|-----------|--------------|-----------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weiteren Fehlercodes auftreten. |           |              |                 |
| 4001   | Technical | Error        | Interner Fehler |

4362

4363

[&lt;=]

4364

4.1.8.4.4 TUC\_KON\_150 „Dokumente QES signieren“

4365

**TIP1-A\_4655 - TUC\_KON\_150 „Dokument QES signieren“,**

4366

Der Konnektor MUSS den technischen Use Case TUC\_KON\_150 „Dokumente QES signieren“ umsetzen.

4367

4368

4369

**Tabelle 214: TAB\_KON\_755 – TUC\_KON\_150 „Dokumente QES signieren“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_150 "Dokumente QES signieren"   |
| Beschreibung   | Im Rahmen von Fachanwendungen werden ein oder mehrere Dokumente mit einer qualifizierten elektronischen Signatur versehen. Es werden die QES_DocFormate unterstützt.  |
| Auslöser       | Aufruf durch ein Clientsystem (Operation SignDocument) oder ein Fachmodul.  |
| Vorbedingungen | Die Signaturkarte muss gesteckt sein.   |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>signRequests<br/>(Liste von Signaturaufträgen)<br/>Jeder Signaturauftrag (SignRequest) kapselt: <ul style="list-style-type: none"> <li>documentsToBeSigned<br/>(Zu signierendes Dokument bzw. zu signierende Dokumente);<br/>darin u.a.</li> </ul> </li> </ul> |

|                |  |
|----------------|--|
|                | <p>documentFormat<br/>(Formatangabe für das zu signierende Dokument)</p> <ul style="list-style-type: none"> <li>optionalInputs<br/>(weitere optionale Eingabeparameter zur Steuerung der Details bei der zu erstellenden Signatur, siehe Operation SignDocument, Parameter dss:OptionalInputs); darin u.a. signatureType (URI für den Signaturtyp XML-, CMS-, S/MIME-, PDF-Signatur)</li> <li>includeRevocationInfo [Boolean]: – optional; Default: true<br/>(Dieser optionale Parameter steuert die Einbettung von OCSP Antworten in die Signatur; siehe Operation SignDocument, Parameter SIG:IncludeRevocationInfo)</li> <li>cardSession<br/>(Kartensitzung. Unterstützte Kartentypen: HBAX)</li> <li>crypt [SIG_CRYPT_QES] - <i>optional</i>;<br/>default und Wertebereich: siehe TAB_KON_862<br/>(Dieser Parameter steuert, ob RSA-basierte oder ECC-basierte Signaturen erzeugt werden.)</li> <li>workplaceId</li> </ul>   |
| Komponenten    | Konnektor, Kartenterminal, Signaturkarte (HBAX)  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>signedDocuments<br/>(Liste der signierten Dokumente)</li> </ul>   |
| Standardablauf | <p>Der Konnektor KANN die Schritte 1 bis 4 in einer beliebigen Reihenfolge durchführen.</p> <ol style="list-style-type: none"> <li>Der Signaturtyp und die Signaturvariante werden für jedes Dokument der Liste entsprechend signatureType und SignatureVariant festgelegt (ggf. in optionalInputs enthalten). Wenn SignatureType oder SignatureVariant nicht übergeben wurden, wird das dem Dokumentformat entsprechende Default-Verfahren gewählt (siehe TAB_KON_583 – Default-Signaturverfahren).</li> <li>Für alle Dokumente des Stapels wird die Zulässigkeit des Kartentyps geprüft. Das für die Signatur zu nutzende Zertifikat wird anhand des Kartentyps und des Parameters crypt ausgewählt.</li> <li>Es werden die Voraussetzungen für die Signatur geprüft. Dies erfolgt im TUC_KON_152 „Signaturvoraussetzungen für QES prüfen“. Wenn includeRevocationInfo=true, dann setze ocsResponses auf Rückgabewert von TUC_KON_152.</li> <li>Die am Signaturvorgang beteiligten Ressourcen (Signaturkarte sowie PIN Pad und Display des PIN-</li> </ol> |

|                            |  |
|----------------------------|--|
|                            | <p>Eingabe-Kartenterminals) werden für die exklusive Nutzung durch diesen Signaturvorgang reserviert. Die Reservierung der Signaturkarte erfolgt durch Aufruf von TUC_KON_023 „Karte reservieren“ {<br/> cardSession;<br/> doLock = true }.</p> <p>5. Zum Vorbereiten der Dokumente für die Signatur wird TUC_KON_155 „Dokumente zur Signatur vorbereiten“ mit ocsResponses aufgerufen.<br/> Die Zugriffe auf die Signaturkarte in den Schritten 6 bis 7 müssen im DF.QES erfolgen.</p> <p>6. Die Signaturerstellung wird durch den Anwender autorisiert. Dies erfolgt durch Aufruf von TUC_KON_012 „PIN verifizieren“ { cardSession;<br/> workplaceId;<br/> pinRef = PIN.QES;<br/> verificationType = Mandatorisch }</p> <p>Wenn nur ein zu signierendes Dokument vorhanden ist und der Einfachsignaturmodus aktiviert ist (siehe Konfigurationsparameter SAK_SIMPLE_SIGNATURE_MODE), wird in Schritt 7 Variante a) durchgeführt, ansonsten Variante b).</p> <p>7. Variante a) Die Signatur wird erstellt. Dies erfolgt gemäß TUC_KON_168 „Einzelsignatur QES erstellen“.<br/> Variante b) Die Signaturen werden erstellt. Dies erfolgt gemäß TUC_KON_154 „QES-Signaturen erstellen“.</p> <p>8. Es wird DF.QES verlassen, um den PIN-Status der PIN.QES zurückzusetzen. Der im Konnektor verwaltete Sicherheitszustand (CARDSESSION.AUTHSTATE) ist zu aktualisieren.</p> <p>9. Die reservierten Ressourcen (Signaturkarte sowie PIN-Pad und Display des PIN-Eingabe-Kartenterminals) werden wieder freigegeben.<br/> Zur Freigabe der Signaturkarte wird TUC_KON_023 „Karte reservieren“<br/> cardSession;<br/> doLock = false }<br/> aufgerufen.</p> <p>10. Die signierten Dokumente werden an den Aufrufer zurückgegeben.</p> |
| Varianten/<br>Alternativen | Der Nutzer kann den Vorgang bei der Autorisierung (Schritt 6) abbrechen. Hierbei sind die gleichen Regeln anzuwenden wie im Fehlerfall (s. Fehlerfälle).   |
| Fehlerfälle                | Fehler in den folgenden Schritten des Standardablaufs führen zum Abbruch mit den ausgewiesenen Fehlercodes:<br>(->1) Ungültige Angabe des Signaturtyps oder Signaturvariante:<br>Fehlercode 4111   |

|                                |   |
|--------------------------------|---|
|                                | <p>Übergabe eines für die QES nicht unterstützten Dokumentformats:<br/> Fehlercode 4110<br/> (-&gt;2) Kartentyp nicht zulässig für Signatur: Fehlercode 4126<br/> (-&gt;5) Fehler bei der Reservierung von Ressourcen: Fehlercode 4060<br/> (-&gt;7b) Karte ist kein HBA, sondern HBA-Vorläuferkarte: Fehlercode 4118<br/> Im Fehlerfall, inklusive Timeout bei der PIN-Eingabe, oder bei Abbruch durch den Benutzer (Fehler 4049):<br/> a) ... MUSS DF.QES verlassen werden<br/> b) ... MÜSSEN alle reservierten Ressourcen freigegeben werden<br/> c) ... MUSS der Fehler immer an das Clientsystem zurückgemeldet werden</p> |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | Abbildung PIC_KON_114 Aktivitätsdiagramm zu „Dokument QES signieren“  |

4370

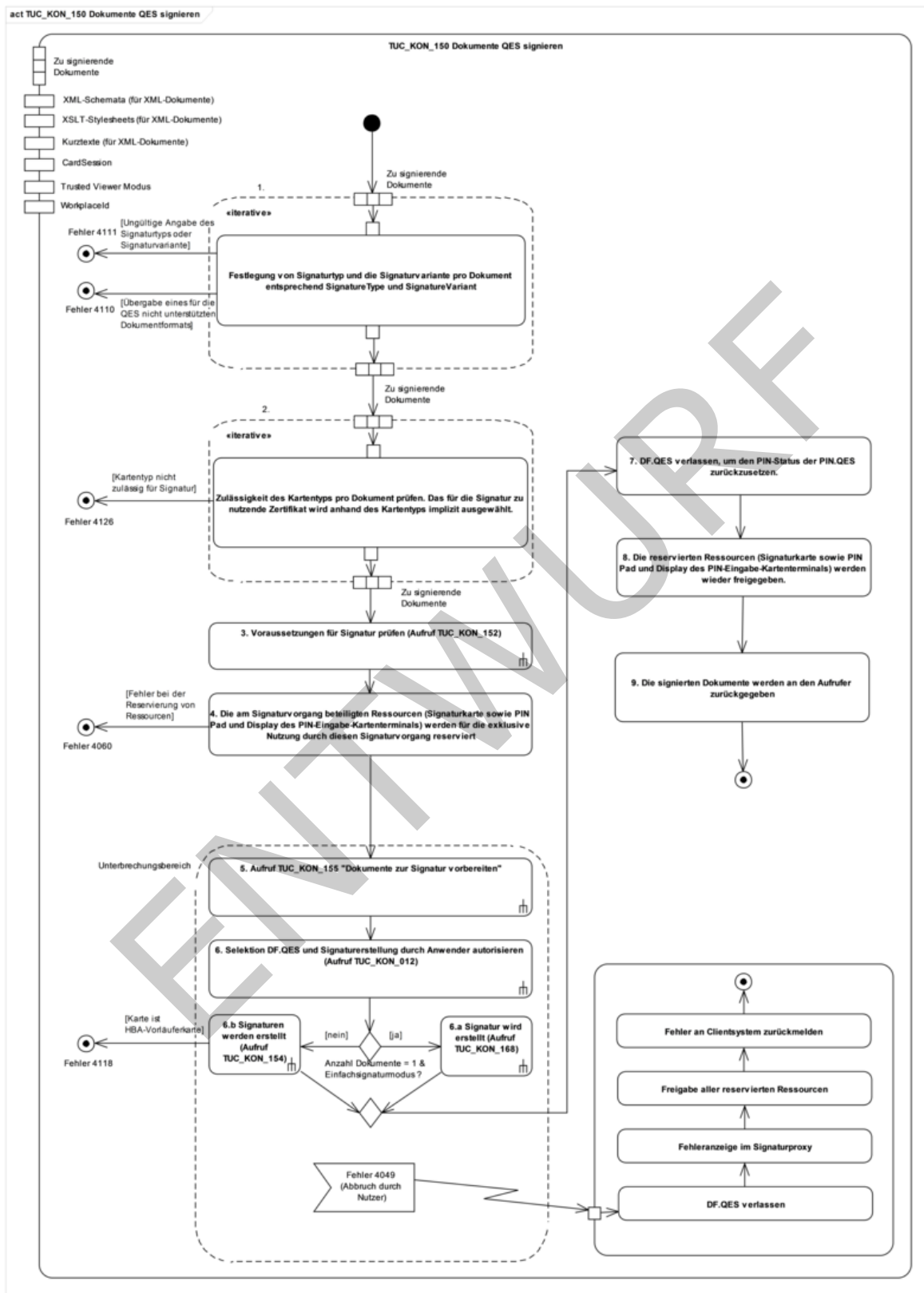


Abbildung 19: PIC\_KON\_114 Aktivitätsdiagramm zu „Dokument QES signieren“

4371

4372

4373

4374 **Tabelle 215: TAB\_KON\_128 Fehlercodes TUC\_KON\_150 „Dokument QES signieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4060  | Technical | Error    | Ressource belegt  |
| 4110  | Technical | Error    | ungültiges Dokumentformat (%Format%)<br>Der Parameter Format enthält das übergebene Dokumentformat.             |
| 4111  | Technical | Error    | ungültiger Signaturtyp oder Signaturvariante  |
| 4118  | Technical | Error    | Stapelsignaturen werden nur für den HBA unterstützt. Mit HBA-Vorläuferkarten sind nur Einzelsignaturen möglich. |
| 4126  | Security  | Error    | Kartentyp nicht zulässig für Signatur   |
| 4049  | Technical | Error    | Abbruch durch den Benutzer  |

4375 [**<=**]4376 **Anforderungen zur XML-Sicherheit:**4377 **TIP1-A\_5113 - Abwehr von XML-Signature-Wrapping Angriffen**

4378 Der Konnektor MUSS XML-Signature-Wrapping-Angriffe (XSW) abwehren.

4379 [**<=**]4380 **4.1.8.4.5 Anforderungen an die Stapelsignatur**

4381 Eine Stapelsignatur definiert sich als „Erstellung einer begrenzten Anzahl Signaturen nach  
 4382 den zeitlich unmittelbar aufeinander folgenden Prozessen der Anzeige der zu  
 4383 signierenden Daten und der einmaligen Authentisierung des Signaturschlüssel-Inhabers  
 4384 gegenüber der qualifizierten elektronischen Signaturerstellungseinheit“ (siehe [BSI-  
 4385 TR03114]).

4386 **TIP1-A\_4669 - QES-Stapelsignatur**

4387 Der Signatordienst MUSS die Möglichkeit bieten, Dokumente eines Stapels einzeln  
 4388 qualifiziert elektronisch zu signieren. Der Signatordienst MUSS als qualifizierte  
 4389 elektronische Signaturerstellungseinheit für die Stapelsignatur den HBA unterstützen.

4390 [**<=**]4391 **TIP1-A\_5664 - Reihenfolge der Dokumente bei Stapelsignatur**

4392 Die zu signierenden Dokumente einer Stapelsignatur MÜSSEN vom Signatordienst im  
 4393 Konnektor in derselben Reihenfolge signiert, in der sie im Signaturauftrag vom  
 4394 Clientsystem geschickt werden.

4395 [**<=**]4396 **TIP1-A\_4670 - Secure Messaging für die DTBS**

4397 Bei der Stapelsignatur MUSS der Signatordienst die zu signierenden Daten (DTBS) über  
 4398 Secure Messaging vom Konnektor zum HBA übertragen. Dieser Secure Messaging-Kanal  
 4399 MUSS über die gSMC-K zum HBA mittels C.SAK.AUTD\_CVC aufgebaut werden.

4400 [**<=**]4401 **TIP1-A\_4671 - Verhalten des Konnektors beim Abbruch einer Stapelsignatur**

4402 Der Signatordienst MUSS dem Benutzer während und nach einer PIN-Eingabe die  
 4403 Möglichkeit zum Abbruch einer Stapelsignatur anbieten.

4404 Das geforderte Verhalten des Konnektors beim Abbruch einer Stapelsignatur wird in der

4405 folgenden Tabelle beschrieben. Hierbei werden die beiden Punkte „Abbruch, während die  
 4406 erneute PIN-Eingabe angefordert wird“ (Nummer 1 bis 4) und „Abbruch, während der  
 4407 Vorgang der Signaturerstellung läuft“ (Nummer 5 bis 6) unterschieden. Zeile Nummer 7  
 4408 beschreibt alle sonstigen Fehlerfälle.  
 4409 Ein Teilstapel einer Stapelsignatur ist durch die maximale Anzahl der Dokumente  
 4410 definiert, welche nach der Eingabe der Signatur-PIN durch den Signaturschlüssel-Inhaber  
 4411 signiert werden kann.  
 4412

4413 **Tabelle 216: TAB\_KON\_192 Verhalten des Konnektors beim Abbruch einer Stapelsignatur**

| Nummer   | Problem/Fehler/Ereignis  | Verhalten des Konnektors  |
|--|--|---|
| Während die erneute PIN-Eingabe angefordert wird | 1 Timeout bei der PIN-Eingabe am KT                            | Der Signaturvorgang (Stapel) wird <u>beendet</u> .<br>Kein „Fehler“<br>Die Signaturen des/der vorherigen Teilstapel(s) bleiben erhalten und werden an das Clientsystem zurückgegeben. Keine weiteren Signaturen des neuen Teilstapels werden erstellt. (Die Weiterverarbeitung bereits erstellter Signaturen des letzten Teilstapels (sofern vorhanden) wird noch abgeschlossen). |
|  | 2 PIN gesperrt (nach mehrfacher Fehleingabe)                   | Siehe Verhalten unter Nummer 1  |
|  | 3 Abbruchkommando „StopSignature“ zur Jobnummer wird empfangen | Der Signaturvorgang (Stapel) wird <u>beendet</u> .<br>Kein „Fehler“<br>Keine weiteren Signaturen des neuen Teilstapels werden erstellt. (Die Weiterverarbeitung bereits erstellter Signaturen des letzten Teilstapels (sofern vorhanden) wird noch abgeschlossen).  |
|  | 4 Abbruchtaste am Kartenterminal wird gedrückt                 | Siehe Verhalten unter Nummer 1  |
| während der Vorgang der Signaturerstellung läuft | 5 Abbruchkommando „StopSignature“ zur Jobnummer wird empfangen | Signaturvorgang (Stapel) wird <u>abgebrochen</u> .<br>Kein „Fehler“<br>Keine weiteren Signaturen des Stapels werden erstellt.<br>Keine weiteren Signaturen des Teilstapels werden erstellt.<br>Bisher erstellte Signaturen des  |

|  |   |   |  |
|--|---|---|--|
|  |   |   | aktuellen Teilstapels werden verworfen.  |
|  | 6 | Abbruchtaste am Kartenterminal wird gedrückt.   | Die „Abbruch“-Taste wird nicht vom Signatordienst fortlaufend überwacht → Keine Aktion seitens des Signatordienstes.   |
|  | 7 | Bei allen anderen Fehlerfällen (z. B.: es kommen zu viele Signaturen zurück, der Hash-Wert einer der Signaturen stimmt nicht, Karte gezogen, etc) | Signaturvorgang (Stapel) wird abgebrochen.<br>Schwerer Fehler.<br>Keine weiteren Signaturen des Stapels werden erstellt.<br>Keine weiteren Signaturen des aktuellen Teilstapels werden erstellt.<br>Bisher erstellte Signaturen aller Teilstapel werden verworfen.<br>Es handelt sich um Probleme/Fehlerfälle, die bei typischen Angriffen auftreten können. |

4414

4415 [ $\leq$ ]

4416 4.1.8.4.6 TUC\_KON\_151 „QES Dokumentensignatur prüfen“

4417 **TIP1-A\_4672-02 - TUC\_KON\_151 „QES-Dokumentensignatur prüfen“**

4418 Der Konnektor MUSS den technischen Use Case TUC\_KON\_151 „QES-Dokumentensignatur prüfen“ umsetzen.

4420

4421 **Tabelle 217: TAB\_KON\_591 - TUC\_KON\_151 „QES-Dokumentensignatur prüfen“**

| Element             | Beschreibung   |
|---------------------|--|
| Name                | TUC_KON_151 „QES-Dokumentensignatur prüfen“  |
| Beschreibung        | Es wird die QES eines Dokuments geprüft. Dabei werden die Signaturverfahren laut Tabelle TAB_KON_582 – Signaturverfahren unterstützt. Sind mehrere Signaturen vorhanden, so werden alle geprüft. Auch Parallel- und Gegensignaturen MÜSSEN unterstützt werden. |
| Eingangsanforderung | keine  |
| Auslöser            | Aufruf durch ein Clientsystem (Operation VerifyDocument) oder durch ein Fachmodul im Konnektor   |
| Vorbedingungen      | keine  |



|                |   |
|----------------|---|
| Eingangsdaten  | <ul style="list-style-type: none"> <li>signedDocument – <i>optional</i><br/>(QES-signiertes Dokument vom Typ QES_DocFormate<br/>-&gt; siehe Definition in Operation VerifyDocument mit<br/>SIG:Document)</li> <li>signatureObject – <i>optional</i><br/>( -&gt; siehe Definition in Operation VerifyDocument mit<br/>dss:SignatureObject.<br/>Es werden Parallel- und Gegensignaturen unterstützt.)</li> <li>optionalInputParams<br/>(optionale Eingabeparameter, siehe Operation<br/>VerifyDocument, Parameter SIG:OptionalInputs)</li> <li>certificates – <i>optional/falls diese nicht im signierten<br/>Dokument enthalten sind, sondern nur referenziert<br/>werden</i><br/>(X.509-Zertifikate ).</li> <li>xmlSchemas – <i>optional/nur für XML-Dokumente</i><br/>(XMLSchema und ggf. weitere vom Hauptschema<br/>benutzte Schemata)</li> <li>includeRevocationInfo [Boolean]: – <i>optional; Default:<br/>false</i><br/>(Dieser optionale Parameter steuert die Einbettung<br/>von OCSP Antworten in die Signatur)</li> </ul> |
| Komponenten    | Konnektor   |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>verificationResult [VerificationResult]<br/>(Ergebnis der Signaturprüfung)</li> <li>optionalOutput – <i>optional</i><br/>(weitere Ausgabedaten gemäß SIG:OptionalOutput)</li> </ul>  |
| Standardablauf | <p><b>1. „DocumentValidation“:</b> Das signierte Dokument wird validiert mit Aufruf TUC_KON_080 „Dokument validieren“{ ... }.</p> <p>Treten Fehler bei der Validierung der Typkonformität auf, wenn die Signatur im Dokument eingebettet ist, wird die Prüfung mit einem Fehler abgebrochen. Treten bei der Typkonformität, wenn die Signatur nicht im Dokument eingebettet ist, Fehler auf, so bricht der TUC nicht ab, sondern führt die folgenden Schritte soweit sinnvoll möglich durch. (Die Entscheidung über das sinnvoll Durchführbare liegt beim Hersteller des Konnektors.)</p> <p><b>2. „CoreValidation“:</b></p> <p>Es erfolgt die mathematische Prüfung der Signatur, bestehend aus der Prüfung der Hash-Kette bis zum signierten Hashwert und der Prüfung der Signatur unter Verwendung des öffentlichen Schlüssels, des Signaturwertes und des signierten Hashwertes.</p>  |

|  |   |
|--|---|
|  | <p>XML-Signatur:<br/>Die Core Validation erfolgt entsprechend [XMLDSig] Kapitel 3.2 Core Validation.</p> <p>CMS-Signatur:<br/>Die Core Validation erfolgt entsprechend Cryptographic Message Syntax (CMS) Kapitel 5.6 Signature Verification Process [RFC5652].</p> <p>PDF-Signatur:<br/>Die Core Validation erfolgt entsprechend [PAdES-3] Kapitel 4.6 Signature Validation aus PAdES-BES Part 3.</p> <p>Auch wenn die Validierung fehlschlägt, werden die folgenden Prüfschritte durchgeführt, so dass ein vollständiges Prüfprotokoll erstellt werden kann.</p> <p><b>3. „CheckSignatureCertificate“:</b></p> <p><b>Teil 1: Signaturzertifikat ermitteln</b></p> <p>XML-Signatur:<br/>Das Signaturzertifikat ist im XMLDSig Element <code>ds:KeyInfo/ds:X509Data</code> gespeichert [XMLDSig] oder wird als Eingangsparemeter übergeben.</p> <p>CMS-Signatur:<br/>Das Signaturzertifikat für CADES ist im Feld <code>certificates</code> im <code>SignedData</code> Container gespeichert [CADES] oder wird als Eingangsparemeter übergeben.</p> <p>PDF-Signatur:<br/>Das PDF Signaturzertifikat für PAdES ist im Feld <code>SignedData.certificates</code> entsprechend Kapitel 6.1.1 „Placements of the signing certificate“ [PAdES Baseline Profile] gespeichert oder wird als Eingangsparemeter übergeben.</p> <p><b>Teil 2: Signaturzeitpunkt bestimmen</b></p> <p>Der Signaturzeitpunkt <code>Ermittelter_Signaturzeitpunkt_Eingebettet</code> wird wie folgt selektiert:</p> <p>XML-Signatur:<br/>Das XML element <code>SigningTime</code> spezifiziert den Signaturzeitpunkt entsprechend Kapitel 7.2.1 XAdES [XAdES].</p> <p>CMS-Signatur:<br/>Das Attribut <code>SigningTime</code> spezifiziert den Signaturzeitpunkt entsprechend Kapitel 11.3 CMS [CMS].</p> <p>PDF-Signatur:<br/>Der Signaturzeitpunkt kann dem M Eintrag des Signature Dictionary entnommen werden [PAdES Baseline Profile] Kapitel</p> |
|--|---|

|  |  |
|--|--|
|  | <p>6.2.1 Signing time.</p> <p>Der Signaturzeitpunkt <code>Benutzerdefinierter_Zeitpunkt</code> liegt gegebenenfalls als Aufrufparameter vor. Der Signaturzeitpunkt <code>Ermittelter_Signaturzeitpunkt_System</code> wird ermittelt.</p> <p><b>Teil 3: Signaturzertifikatsprüfung:</b><br/> Bei der folgenden Signaturzertifikatsprüfung sind die Signaturzeitpunkte gemäß [TIP1-A_5540] zu berücksichtigen. Die Signaturzertifikatsprüfung erfolgt durch Aufruf von TUC_KON_037 „Zertifikat prüfen“ {<br/> certificate = C.HP.QES;<br/> qualifiedCheck = required;<br/> baseTime = Signaturzeitpunkt;<br/> offlineAllowNoCheck = true;<br/> validationMode = OCSP;<br/> ocspResponses = OCSP-Response;<br/> getOCSPResponses = includeRevocationInfo<br/> }.<br/> Sind OCSP-Responses in der Signatur eingebettet, ist die jüngsten OCSP-Response des EE-Zertifikats, die für die Zertifikatsprüfung notwendig ist, beim Aufruf von TUC_KON_037 zu übergeben.<br/> Sofern der Aufruf von TUC_KON_037 ocspResponses zurückgibt, wird die OCSP-Response des EE-Zertifikats in die Signatur eingebettet.<br/> Auch wenn die Zertifikatsprüfung fehlschlägt, werden die folgenden Prüfungen durchgeführt.</p> <p><b>4. „CheckPolicyConstraints“:</b></p> <p>In diesem Schritt wird das signierte Dokument entsprechend der Profilierung der Signaturformate (siehe Anhang B.2) geprüft. Es sind die Vorgaben für die Prüfung von Signaturen aus den Standards für AdES [XAdES], [XAdES Baseline], [CAAdES], [CAAdES Baseline], [PAdES-3] und [PAdES Baseline] umzusetzen. Dabei sind die Vorgaben aus Tabelle TAB_KON_779 „Profilierung der Signaturformate“ und Tabelle TAB_KON_778 „Einsatzbereich der Signaturvarianten“ zu erfüllen.</p> <p>Auch wenn nicht alle Anforderungen an das Format des signierten Dokuments erfüllt werden, wird die Prüfung mit den folgenden Schritten fortgesetzt, um ein vollständiges Prüfungsprotokoll zu erhalten.</p> <p><b>5. Das Prüfergebnis</b> (VerificationResult, OptionalOutput) wird an den Aufrufer zurückgegeben</p> |
|--|--|

|                                |   |
|--------------------------------|---|
|                                | (siehe TAB_KON_593 Übersicht Status für Prüfung einer Dokumentensignatur).  |
| Varianten/Alternativen         | Keine   |
| Fehlerfälle                    | <p>Das Verhalten des TUCs bei einem Fehlerfall ist in TAB_KON_592 Fehlercodes TUC_KON_151 „QES Dokumentensignatur prüfen“ beschrieben.</p> <p>(-&gt;1) keine Signatur in signedDocument und signatureObject vorhanden: 4253.</p> <p>(→ 2 „<b>CoreValidation</b>“) Interner Fehler: 4001, Signatur des Dokuments ungültig: 4115, Signatur umfasst nicht das gesamte Dokument: 4262</p> <p>(→3 „<b>CheckSignatureCertificate</b>“) Interner Fehler: 4001, Signaturzertifikat ermitteln ist fehlgeschlagen: 4206.</p> <p>(→4 „<b>CheckPolicyConstraints</b>“) Interner Fehler: 4001, Dokument nicht konform zu Regeln für QES: 4124, Dokument nicht konform zu Profilierung der Signaturformate: 4208.</p> |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | keine   |

4422

4423

4424

**Tabelle 218: TAB\_KON\_592 Fehlercodes TUC\_KON\_151 „QES Dokumentensignatur prüfen“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4001  | Technical | Error    | interner Fehler  |
| 4115  | Security  | Error    | Signatur des Dokuments ungültig. Prüfung der Hashwertkette bzw. Prüfung der kryptographischen Signatur fehlgeschlagen. |
| 4124  | Technical | Error    | Dokument nicht konform zu Regeln für QES   |
| 4206  | Technical | Error    | Signaturzertifikat ermitteln ist fehlgeschlagen  |
| 4208  | Technical | Error    | Dokument nicht konform zu Profilierung der Signaturformate   |
| 4253  | Technical | Error    | Keine Signatur im Aufruf   |
| 4262  | Technical | Error    | Signatur umfasst nicht das gesamte Dokument  |
| 4264  | Technical | Warning  | Ein oder mehrere Zertifikate ignoriert   |

4425

4426

4427

Das Gesamtergebnis (VerificationResult) für die Prüfung einer Dokumentensignatur fasst die Ergebnisse aller Prüfungsschritte in einem einzelnen Statuswert zusammen.

4428 **Tabelle 219: TAB\_KON\_593 Übersicht Status für Prüfung einer Dokumentensignatur**

| VerificationResult für gesamtes Dokument (VerificationResult/HighLevelResult) |   |
|---|---|
| Wert  | Bedeutung   |
| VALID   | Wenn VerificationResult für alle Signaturen zum Dokument VALID  |
| INVALID   | Wenn VerificationResult für eine Signatur zum Dokument INVALID  |
| INCONCLUSIV<br>E  | in allen anderen Fällen   |
| VerificationResult pro Signatur (VerificationReport/IndividualReport/Result)  |   |
| Wert  | Bedeutung<br>mögliche Ausprägungen im VerificationReport  |
| VALID   | Die Signatur wurde gemäß den Regeln für die QES geprüft und für gültig befunden.  |
|   | ResultMajor = urn:oasis:names:tc:dss:1.0:resultmajor:Success<br>ResultMinor = urn:oasis:names:tc:dss:1.0:resultminor:valid:signature:OnAllDocuments     |
|   | ResultMajor = urn:oasis:names:tc:dss:1.0:resultmajor:Success<br>ResultMinor = urn:oasis:names:tc:dss:1.0:resultminor:valid:signature:HasManifestResults |
| INVALID   | Die Signatur ist ungültig oder aufgrund eines Fehlers konnte die Signaturprüfung nicht durchgeführt werden.   |
|   | ResultMajor = urn:oasis:names:tc:dss:1.0:resultmajor:Success<br>ResultMinor = urn:oasis:names:tc:dss:1.0:resultminor:invalid:IncorrectSignature         |
|   |   |
|   | ResultMajor = urn:oasis:names:tc:dss:1.0:resultmajor:RequesterError   |
|   | ResultMajor = urn:oasis:names:tc:dss:1.0:resultmajor:ResponderError   |
|   | ResultMajor = urn:oasis:names:tc:dss:1.0:resultmajor:InsufficientInformation  |

|              |   |
|--------------|---|
|              | ResultMinor =<br>urn:oasis:names:tc:dss:1.0:resultminor:invalid:IncorrectSignature  |
|              | ResultMajor =<br>urn:oasis:names:tc:dss:1.0:resultmajor:InsufficientInformation<br>ResultMinor =<br>urn:oasis:names:tc:dss:1.0:resultminor:CertificateChainNotComplete  |
| INCONCLUSIVE | <p>Die Signatur wurde gemäß den Regeln für die QES geprüft. Allerdings konnten eine oder mehrere Prüfungen nicht vollständig durchgeführt werden. Einzelheiten finden sich in Result-Detail.<br/>Die Prüfungen, die durchgeführt werden konnten, waren erfolgreich.</p> <p>ResultMajor =<br/>urn:oasis:names:tc:dss:1.0:resultmajor:InsufficientInformation<br/>ResultMinor =<br/>urn:oasis:names:tc:dss:1.0:resultminor:OcspNotAvailable</p> <p>Hinweis: Das Erreichen dieses Zustandes hängt davon ab, ob eine OCSP-Abfrage nicht durchgeführt werden konnte, unabhängig davon, ob die Ursache dafür die Offlineschaltung des Konnektors (MGM_LU_ONLINE = Disabled) oder die Nichterreichbarkeit des OCSP-Responders im Online-Betrieb (MGM_LU_ONLINE = Enabled) ist.</p> |

4429

4430 [ $\leq$ ]4431 **TIP1-A\_5540-01 - QES-Signaturprüfergebnis bezogen auf Signaturzeitpunkt**4432 Der Konnektor MUSS zur QES-Signaturprüfung ein Prüfergebnis, das sich auf genau  
4433 einen angenommenen Signaturzeitpunkt bezieht, an den Aufrufer zurückgeben.4434 Die Auswahl des angenommenen Signaturzeitpunkts, auf den sich das Signaturergebnis  
4435 bezieht, erfolgt hierarchisch:

- 4436 • Benutzerdefinierter\_Zeitpunkt
- 4437 falls vorhanden, sonst
- 4438 • Ermittelter\_Signaturzeitpunkt\_Eingebettet
- 4439 falls vorhanden, sonst
- 4440 • Ermittelter\_Signaturzeitpunkt\_System

4441 [ $\leq$ ]4442 **4.1.8.4.7 TUC\_KON\_170 „Dokumente mit Komfort signieren“**4443 **A\_19103-01 - TUC\_KON\_170 "Dokumente mit Komfort signieren"**4444 Der Konnektor MUSS den technischen Use Case TUC\_KON\_170 „Dokumente mit Komfort  
4445 signieren“ umsetzen.

4446

4447 **Tabelle 220: TAB\_KON\_871 – TUC\_KON\_170 „Dokumente mit Komfort signieren“**

| Element | Beschreibung                                  |
|---------|---|
| Name    | TUC_KON_170 "Dokumente mit Komfort signieren" |

|                |  |
|----------------|--|
| Beschreibung   | Im Rahmen von Fachanwendungen werden ein oder mehrere Dokumente mit einer Komfortsignatur versehen. Es werden die QES_DocFormate unterstützt.  |
| Auslöser       | Aufruf durch ein Clientsystem (Operation SignDocument) oder ein Fachmodul.   |
| Vorbedingungen | Die Signaturkarte muss gesteckt sein.  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>signRequests<br/>(Liste von Signaturaufträgen)<br/>Jeder Signaturauftrag (SignRequest) kapselt: <ul style="list-style-type: none"> <li>documentsToBeSigned<br/>(Zu signierendes Dokument bzw. zu signierende Dokumente);<br/>darin u.a.<br/>documentFormat<br/>(Formatangabe für das zu signierende Dokument)</li> <li>optionalInputs<br/>(weitere optionale Eingabeparameter zur Steuerung der Details bei der zu erstellenden Signatur, siehe Operation SignDocument, Parameter dss:OptionalInputs); darin u.a.<br/>signatureType (URI für den Signatortyp XML-, CMS-, PDF-Signatur)</li> <li>includeRevocationInfo [Boolean]: – optional; Default: true<br/>(Dieser optionale Parameter steuert die Einbettung von OCSP Antworten in die Signatur; siehe Operation SignDocument, Parameter SIG:IncludeRevocationInfo)</li> </ul> </li> <li>cardSession<br/>(Kartensitzung. Unterstützte Kartentypen: HBA)</li> <li>crypt [SIG_CRYPT_QES] - <i>optional</i>;<br/>default und Wertebereich: siehe TAB_KON_862<br/>(Dieser Parameter steuert, ob RSA-basierte oder ECC-basierte Signaturen erzeugt werden.)</li> <li>workplaceId</li> </ul> |
| Komponenten    | Konnektor, Kartenterminal, Signaturkarte (HBA)   |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>signedDocuments<br/>(Liste der signierten Dokumente)</li> </ul>   |
| Standardablauf | <p>Der Konnektor KANN die Schritte 1 bis 4 in einer beliebigen Reihenfolge durchführen.</p> <ol style="list-style-type: none"> <li>1. Prüfe SAK_COMFORT_SIGNATURE = Enabled</li> <li>2. Der Signatortyp und die Signaturvariante werden für jedes Dokument der Liste entsprechend signatureType und SignatureVariant festgelegt (ggf. in optionalInputs enthalten). Wenn SignatureType oder SignatureVariant nicht übergeben wurden, wird das dem Dokumentformat entsprechende Default-Verfahren gewählt (siehe TAB_KON_583 – Default-Signaturverfahren).</li> </ol>   |

|                            |  |
|----------------------------|--|
|                            | <p>3. Für alle Dokumente des Stapels wird die Zulässigkeit des Kartentyps geprüft. Das für die Signatur zu nutzende Zertifikat wird anhand des Kartentyps und des Parameters crypt ausgewählt.</p> <p>4. Es werden die Voraussetzungen für die Signatur geprüft. Dies erfolgt im TUC_KON_152 „Signaturvoraussetzungen für QES prüfen“.<br/>Wenn includeRevocationInfo=true, dann setze ocsResponses auf Rückgabewert von TUC_KON_152.</p> <p>5. Die am Signaturvorgang beteiligte Ressource Signaturkarte wird für die exklusive Nutzung durch diesen Signaturvorgang reserviert. Die Reservierung der Signaturkarte erfolgt durch Aufruf von<br/>TUC_KON_023 „Karte reservieren“ {<br/>    cardSession;<br/>    doLock = true }.</p> <p>6. Zum Vorbereiten der Dokumente für die Signatur wird TUC_KON_155 „Dokumente zur Signatur vorbereiten“ mit ocsResponses aufgerufen.</p> <p>Die Zugriffe auf die Signaturkarte im Schritt 7 müssen im DF.QES erfolgen. DF.QES darf am Ende des TUCs nicht verlassen werden.</p> <p>7. Die Signaturen werden erstellt. Dies erfolgt gemäß TUC_KON_158 „Komfortsignaturen erstellen“.</p> <p>8. Die reservierte Ressource Signaturkarte wird wieder freigegeben. Zur Freigabe der Signaturkarte wird TUC_KON_023 „Karte reservieren“<br/>    cardSession;<br/>    doLock = false }<br/>aufgerufen.</p> <p>9. Die signierten Dokumente werden an den Aufrufer zurückgegeben.</p> |
| Varianten/<br>Alternativen | keine  |
| Fehlerfälle                | <p>Fehler in den folgenden Schritten des Standardablaufs führen zum Abbruch mit den ausgewiesenen Fehlercodes:</p> <p>(-&gt;1) Komfortsignaturfunktion im Konnektor nicht aktiviert:<br/>Fehlercode 4263</p> <p>(-&gt;2) Ungültige Angabe des Signaturtyps oder Signaturvariante:<br/>Fehlercode 4111<br/>Übergabe eines für die QES nicht unterstützten Dokumentformats:<br/>Fehlercode 4110</p> <p>(-&gt;3) Kartentyp nicht zulässig für Signatur: Fehlercode 4126</p> <p>(-&gt;5) Fehler bei der Reservierung der Signaturkarte: Fehlercode 4060</p> <p>(-&gt;7) Karte ist kein HBA, sondern HBA-Vorläuferkarte: Fehlercode 4274</p> <p>Im Fehlerfall:</p>  |



|                          |  |
|--------------------------|--|
|                          | a) ... DARF DF.QES NICHT verlassen werden<br>b) ... MÜSSEN alle reservierten Ressourcen freigegeben werden<br>c) ... MUSS der Fehler immer an das Clientsystem zurückgemeldet werden     |
| Sicherheitsanforderungen | Der Konnektor MUSS sicherstellen, dass der erhöhte Sicherheitszustand der PIN.QES nur für die Komfortsignatur mittels TUC_KON_170 innerhalb einer Kartensitzung nachgenutzt werden darf. |

4448 **Tabelle 221: TAB\_KON\_872 Fehlercodes TUC\_KON\_170 „Dokumente mit Komfort**  
4449 **signieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4060  | Technical | Error    | Ressource belegt  |
| 4110  | Technical | Error    | ungültiges Dokumentformat (%Format%)<br>Der Parameter Format enthält das übergebene Dokumentformat. |
| 4111  | Technical | Error    | ungültiger Signaturtyp oder Signaturvariante  |
|   |           |          |   |
| 4126  | Security  | Error    | Kartentyp nicht zulässig für Signatur   |
| 4049  | Technical | Error    | Abbruch durch den Benutzer  |
| 4263  | Technical | Error    | Komfortsignaturfunktion nicht aktiviert   |
| 4274  | Technical | Error    | Komfortsignaturen werden nur für den HBA unterstützt  |

4450 [**<=**]

4451 4.1.8.4.8 TUC\_KON\_171 „Komfortsignatur einschalten“

4452 **A\_19104-01 - TUC\_KON\_171 „Komfortsignatur einschalten“**

4453 Der Konnektor MUSS den technischen Use Case TUC\_KON\_171 „Komfortsignatur  
4454 einschalten“ umsetzen.

4455

4456 **Tabelle 222: TAB\_KON\_883 – TUC\_KON\_171 „Komfortsignatur einschalten“**

| Element      | Beschreibung   |
|--------------|--|
| Name         | TUC_KON_171 „Komfortsignatur einschalten“  |
| Beschreibung | Zum Einschalten des Komfortsignaturmodus wird die PIN.QES verifiziert und der Signaturmodus „Comfort“ für die cardSession gesetzt. |
| Auslöser     | <ul style="list-style-type: none"> <li>Operation ActivateComfortSignature</li> <li>Aufruf durch ein Fachmodul</li> </ul>           |

|                            |   |
|----------------------------|---|
| Vorbedingungen             | Der Karte muss gesteckt sein.   |
| Eingangsdaten              | <ul style="list-style-type: none"> <li>cardSession (nur HBA erlaubt)</li> </ul>   |
| Komponenten                | Konnektor, Kartenterminal, Karte (HBA)  |
| Ausgangsdaten              | <ul style="list-style-type: none"> <li>signatureMode</li> </ul>   |
| Standardablauf             | <ol style="list-style-type: none"> <li>1. Prüfe <code>SAK_COMFORT_SIGNATURE = Enabled</code></li> <li>2. Die am Vorgang beteiligten Ressourcen (Karte sowie PIN-Pad und Display des PIN-Eingabe-Kartenterminals) werden für die exklusive Nutzung durch diesen Vorgang reserviert. Die Reservierung der Karte erfolgt durch Aufruf von<br/> <code>TUC_KON_023 „Karte reservieren“ {</code><br/>             <code>cardSession;</code><br/>             <code>doLock = true }</code> <p>Der Zugriff auf die Karte im Schritt 3 muss im DF.QES erfolgen. Das DF.QES darf danach nicht verlassen werden, damit der PIN-Status der PIN.QES erhalten bleibt.</p> </li> <li>3. Die Einschaltung der Komfortsignatur wird durch den Anwender autorisiert. Dies erfolgt durch Aufruf von<br/> <code>TUC_KON_012 „PIN verifizieren“ {</code> <code>cardSession;</code><br/>             <code>workplaceId;</code><br/>             <code>pinRef = PIN.QES;</code><br/>             <code>verificationType = Mandatorisch }</code> <p>Für die Anzeige am Kartenterminal ist die Displaymessage für „Komfortsignatur aktivieren“ aus TAB_KON_090 zu verwenden.</p> </li> <li>4. Setze <code>CARDSESSION.SIGNMODE = Comfort</code></li> <li>5. Starte Komfortsignatur-Timer für die cardSession bei „0“</li> <li>6. Die reservierten Ressourcen (Karte sowie PIN-Pad und Display des PIN-Eingabe-Kartenterminals) werden wieder freigegeben.<br/> Zur Freigabe der Karte wird <code>TUC_KON_023</code><br/>             <code>„Karte reservieren“</code><br/>             <code>cardSession;</code><br/>             <code>doLock = false }</code><br/>         aufgerufen. </li> </ol> |
| Varianten/<br>Alternativen | Keine   |
| Fehlerfälle                | <p>Fehler in den folgenden Schritten des Standardablaufs führen zum Abbruch mit den ausgewiesenen Fehlercodes:</p> <ul style="list-style-type: none"> <li>(-&gt;1) Komfortsignaturfunktion im Konnektor nicht aktiviert: Fehlercode 4263</li> <li>(-&gt;2) Fehler bei der Reservierung von Ressourcen: Fehlercode 4060</li> <li>(-&gt;3) Karte ist kein HBA, sondern HBA-Vorläuferkarte: Fehlercode 4274</li> <li>(-&gt;4) Fehler beim Setzen des Signaturmodus: Fehlercode 4267</li> </ul>   |

|  |  |
|--|--|
|  | (->5) Fehler beim Starten des Komfortsignatur-Timers:<br>Fehlercode 4267<br>Im Fehlerfall, inklusive Timeout bei der PIN-Eingabe, oder<br>bei Abbruch durch den Benutzer (Fehler 4049):<br>a) ... MUSS (ab Schritt 3) DF.QES verlassen werden<br>b) ... MÜSSEN alle reservierten Ressourcen freigegeben<br>werden<br>c) ... MUSS der Fehler immer an das Clientsystem<br>zurückgemeldet werden |
|--|--|

4457 **Tabelle 223: TAB\_KON\_886 Fehlercodes TUC\_KON\_171 „Komfortsignatur einschalten“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4049  | Technical | Error    | Abbruch durch den Benutzer                                   |
| 4060  | Technical | Error    | Ressource belegt   |
| 4263  | Technical | Error    | Komfortsignaturfunktion nicht aktiviert                      |
| 4267  | Technical | Error    | Fehler beim Aktivieren des Komfortsignaturmodus <cardHandle> |
| 4274  | Technical | Error    | Komfortsignaturen werden nur für den HBA unterstützt         |

4458 [ $\leq$ ]

4459 4.1.8.4.9 TUC\_KON\_172 „Komfortsignatur ausschalten“

4460 **A\_19105 - TUC\_KON\_172 „Komfortsignatur ausschalten“**

4461 Der Konnektor MUSS den technischen Use Case TUC\_KON\_172 „Komfortsignatur  
4462 ausschalten“ umsetzen.

4463

4464 **Tabelle 224: TAB\_KON\_884 – TUC\_KON\_172 „Komfortsignatur ausschalten“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_172 „Komfortsignatur ausschalten“   |
| Beschreibung   | Zum Ausschalten des Komfortsignaturmodus werden die Sicherheitszustände der Karte(n), die im Konnektor verwalteten Sicherheitszustände und der Signaturmodus der cardSession(s) zurückgesetzt.                            |
| Auslöser       | <ul style="list-style-type: none"> <li>• Operation DeactivateComfortSignature</li> <li>• TUC_KON_158</li> <li>• Der Administrator setzt SAK_COMFORT_SIGNATURE = Disabled</li> <li>• Aufruf durch ein Fachmodul</li> </ul> |
| Vorbedingungen | Die Karten müssen gesteckt sein.  |

|                            |   |
|----------------------------|---|
| Eingangsdaten              | Bei Auslösen des TUCs durch den Administrator: <ul style="list-style-type: none"> <li>Keine</li> </ul> Ansonsten: <ul style="list-style-type: none"> <li>cardHandles : Liste von cardHandles (nur HBA erlaubt)</li> </ul>   |
| Komponenten                | Konnektor, Kartenterminal, Karte (HBA)  |
| Ausgangsdaten              | Keine   |
| Standardablauf             | 1. Wenn der TUC <u>nicht</u> durch den Administrator ausgelöst wurde:<br>Prüfe <code>SAK_COMFORT_SIGNATURE = Enabled</code><br>2. Wenn der TUC durch den Administrator ausgelöst wurde: Ermittle die cardHandles aller gesteckten HBA.<br>3. Für jedes übergebene bzw. ermittelte cardHandle:<br>4. Ermittle cardSessions zu cardHandle<br>5. Für jede ermittelte cardSession: <ol style="list-style-type: none"> <li>Setze den PIN-Status der PIN.QES zurück (z. B. durch Verlassen von DF.QES für alle logischen Kanäle der Karte)</li> <li>Lösche den im Konnektor verwalteten Sicherheitszustand aus <code>CARDSESSION.AUTHSTATE</code> (<code>PINRef=PIN.QES</code>)</li> <li>Setze <code>CARDSESSION.SIGNMODE = PIN</code></li> <li>Stoppe Komfortsignatur-Timer für die cardSession</li> </ol> |
| Varianten/<br>Alternativen | Keine   |
| Fehlerfälle                | (->1) Komfortsignaturfunktion im Konnektor nicht aktiviert: Fehlercode 4263<br>Fehler und Warnungen in den folgenden Schritten werden über alle cardHandle akkumuliert und die <komma-separierte Liste von cardHandle> für den jeweiligen Fehlertext erzeugt.<br>(->3) Bei einem ungültigen cardHandle wird mit dem nächsten cardHandle aus cardHandles fortgesetzt. Fehlercode 4265<br>(->4) Ist zu einem cardHandle keine cardSession vorhanden wird mit dem nächsten cardHandle fortgesetzt. Fehlercode 4266<br>(->5) Tritt in Schritt 4 ein Fehler auf wird mit dem nächsten cardHandle fortgesetzt. Fehlercode 4268  |

4465

4466

**Tabelle 225: TAB\_KON\_887 Fehlercodes TUC\_KON\_172 „Komfortsignatur ausschalten“**

| Fehlercode | ErrorType | Severity | Fehlertext                              |
|------------|-----------|----------|---|
| 4263       | Technical | Fehler   | Komfortsignaturfunktion nicht aktiviert |

|      |           |         |   |
|------|-----------|---------|---|
| 4265 | Technical | Warning | Karten-Handle ungültig <komma-separierte Liste von cardHandle>                            |
| 4266 | Technical | Warning | Keine Kartensitzung vorhanden <komma-separierte Liste von cardHandle>                     |
| 4268 | Technical | Fehler  | Fehler beim Deaktivieren des Komfortsignaturmodus <komma-separierte Liste von cardHandle> |

4467 [ $\leq$ ]

4468

4469 4.1.8.4.10 TUC\_KON\_173 „Liefere Signaturmodus“

4470 **A\_19106 - TUC\_KON\_173 „Liefere Signaturmodus“**4471 Der Konnektor MUSS den technischen Use Case TUC\_KON\_173 „Liefere  
4472 Signaturmodus“ umsetzen.

4473

4474 **Tabelle 226: TAB\_KON\_885 – TUC\_KON\_173 „Liefere Signaturmodus“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_173 „Liefere Signaturmodus“  |
| Beschreibung   | Der aktuell konfigurierte Status der Komfortsignaturfunktion im Konnektor und der aktuelle Signaturmodus für alle dem Konnektor bekannten Aufrufkontexte zu den übergebenen HBA-CardHandles wird ermittelt und an den Aufrufer zurückgegeben.  |
| Auslöser       | <ul style="list-style-type: none"> <li>• Operation GetSignatureMode</li> <li>• Aufruf durch ein Fachmodul</li> </ul>   |
| Vorbedingungen | Keine  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• Liste von cardHandles (nur HBA erlaubt)</li> </ul>  |
| Komponenten    | Konnektor, Kartenterminal, Signaturkarte (HBA)   |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>• comfortSignatureStatus</li> <li>• comfortSignatureMax</li> <li>• comfortSignatureTimer</li> <li>• signatureModes: Struktur aus</li> <li>• Liste von cardHandles (nur HBA erlaubt) <ul style="list-style-type: none"> <li>• Liste von Tupeln (signatureContext, signatureMode, countRemaining, timeRemaining)</li> </ul> </li> </ul> |
| Standardablauf | <ol style="list-style-type: none"> <li>1. Ermittle den Status der Komfortsignaturfunktion:<br/>comfortSignatureStatus=SAK_COMFORT_SIGNATURE</li> <li>2. Ermittle comfortSignatureMax=<br/>SAK_COMFORT_SIGNATURE_MAX</li> </ol>   |

|                            |  |
|----------------------------|--|
|                            | <p>3. Ermittle comfortSignatureTimer=<br/>SAK_COMFORT_SIGNATURE_Timer</p> <p>4. Für jedes übergebene cardHandle:</p> <p>a. Ermittle cardSession zu cardHandle</p> <p>b. Für jede ermittelte cardSession:</p> <p>i. Ermittle den Kontext (signatureContext) der cardSession aus<br/>CARDSESSION.MANDANTID,<br/>CARDSESSION.CSID, CARDSESSION.USERID</p> <p>ii. Ermittle den Signaturmodus (signatureMode) aus<br/>CARDSESSION.SIGNMODE</p> <p>iii. Ermittle Differenz von<br/>SAK_COMFORT_SIGNATURE_MAX und<br/>Komfortsignatur-Zähler der cardSession<br/>(countRemaining)</p> <p>iv. Ermittle verbleibende Zeit<br/>aus SAK_COMFORT_SIGNATURE_TIMER und<br/>Komfortsignatur-Timer der cardSession<br/>(timeRemaining)</p> |
| Varianten/<br>Alternativen | <p>Wenn SAK_COMFORT_SIGNATURE = Disabled</p> <p>(-&gt;4 b iii) countRemaining = 0</p> <p>(-&gt;4 b iv) timeRemaining = 0</p>   |
| Fehlerfälle                | <p>(-&gt;2) Bei einem ungültigen cardHandle wird mit dem nächsten cardHandle aus cardHandles fortgesetzt. Fehlercode 4265</p> <p>(-&gt;2a) Ist zu einem cardHandle keine cardSession vorhanden wird mit dem nächsten cardHandle fortgesetzt. Fehlercode 4266</p> <p>(-&gt;2b) Tritt in Schritt 2b ein Fehler auf wird mit dem nächsten cardHandle fortgesetzt. Fehlercode 4269</p> <p>Die Fehler und Warnungen werden über alle cardHandle akkumuliert und die &lt;komma-separierte Liste von cardHandle&gt; für den jeweiligen Fehlertext erzeugt.</p>  |

4475

4476

**Tabelle 227: TAB\_KON\_888 Fehlercodes TUC\_KON\_173 „Liefere Signaturmodus“**

| Fehlercode | ErrorType | Severity | Fehlertext  |
|------------|-----------|----------|---|
| 4265       | Technical | Warning  | Karten-Handle ungültig <komma-separierte Liste von cardHandle>                  |
| 4266       | Technical | Warning  | Keine Kartensitzung vorhanden <komma-separierte Liste von cardHandle>           |
| 4269       | Technical | Error    | Fehler beim Ermitteln des Signaturmodus <komma-separierte Liste von cardHandle> |

4477

[&lt;=]

4478

#### 4.1.8.5 Operationen an der Außenschnittstelle

##### **TIP1-A\_4676-07** ~~TIP1-A\_4676-05~~ - Basisdienst Signaturdienst (nonQES und QES)

Der Konnektor MUSS Clientsystemen den Basisdienst Signaturdienst (nonQES und QES) anbieten.

**Tabelle 228: TAB\_KON\_197 Basisdienst Signaturdienst (nonQES und QES)**

|                          |   |   |
|--------------------------|---|---|
| <b>Name</b>              | SignatureService  |   |
| <b>Version (KDV)</b>     | 7.4.0 (WSDL-Version), 7.4.2 (XSD-Version)<br>7.4.2 (WSDL-Version), 7.4.4 (XSD-Version)<br>7.5.2 (WSDL- und XSD-Version)<br><a href="#">Siehe Anhang D</a> |   |
| <b>Namensraum</b>        | Siehe <a href="#">Anhang D</a> <a href="#">GitHub</a>   |   |
| <b>Namensraum-Kürzel</b> | SIG für Schema und SIGW für WSDL  |   |
| <b>Operationen</b>       | <b>Name</b>   | <b>Kurzbeschreibung</b>   |
|                          | SignDocument  | Dokument signieren  |
|                          | VerifyDocument  | Signatur verifizieren   |
|                          | StopSignature   | Signieren eines Dokumentenstapels abbrechen   |
|                          | GetJobNumber  | Liefert eine Jobnummer für den nächsten Signiervorgang  |
|                          | ActivateComfortSignature  | Aktiviert die Komfortsignatur für einen HBA   |
|                          | DeactivateComfortSignature  | Deaktiviert die Komfortsignatur für einen oder mehrere HBA                                      |
|                          | GetSignatureMode  | Liefert den Status der Komfortsignaturfunktion und den Signaturmodus für einen oder mehrere HBA |
| <b>WSDL</b>              | SignatureService_V7_5_2.wsdl<br>SignatureService_V7_4_2.wsdl<br>SignatureService.wsdl (WSDL-Version 7.4.0)  |   |
| <b>Schema</b>            | SignatureService_V7_5_2.xsd<br>SignatureService_V7_4_4.xsd<br>SignatureService.xsd (XSD-Version 7.4.2)  |   |

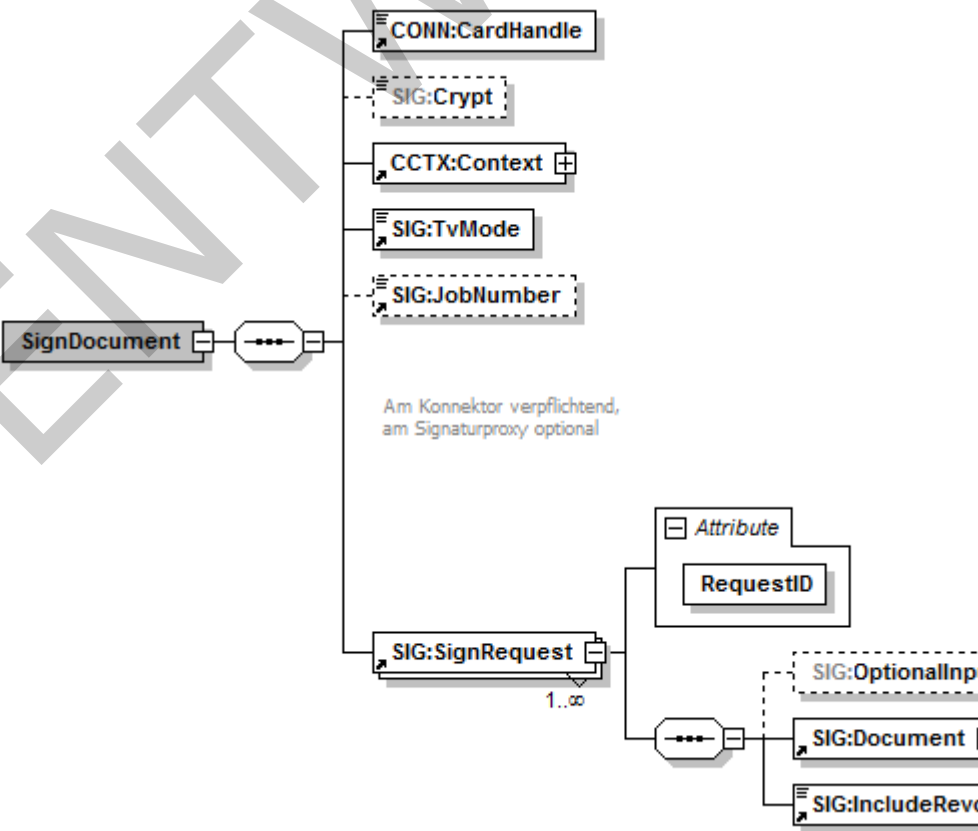
[<=]

##### 4.1.8.5.1 SignDocument (nonQES und QES)

##### **TIP1-A\_5010-04 - Operation SignDocument (nonQES und QES)**

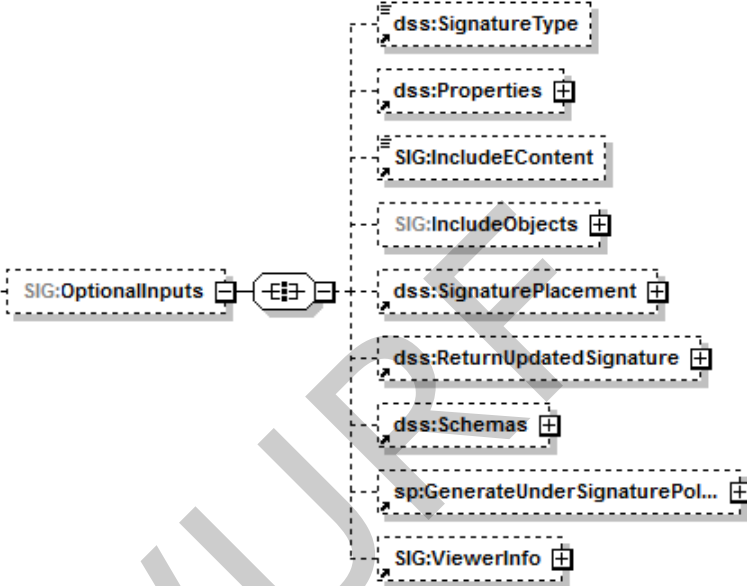
Der Signaturdienst des Konnektors MUSS an der Clientschnittstelle eine an [OASIS-DSS] angelehnte Operation SignDocument anbieten.

4491 Tabelle 229: TAB\_KON\_065 Operation SignDocument (nonQES und QES)

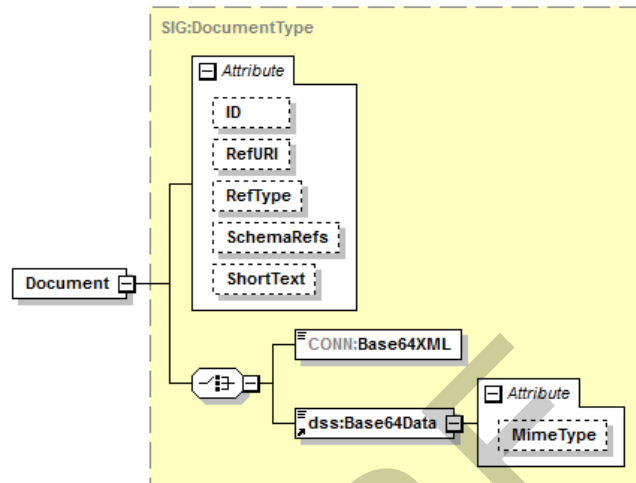
| Name                   | SignDocument  |
|------------------------|---|
| <b>Beschreibung</b>    | <p>Diese Operation lehnt sich an [OASIS-DSS] an. Sie enthält voneinander unabhängige SignRequests. Jeder SignRequest erzeugt eine Signatur für ein Dokument.</p> <p>Für die qualifizierte elektronische Signatur (QES) werden die QES_DocFormate unterstützt. Für nicht-qualifizierte elektronische Signaturen (nonQES) werden die nonQES_DocFormate unterstützt.</p> <p>Zur Signaturerzeugung werden Schlüssel und Zertifikate einer Chipkarte benutzt.</p> <p>Unterstützte Karten sind für die QES der HBAX mit dem QES-Zertifikat. Für die nonQES wird für die Signaturtypen „XML-Signatur, CMS-Signatur, PDF-Signatur, S/MIME-Signatur“ die SM-B mit dem OSIG-Zertifikat unterstützt.</p> <p>Bei der Erstellung von XML-Signaturen MUSS Canonical XML 1.1 verwendet werden [CanonXML1.1].</p> <p>Es soll der Common-PKI-Standard eingesetzt werden, siehe [Common-PKI].</p> <p>In Summe für die Größe der Dokumente in allen SignRequests innerhalb einer SignDocument-Anfrage MUSS der Konnektor eine Gesamtgröße von <math>\leq 250</math> MB unterstützen.</p> |
| <b>Aufrufparameter</b> |  <p>Am Konnektor verpflichtend, am Signaturproxy optional</p>   |



| Name                    | Beschreibung  |
|-------------------------|---|
| CONN:<br>Card<br>Handle | Identifiziert die zu verwendende Signaturkarte.<br>Die Operation DARF die Signatur mit der eGK NICHT unterstützen. Wird die Operation mit einem nicht unterstützten Kartentypen aufgerufen, so MUSS der Konnektor die Bearbeitung mit dem Fehler 4126 abbrechen.  |
| SIG:<br>Crypt           | Der Parameter crypt steuert die Auswahl der Zertifikate und Schlüssel für die Signaturerstellung abhängig von der durch cardHandle adressierten Karte gemäß TAB_KON_900.<br>Defaultwert: <ul style="list-style-type: none"> <li>gemäß TAB_KON_862 für die QES</li> <li>gemäß TAB_KON_863 für die nonQES.</li> </ul>   |
| CCTX:<br>Context        | <u>Aufrufkontext QES mit HBAX:</u><br>MandantId, ClientSystemId, WorkplaceId, UserId verpflichtend<br><u>Aufrufkontext nonQES mit SM-B:</u><br>MandantId, ClientSystemId, WorkplaceId verpflichtend; UserId nicht ausgewertet   |
| TvMode                  | Der Parameter wird im Konnektor nicht ausgewertet.  |
| SIG:<br>JobNumber       | Die Nummer des Jobs, unter der der nächste Signaturvorgang gestartet wird.<br>Parameter ist verpflichtend.  |
| SIG:<br>Sign<br>Request | Ein SignRequest kapselt den Signaturauftrag für ein Dokument.<br>Das verpflichtende XML-Attribut RequestID identifiziert einen SignRequest innerhalb eines Stapels von SignRequests eindeutig. Es dient der Zuordnung der SignResponse zum jeweiligen SignRequest.<br>Enthält der Aufruf mehr als die unterstützte Anzahl von SignRequests, bricht die Operation mit Fehler 4000 ab.<br>Es sind mindestens 50 SignRequests zu unterstützen. |

|  |                            |   |
|--|----------------------------|---|
|  | SIG:<br>Optional<br>Inputs | <p>Enthält optionale Eingangsparameter (angelehnt an dss:OptionalInputs gemäß [OASIS-DSS] Section 2.7):</p>  |
|--|----------------------------|---|

SIG:  
Document



Dieses an das `dss:Document` Element aus [OASIS-DSS] Section 2.4.2 angelehnte Element enthält das zu signierende Dokument, wobei die Kindelemente `CONN:Base64XML` und `dss:Base64Data` auftreten können.

Bei den als `dss:Base64Data` übergebenen Dokumenten werden folgende (Klassen von) MIME-Types unterschieden:

- "application/pdf-a" – für PDF/A-Dokumente,
- "text/plain",  
"text/plain; charset=iso-8859-15" oder  
"text/plain; charset=utf-8" – für Text-Dokumente,
- "image/tiff" – für TIFF-Dokumente und
- ein beliebiger anderer MIME-Type für nicht näher unterschiedene Binärdaten des spezifizierten Typs.

Der MIME-Type „text/plain“ wird interpretiert als „text/plain; charset=iso-8859-15“.

Das Element enthält ein Attribut `ShortText`. Es muss für QES-Signaturen bei jedem Aufruf vom Clientsystem übergeben werden, für nonQES-Signaturen ist es optional.

Über das Attribut `RefURI` kann gemäß [OASIS-DSS] (Abschnitt 2.4.1) ein zu signierender Teilbaum eines XML-Dokuments ausgewählt werden. Wenn die Signatur eines Teilbaums für die Signaturvariante nicht unterstützt wird, muss der Signaturauftrag mit Fehler 4111 abgelehnt werden.

|  |                                       |   |
|--|---------------------------------------|---|
|  | SIG:<br>Include<br>Revocation<br>Info | <p>Durch diesen verpflichtenden Schalter kann der Aufrufer die Einbettung von zum Zeitpunkt der Signaturerstellung vorliegenden Sperrinformationen anfordern. Es wird ausschließlich die zu erstellende Signatur betrachtet, d.h. es erfolgt keine Einbettung von Sperrinformationen für bereits enthaltene Signaturen.</p> <p>Für nicht-qualifizierte elektronische Signaturen (nonQES) wird diese Funktionalität nicht unterstützt. Für PDF-Signaturen werden keine Sperrinformationen eingebettet.</p> |
|--|---------------------------------------|---|

ENTWURF

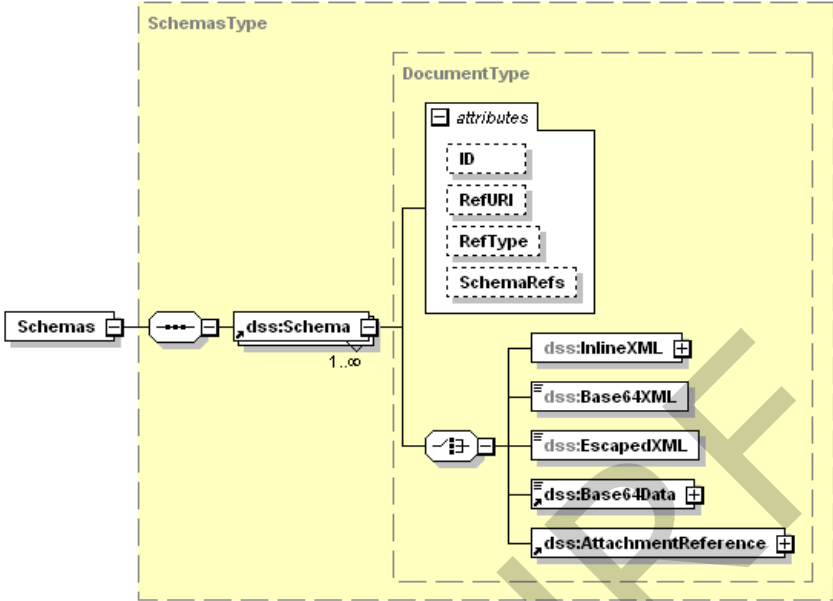
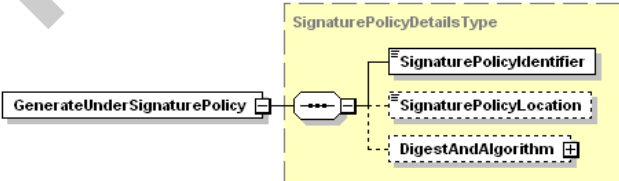
|  |                           |  |
|--|---------------------------|--|
|  | dss:<br>Signature<br>Type | <p>Durch dieses in [OASIS-DSS] (Abschnitt 3.5.1) beschriebene Element kann der generelle Typ der zu erzeugenden Signaturen spezifiziert werden. Hierbei MÜSSEN folgende Signaturtypen unterstützt werden:</p> <ul style="list-style-type: none"> <li> <b>XML-Signatur</b><br/> Durch Übergabe der URI <a href="urn:ietf:rfc:3275">urn:ietf:rfc:3275</a> wird die Erstellung von XML-Signaturen gemäß [RFC3275], [XMLDSig] angestoßen.<br/> Das zu verwendende Profil ist XAdES-BES ([XAdES]). Die Rückgabe einer solchen Signatur erfolgt als <code>ds:Signature</code>-Element. </li> <li> <b>CMS-Signatur</b><br/> Durch Übergabe der URI <a href="urn:ietf:rfc:5652">urn:ietf:rfc:5652</a> wird eine CMS-Signatur gemäß [RFC5652] angestoßen. Das zu verwendende Profil ist CAdES-BES ([CAdES]).<br/> Die Signatur wird als <code>dss:Base64Signature</code> mit der oben genannten URI als <code>Type</code> zurückgeliefert. </li> <li> <b>S/MIME-Signatur</b><br/> Durch Übergabe der URI „urn:ietf:rfc:5751“ wird eine S/MIME-Signatur gemäß [RFC5751] angestoßen.<br/> Die CMS-Signatur der übergebenen MIME-Nachricht erfolgt konform der Vorgaben zur CMS-Signatur. Das Rückgabedokument ist eine MIME-Nachricht vom Typ „application/pkcs7-mime“ mit einer CMS-Struktur vom Typ <code>SignedData</code>.<br/> Ist das übergebene Dokument keine MIME-Nachricht, so wie der Fehler 4111 (Ungültiger Signaturtyp oder Signaturvariante) zurückgeliefert. </li> <li> <b>PDF-Signatur</b><br/> Durch Übergabe der URI <a href="http://uri.etsi.org/02778/3">http://uri.etsi.org/02778/3</a> wird die Erzeugung einer PAdES-Basic Signatur gemäß [PAdES-3] angestoßen, wobei das Dokument mit der integrierten Signatur als <code>dss:Base64Signature</code> mit der oben genannten URI als <code>Type</code> zurückgeliefert wird.<br/> Handelt es sich beim übergebenen Dokument nicht um ein <code>Base64Data</code>-Element mit MIME-Type „application/pdf-a“, so wird ein Fehler 4111 (Ungültiger Signaturtyp oder Signaturvariante) zurückgeliefert. </li> </ul> <p>Andere <code>SignatureType</code>-Angaben führen zu einer Fehlermeldung 4111 (Ungültiger Signaturtyp oder Signaturvariante).</p> |
|--|---------------------------|--|

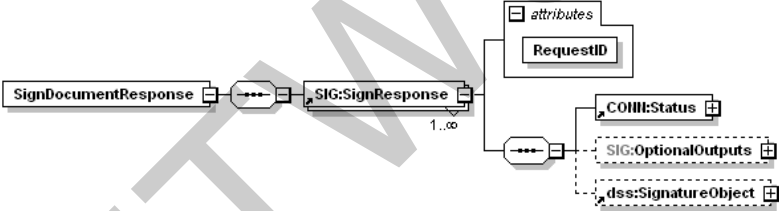
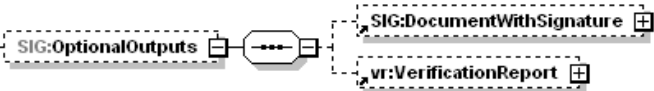
|  |  |  |
|--|--|--|
|  |  | <p>Die Signaturtypen „XML-Signatur, CMS-Signatur, PDF-Signatur, S/MIME-Signatur“ DÜRFEN für QES der HBAX nur mit dem QES-Zertifikat erfolgen, für nonQES nur mit dem OSIG-Zertifikat der SM-B. In jedem diese Anforderung verletzenden Fall MUSS der Fehler 4058 (Aufruf nicht zulässig) zurückgeliefert werden. Fehlt dieses Element, so wird der Signaturtyp gemäß TAB_KON_583 – Default-Signaturverfahren aus dem Dokumententyp abgeleitet.</p> |
|--|--|--|

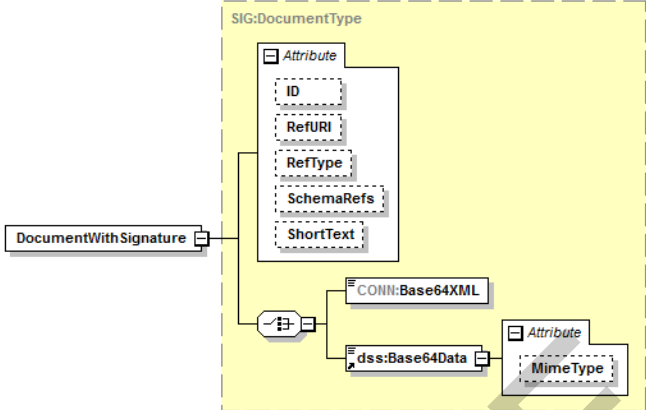
|  |                             |   |
|--|-----------------------------|---|
|  | dss:<br>Properties          | <p>Durch dieses in [OASIS-DSS] (Abschnitt 3.5.5) definierte Element können zusätzliche signierte und unsignierte Eigenschaften (Properties) bzw. Attribute in die Signatur eingefügt werden.</p> <p>Unterstützt werden genau folgende Attribute:<br/>Im CMS-Fall (SignatureType = urn:ietf:rfc:5652) kann es XML-Elemente<br/>./SignedProperties/Property/Value/CMSAttribute<br/>und<br/>./UnsignedProperties/Property/Value<br/>/CMSAttribute<br/>enthalten. Ein solches XML-Element CMSAttribute muss ein vollständiges, base64/DER-kodiertes ASN.1-Attribute enthalten, definiert in [CMS#5.3.SignerInfo Type]. Es muss bei der Erstellung des CMS-Containers unverändert unter SignedAttributes bzw. UnsignedAttributes aufgenommen werden.</p> <p>Die Übergabe der Attribute</p> <ul style="list-style-type: none"> <li>• ContentType</li> <li>• SigningTime</li> <li>• MessageDigest</li> <li>• SigningCertificate und SigningCertificateV2</li> </ul> <p>wird ignoriert und es wird die Warnung 4273 zurück gegeben.</p> |
|  | SIG:<br>Include<br>EContent | <p>Durch dieses in [OASIS-DSS] (Abschnitt 3.5.7), definierte Element kann bei einer CMS-basierten Signatur das Einfügen des signierten Dokumentes in die Signatur angefordert werden.</p> <p>Die Verwendung dieses Parameters bei anderen Signaturtypen führt zu einem Fehler 4111 (Ungültiger Signaturtyp oder Signaturvariante).</p>  |
|  | SIG:<br>Include<br>Object   | <p>Dieses Element enthält zum Anfordern einer Enveloping XML Signatur ein dss:IncludeObject-Element gemäß [OASIS-DSS] (Abschnitt 3.5.6). Ist das Element vorhanden und ein anderer Signaturtyp als eine XML-Signatur angefordert, so wird der Fehler 4111 (Ungültiger Signaturtyp oder Signaturvariante) zurückgeliefert.</p>   |

|  |  |
|--|--|
| dss:<br>Signature<br>Placement         | <p>Durch dieses in [OASIS-DSS] (Abschnitt 3.5.8) definierte Element kann bei XML-basierten Signaturen gemäß [RFC3275] die Platzierung der Signatur im Dokument angegeben werden.</p> <p>Die in [OASIS-DSS] (Abschnitt 2.5, XPath c) beschriebene Deklaration von Namespace-Prefixes im dss:SignaturePlacement-Element muss nicht unterstützt werden.</p> <p>Bei anderen Signaturtypen wird das Element ignoriert und eine Warnung (Fehlercode 4197, Parameter SignaturePlacement wurde ignoriert) zurückgeliefert.</p> <p>dss:SignaturePlacement darf nur zusammen mit einer unterstützten Signaturrichtlinie verwendet werden (sp:SignaturePolicyIdentifier muss entsprechend gesetzt sein).</p>  |
| dss:<br>Return<br>Updated<br>Signature | <p>Durch dieses in [OASIS-DSS] (Abschnitt 4.5.8) definierte Element kann eine übergegebene XML- oder CMS-Signatur mit zusätzlichen Informationen und Signaturen (Parallel- und Gegensignaturen) versehen werden. Hierbei sind folgende Ausprägungen für das Type-Attribut vorgesehen:</p> <ul style="list-style-type: none"> <li>• <a href="http://ws.gematik.de/conn/sig/sigupdate/parallel/">http://ws.gematik.de/conn/sig/sigupdate/parallel/</a><br/>Hierdurch wird eine Parallelsignatur zu einer bereits existierenden Signatur erzeugt und entsprechend zurückgeliefert.</li> <li>• <a href="http://ws.gematik.de/conn/sig/sigupdate/counter/documentexcluding/">http://ws.gematik.de/conn/sig/sigupdate/counter/documentexcluding</a><br/>Hierdurch wird eine dokumentenexkludierende Gegensignatur für alle vorhandenen parallelen Signaturen erzeugt.</li> </ul> <p>Bei anderen Type-Attributen wird der Fehler 4111 (Ungültiger Signaturtyp oder Signaturvariante) zurückgeliefert.</p> |
| dss:<br>Schemas                        | <p>Durch das in [OASIS-DSS] (Abschnitt 2.8.5) definierte Element können eine Menge von XML-Schemata übergeben werden, die zur Validierung der übergebenen XML-Dokumente verwendet werden können.</p>   |



|                                 |   |
|---------------------------------|---|
|                                 |  <p>The diagram shows a <b>SchemasType</b> container. Inside, there is a <b>DocumentType</b> container. The <b>DocumentType</b> contains an <b>attributes</b> group with <b>ID</b>, <b>RefURI</b>, <b>RefType</b>, and <b>SchemaRefs</b>. It also contains a choice of <b>dss:InlineXML</b>, <b>dss:Base64XML</b>, <b>dss:EscapedXML</b>, <b>dss:Base64Data</b>, and <b>dss:AttachmentReference</b>. Outside the <b>DocumentType</b>, there is a <b>Schemas</b> element connected to a <b>dss:Schema</b> element (multiplicity 1..∞).</p>   |
| dss:Schema                      | <p>Dieses Element enthält ein XML-Schema zur Validierung des übergebenen XML-Dokuments. Das Attribut <b>RefURI</b> ist verpflichtend. Es kennzeichnet dabei den Namensraum des XML-Schemas entsprechend [OASIS-DSS] (Abschnitt 2.8.5)</p>   |
| sp:GenerateUnderSignaturePolicy |  <p>The diagram shows a <b>GenerateUnderSignaturePolicy</b> element connected to a <b>SignaturePolicyDetailsType</b> container. The <b>SignaturePolicyDetailsType</b> contains <b>SignaturePolicyIdentifier</b>, <b>SignaturePolicyLocation</b>, and <b>DigestAndAlgorithm</b>.</p> <p>Über dieses in [OASIS-SP], Kapitel 2.2.1.1.1 Optional Input &lt;GenerateUnderSignaturePolicy&gt;, definierte Element wird die erforderliche Singnaturrechtlinie ausgewählt. Die im Element <b>sp:SignaturePolicyIdentifizier</b> übergebene URI identifiziert die Signaturrechtlinie. Die XML-Elemente <b>SignaturePolicyLocation</b> <b>DigestAndAlgorithm</b> werden nicht verwendet. Wenn eine nach TAB_KON_778 notwendige Signaturrechtlinie fehlt oder die übergebene Signaturrechtlinie unbekannt ist, wird Fehler</p> |

|                 |  |   |
|-----------------|--|---|
|                 |  | 4111 zurückgeliefert.   |
|                 | SIG:<br>Viewer<br>Info   | Enthält optional die vom Konnektor in die Signatur einzubeziehende Referenzen für die Stylesheets zur Anzeige.  |
| <b>Rückgabe</b> |  |   |
|                 | SIG:<br>Sign<br>Response   | Eine SignResponse kapselt den ausgeführten Signaturauftrag pro Dokument. Die Zuordnung zwischen SignRequest und SignResponse erfolgt über die RequestID.                  |
|                 | CONN:<br>Status  | Enthält den Status der ausgeführten Operation pro SignRequest.  |
|                 | SIG:<br>Optional<br>Outputs  | Enthält (angelehnt an dss:OptionalOutputs) optionale Ausgangsparameter: <div>  </div> |

|                              |                                       |   |
|------------------------------|---------------------------------------|---|
|                              | SIG:<br>Document<br>With<br>Signature |  <p>Pro SignResponse wird ein Element <code>SIG:DocumentWithSignature</code> gemäß [OASIS-DSS] (Abschnitt 3.5.8) zurückgeliefert, in dem das Dokument mit Signatur enthalten ist. Dabei werden die XML-Attribute des Elements <code>SIG:Document</code> auf dem zugehörigen <code>SignRequest</code> übernommen.</p> <p>Ist die Signatur nicht im Dokument enthalten, wird ein leeres Element <code>Base64XML</code> oder <code>Base64Data</code> zurückgegeben. Die Signatur wird dann im Element <code>dss:SignatureObject</code> abgelegt.</p> <p>Wenn die Signatur im Dokument enthalten ist, wird das signierte Dokument im Feld <code>Base64XML</code> bzw. <code>Base64Data</code> zurückgeliefert. In diesem Fall MUSS die <code>dss:SignaturePtr-Alternative</code> in <code>dss:SignatureObject</code> (vgl. [OASIS-DSS] Abschnitt 2.5) dazu genutzt werden, auf die in den Dokumenten enthaltenen Signaturen zu verweisen.</p> |
|                              | vr:<br>Verifi<br>cation<br>Report     | Vom Konnektor nicht befüllt.  |
|                              | dss:<br>Signature<br>Object           | Enthält im Erfolgsfall die erzeugte Signatur pro <code>SignRequest</code> in Form eines <code>dss:SignatureObject</code> -Elementes gemäß [OASIS-DSS] (Abschnitt 3.2).  |
| <b>Vorbe-<br/>dingungen</b>  | Keine                                 |   |
| <b>Nachbe-<br/>dingungen</b> | Keine                                 |   |

4492 Der Ablauf der Operation SignDocument ist in Tabelle TAB\_KON\_756 Ablauf Operation  
 4493 SignDocument (nonQES und QES) beschrieben:  
 4494

4495 **Tabelle 230: TAB\_KON\_756 Ablauf Operation SignDocument (nonQES und QES)**

| Nr.  | Aufruf<br>Technischer Use<br>Case oder<br>Interne<br>Operation | Beschreibung  |
|--|--|---|
| 1.   | checkArguments   | Anhand des Kartentyps wird ermittelt, ob eine QES oder eine nonQES erzeugt werden soll. Alle übergebenen Parameterwerte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.  |
| 2.   | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“               | Die Prüfung erfolgt durch den Aufruf<br>TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientsystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>userId = \$context.userId;<br>cardHandle = \$cardHandle }<br>Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab. |
| 3.   | TUC_KON_026<br>„Liefere<br>CardSession“                        | Ermittle CardSession über TUC_KON_026 {<br>mandatId = \$context.mandantId;<br>clientsystemId = \$context.clientsystemId;<br>cardHandle = \$context.cardHandle;<br>userId = \$context.userId }   |
| Im Fall QES wird Schritt 4 ausgeführt. Im Fall nonQES wird Schritt 5 ausgeführt.   |  |   |
| 4a)  | Prüfe<br>Signaturdienst-<br>Modul                              | Prüfe, ob MGM_LU_SAK=Enabled. Ist dies nicht der Fall, so bricht die Operation mit Fehler 4125 ab.  |
| Wenn für die CardSession die Komfortsignatur aktiviert ist<br>(CARDESSION.SIGNMODE = Comfort) wird Schritt 4 c) ausgeführt. Andernfalls wird<br>Schritt 4 b) ausgeführt. |  |   |
| 4b)  | TUC_KON_150<br>„Dokumente QES<br>signieren“                    | Die QES wird erzeugt. Tritt hierbei ein Fehler auf, bricht die Operation ab.  |
| 4c)  | TUC_KON_170<br>„Dokumente mit<br>Komfort signieren“            | Eine Komfortsignatur wird erzeugt. Tritt hierbei ein Fehler auf, bricht die Operation ab.   |

|    |  |  |
|----|--|--|
| 5) | TUC_KON_160<br>„Dokumente<br>nonQES signieren“ | Die nonQES wird erzeugt. Tritt hierbei ein Fehler auf,<br>bricht die Operation ab. |
|----|--|--|

4496 **Tabelle 231: TAB\_KON\_757 Fehlercodes „SignDocument (nonQES und QES)“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen TUCs können folgende weiteren Fehlercodes auftreten: |           |          |  |
| 4000  | Technical | Error    | Syntaxfehler   |
| 4111  | Technical | Error    | ungültiger Signaturtyp oder Signaturvariante   |
| 4126  | Security  | Error    | Kartentyp nicht zulässig für Signatur  |
| 4125  | Technical | Error    | LU_SAK nicht aktiviert   |
| 4197  | Technical | Warning  | Parameter SignaturePlacement wurde ignoriert   |
| 4252  | Technical | Error    | Jobnummer wurde in den letzten 1.000 Aufrufen bereits verwendet und ist nicht zulässig |
| 4273  | Technical | Warning  | Attribute im Parameter dss:Properties wurden ignoriert                                 |

4497  
4498 Die zulässigen Zertifikate und Schlüssel sind in TAB\_KON\_900 aufgelistet.  
4499 [**<=**]

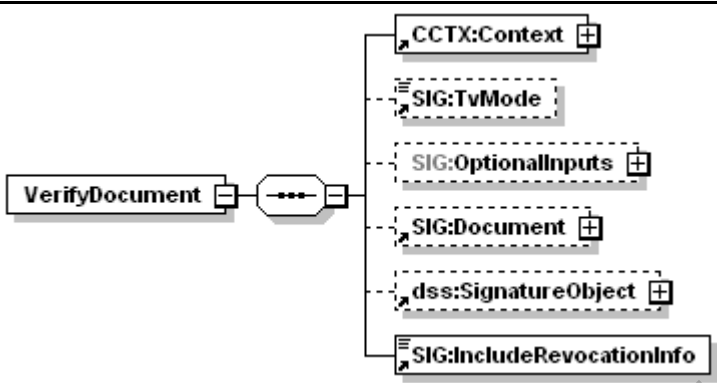
#### 4500 4.1.8.5.2 VerifyDocument (nonQES und QES)

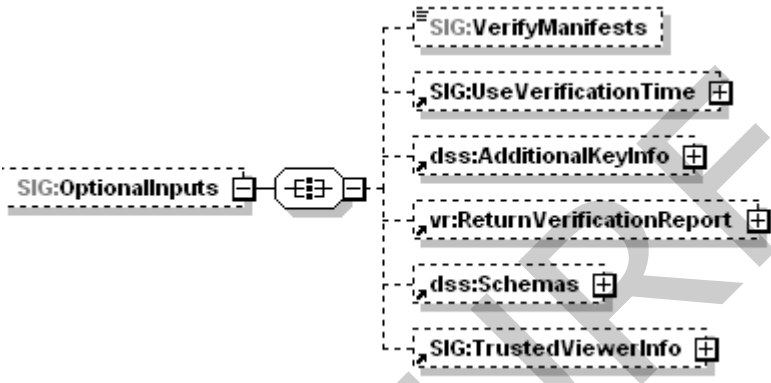
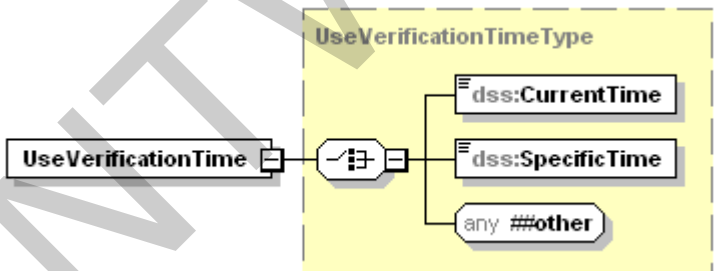
#### 4501 **TIP1-A\_5034-04 - Operation VerifyDocument (nonQES und QES)**

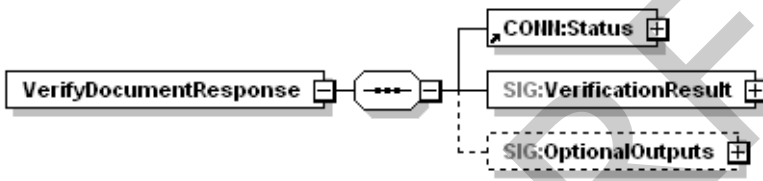
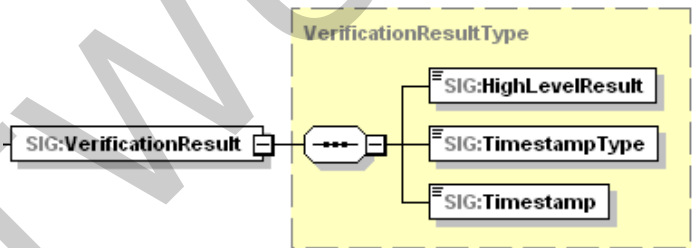
4502 Der Signatordienst des Konnektors MUSS an der Clientschnittstelle eine an [OASIS-DSS]  
4503 angelehnte Operation VerifyDocument (nonQES und QES) anbieten.  
4504

4505 **Tabelle 232: TAB\_KON\_066 Operation VerifyDocument (nonQES und QES)**

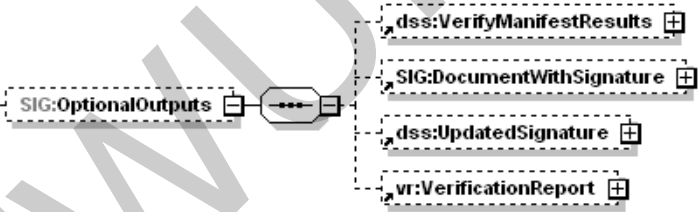
|                     |  |
|---------------------|--|
| <b>Name</b>         | VerifyDocument   |
| <b>Beschreibung</b> | Diese Operation verifiziert die Signatur eines Dokumentes.<br>Der Konnektor MUSS jede konform zur Außenschnittstelle SignDocument erzeugte Signatur durch VerifyDocument prüfen können.<br>Das Ergebnis der Prüfung wird, wenn gefordert, in Form eines standardisierten Prüfberichts in einer <i>VerificationReport</i> -Struktur gemäß [OASIS-VR] zurückgeliefert. |
|                     |  |

|                      |  |  |
|----------------------|--|--|
| Aufruf-<br>parameter |  |  |
|                      | Name   | Beschreibung   |
|                      | CCTX:<br>Context   | MandantId, ClientSystemId, WorkplaceId verpflichtend; UserId nicht ausgewertet   |
|                      | TvMode   | Der Parameter wird im Konnektor nicht ausgewertet.   |
|                      | SIG:<br>Optional<br>Inputs   | Enthält optionale Eingabeparameter (angelehnt an dss:OptionalInputs gemäß [OASIS-DSS] Section 2.7):<br>Die zulässigen optionalen Eingabeparameter sind unten erläutert.  |
|                      | SIG:<br>Document   | Enthält im Fall der Prüfung von detached oder enveloped Signaturen das zur Signatur gehörende bzw. das diese umschließende Dokument (siehe [OASIS-DSS] Section 2.4.2 und oben).  |
|                      | dss:<br>Signature<br>Object  | Enthält die zu prüfende Signatur, wenn sie nicht im Dokument selbst eingebettet ist ([OASIS-DSS] Kapitel 4.1). Hierbei werden <b>XML-Signaturen</b> als ds:Signature Element und alle anderen Signaturen als dss:Base64Signature mit entsprechend gesetztem Type-Attribut (siehe SignatureType, Operation SignDocument) übergeben, wobei die nachfolgenden Werte unterstützt werden müssen: <ul style="list-style-type: none"> <li>• <b>CMS-Signatur</b><br/>urn:ietf:rfc:5652</li> <li>• <b>S/MIME-Signatur</b><br/><a href="http://uri.etsi.org/02778/3">urn:ietf:rfc:5751</a></li> <li>• <b>PDF-Signatur</b><br/><a href="http://uri.etsi.org/02778/3">http://uri.etsi.org/02778/3</a></li> </ul> |
|                      | SIG:<br>Include<br>Revocat<br>ionInfo  | Durch diesen verpflichtenden Schalter kann der Aufrufer die Einbettung von zum Zeitpunkt der Signaturprüfung vorliegenden Sperrinformationen anfordern.<br>Ist bereits eine Sperrinformation eingebettet, so wird die neue Sperrinformation zusätzlich   |

|   |  |  |
|---|--|--|
|   |  | <p>eingebettet.</p> <p>Für in einer Gegensignatur enthaltene Signaturen erfolgt keine Einbettung von Sperrinformationen. Für PDF-Signaturen erfolgt keine Einbettung von Sperrinformationen. Der Konnektor nimmt die Warnung 4261 in die Antwort auf.</p>              |
|   |  |    |
| SIG:<br>Verify<br>Manifests                 |  | <p>Durch das in [OASIS-DSS] (Abschnitt 4.5.1) definierte Element kann die Prüfung eines ggf. vorhandenen Manifests angefordert werden.</p>   |
|   |  |    |
| SIG:<br>Use<br>Verification<br>Time         |  | <p>Durch das in [OASIS-DSS] (Abschnitt 4.5.2) spezifizierte Element kann die Prüfung der Signatur bezüglich eines durch den Aufrufer bestimmten Zeitpunktes (Benutzerdefinierter_Zeitpunkt) erfolgen.</p>  |
| dss:<br>Addit<br>ional<br>KeyInfo           |  | <p>Durch das in [OASIS-DSS] (Abschnitt 4.5.4) spezifizierte Element kann zusätzliches, für die Prüfung benötigtes, Schlüsselmaterial übergeben werden.</p>   |
| vr:<br>Return<br>Verifi<br>cation<br>Report |  | <p>Durch dieses in [OASIS-VR] spezifizierte Element kann die Erstellung eines ausführlichen Prüfberichtes angefordert werden. Der Konnektor MUSS die Anforderungen der Konformitätsstufe 2 („Comprehensive“) erfüllen und die Profilierung aus Anhang B3 beachten.</p> |

|                 |  |  |
|-----------------|--|--|
|                 | dss:<br>Schemas  | Durch das in [OASIS-DSS] (Abschnitt 2.8.5) definierte Element können eine Menge von XML-Schematas übergeben werden, die zur Validierung des übergebenen XML-Dokumentes verwendet werden können.<br>Zur Struktur dieses Elements siehe Beschreibung des Parameters <code>dss:Schemas</code> der Operation <code>SignDocument</code> .       |
|                 | SIG:<br>Viewer<br>Info   | Der Parameter wird im Konnektor nicht ausgewertet.   |
| <b>Rückgabe</b> |  |  |
|                 | Status   | Enthält den Ausführungsstatus der Operation.   |
|                 | SIG:<br>Verifi<br>cation<br>Result   |  <p>Das Element <code>Sig:VerificationResult</code> enthält das Ergebnis der Prüfung als Ampel, den Typ des zugehörigen angenommenen Signaturzeitpunkts und der angenommene Signaturzeitpunkt selbst.</p>  |
|                 | SIG:<br>High<br>Level<br>Result  | <p>Das Ergebnis der Prüfung (Ampelschaltung) mit folgenden Werten:</p> <ul style="list-style-type: none"> <li>• VALID: alle Signaturen sind gültig</li> <li>• INVALID: mindestens eine der Signaturen ist ungültig</li> <li>• INCONCLUSIVE: in allen anderen Fällen</li> </ul>   |
|                 | SIG:<br>Time<br>stamp<br>Type  | <p>Der Typ des angenommenen Signaturzeitpunkts mit folgenden Werten:</p> <ul style="list-style-type: none"> <li>• SIGNATURE_EMBEDDED_TIMESTAMP:<br/>in der Signatur eingebetter Zeitpunkt<br/><code>Ermittelter_Signaturzeitpunkt_Eingebettet</code></li> <li>• </li> <li>• SYSTEM_TIMESTAMP:<br/>Systemzeit des Konnektors bei</li> </ul> |



|  |                              |   |
|--|------------------------------|---|
|  |                              | <p>Signaturprüfung<br/>Ermittelter_Signaturzeitpunkt<br/>_System</p> <ul style="list-style-type: none"> <li>• USER_DEFINED_TIMESTAMP:<br/>benutzerdefinierter Zeitpunkt<br/>Benutzerdefinierter_Zeitpunkt</li> </ul> <p>Als Format darf jedes zum XML-Typ "dateTime" konforme Format verwendet werden (&lt;element name="Timestamp" type="dateTime"/&gt;). Wenn mehrere Signaturen im Dokument vorhanden sind, wird hier der angenommene Signaturzeitpunkt der jüngsten Signatur angegeben.</p> |
|  | SIG: Timestamp               | Im Element SIG:Timestamp wird der zu SIG:TimestampType gehörende Zeitstempel zurückgegeben.   |
|  | SIG: Optional Outputs        | <p>Enthält (angelehnt an dss:OptionalOutputs, wie in Abschnitt 2.7 von [OASIS-DSS] beschrieben) optionale Ausgangelemente:</p>   |
|  | dss: Verify Manifest Results | Dieses in Abschnitt 4.5.1 von [OASIS-DSS] definierte Element enthält Informationen zur Prüfung eines ggf. vorhandenen Signaturmanifests und wird zurückgeliefert, sofern beim Aufruf das dss:VerifyManifest-Element, aber nicht das RequestVerificationReport als optionales Eingabeelement übergeben wurde.  |
|  | SIG: Document With Signature | Dieses in Abschnitt 4.5.8 von [OASIS-DSS] spezifizierte Element wird zurückgeliefert, falls eine in dem Dokument enthaltene Signatur (Enveloped Signature) in Verbindung mit dem SIG:IncludeRevocationInfo-Element geprüft wurde.   |
|  | dss: Updated Signature       | Dieses in Abschnitt 4.5.8 von [OASIS-DSS] spezifizierte Element wird zurückgeliefert, falls eine abgesetzte (Detached Signature) oder umschließende (Enveloping Signature) in Verbindung mit dem SIG:IncludeRevocationInfo- Element geprüft wurde.  |
|  | vr: Verifi cation Report     | Dieses in [OASIS-VR] spezifizierte Element wird zurückgeliefert, falls das ReturnVerificationReport-Element als   |

|                        |       |  |
|------------------------|-------|--|
|                        |       | Eingabeparameter verwendet wurde. Die Profilierung von Anhang B3 MUSS beachtet werden. |
| <b>Vorbedingungen</b>  | Keine |  |
| <b>Nachbedingungen</b> | Keine |  |

4506 **Tabelle 233: TAB\_KON\_760 Ablauf Operation VerifyDocument (nonQES und QES)**

| Nr.  | Aufruf<br>Technischer Use<br>Case oder Interne<br>Operation | Beschreibung   |
|--|---|--|
| 1.   | checkArguments  | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.   |
| 2.   | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“            | Die Prüfung erfolgt durch den Aufruf<br>TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientsystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>needCardSession= false;<br>}<br>Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab. |
| 3.   | prüfe, ob QES oder<br>nonQES                                | Ist im jeweiligen Signaturzertifikat mindestens ein QCStatement mit dem OID id-etsi-qcs-QcCompliance (0.4.0.1862.1.1) enthalten, handelt es sich um eine QES-Signatur, andernfalls liegt eine nonQES-Signatur vor.   |
| Für QES-Signaturen wird Schritt 4 ausgeführt. Für nonQES-Signaturen wird Schritt 5 ausgeführt. |   |  |
| 4.a  | Prüfe<br>Signaturdienst-<br>Modul                           | Prüfe, ob MGM_LU_SAK=Enabled. Ist dies nicht der Fall, so bricht die Operation mit Fehler 4125 ab.   |
| 4.b  | TUC_KON_151 „QES<br>Dokumentensignatur<br>prüfen“           | Die QES wird geprüft. Tritt hierbei ein Fehler auf, bricht die Operation ab.   |
| 5.   | TUC_KON_161<br>„nonQES<br>Dokumentensignatur<br>prüfen“     | Die nonQES wird geprüft. Tritt hierbei ein Fehler auf, bricht die Operation ab.  |

4507

4508 **Tabelle 234: TAB\_KON\_761 Fehlercodes „VerifyDocument (nonQES und QES)“**

| Fehlercode | ErrorType | Severity | Fehlertext |
|------------|-----------|----------|------------|
|------------|-----------|----------|------------|

Neben den Fehlercodes der aufgerufenen TUCs (siehe Tabelle TAB\_KON\_760 Ablauf Operation VerifyDocument) können folgende weiteren Fehlercodes auftreten:

|      |           |         |   |
|------|-----------|---------|---|
| 4000 | Technical | Error   | Syntaxfehler  |
| 4261 | Technical | Warning | Einbettung von Revocation-Informationen nicht unterstützt |
| 4125 | Technical | Error   | LU_SAK nicht aktiviert                                    |

4509  
4510  
4511

[<=]

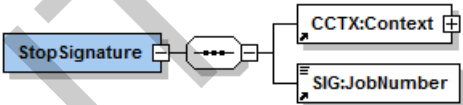
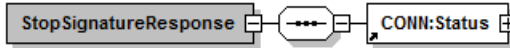
#### 4512 4.1.8.5.3 StopSignature

#### 4513 TIP1-A\_5666 - Operation StopSignature (nonQES und QES)

4514 Der Signaturdienst des Konnektors MUSS an der Clientschnittstelle eine Operation  
4515 StopSignature anbieten.

4516

4517 **Tabelle 235: TAB\_KON\_840 Operation StopSignature**

|                 |   |  |  |
|-----------------|---|--|--|
| Name            | StopSignature   |  |  |
| Beschreibung    | Diese Operation unterbricht die Signatur eines Dokumentenstapels.<br>Der Konnektor MUSS jede Signaturerstellung für ein Dokumentenstapel unterbrechen können. |  |  |
| Aufrufparameter |    |  |  |
|                 | Name  | Beschreibung   |  |
|                 | CCTX:Context  | MandantId, ClientSystemId, WorkplaceId verpflichtend; UserId nicht ausgewertet       |  |
|                 | SIG:JobNumber   | Die Nummer des Jobs, der gestoppt werden soll.                                       |  |
|                 | Rückgabe  |  |  |
| CONN:Status     |   | Enthält den Ausführungsstatus der Operation.   |  |
| Vorbedingungen  | Keine   |  |  |
| Nachbedingungen | Keine   |  |  |

4518 **Tabelle 236: TAB\_KON\_841 Ablauf Operation StopSignature**

| Nr. | Aufruf Technischer Use Case oder Interne Operation | Beschreibung   |
|-----|--|--|
| 1.  | checkArguments                                     | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab. |
| 2.  | Stoppe die Stapelsignaturverarbeitung              | Die Verarbeitung der Stapelsignatur wird abgebrochen   |

4519

4520 **Tabelle 237: TAB\_KON\_842 Fehlercodes „StopSignature“**


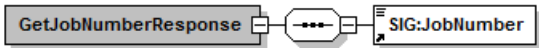
| Fehlercode                             | ErrorType | Severity | Fehlertext          |
|--|-----------|----------|---------------------|
| Folgende Fehlercodes können auftreten: |           |          |                     |
| 4000                                   | Technical | Error    | Syntaxfehler        |
| 4243                                   | Technical | Error    | Jobnummer unbekannt |

4521 **[<=]**4522 **4.1.8.5.4 GetJobNumber**4523 **TIP1-A\_5667 - Operation GetJobNumber**

4524 Der Signatordienst des Konnektors MUSS an der Clientschnittstelle eine Operation  
 4525 GetJobNumber anbieten.

4526

4527 **Tabelle 238: TAB\_KON\_843 Operation GetJobNumber**

|                 |   |  |  |
|-----------------|---|--|--|
| Name            | GetJobNumber  |  |  |
| Beschreibung    | Diese Operation liefert eine Jobnummer zur Verwendung in der Operation SignDocument.<br>Die Jobnummer MUSS nach den Vorgaben von Kapitel 4.1.8.1.4 erstellt werden. |  |  |
| Aufrufparameter |    |  |  |
|                 | Name  | Beschreibung   |  |
|                 | CCTX:Context  | MandantId, ClientSystemId, WorkplaceId verpflichtend; UserId nicht ausgewertet |  |
| Rückgabe        |   |  |  |
|                 | SIG:JobNumber   | Jobnummer zur Verwendung in „SignDocument“                                     |  |
| Vorbedingungen  | Keine   |  |  |

|                        |       |
|------------------------|-------|
| <b>Nachbedingungen</b> | Keine |
|------------------------|-------|

4528 **Tabelle 239: TAB\_KON\_844 Ablauf Operation GetJobNumber**

| Nr. | Aufruf Technischer Use Case oder Interne Operation | Beschreibung  |
|-----|--|---|
| 1.  | checkArguments                                     | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.            |
| 2.  | Generiere und liefere eine Jobnummer               | Eine innerhalb von 1000 Aufrufen eindeutige Jobnummer wird generiert und geliefert. Die Zählung der Aufrufe erfolgt dabei unabhängig vom Aufrufkontext. |

4529

4530 **Tabelle 240: TAB\_KON\_845 Fehlercodes „GetJobNumber“**

| Fehlercode                             | ErrorType | Severity | Fehlertext   |
|--|-----------|----------|--------------|
| Folgende Fehlercodes können auftreten: |           |          |              |
| 4000                                   | Technical | Error    | Syntaxfehler |

4531

4532 [**<=**]

4533 **4.1.8.5.5 ActivateComfortSignature**

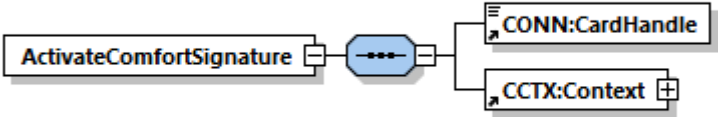
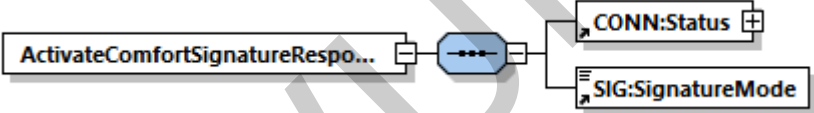
4534 **A\_19107 - Operation ActivateComfortSignature**

4535 Der Signaturdienst des Konnektors MUSS an der Clientschnittstelle eine Operation  
4536 ActivateComfortSignature anbieten.

4537

4538 **Tabelle 241: TAB\_KON\_874 ActivateComfortSignature**

|                     |  |
|---------------------|--|
| <b>Name</b>         | ActivateComfortSignature   |
| <b>Beschreibung</b> | Diese Operation aktiviert die Komfortsignatur für einen HBA bezogen auf einen Aufrufkontext. |

|                        |   |   |
|------------------------|---|---|
| <b>Aufrufparameter</b> |   |   |
|                        | Name  | Beschreibung  |
|                        | CONN:CardHandle   | Identifiziert die zu adressierende Karte. Es wird nur der HBA unterstützt.  |
|                        | CCTX:Context  | MandantId, ClientSystemId, WorkplaceId, UserId verpflichtend zu übergeben; MandantId, WorkplaceId nicht ausgewertet |
| <b>Rückgabe</b>        |  |   |
|                        | CONN:Status   | Enthält den Ausführungsstatus der Operation.  |
|                        | SIG:SignatureMode   | Signaturmodus des HBA<br>Enthält bei erfolgreicher Ausführung der Operation den Wert „COMFORT“                      |
| <b>Vorbedingungen</b>  | Keine   |   |
| <b>Nachbedingungen</b> | Keine   |   |

4539

**Tabelle 242: TAB\_KON\_877 Ablauf ActivateComfortSignature**

| Nr. | Aufruf Technischer Use Case oder Interne Operation | Beschreibung   |
|-----|--|--|
| 1.  | checkArguments                                     | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.   |
| 2.  | TUC_KON_000 „Prüfe Zugriffsberechtigung“           | Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {<br>mandantId = \$context.mandantId;<br>clientsystemId = \$context.clientsystemId;<br>workplaceId = \$context.workplaceId;<br>userId = \$context.userId;<br>cardHandle = \$cardHandle }<br>Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab. |
| 3.  | TUC_KON_026 „Liefere CardSession“                  | Ermittle CardSession über TUC_KON_026 {<br>mandatId = \$context.mandantId;   |

|    |   |   |
|----|---|---|
|    |   | clientsystemId = \$context.clientsystemId;<br>cardHandle = \$context.cardHandle;<br>userId = \$context.userId }                                   |
| 4. | TUC_KON_171<br>„Komfortsignatur<br>einschalten“ | Der Komfortsignaturmodus wird für das Tupel<br>(CardHandle, CardSession) eingeschaltet. Tritt<br>hierbei ein Fehler auf, bricht die Operation ab. |

**Tabelle 243: TAB\_KON\_879 Fehlercodes ActivateComfortSignature**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen TUCs können folgende weiteren Fehlercodes auftreten: |           |          |   |
| 4000  | Technical | Error    | Syntaxfehler  |
| 4270  | Technical | Error    | UserId wurde in den letzten 1.000 Vorgängen bereits verwendet |
| 4272  | Technical | Error    | UserId nicht zulässig   |

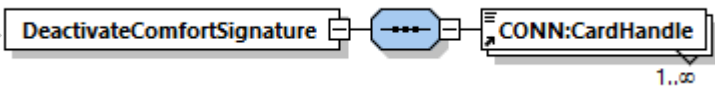
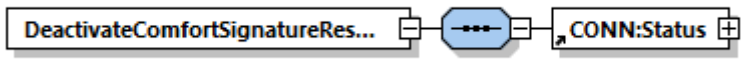
[&lt;=]

#### 4.1.8.5.6 DeactivateComfortSignature

#### A\_19108 - Operation DeactivateComfortSignature

Der Signaturdienst des Konnektors MUSS an der Clientschnittstelle eine Operation DeactivateComfortSignature anbieten.

**Tabelle 244: TAB\_KON\_875 DeactivateComfortSignature**

|                        |  |   |  |
|------------------------|--|---|--|
| <b>Name</b>            | DeactivateComfortSignature   |   |  |
| <b>Beschreibung</b>    | Diese Operation deaktiviert die Komfortsignatur für einen oder mehrere HBA.          |   |  |
| <b>Aufrufparameter</b> |  |   |  |
|                        | Name   | Beschreibung  |  |
|                        | CONN:<br>Card<br>Handle  | Identifiziert die zu adressierende Karte.<br>Es wird nur der HBA unterstützt. |  |
| <b>Rückgabe</b>        |  |   |  |

|                        |             |  |
|------------------------|-------------|--|
|                        | CONN:Status | Enthält den Ausführungsstatus der Operation. |
| <b>Vorbedingungen</b>  | Keine       |  |
| <b>Nachbedingungen</b> | Keine       |  |

4550 **Tabelle 245: TAB\_KON\_878 Ablauf DeactivateComfortSignature**

| Nr. | Aufruf Technischer Use Case oder Interne Operation | Beschreibung   |
|-----|--|--|
| 1.  | checkArguments                                     | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab. |
| 2.  | TUC_KON_172<br>„Komfortsignatur ausschalten“       | Der Komfortsignaturmodus wird für alle Karten aus der CardHandle-Liste ausgeschaltet.  |

4551

4552 **Tabelle 246: TAB\_KON\_880 Fehlercodes DeactivateComfortSignature**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--------------|
| Neben den Fehlercodes der aufgerufenen TUCs können folgende weiteren Fehlercodes auftreten: |           |          |              |
| 4000  | Technical | Error    | Syntaxfehler |

4553 [**<=**]

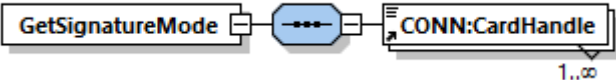
4554

4555 **4.1.8.5.7 GetSignatureMode**4556 **A\_19109 - Operation GetSignatureMode**

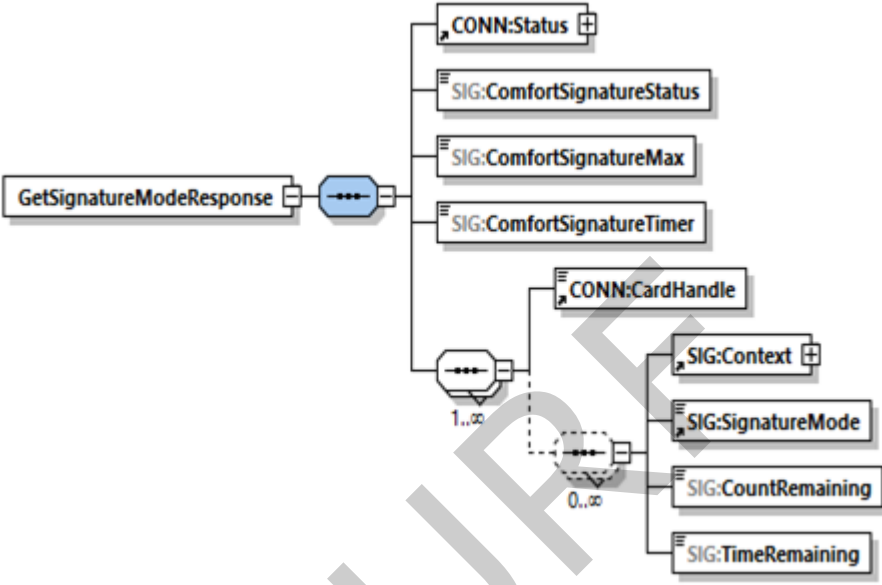
4557 Der Signatordienst des Konnektors MUSS an der Clientschnittstelle eine Operation  
 4558 GetSignatureMode anbieten.

4559

4560 **Tabelle 247: TAB\_KON\_876 GetSignatureMode**

|                        |   |              |
|------------------------|---|--------------|
| <b>Name</b>            | GetSignatureMode  |              |
| <b>Beschreibung</b>    | Diese Operation liefert den aktuell konfigurierten Status der Komfortsignaturfunktion im Konnektor und die im Konnektor aktuell hinterlegten Signaturmodus zu allen HBA aus der übergebenen CardHandle-Liste. |              |
| <b>Aufrufparameter</b> |   |              |
|                        | Name  | Beschreibung |



|                 |  |  |
|-----------------|--|--|
|                 | CONN:<br>Card<br>Handle  | Identifiziert die zu adressierende Karte.<br>Es wird nur der HBA unterstützt.  |
| <b>Rückgabe</b> |  <p>The diagram shows the structure of the <code>GetSignatureModeResponse</code> message. It is a container message (rectangle with a dashed border) containing several elements: <code>CONN:Status</code> (a simple rectangle), <code>SIG:ComfortSignatureStatus</code> (a rectangle with a dashed border), <code>SIG:ComfortSignatureMax</code> (a rectangle with a dashed border), <code>SIG:ComfortSignatureTimer</code> (a rectangle with a dashed border), <code>CONN:CardHandle</code> (a rectangle with a dashed border), and a list of <code>SIG:Context</code> (rectangle with a dashed border), <code>SIG:SignatureMode</code> (rectangle with a dashed border), <code>SIG:CountRemaining</code> (rectangle with a dashed border), and <code>SIG:TimeRemaining</code> (rectangle with a dashed border). The <code>CONN:CardHandle</code> element is connected to a list of <code>SIG:Context</code> elements via a dashed line. The <code>SIG:SignatureMode</code> element is connected to a list of <code>SIG:CountRemaining</code> and <code>SIG:TimeRemaining</code> elements via a dashed line. The cardinalities are 1..∞ for <code>CONN:CardHandle</code> and 0..∞ for the list of <code>SIG:Context</code> elements.</p> |  |
|                 | CONN:Status  | Enthält den Ausführungsstatus der Operation.   |
|                 | SIG:<br>ComfortSignatureStatus   | Komfortsignatur-<br>Konfigurationsstatus des Konnektors  |
|                 | SIG:ComfortSignatureMax  | Im Konnektor konfigurierte Anzahl von Komfortsignaturen, die ohne erneute PIN-Eingabe ausgeführt werden dürfen,  |
|                 | SIG:<br>ComfortSignatureTimer  | Im Konnektor konfiguriertes Zeitintervall, in dem Komfortsignaturen ohne erneute PIN-Eingabe ausgeführt werden dürfen,<br>Format: "PTnHnMnS" (gemäß Datentyp xsd:duration) |
|                 | CONN:CardHandle  | Liste von HBA-CardHandles  |
|                 | SIG:Context  | Liste von im Konnektor hinterlegten Aufrufkontexten für das jeweilige HBA-CardHandle<br>MandantId, ClientSystemId, UserId verpflichtend                                    |
|                 | SIG:SignatureMode  | Im Konnektor hinterlegter Signaturmodus für den jeweiligen Aufrufkontext   |

|                        |                    |   |
|------------------------|--------------------|---|
|                        | SIG:CountRemaining | Verbleibende Anzahl von Komfortsignaturen, die ohne erneute PIN-Eingabe ausgeführt werden dürfen  |
|                        | SIG:TimeRemaining  | Verbleibende Zeit, in der Komfortsignaturen ohne erneute PIN-Eingabe ausgeführt werden dürfen<br>Format: "PTnHnMnS" (gemäß Datentyp xsd:duration) |
| <b>Vorbedingungen</b>  | Keine              |   |
| <b>Nachbedingungen</b> | Keine              |   |

4561 **Tabelle 248: TAB\_KON\_882 Ablauf GetSignatureMode**

| Nr. | Aufruf Technischer Use Case oder Interne Operation | Beschreibung   |
|-----|--|--|
| 1.  | checkArguments                                     | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.   |
| 2.  | TUC_KON_173 „Liefere Signaturmodus“                | Der Komfortsignatur-Konfigurationsstatus des Konnektors und die im Konnektor hinterlegten Signaturmodus werden für alle dem Konnektor bekannten Aufrufkontexte der HBA aus der übergebenen CardHandle-Liste zurückgeliefert. |

4562

4563 **Tabelle 249: TAB\_KON\_881 Fehlercodes GetSignatureMode**

| Fehlercode                             | ErrorType | Severity | Fehlertext   |
|--|-----------|----------|--------------|
| Folgende Fehlercodes können auftreten: |           |          |              |
| 4000                                   | Technical | Error    | Syntaxfehler |

4564 [**<=**]4565 **4.1.8.6 Betriebsaspekte**4566 **TIP1-A\_4680-02 - Konfigurationswerte des Signaturdienstes**

4567 Die Managementschnittstelle MUSS es einem Administrator ermöglichen  
 4568 Konfigurationsänderungen gemäß Tabelle TAB\_KON\_596 vorzunehmen:  
 4569

4570 **Tabelle 250: TAB\_KON\_596 Konfigurationswerte des Signaturdienstes (Administrator)**

| ReferenzID | Belegung | Bedeutung und Administrator-Interaktion |
|------------|----------|---|
|------------|----------|---|

|                                 |                      |  |
|---------------------------------|----------------------|--|
| SAK_SIMPLE_<br>SIGNATURE_ MODE  | SE#1<br>SE#2         | Aktivierung/Deaktivierung des „Einfachsignaturmodus“ für alle HBAx für die Durchführung von Einfachsignaturen im SecurityEnvironment #1 (SE#1) für Dokumentenstapel der Größe 1 anstelle der Verwendung des SE#2.<br>Default-Wert = SE#1<br>Der Parameter ist nur relevant, wenn die Komfortsignaturfunktion nicht aktiviert ist (SAK_COMFORT_SIGNATURE = Disabled). |
| SAK_COMFORT_<br>SIGNATURE       | Enabled/<br>Disabled | Aktivierung/Deaktivierung der Komfortsignaturfunktion im Konnektor<br>Default-Wert = Disabled<br>Die Komfortsignaturfunktion darf nur aktiviert sein, wenn<br>ANCL_TLS_MANDATORY = Enabled und<br>ANCL_CAUT_MANDATORY = Enabled  |
| SAK_COMFORT_<br>SIGNATURE_MAX   | [1 - 250]            | Anzahl von Komfortsignaturen, die ohne erneute PIN-Eingabe ausgeführt werden dürfen<br>Default-Wert = 100<br>Der Parameter ist nur relevant, wenn die Komfortsignaturfunktion aktiviert ist (SAK_COMFORT_SIGNATURE = Enabled).   |
| SAK_COMFORT_<br>SIGNATURE_TIMER | [1 - 24 h]           | Zeitintervall, in dem Komfortsignaturen ohne erneute PIN-Eingabe ausgeführt werden dürfen<br>Der Timer startet mit Eingabe der PIN.QES für die Komfortsignatur.<br>Default-Wert = 6 h<br>Der Parameter ist nur relevant, wenn die Komfortsignaturfunktion aktiviert ist (SAK_COMFORT_SIGNATURE = Enabled).   |

4571

4572

[&lt;=]

## 4.1.9 Zertifikatsdienst

Der Zertifikatsdienst bietet eine Schnittstelle zur Überprüfung der Gültigkeit von Zertifikaten an. Dies geschieht auf Grundlage des durch den Vertrauensanker (TSL-CA-Signer-Zertifikat und eine aktuelle, gültige TSL aufgespannten Vertrauensraums sowie unter Berücksichtigung von aktuellen Statusinformationen (OCSP, CRL). Die Zertifikatsprüfung wird sowohl für nonQES- als auch für QES-Zertifikate unterstützt.

Die für die QES-Zertifikatsprüfung notwendigen QES-Signer-Zertifikate werden durch die Vertrauensliste der Bundesnetzagentur (BNetzA-VL) bereitgestellt. Das Signer-Zertifikat der BNetzA-VL ist in der TSL enthalten.

Im Rahmen der ECC-Migration muss der Konnektor neben RSA auch ECC unterstützen. Hierfür wird eine TSL bereitgestellt, die sowohl die neuen ECC-basierten Zertifikate als auch aus Rückwärtskompatibilitätsgründen die weiterhin benötigten RSA-basierten Zertifikate enthält. Diese neue TSL wird auch als „TSL(ECC-RSA)“ bezeichnet. In dieser Spezifikation wird außerhalb der Regelungen zur ECC-Migration nicht zwischen „TSL(ECC-RSA)“ und „TSL(RSA)“ unterschieden, da die Anforderungslage keine Unterscheidung erfordert.

Innerhalb des Zertifikatsdienstes werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): „CERT“
- Konfigurationsparameter: „CERT\_“

### 4.1.9.1 Funktionsmerkmalweite Aspekte

Bei der Zertifikatsprüfung wird im Rahmen eines Anwendungsfalls u.a. auch der Verwendungszweck des Zertifikats geprüft. Der Verwendungszweck (intendedKeyUsage) wird als Parameter an TUC\_KON\_037 übergeben. Der konkrete Wert von intendedKeyUsage ist abhängig vom kryptographischen Verfahren, auf welchem das Zertifikat basiert. Die Parametrisierung von intendedKeyUsage wird in TAB\_KON\_853 in Abhängigkeit vom zu prüfenden Zertifikat, dem Anwendungsfall und dem kryptographischen Verfahren definiert.

#### A\_17295 - Verwendung der intendedKeyUsage bei der Zertifikatsprüfung (ECC-Migration)

Der Konnektor MUSS bei der Zertifikatsprüfung die intendedKeyUsage in Abhängigkeit vom zu prüfenden Zertifikat, dem Anwendungsfall und dem kryptographischen Verfahren gemäß TAB\_KON\_853 prüfen.

**Tabelle 251: TAB\_KON\_853- intendedKeyUsage bei Zertifikatsprüfung**

| Zertifikat | Anwendungsfall  | intendedKeyUsage bei                 |                  |
|------------|---|--------------------------------------|------------------|
|            |   | RSA                                  | ECC              |
| C.SMKT.AUT | TUC_KON_050<br>„Beginne Kartenterminalsitzung“<br>TUC_KON_053<br>„Paire Kartenterminal“ | digitalSignature<br>&keyEncipherment | digitalSignature |

|  |   |                                      |                  |
|--|---|--------------------------------------|------------------|
| C.CH.AUT<br>C.CH.AUTN  | TUC_KON_161<br>„nonQES<br>Dokumentsignatur<br>prüfen“   | digitalSignature<br>&keyEncipherment | digitalSignature |
| C.CH.ENC<br>C.CH.ENCV<br>C.HCI.ENC<br>C.HP.ENC<br>Zertifikate aus<br>CERT_IMPORTED_CA_<br>LIST | TUC_KON_070<br>„Daten hybrid<br>verschlüsseln“  | keyEncipherment                      | keyAgreement     |
| C.HCI.OSIG   | TUC_KON_161<br>„nonQES<br>Dokumentsignatur<br>prüfen“   | nonRepudiation                       | nonRepudiation   |
| C.FD.TLS-S   | TUC_KON_110<br>„Kartenbasierte<br>TLS-Verbindung<br>aufbauen“   | digitalSignature&keyEncipherment     | digitalSignature |
| C.ZD.TLS-S   | TUC_KON_290<br>„LDAP-Verbindung<br>aufbauen“  | digitalSignature                     | digitalSignature |
| C.ZD.TLS-S   | TIP1-A_5662 -<br>Gesicherte<br>Übertragung von<br>BNetzA-VL und<br>Hashwert<br>TUC_KON_282<br>„UpdateInformationen<br>beziehen“<br>TUC_KON_283<br>Infrastruktur<br>Konfiguration<br>aktualisieren<br>TUC_KON_285<br>„UpdateInformationen<br>für Fachmodul<br>beziehen“<br>TUC_KON_286<br>„Paket für<br>Fachmodul laden“ | digitalSignature&keyEncipherment     | digitalSignature |
| C.FD.AUT   | A_17225   | digitalSignature&keyEncipherment     | digitalSignature |

4606 [ &lt;= ]

Bei der Zertifikatsprüfung wird ein übergebenes Zertifikat oder ein Zertifikat einer referenzierten Karte geprüft. Das konkrete Zertifikatsobjekt einer Karte ist abhängig vom Kartentyp und dem gewählten kryptographischen Verfahren. Die folgende Tabelle führt auf, welche Zertifikatsobjekte einer Karte in Abhängigkeit vom kryptographischen Verfahren für die jeweilige Zertifikatsreferenz ausgewählt werden.

**Tabelle 252: TAB\_KON\_858 Kartenobjekt in Abhängigkeit vom kryptographischen Verfahren**

| CertRef | Kartentyp | Objekt der Karte in Abhängigkeit vom kryptographischen Verfahren (Crypt) |                    |
|---------|-----------|--|--------------------|
|         |           | RSA  | ECC                |
| C.AUT   | HBA-VK    | EF.C.HP.AUT  | -                  |
|         | HBA       | EF.C.HP.AUT.R2048  | EF.C.HP.AUT.E256   |
|         | SM-B      | EF.C.HCI.AUT   | EF.C.HCI.AUT.E256  |
|         | eGK G2    | EF.C.CH.AUT.R2048  | EF.C.CH.AUT.E256   |
| C.ENC   | HBA-VK    | EF.C.HP.ENC  | -                  |
|         | HBA       | EF.C.HP.ENC.2048   | EF.C.HP.ENC.E256   |
|         | SM-B      | EF.C.HCI.ENC.R2048   | EF.C.HCI.ENC.E256  |
| C.SIG   | SM-B      | EF.C.HCI.OSIG.R2048  | EF.C.HCI.OSIG.E256 |
| C.QES   | HBA-VK    | EF.C.HP.QES  | -                  |
|         | HBA       | EF.C.HP.QES.R2048  | EF.C.HP.QES.E256   |

**TIP1-A\_4682 - Sicheres Einbringen des TI-Vertrauensankers**

Der Vertrauensanker der TI MUSS zum Auslieferungszeitpunkt des Konnektors integritätsgeschützt im Konnektor hinterlegt sein. Zur Sicherstellung dieser Integrität MUSS die Dateiablage EF.C.TSL.CA\_1 der Anwendung DF.Sicherheitsanker der gSMC-K [gemSpec\_gSMC-K\_ObjSys#5.7.2] verwendet werden.

[<=]

**TIP1-A\_4684 - Regelmäßige Aktualisierung der CRL und der TSL**

Falls Parameter MGM\_LU\_ONLINE=Enabled, MUSS der Zertifikatsdienst einmal täglich die Aktualisierung der TSL durch Aufruf von TUC\_KON\_032 „TSL aktualisieren“ durchführen und anschließend TUC\_KON\_040 „CRL aktualisieren“ aufrufen.

[<=]

**TIP1-A\_4685 - Vermeidung von Spitzenlasten bei TSL- und CRL-Download**

Der Konnektor MUSS Spitzenlasten durch paralleles Herunterladen der TSL und der CRL vermeiden. Dazu MÜSSEN die im Einsatz befindlichen Konnektoren eines Herstellers ihre Download-Versuche gleichmäßig über den Tag verteilen.

[<=]

Dadurch wird gleichzeitig die Spitzenlast bei OCSP-Anfragen begrenzt.

**A\_17572 - Nutzung der Hash-Datei für TSL (ECC-Migration)**

Falls die TSL(ECC-RSA) verwendet wird, MUSS der Konnektor vor deren Aktualisierung mit TUC\_KON\_032 „TSL aktualisieren“ die Hash-Datei der TSL(ECC-RSA) herunterladen, um zu prüfen, ob die am TSL-Downloadpunkt verfügbare TSL(ECC-RSA) eine andere ist, als die schon zuvor heruntergeladene und bereits ausgewertete TSL(ECC-RSA).

Entspricht der Hash-Wert am Download-Punkt der bereits heruntergeladenen und ausgewerteten TSL(ECC-RSA), MUSS der Konnektor auf den Download verzichten.[<=]

**A\_17661 - Gesicherte Übertragung der Hash-Datei für TSL (ECC-Migration)**

Der Konnektor MUSS für den Download der Hash-Datei der TSL(ECC-RSA) die Verbindung zum TSL-Dienst durch TLS absichern. Der Konnektor MUSS das vom TSL-Dienst beim TLS-Verbindungsaufbau präsentierte Zertifikat C.ZD.TLS-S prüfen. Die Prüfung erfolgt durch Aufruf von TUC\_KON\_037 „Zertifikat prüfen“ {

```
certificate = C.ZD.TLS-S;
qualifiedCheck = not_required;
offlineAllowNoCheck = true;
policyList = oid_zd_tls_s;
intendedKeyUsage = intendedKeyUsage(C.ZD.TLS-S);
intendedExtendedKeyUsage = id-kp-serverAuth;
validationMode = OCSP } .
```

Falls Fehler im TLS-Verbindungsaufbau bzw. bei der Zertifikatsprüfung auftreten MUSS der Konnektor den TLS-Verbindungsaufbau mit Fehlercode 4235 gemäß TAB\_KON\_825 abbrechen.

[<=]

**A\_17781 - Aktualisierung der TSL ohne Hash-Datei für TSL (ECC-Migration)**

Falls im Rahmen der TSL-Aktualisierung beim Download der Hash-Datei der TSL(ECC-RSA) ein Fehler auftritt MUSS der Konnektor die Aktualisierung der TSL mit TUC\_KON\_032 „TSL aktualisieren“ ohne einen ermittelten Hashwert aufrufen.[<=]

**TIP1-A\_6730 - Regelmäßige Aktualisierung der BNetzA-VL**

Falls Parameter MGM\_LU\_ONLINE=Enabled, MUSS der Zertifikatsdienst die Aktualisierung der BNetzA-VL im Zeitintervall CERT\_BNETZA\_VL\_UPDATE\_INTERVAL durch Aufruf von TUC\_KON\_031 „BNetzA-VL aktualisieren“ durchführen.

[<=]

**TIP1-A\_6731 - Regelmäßige Prüfung der BNetzA-VL**

Der Zertifikatsdienst MUSS einmal täglich die zeitliche Gültigkeit der BNetzA-VL prüfen. Wenn das Element NextUpdate in der Vergangenheit liegt MUSS der Konnektor den Betriebszustand EC\_BNetzA\_VL\_not\_valid auslösen.

[<=]

**TIP1-A\_6732 - Vermeidung von Spitzenlasten bei BNetzA-VL-Download**

Der Konnektor MUSS Spitzenlasten durch Herunterladen der BNetzA-VL vermeiden. Dazu MÜSSEN die im Einsatz befindlichen Konnektoren den Zeitpunkt für den Download zufällig wählen unter Beachtung des konfigurierten Zeitintervalls

CERT\_BNETZA\_VL\_UPDATE\_INTERVAL.

[<=]



**TIP1-A\_5662 - Gesicherte Übertragung von BNetzA-VL und Hashwert**

Der Konnektor MUSS für den Download der BNetzA-VL und deren Hashwert die Verbindung zum TSL-Dienst durch TLS absichern. Der Konnektor MUSS das vom TSL-Dienst beim TLS-Verbindungsaufbau präsentierte Zertifikat ID.ZD.TLS\_S prüfen. Die Prüfung erfolgt durch Aufruf von TUC\_KON\_037 „Zertifikat prüfen“ {

```
certificate = ID.ZD.TLS_S;
qualifiedCheck = not_required;
offlineAllowNoCheck = true;
policyList = oid_zd_tls_s;
intendedKeyUsage = intendedKeyUsage(C.ZD.TLS-S);
intendedExtendedKeyUsage = id-kp-serverAuth;
validationMode = OCSP } .
```

Fehler im TLS-Verbindungsaufbau bzw. bei der Zertifikatsprüfung führen zum Abbruch des TLS-Verbindungsaufbaus mit Fehlercode 4235 gemäß TAB\_KON\_825.

**Tabelle 253: TAB\_KON\_825 Fehlercodes „TLS-Verbindungsaufbau zum TSL-Dienst“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4235  | Security  | Error    | TSL-Dienst konnte bei TLS-Verbindungsaufbau nicht authentisiert werden |

[&lt;=]

**TIP1-A\_5663 - Prüfung der technischen Rolle bei TLS-Verbindungsaufbau zum TSL-Dienst**

Der Konnektor MUSS beim TLS-Verbindungsaufbau zum TSL-Dienst prüfen, dass die vom TSL-Dienst in ID.ZD.TLS\_S übergebene technische Rolle gemäß [gemSpec\_OID#GS-A\_4446] dem Wert „oid\_tsl\_ti“ entspricht.

Ein Fehler bei der Prüfung der technischen Rolle führt zum Abbruch des TLS-Verbindungsaufbaus mit Fehlercode 4236 gemäß TAB\_KON\_826.

**Tabelle 254: TAB\_KON\_826 Fehlercodes „TLS-Verbindungsaufbau zum TSL-Dienst bei Prüfung der technischen Rolle“**

| Fehlercode | ErrorType | Severity | Fehlertext  |
|------------|-----------|----------|---|
| 4236       | Security  | Error    | Rollenprüfung bei TLS-Verbindungsaufbau zum TSL-Dienst fehlgeschlagen |

[&lt;=]

**TIP1-A\_4686 - Warnung vor und bei Ablauf der TSL**

Steht der Ablauf der TSL innerhalb von 7 Tagen an, MUSS der Konnektor den Betriebszustand EC\_TSL\_Expiring annehmen.

Mit Ablauf der Gültigkeit der TSL MUSS der Konnektor den Betriebszustand EC\_TSL\_Out\_Of\_Date\_Within\_Grace\_Period annehmen.

Mit Ablauf der Graceperiod der TSL MUSS der Konnektor den kritischen Betriebszustand EC\_TSL\_Out\_Of\_Date\_Beyond\_Grace\_Period annehmen.

[&lt;=]



**TIP1-A\_4687 - Warnung vor und bei Ablauf des TI-Vertrauensankers**

Steht der Ablauf der Gültigkeit des TI-Vertrauensankers innerhalb von 30 Tagen an, MUSS der Konnektor den Betriebszustand EC\_TSL\_Trust\_Anchor\_Expiring annehmen. Mit Ablauf der Gültigkeit des Vertrauensankers MUSS der Konnektor den kritischen Betriebszustand EC\_TSL\_Trust\_Anchor\_Out\_Of\_Date annehmen.

[<=]

**TIP1-A\_4994 - Warnung vor und bei Ablauf der CRL**

Steht der Ablauf der Gültigkeit der CRL innerhalb von 3 Tagen an, MUSS der Konnektor den Betriebszustand EC\_CRL\_Expiring annehmen. Mit Ablauf der Gültigkeit der CRL MUSS der Konnektor den kritischen Betriebszustand EC\_CRL\_Out\_Of\_Date annehmen.

[<=]

**TIP1-A\_4688 - OCSP-Forwarding**

Der Konnektor MUSS alle OCSP-Anfragen über den OCSP-Forwarder (HTTP-Proxy) des Zugangsdienst-Providers schicken, der durch die Konfigurationswerte (CERT\_OCSP\_FORWARDER\_ADDRESS, CERT\_OCSP\_FORWARDER\_PORT) festgelegt ist.

[<=]

**TIP1-A\_4689 - Caching von OCSP-Antworten**

Der Zertifikatsdienst MUSS erhaltene OCSP-Antworten für eine durch CERT\_OCSP\_DEFAULT\_GRACE\_PERIOD\_NONQES angegebene Anzahl an Minuten (nonQES-Zertifikate) zwischenspeichern.

[<=]

**TIP1-A\_4690 - Timeout und Graceperiod für OCSP-Anfragen**

Bei Ausführung von TUC\_PKI\_006 „OCSP-Abfrage“ [gemSpec\_PKI#8.3.2.2] MÜSSEN folgende Parameter verwendet werden:

OCSP-Graceperiod =

CERT\_OCSP\_DEFAULT\_GRACE\_PERIOD\_NONQES

- Timeout-Parameter =

CERT\_OCSP\_TIMEOUT\_NONQES bzw.

CERT\_OCSP\_TIMEOUT\_QES

[<=]

**TIP1-A\_4691 - Ablauf der gSMC-K und der gesteckten Karten regelmäßig prüfen**

Für die gSMC-K sowie für jede gesteckte Karte außer eGK MUSS der Konnektor im Intervall CERT\_EXPIRATION\_CARD\_CHECK\_DAYS genau einmal TUC\_KON\_033 aufrufen.

Der Konnektor MUSS die Gültigkeitsdauer der Zertifikate prüfen mittels Aufruf von: für gSMC-K

TUC\_KON\_033{checkSMCK; doInformClients=Ja; crypt = ECC}

TUC\_KON\_033{checkSMCK; doInformClients=Ja; crypt = RSA}

für jede gesteckte G2.0 Karte außer eGK und außer gSMC-K

TUC\_KON\_033{cardSession; doInformClients=Ja; crypt = RSA}

für jede gesteckte ab G2.1 Karte außer eGK

TUC\_KON\_033{cardSession; doInformClients=Ja; crypt = ECC}

TUC\_KON\_033{cardSession; doInformClients=Ja; crypt = RSA}

[<=]

**TIP1-A\_4692 - Missbrauchserkennung, zu kontrollierende Operationen**

Der Konnektor MUSS zur Unterstützung von Missbrauchserkennungen die in Tabelle TAB\_KON\_597 gelisteten Operationen als Einträge in EVT\_MONITOR\_OPERATIONS berücksichtigen.

4762 **Tabelle 255: TAB\_KON\_597 Operationen in EVT\_MONITOR\_OPERATIONS**

| Operationsname    | OK_Val | NOK_Val | Alarmwert (Default-Grenzwert 10 Minuten- $\Sigma$ ) |
|-------------------|--------|---------|---|
| VerifyCertificate | 1      | 5       | 401   |

4763  
4764 [ $\leq$ ]

4765 **4.1.9.2 Durch Ereignisse ausgelöste Reaktionen**

4766 Keine.

4767 **4.1.9.3 Interne TUCs, nicht durch Fachmodule nutzbar**

4768 4.1.9.3.1 TUC\_KON\_032 „TSL aktualisieren“

4769 **TIP1-A\_4693-02TIP1-A\_4693 - TUC\_KON\_032 „TSL aktualisieren“**

4770 Der Konnektor MUSS den technischen Use Case TUC\_KON\_032 „TSL aktualisieren“  
4771 umsetzen.  
4772

4773 **Tabelle 256: TAB\_KON\_766 TUC\_KON\_032 „TSL aktualisieren“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_032 „TSL aktualisieren“  |
| Beschreibung   | Dieser TUC prüft die Aktualität der TSL und initialisiert ggf. den TSL-spezifischen Bereich des TrustStores neu. Zusätzlich wird bei einem Wechsel des TI-Vertrauensankers das neue TSL-Signer-CA-Zertifikat in einem sicheren Speicherort im Konnektor hinterlegt. Im Fall der Veröffentlichung eines CVC-Root-CA-Zertifikats werden das CVC-Root-CA-Zertifikat und die Cross-CV-Zertifikate aus der TSL in den Truststore eingestellt. |
| Auslöser       | <ul style="list-style-type: none"> <li>Aufruf durch andere TUCs</li> </ul>   |
| Vorbedingungen | <ul style="list-style-type: none"> <li>Ein gültiger TI-Vertrauensanker ist vorhanden</li> <li>Das XML-Schema der TSL-Datei liegt vor</li> </ul>  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>importedTSL – <i>optional</i><br/>(TSL aus manuellem Import) (Optional)</li> <li>baseTime – <i>optional</i>; <i>default</i>: <i>aktuelles Datum</i><br/>(Referenzzeitpunkt) ( )</li> <li>onlineMode [ENABLED   DISABLED]<br/>(Flag „MGM_LU_ONLINE“ für Offline/Online-Modus)</li> <li>hashTSL – <i>optional</i><br/>(Hashwert-Datei der TSL im System; gilt nur für TSL(ECC-RSA))</li> </ul>      |

| Komponenten                | Konnektor   |
|----------------------------|---|
| Ausgangsdaten              | <ul style="list-style-type: none"> <li>• result<br/>(Status der Prüfung)</li> <li>• newHashTSL - <i>optional; verpflichtend für TSL(ECC-RSA)</i><br/>(Hashwert-Datei der TSL im System; gilt nur für TSL(ECC-RSA))</li> </ul>   |
| Nachbedingungen            | <ul style="list-style-type: none"> <li>• Aktuelle TSL-Informationen inkl. des Vertrauensankers der BNetzA VL und sämtlicher CVC-Root-CA- und Cross-CV-Zertifikate liegen im Truststore vor.</li> <li>• Ein ggf. gelieferter neuer Vertrauensanker der TI ist in einem sicheren Speicherort gespeichert</li> </ul>   |
| Standardablauf             | <ol style="list-style-type: none"> <li>1. Der Konnektor prüft und aktualisiert ggf. die TSL durch Aufruf von TUC_PKI_001. <i>Der Konnektor verwendet bei der Aktualisierung der TSL standardmäßig die Download-Punkte in der TI.</i><br/>Der durch den dort aufgerufenen TUC_PKI_011 „Prüfung des TSL-Signer-Zertifikates“ benötigte aktuelle TI-Vertrauensanker befindet sich auf der gSMC-K in der Datei EF.C.TSL_CA_1 oder in einem sicheren Speicherort im Konnektor. Es ist dasjenige Zertifikat zu verwenden, welches zum Referenzzeitpunkt gültig ist und ab dem Aktivierungsdatum (<code>StatusStartingTime</code> des neuen TSL-Signer-CA-Zertifikats) aktiviert ist.</li> <li>2. Ggf. vorhandene CVC-Root-CA-Zertifikat und Cross-CV-Zertifikate werden genauso wie und zusammen mit den anderen CA-Zertifikaten aus der TSL extrahiert.</li> <li>3. Alle Informationen aus der TSL werden im TSL-spezifischen Bereich des TrustStores gespeichert</li> <li>4. Der Konnektor löst TUC_KON_256 {<br/>    topic = „CERT/TSL/UPDATED“;<br/>    eventType = Op;<br/>    severity = Info;<br/>    doLog = true;<br/>    doDisp = false }<br/>aus.</li> <li>5. CERT_CRL_DOWNLOAD_ADDRESS wird mit den CRL-Download-Adressen aus der TSL überschrieben.</li> </ol> |
| Varianten/<br>Alternativen | <p>(-&gt; 1) Wenn der Download der TSL aus der TI fehlschlägt oder wenn der Konnektor im <i>FallonlineMode = ENABLED</i> keine Verbindung zur TI hat, muss der Konnektor die TSL vom Download-Punkt im Internet (CERT_TSL_DOWNLOAD_ADDRESS_INTERNET) gemäß [gemSpec_TSL#A_21182] beziehen. Im Fall <i>onlineMode = DISABLED</i> wird abgebrochen.</p> <p>Wenn die Namensauflösung für CERT_TSL_DOWNLOAD_ADDRESS_INTERNET fehlschlägt, muss der Konnektor die TSL über CERT_TSL_IP_ADDRESS_INTERNET beziehen.</p>  |

|  |   |
|--|---|
|  | <p>Wenn keiner der vorigen Downloadversuche erfolgreich war, muss der Konnektor die TSL von der konfigurierbaren Adresse CERT_TSL_DOWNLOAD_ADDRESS_INTERNET_BU beziehen.</p> <p>Wenn die Namensauflösung für CERT_TSL_DOWNLOAD_ADDRESS_INTERNET_BU fehlschlägt, muss der Konnektor die TSL über CERT_TSL_IP_ADDRESS_INTERNET_BU beziehen.</p> <p>Für eine aus dem Internet bezogene TSL muss der Konnektor auch die vom TSL-Dienst gemäß [gemSpec_TSL#A_21182] bereitgestellte detached-Signatur der TSL herunterladen. Der Konnektor muss dann immer zunächst die detached-Signatur der TSL prüfen, einschließlich vollständiger Prüfung der Zertifikatskette bis zum TI-Vertrauensanker. Die kryptographische Prüfung der Signatur muss entsprechend A_21185 durchgeführt werden.</p> <p>Bezüglich der Prüfung des Sperrstatus des TSL-Signer-Zertifikats muss der Konnektor eines der folgenden Verfahren umsetzen:</p> <ol style="list-style-type: none"> <li>1. Der Konnektor lädt die vorgefertigte OCSP-Response für das TSL-Signer Zertifikat aus dem Internet herunter (vgl. [gemSpec_TSL#A_21182]). Bei der Prüfung dieser OCSP-Response entfällt die Auswertung gegen die im System konfigurierte OCSP-Graceperiod. Der Konnektor prüft, dass die vorgefertigte OCSP-Response nicht älter als 61 Minuten ist. Die OCSP-Abfrage für das TSL-Signer Zertifikat in TUC_PKI_001, Schritt 4 entfällt. oder</li> <li>2. Standard OCSP-Abfrage für das TSL-Signer Zertifikat in TUC_PKI_001, Schritt 4, jedoch unter Verwendung des im Internet verfügbaren OCSP-Responders entsprechend [gemSpec_TSL#TIP1-A_4076-01].</li> </ol> <p>(→1) Wird die <i>importedTSL</i> manuell übergeben (in den Eingangsdaten), so wird diese statt des Downloads verwendet und an TUC_PKI_001 übergeben. Innerhalb der PKI TUCs findet dann kein Download der TSL statt.</p> <p>(→1) Falls <i>onlineMode</i> = DISABLED, kann der Sperrstatus des TSL-Signer-Zertifikats nicht überprüft werden. In diesem Fall wird die Aktivierung der <i>importedTSL</i> auch ohne Prüfung des Sperrstatus durchgeführt.</p> <p>(→1) Wird durch den von TUC_PKI_001 aufgerufenen TUC_PKI_013 „Import neuer Vertrauensanker“ ein neuer TI-Vertrauensanker (ein neues TSL-Signer-CA-Zertifikat) in der <i>importedTSL</i> gefunden, so wird dieser, wie dort beschrieben, extrahiert und in einem sicheren Speicherort gespeichert. Vor Erreichen des Aktivierungsdatums werden die TSLs ausschließlich mit dem alten TSL-Signer-Zertifikat signiert. Ab dem Aktivierungsdatum werden die TSLs mit einem TSL-Signer-Zertifikat signiert, das von der neuen TSL-Signer-CA ausgestellt wurde.</p> |
|--|---|

|                                |  |
|--------------------------------|--|
| Fehlerfälle                    | <p>(→1) Tritt beim manuellen Import der Datei ein Fehler auf, wird TUC_KON_256 {<br/> topic = „CERT/TSL/IMPORT“;<br/> eventType = Op;<br/> severity = Error;<br/> parameters = „\$Fehlerbeschreibung“;<br/> doLog = true;<br/> doDisp = false }<br/> ausgelöst. Fehlercode 4128.</p> <p>(→1) Tritt beim periodischen Update der TSL beim Aufruf des TUC_PKI_001 oder eines durch ihn aufgerufenen TUCs ein Fehler auf, geht der Konnektor in den Betriebszustand EC_TSL_Update_Not_Successful. Der Konnektor geht erst in den Betriebszustand EC_TSL_Update_Not_Successful, wenn weder der Downloadversuch aus der TI noch der Downloadversuch aus dem Internet erfolgreich war. Die vorhandenen TSL-Vertrauensanker werden weiter verwendet. Fehlercode 4127.</p> |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | keine  |

4774 **Tabelle 257: TAB\_KON\_598 Fehlercodes TUC\_KON\_032 „TSL aktualisieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext                                     |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4127  | Security  | Error    | Import der TSL-Datei fehlgeschlagen            |
| 4128  | Technical | Error    | der manuelle Import der TSL-Datei schlägt fehl |

4775  
4776 **[<=]**

4777 Für den Download der TSL über einen HTTP-Server im Internet wird zusätzlich zu der  
4778 bereits mit einer XML-Signatur versehenen TSL eine detached-Signatur als separate  
4779 Datei auf dem Download-Punkt zur Verfügung gestellt. Diese detached-Signatur umfasst  
4780 die TSL in ihrerer Gänze, das heißt die TSL-XML-Datei wird inkl. der dort bereits  
4781 enthaltenen XML-Signatur nochmal durch den TSL-Signer signiert. Ein Konnektor  
4782 verwendet dann bei der Signaturprüfung einer TSL, die über das Internet bezogen wurde,  
4783 die detached-Signatur für die Signaturprüfung. Hintergrund ist die aus  
4784 Sicherheitsperspektive einfachere, im Sinne von sicherer prüfbare detached-Signatur.  
4785 Das heißt, die TSL muss nicht als XML-File verarbeitet und die relativ komplexe XML-  
4786 Signatur - die potentiell von einem Angreifer modifiziert sein könnte - nicht ausgewertet  
4787 werden. Deshalb wird der Weg gewählt, der auch für die Signatur von X.509-Zertifikaten  
4788 und OCSP-Responses verwendet wird.

## A\_21185 - Prüfung der detached Signatur der TSL bei Download aus dem Internet

Der Konnektor MUSS beim Download der TSL aus dem Internet ebenfalls deren detached-Signatur (vgl. [gemSpec\_TSL#A\_21182]) mit herunterladen und immer zunächst folgende Prüfungen durchführen:

1. Prüfung, dass die heruntergeladene detached-Signatur-Datei den folgenden Aufbau aufweist:

Sequence aus drei Elementen:

```
SEQUENCE {
```

```
  a
```

```
  b
```

```
  c}
```

Mit a, b und c wie folgt:

- a. OID für den Signatortyp, bestehend aus

- i. im Falle ECDSA:

```
SEQUENCE {OBJECT IDENTIFIER ecdsaWithSHA256 (1 2 840 10045 4
3 2)}
```

- ii. im Falle RSASSA-PSS:

```
SEQUENCE {OBJECT IDENTIFIER rsaPSS (1 2 840 113549 1 1 10)
```

```
  SEQUENCE {
```

```
    [0] {SEQUENCE {OBJECT IDENTIFIER sha-256 (2 16 840 1
101 3 4 2 1)}}}
```

```
    [1] {SEQUENCE {
```

```
      OBJECT IDENTIFIER pkcs1-MGF (1 2 840 113549 1
```

```
1 8)
```

```
      SEQUENCE {OBJECT IDENTIFIER sha-256 (2 16 840
```

```
1 101 3 4 2 1)}}}
```

```
    [2] {INTEGER 32}}}
```

- b. Kryptographische Signatur

- i. im Falle ECDSA:

einer ECDSA-Signatur nach [BSI-TR-03111#5.2.2.]

- ii. im Falle RSASSA-PSS:

eine RSASSA-PSS-Signatur nach [RFC-8017] (reiner ASN.1-kodierter Signaturwert – die OID ist schon in Teil a.ii. aufgeführt)

- c. Zertifikat des Signierenden (TSL-Signer)

- i. im Falle ECDSA:

genau nur das ECC-TSL-Signer-Zertifikat

- ii. im Falle RSASSA-PSS:

genau nur das RSA-TSL-Signer-Zertifikat

2. Prüfung der Signatur (1b) gegen das TSL-Signer-Zertifikat (1c).

Schlägt eine der Prüfungen fehl, MUSS der Import abgebrochen werden.

Ist die Prüfung erfolgreich, KANN die Prüfung der XML-Signatur der TSL im weiteren fachlichen Ablauf der TSL-Aktualisierung entfallen.

[<=]

Eine erweiterte Übersicht zum Aufbau der detached-Signatur-Datei inkl. Beispiel finden sie unter [gemGithub\_tslSig].

#### A\_20750 - Hinweis auf Betreiber-Verantwortung bei automatischer TSL-Aktualisierung

Der Hersteller des Konnektors MUSS den Betreiber des Konnektors in geeigneter Weise (mindestens per Handbuch-Eintrag und per Hinweis innerhalb der UpdateInformation eines FirmwareUpdates am KSR) darüber informieren, dass im Fall, dass eine TSL-Aktualisierung innerhalb der TI fehlschlägt, automatisch versucht wird, eine TSL-Aktualisierung aus dem Internet vorzunehmen. [ $\leq$ ]

#### 4.1.9.3.2 TUC\_KON\_031 „BNetzA-VL aktualisieren“

##### TIP1-A\_6729 - TUC\_KON\_031 „BNetzA-VL aktualisieren“

Der Konnektor MUSS den technischen Use Case TUC\_KON\_031 „BNetzA-VL aktualisieren“ umsetzen.

**Tabelle 258: TAB\_KON\_618 TUC\_KON\_031 „BNetzA-VL aktualisieren“**

| Element                | Beschreibung   |
|------------------------|--|
| Name                   | TUC_KON_031 „BNetzA-VL aktualisieren“  |
| Beschreibung           | Dieser TUC prüft die Aktualität der BNetzA-VL. Wenn eine neuere BNetzA-VL vorliegt, wird diese heruntergeladen, geprüft und im Truststore gespeichert.   |
| Auslöser               | <ul style="list-style-type: none"> <li>Aufruf durch andere TUCs</li> <li>TIP1-A_6728</li> </ul>  |
| Vorbedingungen         | <ul style="list-style-type: none"> <li>Aktuell gültige TSL im Truststore vorhanden</li> </ul>  |
| Eingangsdaten          | <ul style="list-style-type: none"> <li>BNetzA-VL aus manuellem Import (Optional)</li> <li>Flag „MGM_LU_ONLINE“ für Offline-/Online-Modus</li> <li>Flag „MGM_LU_SAK“ für Signaturdienst-Modus</li> </ul>  |
| Komponenten            | Konnektor  |
| Ausgangsdaten          | <ul style="list-style-type: none"> <li>Status der Prüfung</li> </ul>   |
| Nachbedingungen        | <ul style="list-style-type: none"> <li>Aktuelle BNetzA-VL und deren Hashwert liegen im Truststore vor.</li> </ul>  |
| Standardablauf         | <ol style="list-style-type: none"> <li>Der Konnektor prüft und aktualisiert ggf. die BNetzA-VL durch Aufruf von TUC_PKI_036.</li> <li>Der Konnektor löst TUC_KON_256 {"CERT/BNETZA_VL/UPDATED"; Op; Info; „"; doLog = true; doDisp = false} aus.</li> </ol>  |
| Varianten/Alternativen | <p>(→1) Wird eine zu importierende BNetzA-VL manuell übergeben (in den Eingangsdaten), so wird diese statt des Downloads verwendet und an TUC_PKI_036 {BNetzA-VL Datei} übergeben. Innerhalb der PKI TUCs findet dann kein Download der BNetzA-VL statt.</p> <p>(→1) Ist MGM_LU_SAK=disabled, so wird der TUC ohne Fehler beendet.</p> |



|                                |  |
|--------------------------------|--|
| Fehlerfälle                    | (→1) Tritt beim manuellen Import der Datei ein Fehler auf, wird TUC_KON_256 {„CERT/BNETZA_VL/IMPORT“; Op; Error; „\$Fehlerbeschreibung“; doLog = true; doDisp = false} ausgelöst. Fehlercode 4129.<br>(→1) Tritt beim periodischen Update der BNetzA-VL beim Aufruf des TUC_PKI_036 oder eines durch ihn aufgerufenen TUCs ein Fehler auf, geht der Konnektor in den Betriebszustand EC_BNetzA_VL_Update_Not_Successful. Fehlercode 4133.<br>In beiden Fällen wird eine vorhandene gültige BNetzA-VL weiter verwendet. |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | keine  |

4851 **Tabelle 259: TAB\_KON\_619 Fehlercodes TUC\_KON\_031 „BNetzA-VL aktualisieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4129  | Technical | Error    | der manuelle Import der BNetzA-Vertrauensliste schlägt fehl |
| 4133  | Security  | Error    | Import der BNetzA-Vertrauensliste fehlgeschlagen            |

4852  
4853 **[<=]**

4854 **4.1.9.3.3 TUC\_KON\_040 „CRL aktualisieren“**

4855 **TIP1-A\_4694 - TUC\_KON\_040 „CRL aktualisieren“**

4856 Der Konnektor MUSS den technischen Use Case TUC\_KON\_040 „CRL aktualisieren“  
4857 umsetzen.

4858 **Tabelle 260: TAB\_KON\_767 TUC\_KON\_040 „CRL aktualisieren“**

| Element         | Beschreibung  |
|-----------------|---|
| Name            | TUC_KON_040 „CRL aktualisieren“   |
| Beschreibung    | Dieser TUC aktualisiert die CRL   |
| Auslöser        | <ul style="list-style-type: none"> <li>Aufruf durch andere TUCs</li> </ul>                                    |
| Vorbedingungen  | <ul style="list-style-type: none"> <li>Ein gültiger Vertrauensraum</li> </ul>                                 |
| Eingangsdaten   | <ul style="list-style-type: none"> <li>importedCRL – <i>optional</i><br/>(Manuell importierte CRL)</li> </ul> |
| Komponenten     | Konnektor   |
| Ausgangsdaten   | Keine   |
| Nachbedingungen | <ul style="list-style-type: none"> <li>Eine aktuelle, gültige CRL liegt vor</li> </ul>                        |



|                            |   |
|----------------------------|---|
| Standardablauf             | <ol style="list-style-type: none"> <li>1. Der Konnektor lädt die aktuelle CRL von CERT_CRL_DOWNLOAD_ADDRESS herunter.</li> <li>2. Die Prüfung der CRL-Signatur mit dem CRL-Signer-Zertifikat setzt sich aus folgenden Teilschritten zusammen <ol style="list-style-type: none"> <li>a. Prüfung auf zeitliche Gültigkeit des CRL-Signer-Zertifikats mittels TUC_PKI_002 "Gültigkeitsprüfung des Zertifikats" mit Referenzzeitpunkt = aktuelle Systemzeit</li> <li>b. Auswahl des öffentlichen Schlüssels des CRL-Signer-Zertifikats (CRL-Signer-Zertifikat im Truststore)</li> <li>c. Die Signatur und der verwendete Algorithmus werden aus der CRL ausgelesen</li> <li>d. Verifikation der Signatur und Hashwert-Vergleich (Verfahren siehe [RFC5280]). Falls die Prüfung ein negatives Ergebnis erbracht hat, löst der Konnektor das Ereignis TUC_KON_256 { <pre> topic = „CERT/CRL/INVALID“; eventType = Op; severity = Error; parameters = “”; doLog = true; doDisp = false } </pre> aus. </li> </ol> </li> <li>3. Nach einer erfolgreichen Prüfung speichert der Konnektor die neue CRL und löst das Ereignis TUC_KON_256{ <pre> topic = „CERT/CRL/UPDATED“; eventType = Op; severity = Error; parameters = “”; doLog = true; doDisp=false} </pre> aus. </li> <li>4. Falls die aktuelle Systemzeit den Wert NextUpdate aus der CRL erreicht oder überschritten hat, geht der Konnektor in den Betriebszustand EC_CRL_Out_Of_Date.</li> </ol> |
| Varianten/<br>Alternativen | (→1) Wird eine manuell importierte CRL übergeben, so wird diese verwendet.  |
| Fehlerfälle                | (→1) Tritt beim manuellen Import der Datei ein Fehler auf, wird TUC_KON_256 { <pre> topic = „CERT/CRL/IMPORT“; eventType = Op; severity = Error; parameters = „\${Fehlerbeschreibung}“; doLog = true; doDisp=false} </pre>  |

|                                |  |
|--------------------------------|--|
|                                | ausgelöst.<br>(→2) Signaturprüfung der CRL fehlgeschlagen: Fehlercode 4130 |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | keine  |

4859 **Tabelle 261: TAB\_KON\_599 Fehlercodes TUC\_KON\_040 „CRL aktualisieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4130  | Security  | Error    | Signatur- oder Gültigkeitsprüfung der CRL fehlgeschlagen |

4860

4861 [ $\leq$ ]

4862 4.1.9.3.4 TUC\_KON\_033 „Zertifikatsablauf prüfen“

4863 **TIP1-A\_4695 - TUC\_KON\_033 „Zertifikatsablauf prüfen“**

4864 Der Konnektor MUSS den technischen Use Case TUC\_KON\_033 „Zertifikatsablauf prüfen“  
4865 umsetzen.

4866

4867 **Tabelle 262: TAB\_KON\_768 TUC\_KON\_033 „Zertifikatsablauf prüfen“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_033 „Zertifikatsablauf prüfen“   |
| Beschreibung   | Dieser TUC prüft und meldet das zeitliche Ablaufen eines X.509-Zertifikats einer Karte.  |
| Auslöser       | <ul style="list-style-type: none"> <li>• Aufruf durch andere TUCs des Konnektors oder</li> <li>• über die Managementschnittstelle</li> </ul>   |
| Vorbedingungen | Keine  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• cardSession – <i>optional</i>/für eGK, HBA, SM-B, gSMC-KT</li> <li>• checkSMCK [Boolean] – <i>optional</i>/für gSMC-K;<br/>(Referenz auf eine/die gSMC-K, alternativ zu cardSession)</li> <li>• doInformClients [Boolean]<br/>(Angabe, ob ein Event an die Clients gesendet werden soll)</li> <li>• crypt – <i>optional</i>; default = RSA<br/>(kryptographischer Algorithmus, für welchen das</li> </ul> |

|                |  |
|----------------|--|
|                | Zertifikat ermittelt wird;<br>Wertebereich: ECC, RSA)  |
| Komponenten    | Konnektor  |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>expirationDate<br/>(Ablaufdatum des untersuchten Zertifikats)</li> </ul>  |
| Standardablauf | <p>1. TUC_KON_216 „LeseZertifikat“ für:</p> <ul style="list-style-type: none"> <li>Bei checkSMCK das Zertifikat der gSMC-K (C.NK.VPN) gemäß TAB_KON_856</li> <li>bei CardSession die Zertifikate der identifizierten Karte. <ul style="list-style-type: none"> <li>i. Für die eGK: C.CH.AUT</li> <li>ii. Für den HBAX: C.HP.AUT</li> <li>iii. Für SM-B: C.HCI.AUT</li> </ul> </li> <li>Das konkrete Zertifikatsobjekt der Karte gemäß TAB_KON_858 wird vom Eingangsparameter <code>crypt</code> abgeleitet.</li> </ul> <p>2. Das Ablaufdatum <code>expirationDate</code> wird aus dem Feld <code>validity</code> ausgelesen.</p> <p>3. Falls das Zertifikat abgelaufen ist, Systemereignis absetzen:</p> <ul style="list-style-type: none"> <li>gSMC-K:<br/> <pre>TUC_KON_256 {   topic = „CERT/CARD/EXPIRATION“;   eventType = Op;   severity = Warning;   parameters = („CARD_TYPE=gSMC-K,     ICCSN=\$ICCSN,     Konnektor=\$MGM_KONN_HOSTNAME,     ZertName=\$Name des Zertifikatsobjekts gemäß     TAB_KON_856,     ExpirationDate=\$validity");   doLog = true;   doDisp = \$doInformClients }</pre> </li> <li>Sonstige Karten (mit CARD(CardSession)): <pre>TUC_KON_256 {   topic = „CERT/CARD/EXPIRATION“;   eventType = Op;   severity = Warning;   parameters = („CARD_TYPE=\$Type,     ICCSN=\$ICCSN,     CARD_HANDLE=\$CardHandle,     CardHolderName=\$CardHolderName,     ZertName=\$Name des Zertifikatsobjekts aus     Schritt 1,     ExpirationDate=\$validity");   doLog=false;   doDisp = \$doInformClients }</pre> </li> </ul> |

|                            |   |
|----------------------------|---|
|                            | <p>4. Alternativ bei Ablauf des Zertifikats innerhalb von CERT_EXPIRATION_WARN_DAYS Systemereignis absetzen:</p> <ul style="list-style-type: none"> <li>gSMC-K:<br/>TUC_KON_256 {<br/>  topic = „CERT/CARD/EXPIRATION“;<br/>  eventType = Op;<br/>  severity = Info;<br/>  parameters = („CARD_TYPE=gSMC-K,<br/>    ICCSN=\$ICCSN,<br/>    Konnektor=\$MGM_KONN_HOSTNAME,<br/>    ZertName=\$Name des Zertifikatsobjekts<br/>gemäß TAB_KON_856,<br/>    ExpirationDate=\$validity,<br/>    DAYS_LEFT=\$validity-\$Today“);<br/>  doLog = false;<br/>  doDisp = \$doInformClients}</li> <li>Sonstige mit CARD(CardSession)):<br/>TUC_KON_256 {<br/>  topic = „CERT/CARD/EXPIRATION“;<br/>  eventType = Op;<br/>  severity = Info;<br/>  parameters = („CARD_TYPE=\$Type,<br/>    ICCSN = \$ICCSN,<br/>    CARD_HANDLE = \$CardHandle,<br/>    CardHolderName = \$CardHolderName,<br/>    ZertName=\$Name des Zertifikatsobjekts aus<br/>Schritt 1,<br/>    ExpirationDate = \$validity,<br/>    DAYS_LEFT = \$validity-\$Today“);<br/>  doLog = false;<br/>  doSisp = \$doInformClients}</li> </ul> <p>5. expirationDate wird zurückgegeben.</p> |
| Varianten/<br>Alternativen | Keine   |
| Fehlerfälle                | <p>(→1) Zur angegebenen CardSession keine Karte gefunden: Fehlercode 4131.</p> <p>(→1) Für eGK, HBA, SM-B gilt: Wenn crypt=ECC und Kartengeneration&lt;G2.1, bricht der TUC mit Warnung 4257 ab.</p> <p>(→1) Für gSMC-K gilt: Wenn crypt=ECC und beim Aufruf von TUC_KON_216 wird die Warnung 4256 zurückgegeben, dann wird der TUC nach Schritt 1 beendet und die Warnung 4257 an den Aufrufer zurückgegeben.</p> <p>(→2) Extraktion des Ablaufsdatums fehlgeschlagen: Fehlercode 4132.</p>  |

|                                |       |
|--------------------------------|-------|
| Nichtfunktionale Anforderungen | Keine |
| Zugehörige Diagramme           | Keine |

4868

4869 **Tabelle 263: TAB\_KON\_600 Fehlercodes TUC\_KON\_033 „Zertifikatsablauf prüfen“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4131  | Technical | Error    | Zum angegebenen CardHandle keine Karte gefunden.        |
| 4132  | Security  | Error    | Extraktion des Ablaufsdatums fehlgeschlagen             |
| 4257  | Technical | Warning  | ECC-Zertifikate nicht vorhanden auf Karte: <cardHandle> |

4870

4871 [**<=**]4872 **4.1.9.4 Interne TUCs, auch durch Fachmodule nutzbar**4873 **4.1.9.4.1 TUC\_KON\_037 „Zertifikat prüfen“**4874 **TIP1-A\_4696 - TUC\_KON\_037 „Zertifikat prüfen“**

4875 Der Konnektor MUSS den technischen Use Case „Zertifikat prüfen“ gemäß TUC\_KON\_037  
 4876 „Zertifikat prüfen“ umsetzen.[**<=**]

4877 **Tabelle 264: TAB\_KON\_769 TUC\_KON\_037 „Zertifikat prüfen“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_037 „Zertifikat prüfen“   |
| Beschreibung   | Der TUC beschreibt <ul style="list-style-type: none"> <li>die Prüfung eines X.509-Zertifikats gegen den Vertrauensraum</li> </ul>                                   |
| Auslöser       | <ul style="list-style-type: none"> <li>Aufruf in einem Fachmodul oder</li> <li>technischen Use Case</li> </ul>  |
| Vorbedingungen | <ul style="list-style-type: none"> <li>aktuelle TSL-Informationen im Truststore vorhanden</li> <li>für QES X.509-Prüfung: eine aktuell gültige BNetzA-VL</li> </ul> |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>certificate<br/>(ein X.509-Zertifikat (nonQES- oder QES-X.509-Zertifikat))</li> </ul>  |

|  |  |
|--|--|
|  | <ul style="list-style-type: none"> <li>• EECertificateContainedInTSL - <i>optional; default: false</i><br/>(true: Prüfung, ob ein EE-Zertifikat in der TSL vorhanden und zeitlich gültig ist; EE-Zertifikat wird in der TSL innerhalb eines "TSPService"-Eintrags ServiceTypeIdentifier "http://uri.etsi.org/TrstSvc/Svctype/unspecified" erwartet. false: vollständige Prüfung eines X.509-Zertifikats mit TUC_PKI_018 bzw. TUC_PKI_030)</li> <li>• qualifiedCheck [not_required   required   if_QC_present] – (Art der Zertifikatsprüfung)</li> <li>• baseTime – <i>optional/verpflichtend, wenn ein Zeitpunkt zur Prüfung vorgegeben werden soll; default: Verwendung der Systemzeit des Konnektors</i><br/>(Referenzzeitpunkt: Zeitpunkt, für den das Zertifikat geprüft werden soll)</li> <li>• offlineAllowNoCheck [Boolean] – <i>optional; default: false</i><br/>(Angabe, ob es als Fehler (false) oder als Warnung (true) interpretiert werden soll, wenn eine OCSP-Prüfung nicht durchgeführt werden konnte.)</li> <li>• intendedKeyUsage – <i>optional/verpflichtend, wenn certificate ein nonQES-X.509-Zertifikat ist; wird bei QES nicht ausgewertet</i><br/>(Vorgesehene KeyUsage)</li> <li>• nur für nonQES-Zertifikate: <ul style="list-style-type: none"> <li>• policyList<br/>(Liste der zugelassenen Zertifikatstyp-OIDs gemäß [gemSpec_OID#GS-A_4445])</li> <li>• intendedExtendedKeyUsage – <i>optional/verpflichtend, wenn certificate ein nonQES-X.509-Zertifikat ist; wird bei QES nicht ausgewertet</i><br/>(Vorgesehene ExtendedKeyUsage)</li> <li>• gracePeriod – <i>optional/nur für nonQES-X.509-Zertifikat und wenn vom Standard abgewichen werden soll; wird bei QES nicht ausgewertet; default: CERT_OCSP_DEFAULT_GRACE_PERIOD_NONQES</i><br/>(OCSP-GracePeriod: maximal zulässiger Zeitraum, den letzte OCSP-Antwort aus dem Cache bezüglich des Referenzzeitpunkts zurückliegen darf;)</li> <li>• validationMode [OCSP   CRL   NONE] – <i>optional/verpflichtend, wenn certificate ein nonQES-X.509-Zertifikat ist</i><br/>(Prüfmodus: <ul style="list-style-type: none"> <li>• OCSP: Es wird mittels OCSP geprüft. Dabei wird, falls die OCSP-GracePeriod noch nicht abgelaufen ist, die OCSP-Antwort aus dem Cache des Konnektors verwendet. Für QES einzig erlaubter validationMode.</li> </ul> </li> </ul> </li> </ul> |
|--|--|

|               |   |
|---------------|---|
|               | <ul style="list-style-type: none"> <li>• CRL: Es wird gegen die aktuelle CRL auf dem Konnektor geprüft.</li> <li>• NONE: Keine Prüfung von Statusinformationen)</li> <li>• ocspResponse – <i>optional</i><br/>(OCSPResponse des EE-Zertifikats)</li> <li>• getOCSPResponses [Boolean]– <i>optional; default: false</i><br/>(true – OCSPResponse des geprüften Zertifikats soll an den Aufrufer zurückgegeben werden)</li> </ul>   |
| Komponenten   | Konnektor   |
| Ausgangsdaten | <ul style="list-style-type: none"> <li>• Status und Liste von Warnungen/Fehlern bei der Zertifikatsprüfung</li> <li>• role<br/>(aus dem Zertifikate ermittelte Rolle oder Berufsgruppe; siehe „Tab_PKI_406 OID-Festlegung technische Rolle in X.509-Zertifikaten“ oder „Tab_PKI_402 OID-Festlegung Rolle im X.509-Zertifikat für Berufsgruppen“ oder Tab_PKI_403 OID-Festlegung Institutionen im X.509-Zertifikat der SMC-B [gemSpec_OID])</li> <li>• qcStatement – <i>optional/verpflichtend, wenn certificate ein QES-X.509-Zertifikat ist; nicht relevant bei EECertificateContainedInTSL=true.</i><br/>(QCStatements des Zertifikats)</li> <li>• ocspResponsesRenewed – <i>optional/verpflichtend, wenn Eingabeparameter getOCSPResponses = true; nicht relevant bei EECertificateContainedInTSL=true.</i><br/>(OCSP-Response des geprüften Zertifikats)</li> </ul> |

|                |  |
|----------------|--|
| Standardablauf | <p>1. Falls <code>EECertificateContainedInTSL=false</code>:</p> <ol style="list-style-type: none"> <li>Wenn das X.509-Zertifikat von einem CA-Zertifikat ausgestellt wurde, das in <code>CERT_IMPORTED_CA_LIST</code> enthalten ist, erfolgt eine Zertifikatsprüfung analog zu den Festlegungen in <code>TUC_PKI_018 „Zertifikatsprüfung“</code>. Dabei sind zu prüfen: <ul style="list-style-type: none"> <li>- Zeitliche Gültigkeit,</li> <li>- mathematische Prüfung der Zertifikatssignatur,</li> <li>- die Prüfung der Zweckbindung gemäß der im Zertifikat hinterlegten <code>keyUsage</code></li> </ul> TSL-bezogene Prüfungen im <code>TUC_PKI_018</code> werden in diesem Fall nicht durchgeführt. Ebenso erfolgt keine OCSP-Prüfung. </li> <li>Wenn das zum X.509-Zertifikat gehörende CA-Zertifikat nicht in <code>CERT_IMPORTED_CA_LIST</code> enthalten ist, werden, abhängig vom Parameter <i>qualifiedCheck</i> folgende TUCs unter Weitergabe aller Eingangsparameter sowie der Negation des Werts von <code>MGM_LU_ONLINE</code> als Parameter „Offline-Modus“ aufgerufen: <ol style="list-style-type: none"> <li>Für <code>qualifiedCheck = not_required</code>: <code>TUC_PKI_018 „Zertifikatsprüfung in der TI“</code><br/>Ist der Eingangsparameter <code>ocspResponses</code> mit einer OCSP-Antwort gefüllt, so wird dieser übergeben. Die aktuell aus der OCSP-Abfrage resultierte OCSP-Antwort, falls vorhanden, wird an den Aufrufer weitergegeben.</li> <li>Für <code>qualifiedCheck = required</code>: <code>TUC_PKI_030 „QES-Zertifikatsprüfung“</code><br/>Dabei wird das Basiszertifikat übergeben. Ist Eingangsparameter <code>ocspResponses</code> mit einer OCSP-Response gefüllt, so wird dieser übergeben. Die aktuell aus der OCSP-Abfrage resultierende OCSP-Response, falls vorhanden, wird an den Aufrufer weitergegeben.</li> <li>Für <code>qualifiedCheck = if_QC_present</code>: Ist im jeweiligen Signaturzertifikat mindestens ein <code>QCStatement</code> mit dem OID <code>id-etsi-qcs-QcCompliance (0.4.0.1862.1.1)</code> enthalten, handelt es sich um eine QES-Zertifikatsprüfung mittels <code>TUC_PKI_030 „QES-Zertifikatsprüfung“</code>, sonst um eine nonQES-Zertifikatsprüfung mittels <code>TUC_PKI_018 „Zertifikatsprüfung“</code>.</li> </ol> </li> </ol> <p>Als Timeout wird beim Aufruf von <code>TUC_PKI_018</code> der Wert von <code>CERT_OCSP_TIMEOUT_NONQES</code> bzw. beim Aufruf von <code>TUC_PKI_030</code> der Wert von <code>CERT_OCSP_TIMEOUT_QES</code> übergeben (siehe auch Eingangsdaten von diesen TUCs in <code>[gemSpec_PKI]</code>).</p> |
|----------------|--|



|  |   |
|--|---|
|  | <p>Für die QES-Zertifikatsprüfung wird das zu prüfende QES-Zertifikat an TUC_PKI_030 „QES-Zertifikatsprüfung“ übergeben.</p> <p>Wird im Aufruf der Eingangsparameter <code>getOCSPResponses = false</code> mit übergeben, wird keine OCSP-Response an den Aufrufer zurückgegeben.</p> <p>Als <code>TOLERATE_OCSP_FAILURE</code> wird beim Aufruf von TUC_PKI_018 <code>offlineAllowNoCheck</code> verwendet.</p> <p>Wenn der Eingangsparameter <code>validationMode</code> („Prüfmodus“) den Wert <code>NONE</code> hat, werden die TUC_PKI_018-Eingangsparameter</p> <ul style="list-style-type: none"> <li>• „Offline-Modus“ unabhängig von <code>MGM_LU_ONLINE</code> auf „ja“ gesetzt und</li> <li>• „Prüfmodus“ auf „OCSP“.</li> </ul> <p>2. Falls <code>EECertificateContainedInTSL=true</code></p> <ol style="list-style-type: none"> <li>Prüfe, ob das in <code>certificate</code> übergebene X.509-Zertifikat in der TSL innerhalb eines "TSPService"-Eintrags mit dem <code>ServiceTypeIdentifier</code> <code>"http://uri.etsi.org/TrstSvc/Svctype/unspecified"</code> aufgeführt ist.</li> <li>Prüfe zeitliche Gültigkeit von <code>certificate</code> zum Prüfzeitpunkt aktuelle Systemzeit durch Aufruf von TUC_PKI_002.</li> <li>Ermittle <code>role</code> von <code>certificate</code> durch Aufruf von TUC_PKI_009.</li> </ol> <p>3. Der Status der Prüfung und die ermittelten Ausgangsdaten werden zurückgegeben.</p> |
|--|---|

|                                   |   |
|-----------------------------------|---|
| Varianten/<br>Alternativen        |   |
| Fehlerfälle                       | TUC_KON_037 im kritischen Betriebszustand<br>EC_TSL_Out_Of_Date_Beyond_Grace_Period aufgerufen:<br>Fehlercode 4002.<br>-> 2a) certificate ist nicht in der TSL enthalten  |
| Nichtfunktionale<br>Anforderungen | Der Konnektor MUSS unter Einhaltung aller anderen<br>Anforderungen an die Zertifikatsprüfung die Anzahl der OCSP-<br>Abfragen minimieren. Dies MUSS durch Caching (unter<br>Berücksichtigung der Grace Period) und DARF NICHT durch<br>Bündelung von OCSP-Anfragen geschehen. |
| Zugehörige<br>Diagramme           | keine   |

4878  
4879

4880 **Tabelle 265: TAB\_KON\_601 Fehlercodes TUC\_KON\_037 „Zertifikat prüfen“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases treten folgende Fehlercodes auf. |           |          |   |
| 4002  | Security  | Fatal    | Der Konnektor befindet sich in einem kritischen Betriebszustand |
| 4260  | Security  | Error    | Zertifikat nicht vorhanden in TSL                               |

4881  
4882  
4883

#### 4884 4.1.9.4.2 TUC\_KON\_042 „CV-Zertifikat prüfen“

#### 4885 **TIP1-A\_5482 - TUC\_KON\_042 „CV-Zertifikat prüfen“**

4886 Der Konnektor MUSS den technischen Use Case „CV-Zertifikat prüfen“ gemäß  
4887 TUC\_KON\_042 „CV-Zertifikat prüfen“ umsetzen.  
4888 [ $\leq$ ]

4889

4890 **Tabelle 266: TAB\_KON\_818 TUC\_KON\_042 „CV-Zertifikat prüfen“**

| Element      | Beschreibung   |
|--------------|--|
| Name         | TUC_KON_042 „CV-Zertifikat prüfen“   |
| Beschreibung | Die Gültigkeit eines (EndEntity-)CV-Zertifikats wird geprüft.<br>Es werden folgende Prüfungen durchgeführt:<br>Kryptographische Prüfung der Signaturen des End-Entity-<br>CV-Zertifikats und des CVC-CA-Zertifikats <ul style="list-style-type: none"> <li>Zeitliche Gültigkeit nach dem Schalenmodell (nur CV-Zertifikate der Generation 2).</li> </ul> |
| Auslöser     | <ul style="list-style-type: none"> <li>Aufruf in einem Fachmodul oder</li> </ul>   |

|                |   |
|----------------|---|
|                | <ul style="list-style-type: none"> <li>Technischen Use Case</li> </ul>  |
| Vorbedingungen | <ul style="list-style-type: none"> <li>keine</li> </ul>   |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>eeCertificate<br/>(zu prüfendes kartenindividuelles CV-Zertifikat)</li> <li>caCertificate<br/>(das CVC-CA-Zertifikat mit dem öffentlichen Schlüssel der zugehörigen ausstellenden CA)</li> </ul>   |
| Komponenten    | Konnektor   |
| Ausgangsdaten  | <ul style="list-style-type: none"> <li>status [Boolean]<br/>(Ergebnis der Prüfung;<br/>true: CV-Zertifikat ist gültig<br/>false: CV-Zertifikat ist ungültig)</li> </ul>   |
| Standardablauf | <p>1. Abhängig von der Zertifikats-Generation wird Vorgehen A oder B gewählt.</p> <p>A. Prüfung von CV-Zertifikaten der Generation 1:<br/>Die CVC-Prüfung setzt sich gemäß GS-A_4668 [gemSpec_PKI#8.7] aus folgenden Schritten zusammen.</p> <p>i. Prüfe die Signatur des CA-Zertifikats caCertificate mit dem öffentlichen Root-Schlüssel der ausstellenden CVC-Root-CA. Der benötigte Root-Key befindet sich auf der gSMC-K in der Datei EF.PuK.RCA.CS.R2048.</p> <p>ii. Prüfe die Signatur des (EndEntity-)CV-Zertifikats eeCertificate mit dem öffentlichen Schlüssel der ausstellenden CVC-CA (aus dem CVC-CA-Zertifikat extrahiert).</p> <p>B. Prüfung von CV-Zertifikaten der Generation 2:<br/>Die CVC-Prüfung setzt sich gemäß GS-A_5009, ... GS-A_5012 [gemSpec_PKI#8.8] aus folgenden Schritten zusammen:</p> <p>i. Prüfe die kryptographische Korrektheit der Signatur des CA-Zertifikats caCertificate mit dem öffentlichen Root-Schlüssel der ausstellenden CVC-Root-CA. Der benötigte Root-Key befindet sich im Truststore des Konnektors.</p> <p>ii. Prüfe die kryptographische Korrektheit der Signatur des (EndEntity-)CV-Zertifikats certificate mit dem öffentlichen Schlüssel der ausstellenden CVC-CA (aus dem CVC-CA-Zertifikat extrahiert).</p> <p>iii. Prüfe die zeitliche Gültigkeit des (EndEntity-)CV-Zertifikates,</p> |

|                                |   |
|--------------------------------|---|
|                                | des CVC-CA-Zertifikates und CVC-Root-CA-Zertifikates<br>nach dem Schalenmodell.<br>2. Der Status <i>status</i> der Prüfung wird zurückgegeben.  |
| Varianten/Alternativen         | (→ B.i) Mathematische Korrektheitsprüfung CV-Zertifikate mit Cross-CV-Zertifikat (vgl. Varianten/Alternativen von TUC_KON_005)  |
| Fehlerfälle                    | (→ A.i) kryptographische (mathematische) Prüfung des CVC-CA-Zertifikats fehlgeschlagen, Fehlercode 4196.<br>(→ A.ii) kryptographische (mathematische) Prüfung des (EndEntity-) CV-Zertifikats fehlgeschlagen, Fehlercode 4196.<br>(→ B.i) das benötigte Cross-CV-Zertifikat ist nicht vorhanden, Fehlercode 4228<br>(→ B.i) kryptographische (mathematische) Prüfung des CVC-CA-Zertifikats fehlgeschlagen, Fehlercode 4196.<br>(→ B.ii) kryptographische Prüfung des (EndEntity-)CV-Zertifikats fehlgeschlagen, Fehlercode 4196.<br>(→ B.iii) zeitliche Gültigkeit eines der CV-Zertifikate ist abgelaufen, Fehlercode 4196. |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | keine   |

4891

4892

**Tabelle 267: TAB\_KON\_819 Fehlercodes TUC\_KON\_042 „CV-Zertifikat prüfen“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4196  | Technical | Error    | Fehler bei der CV-Zertifikatsprüfung                  |
| 4228  | Technical | Error    | das benötigte Cross-CV-Zertifikat ist nicht vorhanden |

4893

**4.1.9.4.3 TUC\_KON\_034 „Zertifikatsinformationen extrahieren“**

4894

**TIP1-A\_4697 - TUC\_KON\_034 „Zertifikatsinformationen extrahieren“**

4895

Der Konnektor MUSS den technischen Use Case TUC\_KON\_034 „Zertifikatsinformationen extrahieren“ umsetzen.

4896

4897

4898

**Tabelle 268: TAB\_KON\_770 TUC\_KON\_034 „Zertifikatsinformationen extrahieren“**

| Element | Beschreibung                                       |
|---------|--|
| Name    | TUC_KON_034 „Zertifikatsinformationen extrahieren“ |

|                 |  |
|-----------------|--|
| Beschreibung    | Dieser TUC beschreibt die Extraktion der fachlich zentralen Informationen aus bestimmten Zertifikaten einer gesteckten Karte eines Mandanten.  |
| Auslöser        | <ul style="list-style-type: none"> <li>Aufruf durch ein Fachmodul oder eine Basisanwendung des Konnektors</li> </ul>   |
| Vorbedingungen  | Keine  |
| Eingangsdaten   | <ul style="list-style-type: none"> <li>cardSession – <i>optional/verpflichtend für den Zugriff auf eGK, HBA, SM-B oder gSMC-KT</i></li> <li>checkSMCK [Boolean] – <i>optional/verpflichtend für gSMC-K;</i><br/>(Referenz auf eine/die gSMC-K, alternativ zu cardSession)</li> <li>qes [Boolean] - <i>optional; default: false</i> –<br/>(Angabe, ob die QES-Identität oder die nonQES-Identität der Karte interessiert)</li> <li>crypt - <i>optional; default = RSA</i><br/>(kryptographischer Algorithmus, für welchen das Zertifikat ermittelt wird;<br/>Wertebereich: ECC, RSA)</li> </ul> |
| Komponenten     | Konnektor  |
| Ausgangsdaten   | <ul style="list-style-type: none"> <li>certType [C.CH.AUT   C.HP.AUT   C.HCI.AUT   C.HP.QES]<br/>(Zertifikatstyp)</li> <li>certInfo<br/>(Zertifikatsinformationen, bestehend aus<br/>SerialNumber, Issuer, Subject, Rollen,<br/>registrationNumber und ggf. id-etsi-qcs-<br/>QcCompliance, siehe Standardablauf)</li> <li>qcStatements – <i>optional/nur wenn certType = C.HP.QES</i><br/>(QCStatements)</li> </ul>  |
| Nachbedingungen | Keine  |
| Standardablauf  | <ol style="list-style-type: none"> <li>Je nach Kartentyp wird aus der Karte das passende Zertifikat über TUC_KON_216 "LeseZertifikat" {selektiertes Zertifikat} ausgelesen.<br/>Das Zertifikatsobjekt (fileIdentifier und folder)/Zertifikatsbezeichnung wird für die jeweilige Karte unter Berücksichtigung des kryptographischen Verfahrens crypt gemäß TAB_KON_858 bzw. TAB_KON_856 ermittelt. <ol style="list-style-type: none"> <li>Bei qes = false: <ol style="list-style-type: none"> <li>Für die eGK: C.CH.AUT</li> </ol> </li> </ol> </li> </ol>                                      |

|                        |   |
|------------------------|---|
|                        | <ul style="list-style-type: none"> <li>ii. Für den HBAX: C.HP.AUT</li> <li>iii. Für SM-B: C.HCI.AUT</li> <li>iv. Für gSMC-K: C.NK.VPN</li> </ul> <p>b. Bei qes = true:</p> <ul style="list-style-type: none"> <li>i. Für den HBAX: C.HP.QES</li> </ul> <p>2. Die Zertifikatsbezeichnung aus Schritt 1 („C.XXX.YYY.ZZZZ“) wird als Ausgangsdatum „certType“ zurückgegeben.</p> <p>3. Zusätzlich werden aus dem Zertifikat folgende Informationen extrahiert und zurückgegeben:</p> <ul style="list-style-type: none"> <li>a. X509SerialNumber</li> <li>b. Issuer (DistinguishedName) nach RFC 2253</li> <li>c. Subject (DistinguishedName) nach RFC 2253</li> <li>d. Aus der Extension Admission: <ul style="list-style-type: none"> <li>i. eine Liste von Rollen durch Aufruf von TUC_PKI_009 „Rollenermittlung“</li> <li>ii. registrationNumber (=Telematik-ID; falls vorhanden)</li> </ul> </li> <li>e. id-etsi-qcs-QcCompliance (0.4.0.1862.1.1) in QCStatements, falls vorhanden</li> <li>f. Restriction, falls vorhanden</li> <li>g. validity</li> </ul> |
| Varianten/Alternativen | Keine   |
| Fehlerfälle            | <p>(→1) Wenn im Aufrufkontext (also erreichbar durch den Mandanten) zum angegebenen CardHandle keine Karte gefunden werden kann, bricht der TUC mit Fehlercode 4146 ab.</p> <p>(→1b) Ist bei Angabe von QES=true auf der Karte keine QES-Identität zu finden, bricht der TUC mit Fehlercode 4147 ab. Für die Kombination QES=true mit einer eGK bricht der TUC mit Fehlercode 4148 ab (QES-Zertifikate der eGK werden noch nicht unterstützt).</p> <p>(→1) Für eGK, HBA, SM-B gilt: Wenn crypt=ECC und Karte vom Typ &lt;G2.1, bricht der TUC mit Warnung 4257 ab.</p> <p>(→1) Für gSMC-K gilt: Wenn crypt=ECC und TUC_KON_216 Warnung 4256 liefert, bricht der TUC mit Warnung 4257 ab.</p> <p>(→1) Wenn aus anderen Gründen die Extraktion der Zertifikatsinformationen fehlschlägt, bricht der TUC mit Fehlercode 4148 ab.</p>   |

|                                |       |
|--------------------------------|-------|
| Nichtfunktionale Anforderungen | keine |
| Zugehörige Diagramme           | keine |

**Tabelle 269: TAB\_KON\_602 Fehlercodes TUC\_KON\_034 „Zertifikatsinformationen extrahieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4146  | Technical | Error    | Kartenhandle existiert nicht                                   |
| 4147  | Technical | Error    | Zertifikat nicht vorhanden (z. B. kein QES-Zertifikat in SM-B) |
| 4148  | Technical | Error    | Fehler beim Extrahieren von Zertifikatsinformationen           |
| 4257  | Technical | Warning  | ECC-Zertifikate nicht vorhanden auf Karte: <cardHandle>        |

[<=]

#### 4.1.9.5 Operationen an der Außenschnittstelle

##### **TIP1-A\_4698-03** ~~TIP1-A\_4698-02~~ - Basisanwendung Zertifikatsdienst

Der Konnektor MUSS Clientsystemen eine Basisanwendung Zertifikatsdienst zur Verfügung stellen

**Tabelle 270: TAB\_KON\_771 Basisanwendung Zertifikatsdienst**

|                          |  |                                       |
|--------------------------|--|---------------------------------------|
| <b>Name</b>              | CertificateService   |                                       |
| <b>Version (KDV)</b>     | 6.0.0 (WSDL-Version), 6.0.1 (XSD-Version)<br>6.0.1 (WSDL-Version), 6.0.2 (XSD-Version) |                                       |
| <b>Namensraum</b>        | Siehe <a href="#">Anhang D GitHub</a>  |                                       |
| <b>Namensraum-Kürzel</b> | CERT für Schema und CERTW für WSDL   |                                       |
| <b>Operationen</b>       | Name   | Kurzbeschreibung                      |
|                          | ReadCardCertificate  | Zertifikat von einer Karte lesen      |
|                          | CheckCertificateExpiration   | Ablaufdatum von Zertifikaten erfragen |

|               |  |                                      |
|---------------|--|--------------------------------------|
|               | VerifyCertificate  | Prüfung des Status eines Zertifikats |
| <b>WSDL</b>   | CertificateService.wsdl (WSDL-Version 6.0.0)<br>CertificateService_v6_0_1.wsdl |                                      |
| <b>Schema</b> | CertificateService.xsd (XSD-Version 6.0.1)<br>CertificateService_v6_0_2.xsd    |                                      |

4909

4910 [**<=**]

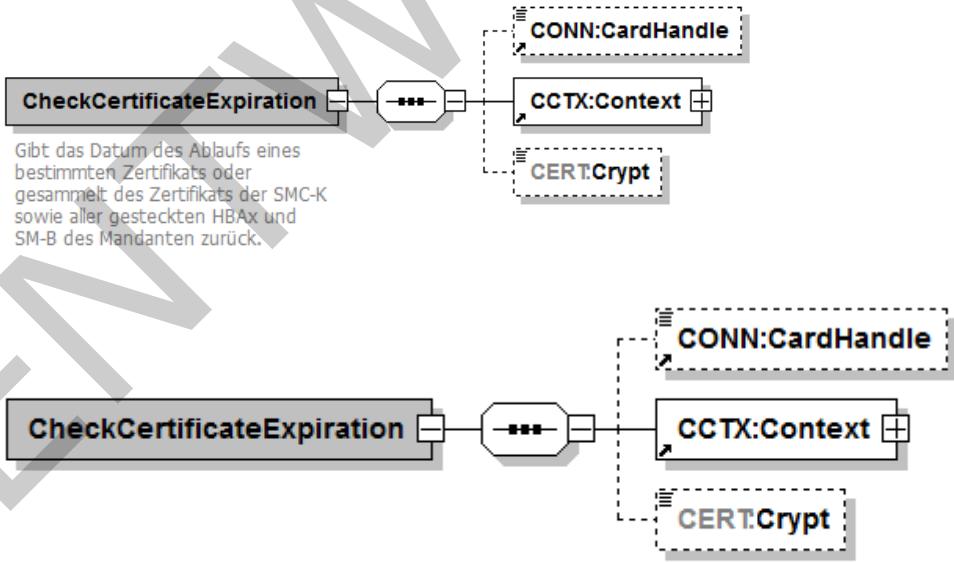
4911 4.1.9.5.1 CheckCertificateExpiration

4912 **TIP1-A\_4699-02TIP1-A\_4699 - Operation CheckCertificateExpiration**

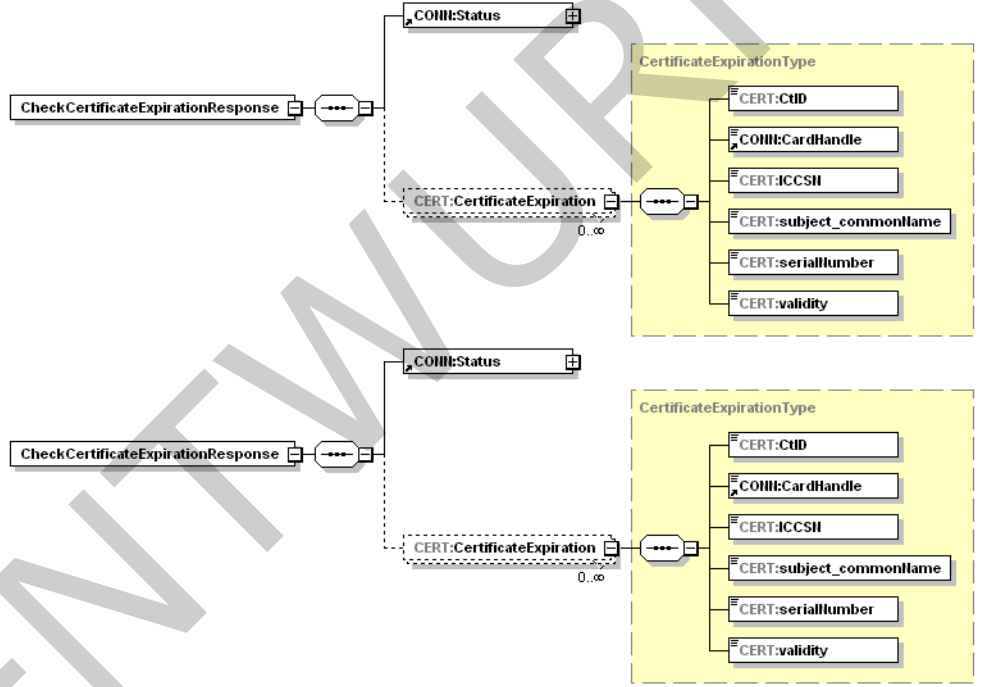
4913 Die Basisanwendung Zertifikatsdienst des Konnektors MUSS an der Clientschnittstelle  
 4914 eine Operation CheckCertificateExpiration anbieten.

4915

4916 **Tabelle 271: TAB\_KON\_676 Operation CheckCertificateExpiration**

|                         |   |   |
|-------------------------|---|---|
| <b>Name</b>             | CheckCertificateExpiration  |   |
| <b>Beschreibung</b>     | Gibt das Datum des Ablaufs eines bestimmten Zertifikats oder gesammelt des Zertifikats der gSMC-K, der gSMC-KT's sowie aller gesteckten HBAX und SM-B des Mandanten zurück.   |   |
| <b>Aufruf-parameter</b> |  <p>Gibt das Datum des Ablaufs eines bestimmten Zertifikats oder gesammelt des Zertifikats der SMC-K sowie aller gesteckten HBAX und SM-B des Mandanten zurück.</p> |   |
|                         | <b>Name</b>   | <b>Beschreibung</b>   |
|                         | CardHandle  | Optional. Identifiziert die Karte, deren Zertifikate geprüft werden sollen. Wird der Parameter nicht angegeben, so werden alle für den Konnektor erreichbaren Karten (inkl. gSMC-K und aller gSMC-KT's), die zum Mandanten passen, berücksichtigt.<br>Die Operation CheckCertificateExpiration DARF |



|                        |   |   |
|------------------------|---|---|
|                        |   | das Lesen von Zertifikaten der eGK NICHT unterstützen.  |
|                        | Context   | MandantId, CsId, WorkplaceId verpflichtend;<br>UserId optional  |
|                        | Crypt   | Optional; Default: RSA<br>Gibt den kryptographischen Algorithmus vor, für den das Zertifikat ermittelt werden soll.<br>Wertebereich: RSA, ECC <ul style="list-style-type: none"> <li>• RSA: Zertifikat für RSA-2048</li> <li>• ECC: Zertifikat für ECC-256</li> </ul> |
| <b>Rückgabe</b>        |  |   |
|                        | Status  | Enthält den Ausführungsstatus der Operation.  |
|                        | CertificateExpiration   | Eine Liste von Tupeln aus (CtID, CardHandle, ICCSN, subject.CommonName, serialNumber, validity) der Zertifikate der Karten.<br>Für die gSMC-K soll in CertificateExpiration/CtID und CertificateExpiration/CardHandle jeweils ein Leerstring zurückgegeben werden.    |
|                        |   |   |
| <b>Vorbedingungen</b>  | Keine   |   |
| <b>Nachbedingungen</b> | Keine   |   |

4917 Der Ablauf der Operation CheckCertificateExpiration ist in Tabelle TAB\_KON\_677  
 4918 beschrieben:  
 4919

4920 **Tabelle 272: TAB\_KON\_677 Ablauf CheckCertificateExpiration**

| Nr.  | Aufruf Technischer Use Case oder Interne Operation | Beschreibung   |
|--|--|--|
| 1.   | checkArguments                                     | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab. Wird die Operation für einen nicht unterstützten Kartentypen aufgerufen, so bricht die Operation mit Fehler 4058 ab.  |
| 2.   | TUC_KON_000 „Prüfe Zugriffsberechtigung“           | Die Prüfung erfolgt durch den Aufruf<br><pre>TUC_KON_000 {   mandantId = \$context.mandantId;   clientSystemId = \$context.clientsystemId;   workplaceId = \$context.workplaceId;   userId = \$context.userId;   cardHandle = \$cardHandle }</pre> Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab. |
| 3.   | enumerateCardHandles                               | Wenn der Parameter CardHandle übergeben wurde, wird dieser als einziges Element in eine Liste gepackt.<br>Wenn der Parameter CardHandle leer war, wird eine Liste der CardHandles aller für den Konnektor erreichbaren Karten (inkl. gSMC-K und gSMC-KT's), die zum Mandanten passen, erstellt.  |
| Für jedes CardHandle der in Schritt 3 erzeugten Liste werden folgende Schritte ausgeführt, für die gSMC-Ks die Schritte 5 und 6:<br>Falls Schritt 5 der TUC_KON_033 die Warnung 4257 zurückgibt, wird Schritt 6 nicht ausgeführt und die Schritte für das CardHandle der in Schritt 3 erzeugten Liste weiter ausgeführt. Die Warnung 4257 wird über alle CardHandle akkumuliert und <komma-separierte List von cardHandle> für den Fehlertext erzeugt. |  |  |
| 4.   | TUC_KON_026 „Liefere CardSession“                  | Ermittle CardSession über TUC_KON_026 {<br><pre>mandatId = MandantId; clientSystemId = ClientSystemId; cardHandle = CardHandle; userId = UserId }</pre>  |
| 5.   | TUC_KON_033 „Zertifikatsablauf“                    | Das Gültigkeitsdatum des Zertifikats wird geprüft mit TUC_KON_033 {<br><pre>cardSession;</pre>   |

|    |  |  |
|----|--|--|
|    | prüfen"  | <pre>doInformClients = false; Crypt; } bzw. TUC_KON_033 {   checkSMCK = true;   doInformClients = false;   Crypt; }</pre>  |
| 6. | TUC_KON_034<br>„Zertifikatsinformationen<br>extrahieren" | <p>Beim Aufruf des TUC_KON_034 ist der Parameter qes = false zu setzen.</p> <p>Aus den jeweiligen Rückgabewerten entsteht eine Liste aus Tupeln (CtID, CardHandle, ICCSN, subject.CommonName, serialNumber, validity). Diese wird von der Operation zurückgegeben.</p> |

4921 **Tabelle 273: TAB\_KON\_603 Fehlercodes „CheckCertificateExpiration“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4000  | Technical | Error    | Syntaxfehler   |
| 4058  | Security  | Error    | Aufruf nicht zulässig  |
| 4257  | Technical | Warning  | ECC-Zertifikate nicht vorhanden auf Karte:<br><komma-separierte List von cardHandle> |

4922 [**<=**]

4925 **4.1.9.5.2 ReadCardCertificate**


4926 **TIP1-A\_4700 - Operation ReadCardCertificate**

4927 Die Basisanwendung Zertifikatsdienst des Konnektors MUSS an der Clientschnittstelle  
 4928 eine Operation ReadCardCertificate wie in Tabelle TAB\_KON\_678 Operation  
 4929 ReadCardCertificate beschrieben anbieten.

4931 **Tabelle 274: TAB\_KON\_678 Operation ReadCardCertificate**

|                     |  |
|---------------------|--|
| <b>Name</b>         | ReadCardCertificate                      |
| <b>Beschreibung</b> | Liest X.509-Zertifikate von einer Karte. |

|                              |  |
|------------------------------|--|
| <b>Aufruf-<br/>parameter</b> |  <pre> sequenceDiagram     participant Conn as CONN:CardHandle     participant CCTX as CCTX:Context     participant CertRefList as CERT:CertRefList     participant CertRef as CERT:CertRef     participant Crypt as CERT:Crypt      Conn-&gt;&gt;CCTX     CCTX-&gt;&gt;CertRefList     CertRefList-&gt;&gt;CertRef     CertRef-&gt;&gt;Crypt     Crypt--&gt;&gt;CertRefList     CertRefList--&gt;&gt;CCTX     CCTX--&gt;&gt;Conn     </pre> <p>ReadCardCertificate<br/>Liest ein X.509-Zertifikat von einer Karte</p>   |
| <b>Name</b>                  | <b>Beschreibung</b>  |
| CardHandle                   | <p>Gibt die Karte an, von der das Zertifikat gelesen werden soll.</p> <p>Es können Zertifikate von HBAX (HBA, HBA-VK), SM-B ausgelesen werden.</p> <p>Die Operation ReadCardCertificate DARF das Lesen von Zertifikaten der eGK NICHT unterstützen.</p>  |
| Context                      | Aufrufkontext (Mandant)  |
| CertRefList                  | <p>Gibt an, welche(s) Zertifikat(e) gelesen werden soll.</p> <p>Mögliche Werte für CertRef sind:</p> <p>C.AUT, C.ENC, C.SIG, C.QES</p>   |
| Crypt                        | <p>Optional; Default: RSA</p> <p>Gibt den kryptographischen Algorithmus vor, für den das Zertifikat ermittelt werden soll.</p> <p>Wertebereich: RSA, ECC</p> <ul style="list-style-type: none"> <li>• RSA: Zertifikat für RSA-2048</li> <li>• ECC: Zertifikat für ECC-256</li> </ul>   |
| <b>Rückgabe</b>              |  <pre> sequenceDiagram     participant Conn as CONN:Status     participant ReadCardCertificateResponse as ReadCardCertificateResponse     participant CertCMN as CERTCMN:X509DataInfoList     participant CertCMNData as CERTCMN:X509DataInfo     participant CertCMNRef as CERTCMN:CertRef     participant CertCMNData2 as CERTCMN:X509Data      Conn-&gt;&gt;ReadCardCertificateResponse     ReadCardCertificateResponse-&gt;&gt;CertCMN     CertCMN-&gt;&gt;CertCMNData     CertCMNData-&gt;&gt;CertCMNRef     CertCMNRef-&gt;&gt;CertCMNData2     CertCMNData2--&gt;&gt;CertCMNData     CertCMNData--&gt;&gt;CertCMN     CertCMN--&gt;&gt;Conn     </pre> <p>ReadCardCertificateResponse<br/>Antwort der Operation ReadCardCertificate. Enthält das gelesene Zertifikat.</p> |

|  |   |   |
|--|---|---|
|  |   |   |
| Status   | Enthält den Ausführungsstatus der Operation.  |   |
| CertRef  | Dieses Element beinhaltet die Referenz des Zertifikats, welches bei der Anfrage übergeben wurde.  |   |
| X509Data   | Inhalt des über die CertRef referenzierten Zertifikats. Ist das referenzierte Zertifikat nicht vorhanden, so wird dieses Element nicht vom Konnektor gefüllt. |   |
|  | X509Issuer<br>Name  | Enthält den Issuer-Name des Zertifikats. Bezüglich des Encodings sind die in [XMLDSig#4.4.4.4.1] angegebenen Regeln zu beachten (Escaping von Sonderzeichen etc.) |
|  | X509Serial<br>Number  | Enthält die serialNumber des Zertifikats.   |
|  | X509Subject<br>Name   | Enthält das Feld subject.CommonName. Bezüglich des Encodings sind die in [XMLDSig#4.4.4.4.1] angegebenen Regeln zu beachten (Escaping von Sonderzeichen etc.)     |
|  | X509<br>Certificate   | Enthält das base64-codierte Zertifikat, dessen Binärstruktur wiederum ASN.1-codiert (gemäß [COMMON_PKI]) vorliegt.  |
| <b>Vorbe-<br/>dingungen</b>  | Keine   |   |

|                        |       |
|------------------------|-------|
| <b>Nachbedingungen</b> | Keine |
|------------------------|-------|

4932 Der Ablauf der Operation ReadCardCertificate ist in Tabelle TAB\_KON\_679 Ablauf  
 4933 ReadCardCertificate beschrieben:  
 4934

4935 **Tabelle 275: TAB\_KON\_679 Ablauf ReadCardCertificate**

| Nr. | Aufruf<br>Technischer Use<br>Case oder Interne<br>Operation | Beschreibung   |
|-----|---|--|
| 1.  | checkArguments  | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab. Wurde als Zielkarte eine eGK adressiert, wird Fehlercode 4090 zurückgeliefert.  |
| 2.  | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“            | Die Prüfung erfolgt durch den Aufruf<br><pre>TUC_KON_000 {   mandantId = \$context.mandantId;   clientsystemId = \$context.clientsystemId;   workplaceId = \$context.workplaceId;   userId = \$context.userId;   cardHandle = \$cardHandle }</pre> Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab. |
| 3.  | TUC_KON_026<br>„Liefere<br>CardSession“                     | Ermittle CardSession über TUC_KON_026 {<br><pre>mandatId = \$context.mandantId; clientsystemId = \$context.clientsystemId; cardHandle = \$context.cardHandle; userId = \$context.userId }</pre>  |
| 4.  | getEF   | Für jedes Paar von CertRef und CardHandle wird in Abhängigkeit des Parameters Crypt gemäß Tabelle TAB_KON_858 das zu lesende File (EF) bestimmt: Ist die übergebene Zertifikatsreferenz ungültig, wird Fehlercode 4149 zurückgegeben. Das Lesen von Zertifikaten der eGK ist aus Sicherheitsgründen für Clientsysteme nicht zulässig.            |
|     | TUC_KON_216<br>„LeseZertifikat“                             | Für jedes Paar von CardHandle und EF wird nun durch Aufruf von TUC_KON_216 „LeseZertifikat“ das Zertifikat ausgelesen. Falls TUC_KON_216 die Warnung 4256 zurückgibt, wird die Operation abgebrochen und Fehler 4258   |

|    |                                  |  |
|----|----------------------------------|--|
|    |                                  | zurückgegeben.   |
| 6. | Zertifikatsattribute extrahieren | Aus jedem Zertifikat werden die zu liefernden Attribute extrahiert. Die Ergebnisstruktur wird mit den erhaltenen Rückgabewerten gefüllt. |

4936

4937

4938

**Tabelle 276: TAB\_KON\_604 Fehlercodes „ReadCardCertificate“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4000  | Technical | Error    | Syntaxfehler   |
| 4149  | Technical | Error    | Ungültige Zertifikatsreferenz                              |
| 4090  | Security  | Error    | Zugriff auf eGK nicht gestattet                            |
| 4258  | Technical | Error    | ECC-Zertifikate nicht vorhanden auf Karte:<br><cardHandle> |

4939

4940

[&lt;=]

4941

**4.1.9.5.3 VerifyCertificate**

4942

**TIP1-A\_5449 - Operation VerifyCertificate**

4943

Die Basisanwendung Zertifikatsdienst des Konnektors MUSS an der Clientschnittstelle eine Operation VerifyCertificate wie in Tabelle TAB\_KON\_795 Operation VerifyCertificate beschrieben anbieten.

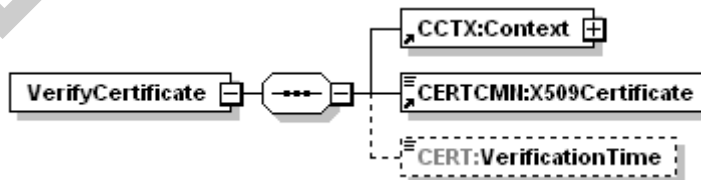
4944

4945

4946

4947

**Tabelle 277: TAB\_KON\_795 Operation VerifyCertificate**

|                        |  |  |
|------------------------|--|--|
| <b>Name</b>            | VerifyCertificate  |  |
| <b>Beschreibung</b>    | Prüft den Status eines Zertifikats.  |  |
| <b>Aufrufparameter</b> |  |  |
|                        | <b>Name</b>  | <b>Beschreibung</b>  |
|                        | CCTX:Context   | Aufrufkontext (Mandant)  |
|                        | CERTCMN:X509Certificate  | Zu prüfendes Zertifikat (base64 kodiert), wie in Response zur Operation ReadCardCertificate enthalten. |

|                              |                                 |   |
|------------------------------|---------------------------------|---|
|                              | CERT:<br>VerificationTime       | Der für die Prüfung zu verwendende Referenzzeitpunkt. Falls der Parameter nicht angegeben ist, wird als Referenzzeitpunkt die Systemzeit verwendet.   |
| <b>Rückgabe</b>              |                                 |   |
|                              | CONN:Status                     | Enthält den Ausführungsstatus der Operation.  |
|                              | CERT:<br>Verification<br>Status | Enthält eines der drei möglichen Prüfungsergebnisse in CERT:VerificationResult <ul style="list-style-type: none"> <li>• VALID</li> <li>• INCONCLUSIVE</li> <li>• INVALID</li> </ul> sowie weiter Details zu den Zuständen „INCONCLUSIVE“ und „INVALID“ in GERROR:Error. |
|                              | CERT:RoleList                   | OIDs der im Zertifikat gespeicherten Rollen.  |
| <b>Vorbe-<br/>dingungen</b>  | Keine                           |   |
| <b>Nachbe-<br/>dingungen</b> | Keine                           |   |

4948 Der Ablauf der Operation VerifyCertificate ist in Tabelle TAB\_KON\_797 Ablauf  
4949 VerifyCertificate beschrieben:  
4950

4951 **Tabelle 278: TAB\_KON\_797 Ablauf VerifyCertificate**

| Nr. | Aufruf<br>Technischer<br>Use Case oder<br>Interne<br>Operation | Beschreibung  |
|-----|--|---|
| 1.  | checkArguments   | Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.  |
| 2.  | TUC_KON_037<br>„Zertifikat<br>prüfen“                          | Die Zertifikatsprüfung erfolgt durch Aufruf von TUC_KON_037. Als Parameter des TUC-Aufrufs gilt für Zertifikate aus CERT_IMPORTED_CA_LIST: {<br>certificate = CERTCMN:X509Certificate<br>qualifiedCheck = not_required; |



|    |  |  |
|----|--|--|
|    |  | <pre> baseTime = CERT:VerificationTime; offlineAllowNoCheck = true; policyList= keine Einschränkung; intendedKeyUsage=empty; intendedExtendedKeyUsage=empty; gracePeriod = empty; validationMode = NONE; ocspResponses (OCSP-Response/Liste von OCSP- Responses = empty } für alle anderen Zertifikate gilt: { certificacate = CERTCMN:X509Certificate qualifiedCheck =if_QC_present; baseTime = CERT:VerificationTime; offlineAllowNoCheck = true; policyList = alle zugelassenen Zertifikatstyp-OIDs; intendedKeyUsage = empty; intendedExtendedKeyUsage = empty; gracePeriod = empty; validationMode = OCSP; ocspResponses = empty}. </pre> |
| 3. |  | <p>Wenn der Prüfprozess fehlerhaft war und nicht zu einem Ergebnis im Sinne eines VerificationResult führt, wird eine FaultMessage erzeugt.</p> <p>War der Prüfprozess erfolgreich, wird eine VerifyCertificateResponse mit CONN:Status/CONN:Result=OK, dem VerificationStatus (als Ergebnis der Zertifikatsprüfung) und den ermittelten Rollen-OIDs erzeugt. Ein Prüfergebnis „INCONCLUSIVE“ bzw. „INVALID“ wird in CERT:VerificationStatus/GERROR:Error mit den zugehörigen Fehlermeldungen detailliert (in diesem Fall kann CONN:Status/CONN:Result=OK oder CONN:Status/CONN:Result=Warning gesetzt sein).</p>                              |

4952 **Tabelle 279: TAB\_KON\_800 Fehlercodes „VerifyCertificate“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |              |
| 4000  | Technical | Error    | Syntaxfehler |

4953

4954

4955 [**<=**]

#### 4956 **4.1.9.6 Betriebsaspekte**

##### 4957 **4.1.9.6.1 TUC\_KON\_035 „Zertifikatsdienst initialisieren“**

##### 4958 **TIP1-A\_4701 - TUC\_KON\_035 „Zertifikatsdienst initialisieren“**

4959 In der Bootup-Phase MUSS der Konnektor den Zertifikatsdienst durch Aufruf des  
 4960 TUC\_KON\_035 „Zertifikatsdienst initialisieren“ initialisieren.

4961

4962

**Tabelle 280: TAB\_KON\_772 TUC\_KON\_035 „Zertifikatsdienst initialisieren“**

| Element         | Beschreibung  |
|-----------------|---|
| Name            | TUC_KON_035 „Zertifikatsdienst initialisieren“  |
| Beschreibung    | Der TUC beschreibt den gesamten Ablauf der Initialisierung des TrustStore im Rahmen der betrieblichen Prozesse: Prüfung der Aktualität, Integrität und Authentizität der Einträge im TrustStore.  |
| Auslöser        | <ul style="list-style-type: none"> <li>• Bootup des Konnektors</li> </ul>   |
| Vorbedingungen  | keine   |
| Eingangsdaten   | keine   |
| Komponenten     | Konnektor   |
| Ausgangsdaten   | <ul style="list-style-type: none"> <li>• Status der Initialisierung des TrustStore</li> </ul>   |
| Nachbedingungen | Keine   |
| Standardablauf  | <p>Für den übergebenen Status der Initialisierung des TrustStore werden folgende Schritte durchgeführt:</p> <ol style="list-style-type: none"> <li>1. Durch eine DNS-Anfrage an den DNS-Forwarder zur Auflösung der SRV-RR mit dem Bezeichner "_ocsp._tcp.&lt;DOMAIN_SRVZONE_TI&gt;„ erhält der Konnektor Adressen des http-Forwarders des VPN-Zugangsdienststandortes.</li> <li>2. Falls in den letzten 24 Stunden keine Aktualisierung der TSL und CRL im Truststore stattgefunden hat, aktualisiert der Konnektor die TSL durch den Aufruf von TUC_KON_032 „TSL aktualisieren“ und die CRL durch den Aufruf von TUC_KON_040 „CRL aktualisieren“.</li> <li>3. Falls im Zeitraum von CERT_BNETZA_VL_UPDATE_INTERVAL keine Aktualisierung der BNetZA VL stattgefunden hat, aktualisiert der Konnektor die BNetZA VL durch den Aufruf von TUC_KON_031 „BNetZA-VL aktualisieren“.</li> <li>4. Der Konnektor prüft die Gültigkeitsdauer der Zertifikate aller gesteckten Karten (inkl. gSMC-K) mittels Aufruf von: <ul style="list-style-type: none"> <li>für gSMC-K<br/>TUC_KON_033{checkSMCK; doInformClients=Ja; crypt = ECC}</li> <li>TUC_KON_033{checkSMCK; doInformClients=Ja; crypt = RSA}</li> <li>für jede gesteckte G2.0 Karte<br/>TUC_KON_033{cardSession; doInformClients=Ja; crypt = RSA}</li> <li>für jede gesteckte ab G2.1 Karte<br/>TUC_KON_033{cardSession; doInformClients=Ja; crypt = ECC}</li> <li>TUC_KON_033{cardSession; doInformClients=Ja; crypt = RSA}</li> </ul> </li> </ol> |

|                                   |  |
|-----------------------------------|--|
|                                   | 5. Der Konnektor liest von der gSMC-K den öffentlichen Schlüssel des CVC-Root-Zertifikats und speichert diesen im TrustStore [gemSpec_gSMC-K_ObjSys#5.3.10]. |
| Varianten/<br>Alternativen        | Keine  |
| Fehlerfälle                       | Keine  |
| Nichtfunktionale<br>Anforderungen | Keine  |
| Zugehörige<br>Diagramme           | Keine  |

4963 **Tabelle 281: TAB\_KON\_605 Fehlercodes TUC\_KON\_035 „Zertifikatsdienst initialisieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext |
|---|-----------|----------|------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können keine weiteren Fehlercodes auftreten. |           |          |            |

4964  
4965 [ $\leq$ ]

4966 **TIP1-A\_4702-03TIP1-A\_4702-02 - Konfigurierbarkeit des Zertifikatsdienstes**

4967 Der Administrator MUSS die in TAB\_KON\_606 aufgelisteten Parameter über die  
4968 Managementschnittstelle konfigurieren und die in TAB\_KON\_733 aufgelisteten Parameter  
4969 ausschließlich einsehen können.  
4970

4971 **Tabelle 282: TAB\_KON\_606 Konfiguration des Zertifikatsdienstes**

| ReferenzID                             | Belegung | Bedeutung  |
|--|----------|--|
| CERT_TSL_DEFAULT_<br>GRACE_PERIOD_DAYS | X Tage   | Default Grace Period TSL in Tagen<br>Gibt an, wie viele Tage der Konnektor mit einer zeitlich abgelaufenen TSL weiter betrieben werden kann.<br>Der Wert MUSS zwischen 1 und 30 Tagen liegen.<br>Default-Wert = 30 Tage<br><i>Hinweis: Vor dem zeitlichen Ablauf einer TSL wird mit ausreichendem Vorlauf eine neue TSL verteilt. Sollte die TSL dennoch ablaufen und der Konfigurationswert überschritten werden, kann eine neue TSL immer noch lokal geladen werden (TIP1-A_4705 „TSL manuell importieren“).</i> |

|                                       |  |  |
|---------------------------------------|--|--|
| CERT_OCSP_DEFAULT_GRACE_PERIOD_NONQES | X<br>Minute<br>n   | Default Grace Period OCSP für nonQES in Minuten.<br>Der Wert MUSS zwischen 0 und 20 Minuten liegen.<br>Default-Wert = 10 Minuten   |
| CERT_OCSP_TIMEOUT_NONQES              | X<br>Sekund<br>en  | Timeout für OCSP-Abfragen bei der Prüfung von nonQES-Zertifikaten.<br>Der Wert MUSS zwischen 1 und 120 Sekunden liegen.<br>Default-Wert = 10 Sekunden  |
| CERT_OCSP_TIMEOUT_QES                 | X<br>Sekund<br>en  | Timeout für OCSP-Abfragen bei der Prüfung von QES-Zertifikaten.<br>Der Wert muss zwischen 1 und 120 Sekunden liegen.<br>Default-Wert = 10 Sekunden   |
| CERT_EXPIRATION_WARN_DAYS             | X<br>Tag (e)   | Warnung X Tage vor Ablauf von Zertifikaten im Managementinterface und per Ereignis.<br>Der Wert muss zwischen 0 und 180 Tagen (0=keine Warnung) liegen.<br>Default-Wert = 90 Tage                        |
| CERT_EXPIRATION_CARD_CHECK_DAYS       | X<br>Tag (e)   | Alle X Tage wird der Ablauf aller gesteckten Karten überprüft.<br>Der Wert muss zwischen 0 und 365 liegen (0=kein Check).<br>Default-Wert = 1 Tag  |
| CERT_IMPORTED_CA_LIST                 | Liste<br>von<br>manuell<br>importierten<br>CA-Zertifikaten | Der Administrator MUSS CA-Zertifikate importieren, anzeigen und löschen können.<br>Der Konnektor DARF CA-Zertifikate zur Ableitung von QES-Zertifikaten NICHT importieren.<br>Default-Wert = leere Liste |
| CERT_BNETZA_VL_UPDATE_INTERVAL        | X<br>Stunde<br>n   | Intervall, in dem die BNetZA VL auf Aktualität geprüft werden muss. Der Wert MUSS zwischen 1 Stunde und 168 Stunden (7 Tage) liegen.<br>Default-Wert = 24 Stunden  |

|  |       |  |
|--|-------|--|
| CERT_TSL_DOWNLOAD_ADDRESS_INTERNET_BU                    | 1 URI | Konfigurierbare Backup Adresse der TSL im Internet   |
| CERT_ECC_RSA_TSL_SIGNER_CA_CERTIFICATE_INTERNET_BU       | 1 URI | Konfigurierbare Backup Adresse der TSL-Signer-CA Zertifikate im Internet (gemäß gemSpec_TSL#A_17680-02 und gemSpec_PKI#(5.15.3 X.509 Zertifikatsprofil der TSL-Signer-CA)                                  |
| CERT_ECC_RSA_TSL_SIGNER_CA_CROSS_CERTIFICATE_INTERNET_BU | 1 URI | Konfigurierbare Backup Adresse der TSL-Signer-CA-Cross Zertifikate im Internet (gemäß gemSpec_TSL#A_17680-02 und gemSpec_PKI#(5.15.3 X.509 Zertifikatsprofil der TSL-Signer-CA)                            |
| CERT_TSL_IP_ADDRESS_INTERNET_BU                          | 1 URI | Konfigurierbare Backup Adresse der TSL im Internet (enthält IP-Adresse des Hosts statt FQDN). Wird verwendet, falls Auflösen der FQDN mittels DNS bei CERT_TSL_DOWNLOAD_ADDRESSES_INTERNET_BU fehlschlägt. |

4972 **Tabelle 283: TAB\_KON\_733 Einsehbare Konfigurationsparameter des Zertifikatsdienstes**

| ReferenzID                  | Belegung | Bedeutung   |
|-----------------------------|----------|---|
| CERT_CRL_DOWNLOAD_ADDRESS   | 2 URIs   | Download-Adressen für die CRL   |
| CERT_OCSP_FORWARDER_ADDRESS | 2 FQDNs  | Adressen der OCSP-Forwarder (HTTPS-Proxy) beim Zugangsdienstprovider<br>Der Administrator muss in geeigneter Weise einen Test auslösen können, ob einer der Server per ICMP-Echo (ping) erreichbar ist und ob ein (beliebiger) OCSP-Request zu einer erhaltenen OCSP-Antwort führt. |
| CERT_OCSP_FORWARDER_PORT    | TCP-Port | TCP-Port des OCSP-Forwarders (HTTPS-Proxy) beim Zugangsdienstprovider   |

|  |                                  |  |
|--|----------------------------------|--|
| CERT_TSL_DOWNLOAD_ADDRESS_ <del>TI</del> INTERNET                    | <del>2</del><br>URIs<br>1<br>URI | <del>Primäre und Backup</del> Adresse der TSL <del>in der TI</del> im Internet gemäß gemSpec_TSL #A_17680-01   |
| CERT_ECC_RSA_TSL_SIGNER_CA_CERTIFICATE_ <del>TI</del> INTERNET       | URIs                             | <del>Adressen</del> Adresse der TSL-Signer-CA Zertifikate <del>in der TI</del> im Internet (gemäß gemSpec_TSL#A_17680-0102 und gemSpec_PKI#(5.15.3 X.509 Zertifikatsprofil der TSL-Signer-CA))       |
| CERT_ECC_RSA_TSL_SIGNER_CA_CROSS_CERTIFICATE_ <del>TI</del> INTERNET | URIs                             | <del>Adressen</del> Adresse der TSL-Signer-CA-Cross Zertifikate <del>in der TI</del> im Internet (gemäß gemSpec_TSL#A_17680-0102 und gemSpec_PKI#(5.15.3 X.509 Zertifikatsprofil der TSL-Signer-CA)) |
| CERT_TSL_IP_ADDRESS_INTERNET   | 1 URI                            | Adresse der TSL im Internet gemäß gemSpec_TSL (enthält IP-Adresse des Hosts statt FQDN). Wird verwendet, falls Auflösen der FQDN mittels DNS bei CERT_TSL_DOWNLOAD_ADDRESS_INTERNET fehlschlägt.     |

[<=]

#### TIP1-A\_4703-01 - Vertrauensraumstatus anzeigen

Das Managementinterface des Konnektors MUSS einem authentisierten Administrator die Anzeige des Status des Vertrauensraums in Form folgender Daten anbieten:  
Sequenznummer der aktuellen TSL, StatusStartingTime (des TSPService (TSL-Signer-CA-Dienst) zum aktuell gültigen, aktiven TI-Vertrauensanker), NextUpdate, Gültigkeit der TSL, Typ der TSL (RSA oder ECC-RSA)) sowie optional für den Administrator einsehbar der Fingerprint des TSL-Signer-Zertifikats. [<=]

Der Typ der TSL liefert dem Administrator die Information, ob es sich um eine TSL handelt, die den TI-Vertrauensraum ausschließlich für Zertifikate mit kryptographischen Verfahren nach RSA-2048 (TSL(RSA)) oder für Zertifikate mit kryptographischen Verfahren nach RSA-2048 und ECC-256 (TSL(ECC-RSA)) bereitstellt. Die Information kann aus der Signatur der TSL ermittelt werden.

**TIP1-A\_6733 - Aktive BNetzA-VL anzeigen**

Das Managementinterface des Konnektors MUSS einem authentisierten Administrator die Anzeige des Status der BNetzA-VL in Form folgender Daten anbieten: Sequenznummer, NextUpdate, Gültigkeitsstatus und Zeitpunkt der letzten Prüfung der Aktualität durch TUC\_KON\_031.

[<=]

**TIP1-A\_4704 - Zertifikatsablauf anzeigen**

Der Administrator MUSS einen Prüflauf auf den innerhalb von CERT\_EXPIRATION\_WARN\_DAYS Tagen bevorstehenden Ablauf von Zertifikaten aller für den Konnektor erreichbaren Karten (inkl. gSMC-K) an zentraler Stelle in der Managementschnittstelle auslösen können und das Ergebnis angezeigt bekommen.

Der Konnektor MUSS die Gültigkeitsdauer der Zertifikate aller gesteckten Karten (inkl. gSMC-K) prüfen mittels Aufruf von:

für gSMC-K

TUC\_KON\_033{checkSMCK; doInformClients=Nein; crypt = ECC}

TUC\_KON\_033{checkSMCK; doInformClients=Nein; crypt = RSA}

für jede gesteckte G2.0 Karte außer gSMC-K

TUC\_KON\_033{cardSession; doInformClients=Nein; crypt = RSA}

für jede gesteckte ab G2.1 Karte außer gSMC-K

TUC\_KON\_033{cardSession; doInformClients=Nein; crypt = ECC}

TUC\_KON\_033{cardSession; doInformClients=Nein; crypt = RSA}[<=]

**A\_18931 - Anzeige Personalisierungs-Status gSMC-K-X.509-Zertifikate**

Der Konnektor MUSS dem Administrator die X.509-Zertifikate der verbauten gSMC-Ks gemäß TIP1-A\_4506 anzeigen. Aus der Anzeige MUSS der Personalisierungs-Status der X.509-Zertifikate ersichtlich sein (dual RSA- und ECC-personalisiert oder nur RSA-personalisiert).

[<=]

**TIP1-A\_4705 - TSL manuell importieren**

Der Konnektor MUSS es dem Administrator ermöglichen, eine TSL manuell von lokaler Datenquelle einzuspielen. Dabei MUSS der Konnektor TUC\_KON\_032{TSL-Datei} mit der manuell importierten TSL aufrufen.

Der Konnektor MUSS den manuellen Import einer TSL auch ermöglichen, wenn er sich im kritischen Betriebszustand EC\_TSL\_Out\_Of\_Date\_Beyond\_Grace\_Period befindet.

Der Konnektor MUSS den manuellen Import einer zeitlich abgelaufenen TSL

zulassen. [<=]

Auch im Fall des automatischen Imports der TSL muss dies im kritischen Betriebszustand EC\_TSL\_Out\_Of\_Date\_Beyond\_Grace\_Period unterstützt werden.

**A\_20536 - TSL im kritischen Betriebszustand**

Der Konnektor MUSS den automatischen Import einer TSL auch ermöglichen, wenn er sich im kritischen Betriebszustand EC\_TSL\_Out\_Of\_Date\_Beyond\_Grace\_Period befindet. [<=]

**A\_20748 - Automatischer TSL Download im kritischen Zustand**

Falls der Konnektor im kritischen

Betriebszustand EC\_TSL\_Out\_Of\_Date\_Beyond\_Grace\_Period ist, und falls *onlineMode* = ENABLED ist, MUSS der Konnektor periodisch und in der BootUp Phase die TSL

aktualisieren. [<=]

**TIP1-A\_6728 - BNetzA-VL manuell importieren**

Der Konnektor MUSS es dem Administrator ermöglichen, die BNetzA-VL manuell von lokaler Datenquelle einzuspielen. Dabei MUSS der Konnektor TUC\_KON\_031{BNetzA-VL-



5040 Datei} mit der manuell importierten BNetzA-VL-Datei aufrufen.  
5041 [ $\leq$ ]

#### 5042 **TIP1-A\_4706 - CRL manuell importieren**

5043 Der Konnektor SOLL es dem Administrator ermöglichen, eine CRL manuell von einer  
5044 lokalen Datenquelle einzuspielen. In dem Fall MUSS der Konnektor TUC\_KON\_040{CRL-  
5045 Datei} mit der manuell importierten CRL aufrufen. [ $\leq$ ]

5046 Für die ECC-Migration ist es notwendig den ECC-RSA-Vertrauensraum zu etablieren. Dies  
5047 erfolgt durch das Einspielen eines TSL-Signer-CA Cross-Zertifikats und das neue TSL-  
5048 Signer-CA-Zertifikat, wodurch der ECC-Vertrauensanker im Konnektor im sicheren  
5049 Datenspeicher gespeichert wird. Die Prüfung des Cross-Zertifikats erfolgt durch  
5050 A\_17821 - Wechsel des Vertrauensraumes mittels Cross-Zertifikaten (ECC-Migration).  
5051 Danach kann die TSL(ECC-RSA) importiert werden. Das Ergebnis ist ein etablierter TI-  
5052 Vertrauensraum für ECC und RSA.  
5053 Konnektoren müssen den ECC-Vertrauensraum automatisiert und im Rahmen des  
5054 Upgrades auf PTV4 etablieren. Manuelle Schritte durch den Administrator sind für den  
5055 Regelfall zu vermeiden und sollten nur im Fehlerfall nötig werden. Als Fallback-Lösung  
5056 muss das manuelle Verfahren dennoch unterstützt werden.

#### 5057 **A\_20469-01A\_20469 - Automatisierte Etablierung des ECC-RSA- 5058 Vertrauensraums (ECC-Migration)**

5059 In der BootUp-Phase MUSS ein Konnektor, der den RSA-Vertrauensraum (RSA)  
5060 verwendet, überprüfen, ob die TSL(ECC-RSA) und die entsprechenden TSL-Signer-CA  
5061 Cross-Zertifikate sowie TSL-Signer-CA-Zertifikate verfügbar sind und MUSS sie im  
5062 positiven Fall automatisiert herunterladen, nach erfolgreicher Prüfung verwenden und  
5063 dadurch den ECC-Vertrauensraum (ECC-RSA) etablieren.

5064 Der Konnektor MUSS hierzu die Downloadpunkte, die mit A\_17680-0102 in  
5065 [gemSpec\_TSL#6.3.1.2] definiert sind, verwenden. Dabei MUSS der Konnektor zunächst  
5066 die Downloadpunkte innerhalb der TI verwenden. Wenn der Download aus der TI  
5067 fehlschlägt, MUSS der Konnektor einen der definierten Downloadpunkte im Internet  
5068 verwenden.

5069 Falls beim Wechsel auf den ECC-RSA Vertrauensraum ein Fehler auftritt, MUSS der  
5070 Konnektor weiterhin den RSA-Vertrauensraum (RSA) verwenden.

5071  
5072 [ $\leq$ ]

#### 5073 5074 **A\_20508 - Protokollierung der Etablierung des ECC-RSA-Vertrauensraums (ECC- 5075 Migration)**

5076 Der Konnektor MUSS alle Schritte, die er zur Etablierung des ECC-RSA-Vertrauensraums  
5077 durchläuft, im Systemprotokoll des Konnektors mit dem Log-Level "Info"  
5078 protokollieren. [ $\leq$ ]

5079

5080

5081

#### 5082 **A\_17345 - TSL-Signer-CA Cross-Zertifikat manuell importieren (ECC-Migration)**

5083 Der Konnektor MUSS es dem Administrator ermöglichen, ein TSL-Signer-CA Cross-  
5084 Zertifikat und das TSL-Signer-CA-Zertifikat für den neuen TI-Vertrauensanker manuell  
5085 von lokaler Datenquelle einzuspielen. [ $\leq$ ]



#### **A\_17837-01 - Nutzung von Cross-Zertifikaten für Vertrauensraum-Wechsel nach ECC-RSA (ECC-Migration)**

Um auf Basis des bereits etablierten Vertrauensankers (RSA) in den Vertrauensraum (ECC-RSA) zu wechseln MUSS der Konnektor bei der Initialisierung des neuen Vertrauensankers (ECC-RSA) Cross-Zertifikate verwenden. Das Ergebnis ist ein neuer etablierter TI-Vertrauensanker (ECC-RSA). [ $\leq$ ]

#### **A\_17548-01 - TLS-Signer-CA Zertifikat sicher speichern (ECC-Migration)**

Der Konnektor MUSS den neuen TI-Vertrauensanker im sicheren Datenspeicher speichern. [ $\leq$ ]

#### **A\_17549-01 - TLS-Signer-CA Cross-Zertifikat im kritischen Betriebszustand (ECC-Migration)**

Der Konnektor MUSS den Import des TLS-Signer-CA Cross-Zertifikats auch ermöglichen, wenn er sich im kritischen Betriebszustand EC\_TSL\_Out\_Of\_Date\_Beyond\_Grace\_Period befindet. [ $\leq$ ]

#### **A\_17550-01 - TLS-Signer-CA Cross-Zertifikat importieren - Fehler (ECC-Migration)**

Falls der Import des TLS-Signer-CA Cross-Zertifikats nicht erfolgreich durchgeführt werden konnte, MUSS der Konnektor den Vorgang abbrechen und einen Fehler gemäß TAB\_KON\_857 dem Administrator zur Anzeige bringen und protokollieren.

#### **Tabelle 284: TAB\_KON\_857 - Fehlercodes beim Import des Cross-Zertifikats für TI-Vertrauensanker ECC**

| Fehlercode | ErrorType | Severity | Fehlertext   |
|------------|-----------|----------|--|
| 4255       | Security  | Error    | Fehler beim Import des TLS-Signer-CA Cross-Zertifikats |

[ $\leq$ ]

#### **TIP1-A\_5700 - Ereignisbasiert http-Forwarder Adressen ermitteln**

Beim Auftreten des Events NETWORK/VPN\_TI/UP MUSS der Konnektor über DNS die Adressen des http-Forwarders des VPN-Zugangsdienststandortes ermitteln (SRV-RR mit Bezeichner "\_ocsp.\_tcp.<DOMAIN\_SRVZONE\_TI>").

[ $\leq$ ]

#### **A\_21158 - Mindestanzahl von CV-Crosszertifikaten**

Der Konnektor MUSS mindestens 20 CV-Crosszertifikate verwalten und verarbeiten können. [ $\leq$ ]

### **4.1.10 Protokollierungsdienst**

Der Protokollierungsdienst protokolliert system- und sicherheitsrelevante Ereignisse, sowie Ereignisse im Kontext der Performancemessung (siehe [gemSpec\_Perf#4.1.2]), innerhalb des Konnektors. Auch Ereignisse von Fachmodulen können protokolliert werden. Im Sicherheitsprotokoll werden alle Ereignisse eingetragen, die Auswirkungen auf Sicherheitsmerkmale des Konnektors haben können (Änderungen an der Firewall-Konfiguration, Authentisierungsfehler etc.). Ereignisse im Kontext der Performancemessung innerhalb des Konnektors werden in das Konnektor-Performanceprotokoll geschrieben. Ereignisse im Kontext der Performancemessung von

5127 Fachmodulen werden in das Fachmodul-Performanceprotokoll geschrieben. Alle anderen  
 5128 Ereignisse werden in das Systemprotokoll oder die Fachmodulprotokolle geschrieben  
 5129 (grundsätzlich trifft die Entscheidung über den zu verwendenden Protokollspeicher der  
 5130 Aufrufer des Protokolldienstes).

5131 Die Protokolle werden persistiert.

5132 Hinweis:

5133 Ereignisse im Protokollierungsdienst adressieren nicht nur zu protokollierende Events im  
 5134 Sinne des Systeminformationsdienstes sondern alles, was zu einem Protokolleintrag  
 5135 führen soll (z.B. Fehler, Informationen zu Ablauf, Debug, Performance).

5136 Innerhalb des Protokollierungsdienstes werden folgende Präfixe für Bezeichner  
 5137 verwendet:

- 5138 • Events (Topic Ebene 1): „LOG“
- 5139 • Konfigurationsparameter: „LOG\_“

## 5140 4.1.10.1 Funktionsmerkmalweite Aspekte

### 5141 TIP1-A\_4708 - Protokollierungsfunktion

5142 Der Konnektor MUSS einen Protokollierungsdienst anbieten. Dabei MUSS der Konnektor  
 5143 zwischen System- und Sicherheitsprotokoll, sowie Fachmodulprotokollen unterscheiden.  
 5144 Je Fachmodul MUSS ein getrenntes Protokoll vorhanden sein.  
 5145 Die Protokolleinträge MÜSSEN durch den Konnektor lokal persistiert werden.

5146 [ $\leq$ ]

### 5147 TIP1-A\_5654 - Sicherheits-Protokollierung

5148 Der Konnektor MUSS herstellerspezifische Fehler, die Auswirkungen auf  
 5149 Sicherheitsmerkmale des Konnektors haben, in das Sicherheitsprotokoll schreiben.

5150 [ $\leq$ ]

### 5151 TIP1-A\_4709 - Integrität des Sicherheitsprotokolls

5152 Der Konnektor MUSS sicherstellen, dass Einträge in das Sicherheitsprotokoll nicht von  
 5153 außen und nicht durch den Administrator verändert und gelöscht werden können.

5154 [ $\leq$ ]

### 5155 TIP1-A\_4710 - Protokollierung personenbezogener und medizinischer Daten

5156 Der Konnektor DARF medizinische Daten NICHT in die Protokolldateien schreiben.  
 5157 Personenbezogene Daten DÜRFEN NICHT in Protokolleinträgen gespeichert werden.  
 5158 KVN, ICCSN und CardHolderName MÜSSEN als personenbezogene Daten behandelt  
 5159 werden.

5160 Die ICCSN DARF Im Fehlerfall durch Fachmodule in Protokolleinträgen gespeichert  
 5161 werden. Die ICCSN DARF NICHT im Sicherheitsprotokoll gespeichert werden.

5162 [ $\leq$ ]

### 5163 TIP1-A\_6479 - Keine Protokollierung vertraulicher Daten

5164 Der Konnektor DARF vertrauliche Daten NICHT in die Protokolldateien schreiben.

5165 [ $\leq$ ]

### 5166 TIP1-A\_4711 - Kapazität der Protokolldateien

5167 Der Konnektor MUSS über eine Speichergröße für Protokolldateien verfügen, so dass  
 5168 Einträge (protokollierte Ereignisse ab der Schwere „Warning“) über einen Zeitraum von  
 5169 bis zu einem Jahr darin vorgehalten werden können.

5170 [ $\leq$ ]

5171 Da sich die Menge an Einträgen nach der Größe der Einsatzumgebung richtet, ist die  
 5172 Speichergröße nach den in [gemSpec\_Perf#3.1.1] beschriebenen Einsatzumgebungen  
 5173 (LE-Ux, x=1,2,3,4) ausreichend zu wählen.

**TIP1-A\_4712 - Protokollierung erfolgreicher Kryptooperationen**

Wenn LOG\_SUCCESSFUL\_CRYPTOPS = Enabled MUSS der Konnektor die folgenden erfolgreich durchlaufenen Außenoperationen protokollieren:

- SignDocument,
- VerifyDocument,
- ExternalAuthenticate,
- EncryptDocument,
- DecryptDocument.

Dazu MUSS er

```
TUC_KON_256 {
  topic = „LOG/CRYPTO_OP“;
  eventType = Sec;
  severity = Info;
  parameters = („Operation=$Operationsname,
    <für alle betroffenen Schlüssel:>
      Karte=$ICCSN,
      Keyref=<Referenz auf den Schlüssel>,
      CARD_HANDLE=$CardHandle,
      CardHolderName=$CardHolderName“);
  doDisp = false}
```

aufrufen.

[<=]

**TIP1-A\_4713 - Herstellerspezifische Systemprotokollierung**

Wenn LOG\_LEVEL\_SYSLOG = Info MUSS der Konnektor herstellerspezifische Informationen über den laufenden Betrieb in das Systemprotokoll eintragen, um im Bedarfsfall das Verhalten des Konnektors analysieren zu können (Unterstützung der Fehlersuche etc.). Die Häufigkeit und der Inhalt der protokollierten Informationen sind herstellerspezifisch.

[<=]

**TIP1-A\_4714 - Art der Protokollierung**

Der Konnektor MUSS Protokolleinträge so anlegen, dass eine Analyse der Einträge unterstützt wird:

- Die Protokolleinträge MÜSSEN eine patternbasierte Filterung unterstützen. Protokollwert/-texte sowie Attribute MÜSSEN in ihren Namensstrukturen hierauf abgestimmt sein.
- „;“ MUSS als Trennzeichen zwischen Key/Value-Paaren verwendet werden.
- „=“ MUSS als Trennzeichen zwischen Key und Value in einem Key/Value-Paar verwendet werden.
- Es MUSS durchgängig dasselbe Zeitstempelformat verwendet werden, entweder „timestamp=%d{dd.MM.yyyy HH:mm:ss.SSS}“ (Beispiel „30.08.2017 13:44:12.436“) und als Wert die gesetzliche Zeit (§4 EinhZeitG) oder „timestamp=%d{dd.MM.yyyy HH:mm:ss.SSSZ}“, wobei „Z“ die Zeitzoneangabe nach RFC 822 mit („+“ / „-“) 4DIGIT bezeichnet (Beispiel „30.08.2017 13:44:12.436+0200“).

[<=]

5222 **4.1.10.2 Durch Ereignisse ausgelöste Reaktionen**

5223 Keine.

5224 **4.1.10.3 Interne TUCs, nicht durch Fachmodule nutzbar**

5225 Keine.

5226 **4.1.10.4 Interne TUCs, auch durch Fachmodule nutzbar**

5227 *4.1.10.4.1 TUC\_KON\_271 „Schreibe Protokolleintrag“*

5228 **TIP1-A\_4715 - TUC\_KON\_271 „Schreibe Protokolleintrag“**

5229 Der Konnektor MUSS den technischen Use Case TUC\_KON\_271 „Schreibe  
5230 Protokolleintrag“ umsetzen.

5231

5232 **Tabelle 285: TAB\_KON\_607 – TUC\_KON\_271 „Schreibe Protokolleintrag“**

| Element                  | Beschreibung   |
|--------------------------|--|
| Name                     | TUC_KON_271 „Schreibe Protokolleintrag“  |
| Beschreibung             | Dieser TUC schreibt einen Eintrag in ein Protokoll.  |
| Auslöser                 | Aufruf durch Basisdienst, Fachmodul oder TUC_KON_256<br>„Systemereignis absetzen“  |
| Vorbedingungen           | <p>Im Fall eines zu protokollierenden Ereignisses des Systeminformationsdienstes wird</p> <ul style="list-style-type: none"> <li>• eventType = "Op" gesetzt, wenn Event.Type gleich "Operation", "Infrastructure", "Business" oder "Other" bzw.</li> <li>• eventType = "Sec", wenn Event.Type gleich "Security". Die Schwere entspricht der Event.Severity gemäß Schema EventService.xsd.</li> </ul> <p>Im Fall eines zu protokollierenden Fehlers wird</p> <ul style="list-style-type: none"> <li>• eventType = "Op" gesetzt, wenn ErrorType gleich "Technical", "Business", "Infrastructure" oder "Other" bzw.</li> <li>• eventType = "Sec", wenn ErrorType gleich "Security". Die Schwere entspricht der Severity des Fehlers.</li> </ul> |
| Eingangs-<br>anforderung | Keine  |
| Eingangsdaten            | <ul style="list-style-type: none"> <li>• Zu protokollierendes Ereignis</li> <li>• fmName – <i>optional/verpflichtend für Aufruf durch Fachmodule; default = ""</i></li> </ul>  |

|                 |  |
|-----------------|--|
|                 | <p>(Name des aufrufenden Fachmoduls; das Ereignis wird in das entsprechende Konnektor-Protokoll geschrieben)</p> <ul style="list-style-type: none"> <li>• eventType [EventType]<br/>definiert den Protokolltyp, in welchen das Ereignis geschrieben wird;<br/>Sec = Security: Ereignis wird in das Securityprotokoll geschrieben<br/>Op = Operation: Wenn fmName = „“, wird das Ereignis in das Systemprotokoll geschrieben.<br/>Wenn fmName gesetzt ist, wird das Ereignis in das durch fmName definierte Fachmodul-Protokoll geschrieben.<br/>Perf = Performance: Wenn fmName = „“, wird das Ereignis in das Konnektor-Performanceprotokoll geschrieben.<br/>Wenn fmName gesetzt ist, wird das Ereignis in das durch fmName definierte Fachmodul-Performanceprotokoll geschrieben.</li> <li>• severity { [EventSeverity] , Debug}<br/>(Schwere mit: Debug = Debug Information, Info = Information, Warn = Warning, Err = Error, Fatal)</li> <li>• parameters<br/>beinhaltet die Daten des Ereignisses, die im Protokolleintrag geschrieben werden</li> </ul> |
| Komponenten     | Konnektor  |
| Ausgangsdaten   | Keine  |
| Nachbedingungen |  |
| Standardablauf  | <ol style="list-style-type: none"> <li>1. Wenn eventType = Sec, so wird das Ereignis in das Sicherheitsprotokoll geschrieben. Falls fmName angegeben ist, wird er dem Eintrag hinzugefügt.</li> <li>2. fmName ist angegeben (durch ein Fachmodul aufgerufen) und eventType = „Op“, so wird das Ereignis in das zugehörige Fachmodulprotokoll geschrieben. <ol style="list-style-type: none"> <li>a. Gemäß den Festlegungen in den jeweiligen Fachmodulspezifikationen (FM_&lt;fmName&gt;_LOG_LEVEL), werden nur Ereignisse in das Fachmodulprotokoll geschrieben, deren severity mindestens dem jeweils dort festgelegten Wert entsprechen.</li> </ol> </li> <li>3. fmName ist nicht angegeben (Aufruf durch ein Fachmodul) und eventType = „Op“, dann wird das Ereignis in das Systemprotokoll geschrieben. <ol style="list-style-type: none"> <li>a. Gemäß den Festlegungen in LOG_LEVEL_SYSLOG werden nur Ereignisse in das Systemprotokoll geschrieben, deren Schwere mindestens dem Wert von LOG_LEVEL_SYSLOG entsprechen.</li> </ol> </li> </ol>         |

|                                |  |
|--------------------------------|--|
|                                | <p>4. Wurde der TUC durch ein Fachmodul aufgerufen (fmName ist angegeben) und ist eventType = Perf, so wird das Ereignis in das zugehörige Fachmodul-Performanceprotokoll geschrieben.</p> <p>5. Wurde der TUC nicht durch ein Fachmodul aufgerufen (fmName ist nicht angegeben) und ist eventType = Perf, so wird das Ereignis in das Konnektor-Performanceprotokoll geschrieben.<br/>Die geschriebenen Protokolleinträge MÜSSEN mindestens folgende Attribute beinhalten:</p> <ul style="list-style-type: none"> <li>• Datum und Uhrzeit</li> <li>• Übergebenes Ereignis</li> </ul> <p>Die Speicherung erfolgt rollierend.<br/>Übersteigt die Anzahl der Einträge im Sicherheitsprotokoll SECURITY_LOG_SIZE, so werden ältere Einträge überschrieben. Für die anderen Protokolle beginnt das Überschreiben, wenn der jeweilige Speicherplatz für das Protokoll erschöpft ist. Dabei werden die nach der Reihenfolge der Erstellung ältesten Einträge überschrieben, unabhängig vom Zeitstempel des Logeintrags.<br/>Ist der Zeitstempel eines überschriebenen Logeintrags jünger als LOG_DAYS bzw. FM_&lt;fmName&gt;_LOG_DAYS bzw. SECURITY_LOG_DAYS, so wird der Fehlerzustand EC_LOG_OVERFLOW ausgelöst.</p> |
| Fehlerfälle                    | <p>Fehler in den folgenden Schritten des Standardablaufs führen zu:</p> <p>a) Aufruf von<br/>TUC_KON_256 {<br/>  topic = „LOG/ERROR“;<br/>  eventType = \$ErrorType;<br/>  severity = \$Severity;<br/>  parameters = („Error=\$Fehlercode, Bedeutung=\$Fehlertext“);<br/>  doLog = false }</p> <p>b) Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes</p> <p>(→1) In das Sicherheitsprotokoll kann nicht geschrieben werden:<br/>Fehlercode: 4152</p> <p>(→2) In das Fachmodulprotokoll kann nicht geschrieben werden:<br/>Fehlercode: 4151</p> <p>(→3) In das Systemprotokoll kann nicht geschrieben werden:<br/>Fehlercode: 4150</p> <p>(→4) In das Fachmodul-Performanceprotokoll kann nicht geschrieben werden: Fehlercode: 4217</p> <p>(→5) In das Konnektor-Performanceprotokoll kann nicht geschrieben werden: Fehlercode: 4216</p>   |
| Nichtfunktionale Anforderungen | Keine  |

|                      |       |
|----------------------|-------|
| Zugehörige Diagramme | Keine |
|----------------------|-------|

5233 **Tabelle 286: TAB\_KON\_608 Fehlercodes TUC\_KON\_271 „Schreibe Protokolleintrag“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4150  | Technical | Fatal    | Fehler beim Schreiben des Systemprotokolls                  |
| 4151  | Technical | Fatal    | Fehler beim Schreiben eines Fachmodulprotokolls             |
| 4152  | Security  | Error    | Fehler beim Schreiben des Sicherheitsprotokolls             |
| 4216  | Technical | Fatal    | Fehler beim Schreiben des Konnektor-Performanceprotokolls   |
| 4217  | Technical | Fatal    | Fehler beim Schreiben eines Fachmodul-Performanceprotokolls |

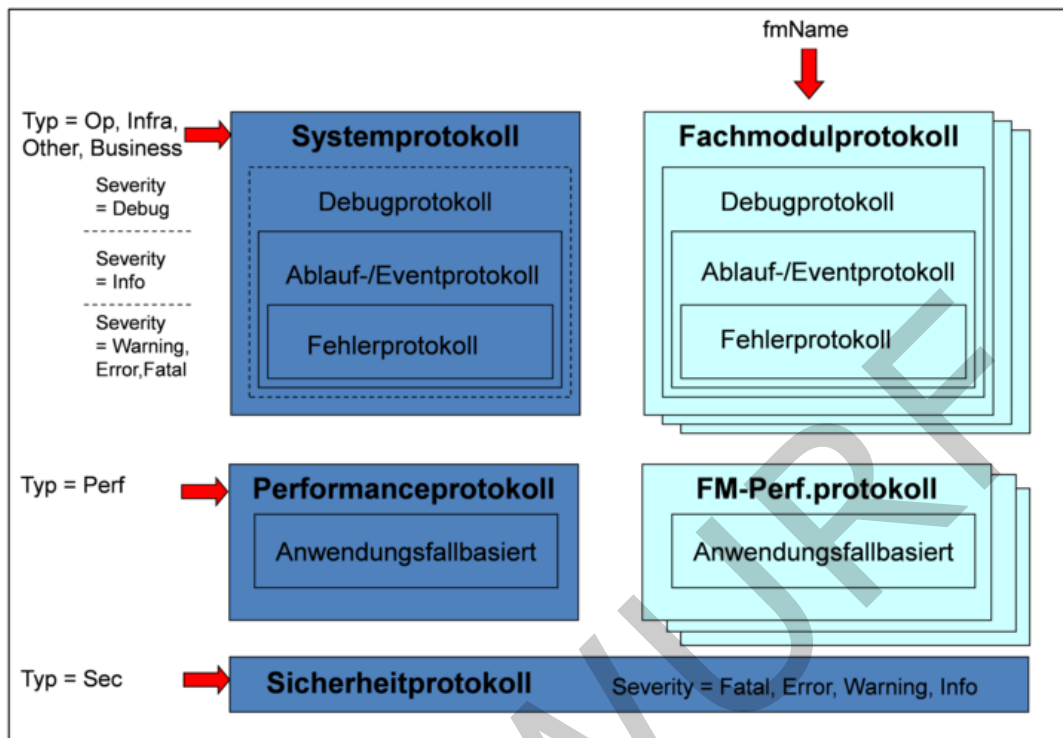
5234

5235 **[<=]**

5236 Die Darstellung PIC\_KON\_118 veranschaulicht den Aufbau der Protokolle für Plattform  
5237 und Fachmodule und die Steuerung der Protokolleinträge in TUC\_KON\_271 „Schreibe

5238

Protokolleintrag".



5239

5240

5241

**Abbildung 20: PIC\_KON\_118 Aufbau und Struktur der Protokolldateien für Plattform und Fachmodule**

5242

#### 4.1.10.5 Operationen an der Außenschnittstelle

5243

Keine

5244

#### 4.1.10.6 Betriebsaspekte

5245

##### **TIP1-A\_4716 - Einsichtnahme und Veränderung der Protokolle**

5246

Der Administrator MUSS die durch den Protokollierungsdienst geschriebenen Protokolle über die Managementschnittstelle einsehen können.

5247

Eine Veränderung des Sicherheitsprotokolls DARF für den Administrator NICHT möglich sein.

5248

Das Löschen folgender Protokolle MUSS für den Administrator möglich sein:

5249

5250

5251

- Systemprotokoll

5252

- das jeweils durch <fmName> spezifizierte Fachmodulprotokoll

5253

- Konnektor-Performanceprotokoll

5254

- das jeweils durch <fmName> spezifizierte Fachmodul-Performanceprotokoll

5255

Der Konnektor MUSS den Export von Protokolleinträgen oder ganzen Protokolldateien unterstützen.

5256

Der Konnektor SOLL das Sortieren und Filtern der Protokolleinträge sowie das Suchen in den Protokolleinträgen unterstützen.

5257

[&lt;=]



**TIP1-A\_4996 - Hinweis auf neue Sicherheitsprotokolleinträge**

Nachdem sich der Administrator an der Managementschnittstelle angemeldet hat, MUSS der Konnektor ihn automatisch auf Sicherheitsprotokolleinträge hinweisen, die seit dem Ausloggen dieses Administrator aufgelaufen sind.

[<=]

**TIP1-A\_4717 - Konfigurationswerte des Protokollierungsdienstes**

Die Managementschnittstelle MUSS es einem Administrator ermöglichen Konfigurationsänderungen gemäß Tabelle TAB\_KON\_609 vorzunehmen:

**Tabelle 287: TAB\_KON\_609 Konfigurationswerte des Protokollierungsdienstes (Administrator)**

| ReferenzID                | Belegung                                 | Bedeutung und Administrator-Interaktion   |
|---------------------------|--|---|
| LOG_LEVEL_<br>SYSLOG      | Info, Warning,<br>Error, Fatal           | Der Administrator MUSS den Detaillierungsgrad des Systemprotokolls durch Festlegung der Mindest-Schwere zu protokollierender Einträge festlegen können.<br>Default-Wert: Warning  |
| FM_<fmName>_<br>LOG_LEVEL | Debug, Info,<br>Warning, Error,<br>Fatal | Der Administrator MUSS den Detaillierungsgrad des Fachmodulprotokolls durch Festlegung der Mindest-Schwere zu protokollierender Einträge festlegen können.<br>Default-Wert: Warning   |
| SECURITY_LOG_SIZE         | X Einträge                               | Der Administrator MUSS die Größe des Sicherheitsprotokolls angeben können (Anzahl der Einträge im Ringbuffer).<br>Mindestgröße: >= 10.000<br>Maximalgröße: herstellerspezifisch<br>Default-Wert: >= 50.000  |
| SECURITY_LOG_DAYS         | X Tage                                   | Der Administrator MUSS die erwartete Anzahl der im Sicherheitsprotokoll gespeicherten Tage im Bereich 10 bis 365 konfigurieren können.<br>Default-Wert: 180   |
| LOG_DAYS                  | X Tage                                   | Der Administrator MUSS die Anzahl der gespeicherten Tage für das Systemprotokoll und das Performanceprotokoll festlegen können. Der Konnektor kann Protokolleinträge, die älter als LOG_DAYS sind, zyklisch löschen. Dabei DARF der eingestellte Wert NICHT unter der Mindestgröße von 10 Tagen oder über der Maximalgröße von einem Jahr (365 Tage) liegen.<br>Default-Wert: 180 |

|                               |                  |   |
|-------------------------------|------------------|---|
| FM_<fmName>_<br>LOG_DAYS      | X Tage           | <p>Der Administrator MUSS die Anzahl der gespeicherten Tage für die fachmodulspezifischen Protokolle festlegen können. Es kann je Fachmodul einen Konfigurationsparameter für LOG_DAYS geben, der gemeinsam für das Fachmodulprotokoll und das Fachmodul-Performanceprotokoll gilt.</p> <p>Der Konnektor kann Protokolleinträge, die älter als FM_&lt;fmName&gt;LOG_DAYS sind, zyklisch löschen.</p> <p>Dabei DARF der eingestellte Wert NICHT unter der Mindestgröße von 10 Tagen oder über der Maximalgröße von einem Jahr (365 Tage) liegen.</p> <p>Default-Wert: 180</p> <p>Die Definition des fachmodulspezifischen Konfigurationswertes ist Bestandteil der entsprechenden Fachmodulspezifikation. Ist kein fachmodulspezifischer Konfigurationsparameter spezifiziert, dann gilt LOG_DAYS.</p> |
| LOG_SUCCESSFUL_<br>CRYPTO_OPS | Enabled/Disabled | <p>Der Administrator MUSS festlegen können, ob auch erfolgreich ausgeführte Kryptooperationen im Sicherheitslog protokolliert werden sollen.</p> <p>Default-Wert: Disabled</p>  |

5271

5272 [ $\leq$ ]

5273 4.1.10.6.1 TUC\_KON\_272 „Initialisierung Protokollierungsdienst“

5274 **TIP1-A\_4718 - TUC\_KON\_272 „Initialisierung Protokollierungsdienst“**

5275 Der Konnektor MUSS den technischen Use Case TUC\_KON\_272 „Initialisierung

5276 Protokollierungsdienst“ umsetzen.

5277

5278 **Tabelle 288: TAB\_KON\_610 – TUC\_KON\_272 „Initialisierung Protokollierungsdienst“**

| Element                     | Beschreibung   |
|-----------------------------|--|
| Name                        | TUC_KON_272 „Initialisierung Protokollierungsdienst“   |
| Beschreibung                | Der Konnektor muss zum Bootup den Protokollierungsdienst starten und die Existenz und Schreibbarkeit der Protokolle sicherstellen. |
| Eingangs anforderung        | Keine  |
| Auslöser und Vorbedingungen | Bootup   |
| Eingangsdaten               | Keine  |
| Komponenten                 | Konnektor  |

|                                   |   |
|-----------------------------------|---|
| Ausgangsdaten                     | Keine   |
| Standardablauf                    | <ol style="list-style-type: none"> <li>1. Prüfen, ob Schreib-/Lesezugriff auf Sicherheitsprotokoll möglich ist</li> <li>2. Prüfen, ob Schreib-/Lesezugriff auf Systemprotokoll möglich ist</li> <li>3. Prüfen, ob Schreib-/Lesezugriff auf Fachmodulprotokolle möglich ist</li> <li>4. Prüfen, ob Schreib-/Lesezugriff auf Konnektor-Performanceprotokoll möglich ist</li> <li>5. Prüfen, ob Schreib-/Lesezugriff auf Fachmodul-Performanceprotokolle möglich ist</li> </ol>  |
| Varianten/<br>Alternativen        | Keine   |
| Fehlerfälle                       | <p>Fehler in den folgenden Schritten des Standardablaufs führen zu:</p> <p>a) Aufruf von TUC_KON_256 mit folgenden Parametern {<br/> topic = „LOG/ERROR“;<br/> eventType = \$ErrorType;<br/> severity = \$Severity;<br/> parameters = („Error=\$Fehlercode, Bedeutung=\$Fehlertext“);<br/> doLog = false }</p> <p>b) Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes</p> <p>(→1) Zugriff nicht möglich: Fehlercode: 4153<br/> (→2) Zugriff nicht möglich: Fehlercode: 4154<br/> (→3) Zugriff nicht möglich: Fehlercode: 4155<br/> (→4) Zugriff nicht möglich: Fehlercode: 4218<br/> (→5) Zugriff nicht möglich: Fehlercode: 4219</p> |
| Nichtfunktionale<br>Anforderungen | Keine   |
| Zugehörige<br>Diagramme           | Keine   |

**Tabelle 289: TAB\_KON\_611 Fehlercodes TUC\_KON\_272 „Initialisiere Protokollierungsdienst“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4153  | Technical | Fatal    | Zugriff auf Sicherheitsprotokoll nicht möglich           |
| 4154  | Technical | Fatal    | Zugriff auf Systemprotokoll nicht möglich                |
| 4155  | Technical | Fatal    | Zugriff auf Fachmodulprotokolle nicht möglich            |
| 4218  | Technical | Fatal    | Zugriff auf Konnektor-Performanceprotokoll nicht möglich |
| 4219  | Technical | Fatal    | Zugriff auf Fachmodul-Performanceprotokoll nicht möglich |

[<=]

#### 5283 4.1.11 TLS-Dienst

5284 Fachmodule müssen gesicherte Verbindungen zu Fachdiensten in der TI aufbauen  
 5285 können. Dabei sollen sie sich mit einer Organisationsidentität (einer SM-B) authentisieren  
 5286 können. Der TLS-Dienst stellt hierfür TUCs für einen TLS-Verbindungsaufbau und -  
 5287 Verbindungsabbau zur Verfügung. Die gesicherte Kommunikation selbst erfolgt dann  
 5288 durch das Fachmodul unter Nutzung der etablierten Verbindung.

5289 Die Funktionalität steht nur zur Verfügung, wenn MGM\_LU\_ONLINE aktiv ist (siehe  
 5290 Kapitel 4.3.6)

#### 5291 4.1.11.1 Funktionsmerkmalweite Aspekte

##### 5292 4.1.11.2 Durch Ereignisse ausgelöste Reaktionen

##### 5293 TIP1-A\_4719 - TLS-Dienst reagiert auf Veränderung LU\_ONLINE

5294 Tritt das Ereignis „MGM/LU\_CHANGED/LU\_ONLINE“ ein, so MUSS

- 5295 • wenn „Active=Enabled“ der Dienst bereitgestellt werden
- 5296 • wenn „Active=Disabled“ der Dienst gestoppt werden.
- 5297 Sind TLS-Verbindungen aktiv, so MUSS für jede TUC\_KON\_111 "Kartenbasierte
- 5298 TLS-Verbindung abbauen" gerufen werden.

5299 [ $\leq$ ]

#### 5300 4.1.11.3 Interne TUCs, nicht durch Fachmodule nutzbar

5301 Keine.

#### 5302 4.1.11.4 Interne TUCs, auch durch Fachmodule nutzbar

##### 5303 4.1.11.4.1 TUC\_KON\_110 „Kartenbasierte TLS-Verbindung aufbauen“

##### 5304 TIP1-A\_4720 - TUC\_KON\_110 „Kartenbasierte TLS-Verbindung aufbauen“

5305 Der Konnektor MUSS den technischen Use Case "Kartenbasierte TLS-Verbindung  
 5306 aufbauen" gemäß TUC\_KON\_110 umsetzen.

5307

#### 5308 Tabelle 290: TAB\_KON\_773 – TUC\_KON\_110 „Kartenbasierte TLS-Verbindung aufbauen“

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_110 „Kartenbasierte TLS-Verbindung aufbauen“  |
| Beschreibung   | Der TUC_KON_110 „Kartenbasierte TLS-Verbindung aufbauen“ baut eine TLS-Verbindung zur angegebenen Zieladresse auf. Dabei kann für eine gegenseitige Authentisierung eine SM-B verwendet werden. |
| Auslöser       | Aufruf durch ein Fachmodul  |
| Vorbedingungen | Die für die Authentisierung adressierte Karte muss freigeschaltet sein  |
| Eingangsdaten  | <ul style="list-style-type: none"> <li>• roleToMatch – <i>optional/verpflichtend, wenn Rollenprüfung durchgeführt werden soll</i></li> </ul>  |

|                                |   |
|--------------------------------|---|
|                                | <ul style="list-style-type: none"> <li>cardSession – <i>optional/verpflichtend, wenn Clientauthentisierung durchgeführt werden soll</i> (CardSession einer SM-B)</li> <li>targetUri (URI des Verbindungsziels)</li> </ul>   |
| Komponenten                    | Konnektor, eHealth-Kartenterminal, Karte, Server des Fachdienstes   |
| Ausgangsdaten                  | <ul style="list-style-type: none"> <li>tlsConnectionId (ConnectionIdentifier der TLS-Verbindung)</li> </ul>   |
| Standardablauf                 | <p>Der Konnektor MUSS folgende Schritte durchlaufen:</p> <ol style="list-style-type: none"> <li>Auflösen des FQDN im targetUri per 'TUC_KON_361 „DNS Namen auflösen“</li> <li>TLS-Verbindung mit ermittelter Adresse aufbauen: <ol style="list-style-type: none"> <li>Prüfe Server-Zertifikat mittels TUC_KON_037 „Zertifikat prüfen“ { <pre>certificate = C.FD.TLS-S; qualifiedCheck = not_required; offlineAllowNoCheck = true; policyList = oid_fd_tls_s; intendedKeyUsage= intendedKeyUsage(C.FD.TLS-S); intendedExtendedKeyUsage = id-kp-serverAuth; validationMode = OCSP}</pre> Das Server-Zertifikat MUSS C.FD.TLS-S sein </li> <li>Prüfe in a) zurückgegebene Rolle („ermittelte Rolle“) == roleToMatch</li> <li>Wenn cardSession übergeben: Clientauthentisierung mittels ID.HCI.AUT</li> </ol> </li> <li>tlsConnectionId der erzeugten Verbindung zurückgeben</li> </ol> |
| Varianten/ Alternativen        | <ul style="list-style-type: none"> <li>Keine</li> </ul>   |
| Fehlerfälle                    | <p>Fehler in den folgenden Schritten des Ablaufs führen zu einem Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes</p> <p>(→1) Der Name der Gegenstelle kann nicht aufgelöst werden</p> <p>(→2b) Rollenprüfung fehlgeschlagen: Fehlercode 4220</p> <p>(→2) Server konnte nicht authentisiert werden: Fehlercode 4156</p> <p>(→2) Clientauthentisierung fehlgeschlagen: Fehlercode 4157</p>   |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | keine   |

5309  
5310**Tabelle 291: TAB\_KON\_612 Fehlercodes TUC\_KON\_110 „Kartenbasierte TLS-Verbindung aufbauen“**

| Fehlercode | ErrorType | Severity | Fehlertext |
|------------|-----------|----------|------------|
|------------|-----------|----------|------------|

|   |          |       |  |
|---|----------|-------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |          |       |  |
| 4156  | Security | Error | Server konnte bei TLS-Verbindungsaufbau nicht authentisiert werden |
| 4157  | Security | Error | Clientauthentisierung bei TLS-Verbindungsaufbau fehlgeschlagen     |
| 4220  | Security | Error | Rollenprüfung bei TLS-Verbindungsaufbau fehlgeschlagen             |

5311

5312 [ $\leq$ ]

5313 4.1.11.4.2 TUC\_KON\_111 „Kartenbasierte TLS-Verbindung abbauen“

5314 **TIP1-A\_4721 - TUC\_KON\_111 „Kartenbasierte TLS-Verbindung abbauen“**

5315 Der Konnektor MUSS den technischen Use Case "Kartenbasierte TLS-Verbindung abbauen" gemäß TUC\_KON\_111 umsetzen.

5316

5317

5318 **Tabelle 292: TAB\_KON\_774 - TUC\_KON\_111 „Kartenbasierte TLS-Verbindung abbauen“**

| Element                        | Beschreibung   |
|--------------------------------|--|
| Name                           | TUC_KON_111 „Kartenbasierte TLS-Verbindung abbauen“  |
| Beschreibung                   | Der TUC_KON_111 „Kartenbasierte TLS-Verbindung abbauen“ dient der geregelten Beendigung einer TLS-Verbindung, die zuvor über TUC_KON_110 aufgebaut wurde.  |
| Auslöser                       | Aufruf durch ein Fachmodul   |
| Vorbedingungen                 | Mittels TUC_KON_110 wurde eine TLS-Verbindung aufgebaut  |
| Eingangsdaten                  | <ul style="list-style-type: none"> <li>tlsConnectionId (ConnectionIdentifier der TLS-Verbindung)</li> </ul>  |
| Komponenten                    | Konnektor, Server des Fachdienstes   |
| Ausgangsdaten                  | Keine  |
| Standardablauf                 | Der Konnektor MUSS folgende Schritte durchlaufen:<br>1. Trennen der über tlsConnectionId adressierten TLS-Verbindung   |
| Varianten/<br>Alternativen     | keine  |
| Fehlerfälle                    | Fehler in den folgenden Schritten des Ablaufs führen zu einem Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes:<br>(→1) Keine Verbindung mit angegebenem TLSConnectionIdentifier<br>vorhanden: Fehlercode 4158 |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | keine  |

5319

5320 **Tabelle 293: TAB\_KON\_613 Fehlercodes TUC\_KON\_111 „Kartenbasierte TLS-Verbindung**  
 5321 **abbauen“**

| Fehlercode  | ErrorType | Severity | Fehlertext                                 |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4158  | Technical | Error    | Adressierte TLS-Verbindung nicht vorhanden |

5322

5323 [ $\leq$ ]

#### 5324 **4.1.11.5 Operationen an der Außenschnittstelle**

5325 Keine.

#### 5326 **4.1.11.6 Betriebsaspekte**

##### 5327 **TIP1-A\_4722 - TLS-Dienst initialisieren**

5328 Wenn MGM\_LU\_ONLINE = „Enabled“, MUSS der Basisdienst TLS-Dienst nach dem Bootup zur Nutzung zur Verfügung stehen.

5330 Wenn MGM\_LU\_ONLINE = „Disabled“, DARF der Basisdienst TLS-Dienst nach dem Bootup NICHT zur Nutzung zur Verfügung stehen.

5332 [ $\leq$ ]

#### 5333 **4.1.12 LDAP-Proxy**

5334 Der Konnektor ermöglicht es Clientsystemen und Fachmodulen durch Nutzung des LDAP-  
 5335 Proxies Daten aus dem Verzeichnisdienst der TI-Plattform (VZD) abzufragen. Die  
 5336 Kommunikation erfolgt über das LDAPv3 Protokoll.

5337 Die Funktionalität steht nur zur Verfügung, wenn MGM\_LU\_ONLINE=Enabled ist (siehe  
 5338 Kapitel 4.3.6)

#### 5339 **4.1.12.1 Funktionsmerkmalweite Aspekte**

5340 Keine.

#### 5341 **4.1.12.2 Durch Ereignisse ausgelöste Reaktionen**

##### 5342 **TIP1-A\_5516 - LDAP-Proxy reagiert auf Veränderung LU\_ONLINE**

5343 Tritt das Ereignis „MGM/LU\_CHANGED/LU\_ONLINE“ ein, so MUSS

- 5344 • wenn „Active=Enabled“ der Dienst bereitgestellt werden
- 5345 • wenn „Active=Disabled“ der Dienst gestoppt werden.
- 5346 Ist eine Verbindung zum VZD aktiv, so MUSS diese abgebaut werden.

5347 [ $\leq$ ]

#### 5348 **4.1.12.3 Interne TUCs, nicht durch Fachmodule nutzbar**

5349 Keine.

#### 5350 4.1.12.4 Interne TUCs, auch durch Fachmodule nutzbar

##### 5351 4.1.12.4.1 TUC\_KON\_290 „LDAP-Verbindung aufbauen“

#### 5352 **TIP1-A\_5517-02 - Konnektor, TUC\_KON\_290 „LDAP-Verbindung aufbauen“**

5353 Der Konnektor MUSS den technischen Use Case TUC\_KON\_290 „LDAP-Verbindung  
5354 aufbauen“ gemäß TAB\_KON\_805 umsetzen.

5355

#### 5356 **Tabelle 294: TAB\_KON\_805 - TUC\_KON\_290 „LDAP-Verbindung aufbauen“**

| Element                        | Beschreibung  |
|--------------------------------|---|
| Name                           | TUC_KON_290 „LDAP-Verbindung aufbauen“  |
| Beschreibung                   | Initiiert durch einen Verbindungsaufbau des LDAP-Clients zum Konnektor baut der Konnektor eine TLS-gesicherte Verbindung zum VZD auf.   |
| Auslöser                       | LDAP (oder LDAPS wenn ANCL_TLS_MANDATORY=Enabled) Verbindungsaufbau von einem Fachmodul oder einem Clientsystem ist abgeschlossen. Bei Verwendung von LDAPS authentisiert sich der Konnektor beim LDAP-Client mit der Identität ID.AK.AUT.  |
| Vorbedingungen                 | <ul style="list-style-type: none"> <li>MGM_LU_ONLINE=Enabled</li> </ul>   |
| Eingangsdaten                  | keine   |
| Komponenten                    | Konnektor, VZD  |
| Ausgangsdaten                  | keine   |
| Standardablauf                 | <ol style="list-style-type: none"> <li>Der Konnektor ermittelt den FQDN und Port des VZD durch eine DNS-SD Namensauflösung gemäß [RFC6763] mit dem Bezeichner<br/>"_ldap._tcp.vzd.&lt;DNS_TOP_LEVEL_DOMAIN_TI&gt;."</li> <li>Der Konnektor baut eine LDAPS-Verbindung zum VZD auf. Dabei wird das Serverzertifikat des Verzeichnisdienst C.ZD.TLS-S nach TUC_PKI_018 geprüft (PolicyList: oid_zd_tls_s (gemäß gemSpec_OID), intendedKeyUsage: intendedKeyUsage(C.ZD.TLS-S), ExtendedKeyUsages: serverAuth (1.3.6.1.5.5.7.3.1), Offlinemodus: nein, TOLERATE_OCSP_FAILURE: false , Prüfmodus: OCSP)</li> </ol> |
| Varianten/Alternativen         | keine   |
| Fehlerfälle                    |   |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | keine   |

5357

5358 [**<=**]



5359

5360 4.1.12.4.2 TUC\_KON\_291 „Verzeichnis abfragen“

5361 **TIP1-A\_5518 - Konnektor, TUC\_KON\_291 „Verzeichnis abfragen“**5362 Der Konnektor MUSS den technischen Use Case TUC\_KON\_291 „Verzeichnis abfragen“  
5363 gemäß TAB\_KON\_815 umsetzen.

5364

5365 **Tabelle 295: TAB\_KON\_815 – TUC\_KON\_291 „Verzeichnis abfragen“**

| Element                        | Beschreibung   |
|--------------------------------|--|
| Name                           | TUC_KON_291 „Verzeichnis abfragen“   |
| Beschreibung                   | Der Konnektor leitet als LDAP-Proxy einen Search Request des LDAP-Clients an den VZD weiter. Vom VZD empfängt der Konnektor eine Search Response und leitet diese an den LDAP-Client weiter.                                     |
| Auslöser                       | Aufruf durch einen LDAPv3 Search Request von einem Fachmodul-TUC oder einem Clientsystem   |
| Vorbedingungen                 | <ul style="list-style-type: none"> <li>MGM_LU_ONLINE=Enabled</li> <li>Eine LDAPv3-Verbindung LDAP-Client sowie eine LDAPv3 Verbindung vom Konnektor zum VZD wurden aufgebaut (TUC_KON_290 „LDAP-Verbindung aufbauen“)</li> </ul> |
| Eingangsdaten                  | <ul style="list-style-type: none"> <li>LDAPv3 Search Request gemäß [RFC4511]#4.5.1</li> </ul>  |
| Komponenten                    | Konnektor, VZD   |
| Ausgangsdaten                  | <ul style="list-style-type: none"> <li>LDAPv3 Search Response gemäß [RFC4511]#4.5.2</li> </ul>   |
| Standardablauf                 | 1. Der Konnektor führt TUC_VZD_0001 „search_Directory“ mit dem vom LDAP-Client empfangenen Search Request als Eingangsdaten aus und empfängt die LDAPv3 Search Response vom VZD (entspricht den Ausgangsdaten).                  |
| Varianten/Alternativen         | keine  |
| Fehlerfälle                    | Auftretende Fehler werden gemäß [RFC4511]# Appendix A behandelt.   |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | keine  |

5366 [**<=**]

5367 4.1.12.4.3 TUC\_KON\_292 „LDAP-Verbindung trennen“

5368 **TIP1-A\_5519 - Konnektor, TUC\_KON\_292 „LDAP-Verbindung trennen“**5369 Der Konnektor MUSS den technischen Use Case „LDAP-Verbindung trennen“ gemäß  
5370 TAB\_KON\_816 umsetzen.

5371

5372 **Tabelle 296: TAB\_KON\_816 – TUC\_KON\_292 „LDAP-Verbindung trennen“**

| Element                        | Beschreibung   |
|--------------------------------|--|
| Name                           | TUC_KON_292 „LDAP-Verbindung trennen“  |
| Beschreibung                   | Der Konnektor beendet die Verbindung zum VZD und zum LDAP-Client.  |
| Auslöser                       | Aufruf durch einen LDAPv3 Unbind Request von einem Fachmodul-TUC oder einem Clientsystem   |
| Vorbedingungen                 | <ul style="list-style-type: none"> <li>• MGM_LU_ONLINE=Enabled</li> <li>• Eine LDAPv3-Verbindung LDAP-Client sowie eine LDAPv3 Verbindung vom Konnektor zum VZD wurden aufgebaut (TUC_KON_290 „Verbindungsaufbau zum VZD“)</li> </ul>  |
| Eingangsdaten                  | keine  |
| Komponenten                    | Konnektor, VZD   |
| Ausgangsdaten                  | keine  |
| Standardablauf                 | <ol style="list-style-type: none"> <li>1. Der Konnektor empfängt vom LDAP-Client einen Unbind Request gemäß [RFC4511]#4.3.</li> <li>2. Der Konnektor sendet zum VZD einen Unbind Request.</li> <li>3. Der Konnektor beendet die Verbindung zum VZD und zum LDAP Client gemäß [RFC4511]#5.3.</li> </ol> |
| Varianten/Alternativen         | keine  |
| Fehlerfälle                    | Auftretende Fehler werden gemäß [RFC4511]# Appendix A behandelt.   |
| Nichtfunktionale Anforderungen | keine  |
| Zugehörige Diagramme           | keine  |

5373  
5374 [**<=**]

5375 4.1.12.4.4 TUC\_KON\_293 „Verzeichnisabfrage abbrechen“

5376 **TIP1-A\_5520 - Konnektor, TUC\_KON\_293 „Verzeichnisabfrage abbrechen“**

5377 Der Konnektor MUSS den technischen Use Case TUC\_KON\_293 „Verzeichnisabfrage  
5378 abbrechen“ gemäß TAB\_KON\_817 umsetzen.

5379

5380 **Tabelle 297: TAB\_KON\_817 – TUC\_KON\_293 „Verzeichnisabfrage abbrechen“**

| Element                        | Beschreibung  |
|--------------------------------|---|
| Name                           | TUC_KON_293 „Verzeichnisabfrage abbrechen“  |
| Beschreibung                   | Der Konnektor bricht einen unbeantworteten Search Request ab.   |
| Auslöser                       | Aufruf durch einen LDAPv3 Abandon Request von einem Fachmodul-TUC oder einem Clientsystem   |
| Vorbedingungen                 | <ul style="list-style-type: none"> <li>• MGM_LU_ONLINE=Enabled</li> <li>• Ein Search Request wurde vom Konnektor empfangen und an den VZD weitergeleitet (TUC_KON_291 „Verzeichnis Abfragen“). Der Request wurde vom VZD noch nicht beantwortet.</li> </ul> |
| Eingangsdaten                  | keine   |
| Komponenten                    | Konnektor, VZD  |
| Ausgangsdaten                  | keine   |
| Standardablauf                 | <ol style="list-style-type: none"> <li>1. Der Konnektor empfängt vom LDAP-Client einen Abandon Request gemäß [RFC4511]#4.11.</li> <li>2. Der Konnektor sendet zum VZD einen Abandon Request gemäß [RFC4511]#4.11</li> </ol>                                 |
| Varianten/Alternativen         | keine   |
| Fehlerfälle                    | Auftretende Fehler werden gemäß [RFC4511]# Appendix A behandelt.  |
| Nichtfunktionale Anforderungen | keine   |
| Zugehörige Diagramme           | keine   |

5381

5382 [ $\leq$ ]

5383 **4.1.12.5 Operationen an der Außenschnittstelle**

5384 4.1.12.5.1 Unterstützte LDAPv3 Operationen

5385 **TIP1-A\_5521 - Konnektor, LDAPv3 Operationen**

5386 Der Konnektor MUSS an der Client-Schnittstelle die folgenden LDAPv3 Operationen  
5387 gemäß [RFC4511] anbieten.

5388 

- Bind Operation

5389 

- Unbind Operation

- 5390 • Search Operation
- 5391 • Abandon Operation

5392 Andere LDAPv3 Operationen werden mit dem LDAP-Fehler unwillingToPerform (53)  
5393 beantwortet.

5394 Wenn ANCL\_TLS\_MANDATORY=Enabled, muss der Konnektor sicherstellen, dass nur  
5395 über eine LDAPS-Verbindung (Voreinstellung TCP Port 636) Daten abgefragt werden  
5396 können.

5397 Wenn ANCL\_TLS\_MANDATORY=Disabled, muss der Konnektor sicherstellen, dass über  
5398 eine LDAP-Verbindung (Voreinstellung TCP Port 389) und über eine LDAPS-Verbindung  
5399 (Voreinstellung TCP Port 636) Daten abgefragt werden können.

5400 Fehler müssen gemäß [RFC4511]#Appendix A behandelt werden.

5401 [ $\leq$ ]

#### 5402 4.1.12.6 Betriebsaspekte

5403 keine

#### 5404 4.1.13 Authentifizierungsdienst

5405 Der Authentifizierungsdienst bietet Clientsystemen und Fachmodulen eine Schnittstelle  
5406 zum Signieren von Binärstrings zum Zweck der externen Authentisierung.

5407 Innerhalb des Authentifizierungsdienstes werden folgende Präfixe für Bezeichner  
5408 verwendet:

- 5409 • Events (Topic Ebene 1): *keine Events vorhanden*
- 5410 • Konfigurationsparameter: *keine Konfigurationsparameter vorhanden*

5411 Eine Prüfung der Signatur bietet der Konnektor nicht an.

#### 5412 4.1.13.1 Funktionsmerkmalweite Aspekte

##### 5413 4.1.13.1.1 Externe Authentisierung

#### 5414 TIP1-A\_5437-02 - Signaturverfahren für externe Authentisierung

5415 Der Signatordienst MUSS für die Operation ExternalAuthenticate die Signaturverfahren  
5416 entsprechend TAB\_KON\_780 – Signaturverfahren Externe Authentisierung unterstützen.

#### 5417 Tabelle 298: TAB\_KON\_780 – Signaturverfahren Externe Authentisierung

| Signaturformat           | Standard           | Dokument<br>formate | QES/<br>nonQES | Bemerkung  |
|--------------------------|--------------------|---------------------|----------------|--|
| <b>PKCS#1<br/>(V2.1)</b> | [RFC3447]          | Binär               | nonQES         | Die Low-Level-Signatur<br>von Bitstrings DARF<br>NUR in Verbindung mit<br>dem zur<br>Authentisierung<br>vorgesehenen Schlüssel<br>des HBAX und des SM-<br>B genutzt werden.<br>Die Nutzung ist auf<br>Bitstrings (Hash-Werte)<br>von maximal 512 bit |
| <b>ECDSA</b>             | [BSI-TR-<br>03111] | Binär               | nonQES         |  |

|  |  |  |  |                   |
|--|--|--|--|-------------------|
|  |  |  |  | Länge beschränkt. |
|--|--|--|--|-------------------|

5418

5419 [ $\leq$ ]

5420 **TIP1-A\_5149-01 - ExternalAuthenticate nur für Authentisierung mit HBAX und**  
 5421 **SM-B nutzen**

5422 Der Hersteller des Konnektors MUSS den Anwender (Clientsystem) im Handbuch des  
 5423 Konnektors geeignet und ausreichend darüber informieren, dass die Operation  
 5424 ExternalAuthenticate nur zu Authentisierungszwecken mit dem Authentisierungsschlüssel  
 5425 des HBAX und des SM-B verwendet werden darf. [ $\leq$ ]

5426

#### 5427 4.1.13.2 Durch Ereignisse ausgelöste Reaktionen

5428 keine

#### 5429 4.1.13.3 Interne TUCs

5430 keine

#### 5431 4.1.13.4 Operationen an der Außenschnittstelle

5432 **TIP1-A\_5665-03** ~~TIP1-A\_5665-02~~ - Basisdienst Authentifizierungsdienst

5433 Der Konnektor MUSS Clientsystemen den Basisdienst Authentifizierungsdienst anbieten.

5434

5435 **Tabelle 299: TAB\_KON\_839 Basisdienst Authentifizierungsdienst**

|                          |  |                                |
|--------------------------|--|--------------------------------|
| <b>Name</b>              | AuthSignatureService   |                                |
| <b>Version (KDV)</b>     | 7.4.0 (WSDL-Version)<br>7.4.1 (WSDL-Version)                                       |                                |
| <b>Namensraum</b>        | Siehe <a href="#">Anhang D</a> <a href="#">GitHub</a>                              |                                |
| <b>Namensraum-Kürzel</b> | SIG für Schema und SIGW für WSDL   |                                |
| <b>Operationen</b>       | <b>Name</b>  | <b>Kurzbeschreibung</b>        |
|                          | ExternalAuthenticate   | Binärstring signieren (nonQES) |
| <b>WSDL</b>              | AuthSignatureService_v7_4_1.wsdl<br>AuthSignatureService.wsdl (WSDL-Version 7.4.0) |                                |
| <b>Schema</b>            | Kein   |                                |

5436

5437 [ $\leq$ ]

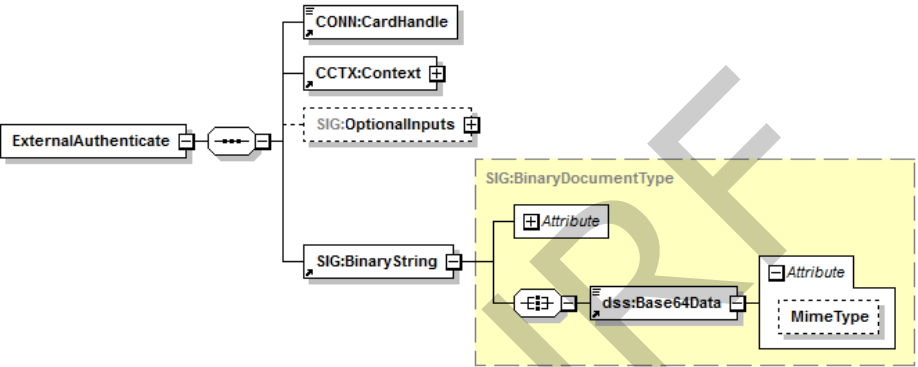
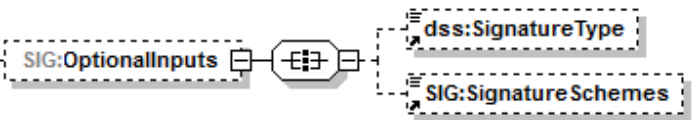
#### 5438 4.1.13.4.1 ExternalAuthenticate

5439 **TIP1-A\_5439 - Operation ExternalAuthenticate**

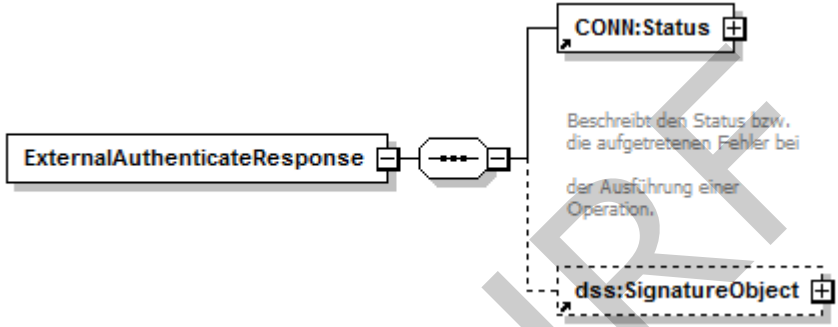
5440 Der Authentifizierungsdienst des Konnektors MUSS an der Clientschnittstelle eine  
 5441 Operation ExternalAuthenticate anbieten.

5442

5443 Tabelle 300: TAB\_KON\_781 Operation ExternalAuthenticate

|                        |  |  |
|------------------------|--|--|
| <b>Name</b>            | ExternalAuthenticate   |  |
| <b>Beschreibung</b>    | Diese Operation versieht einen Binärstring der maximalen Länge 512 Bit mit einer nicht-qualifizierten elektronischen Signatur (nonQES). Dazu wird das Signaturverfahren PKCS#1 oder ECDSA verwendet. Das AUT-Zertifikat der SM-B und das AUT-Zertifikat des HBax werden unterstützt. |  |
| <b>Aufrufparameter</b> |    |  |
|                        | <b>Name</b>  | <b>Beschreibung</b>  |
|                        | CONN: CardHandle   | Identifiziert die zu verwendende Signaturkarte. Die Operation unterstützt HBax und SM-B.   |
|                        | CCTX: Context  | <u>Aufrufkontext für HBax:</u><br>MandantId, ClientSystemId, WorkplaceId, UserId verpflichtend<br><u>Aufrufkontext für SM-B:</u><br>MandantId, ClientSystemId, WorkplaceId verpflichtend; UserId nicht ausgewertet   |
|                        | SIG: Optional Inputs   | Enthält optionale Eingangsparameter:<br>   |
|                        | SIG: Binary String   | Dieses Element enthält im Kindelement dss:Base64Data den zu signierenden Binärstring. Das XML Attribut SIG:BinaryString/dss:Base64Data/@MimeType MUSS den Wert "application/octet-stream" haben. Die maximale Länge des Binärstrings beträgt 512 Bit entsprechend der maximal zu erwartenden Hash-Größe.<br>Aus der Länge des Binärstrings wird auf das verwendete Hashverfahren geschlossen. Es werden folgende Längen unterstützt: <ul style="list-style-type: none"> <li>256 Bit: SHA-256 (OID 2.16.840.1.101.3.4.2.1)</li> </ul> |

|  |                   |  |
|--|-------------------|--|
|  |                   | <ul style="list-style-type: none"> <li>• 384 Bit: SHA-384 (OID 2.16.840.1.101.3.4.2.2)</li> <li>• 512 Bit: SHA-512 (OID 2.16.840.1.101.3.4.2.3)</li> </ul> <p>Im Falle des Signaturverfahrens RSASSA-PKCS1-v1_5 werden SHA-256, SHA-384 und SHA-512 unterstützt.</p> <p>Im Falle des Signaturverfahrens RSASSA-PSS wird SHA-256 unterstützt.</p> <p>Im Falle des Signaturverfahrens ECDSA wird SHA-256 unterstützt.</p> <p>Für die Signaturerstellung gilt:</p> <ul style="list-style-type: none"> <li>• Im Falle des Signaturverfahrens RSASSA-PKCS1-v1_5 beginnt der Konnektor die Ausführung der Methode EMSA-PKCS1-v1_5-ENCODE nach [RFC3447], Abschnitt 9.2, mit Schritt 2, Erstellung des DigestInfo-Datenfeldes.</li> <li>• Im Falle des Signaturverfahrens RSASSA-PSS beginnt der Konnektor die Ausführung der Methode EMSA-PSS-ENCODE nach [RFC3447], Abschnitt 9.1.1, mit Schritt 3.</li> <li>• Im Falle des Signaturverfahrens ECDSA erfolgt die Signaturerstellung gemäß [BSI-TR-03111]#4.2.1. Als Eingangsparameter wird der Hash vom Aufrufer in SIG: BinaryString übergeben.</li> </ul> |
|  | dss:SignatureType | <p>Durch dieses in [OASIS-DSS] (Abschnitt 3.5.1) beschriebene Element wird der Typ der zu erzeugenden Signatur bestimmt. Als Signaturtyp wird unterstützt :</p> <ul style="list-style-type: none"> <li>• <b>PKCS#1-Signatur</b><br/>Durch Übergabe der URI <a href="urn:ietf:rfc:3447">urn:ietf:rfc:3447</a> wird eine PKCS#1 (Version 2.1) Signatur gemäß [RFC3447] erzeugt, die als <code>dss:Base64Signature</code> mit der oben genannten URI zurückgeliefert wird.</li> <li>• <b>ECDSA-Signatur</b><br/>Durch Übergabe der URI <a href="urn:bsi:tr:03111:ecdsa">urn:bsi:tr:03111:ecdsa</a> wird eine ECDSA-Signatur gemäß [BSI-TR-03111]#4.2.1 erzeugt, die als <code>dss:Base64Signature</code> mit der oben genannten URI zurückgeliefert wird.</li> </ul> <p>Andere <code>SignatureType</code>-Angaben führen zu einer Fehlermeldung 4111 (Ungültiger Signaturtyp oder Signaturvariante).</p> <p>Fehlt dieses Element, so wird ebenfalls der Signaturtyp PKCS#1-Signatur verwendet.</p>  |

|                             |  |  |
|-----------------------------|--|--|
|                             | SIG:<br>Signature<br>Schemes   | <p>Durch dieses Element wird für PKCS#1-Signaturen zwischen den folgenden SignatureScheme-Optionen unterschieden:</p> <ul style="list-style-type: none"> <li>• RSASSA-PSS</li> <li>• RSASSA-PKCS1-v1_5</li> </ul> <p>Fehlt dieses Element, so wird als Default-SignatureScheme RSASSA-PSS gewählt.</p>   |
| <b>Rückgabe</b>             |  <p>The diagram shows a box labeled 'ExternalAuthenticateResponse' connected to a dashed box labeled 'dss:SignatureObject'. A solid box labeled 'CONN:Status' is also connected to the 'dss:SignatureObject' box. A text box explains: 'Beschreibt den Status bzw. die aufgetretenen Fehler bei der Ausführung einer Operation.'</p> |  |
|                             | CONN:<br>Status  | Enthält den Status der ausgeführten Operation.   |
|                             | dss:<br>Signature<br>Object  | <p>Enthält im Erfolgsfall die erzeugte Signatur in Form eines dss:SignatureObject-Elements gemäß [OASIS-DSS] (Abschnitt 3.2). Der Signaturwert wird im XML-Element dss:SignatureObject/dss:Base64Signature übergeben. Die Signatur wird binär gemäß [BSI-TR-03111]#5.2.2 in der ASN.1 Struktur ECDSA-Sig-Value zurückgegeben. Das XML-Attribut dss:SignatureObject/dss:Base64Signature/@Type kennzeichnet durch den Wert:</p> <ul style="list-style-type: none"> <li>• <a href="urn:ietf:rfc:3447">urn:ietf:rfc:3447</a> den Signatur-Typ PKCS#1 bzw.</li> <li>• <a href="urn:bsi:tr:03111:ecdsa">urn:bsi:tr:03111:ecdsa</a> den Signatur-Typ ECDSA.</li> </ul> <p>Die XML-Elemente<br/> dss:SignatureObject/ds:Signature<br/> dss:SignatureObject/dss:Timestamp<br/> dss:SignatureObject/dss:SignaturePtr<br/> dss:SignatureObject/dss:Other<br/> werden nicht verwendet.</p> |
| <b>Vorbeding<br/>ungen</b>  | Keine  |  |
| <b>Nachbeding<br/>ungen</b> | Keine  |  |

5444 Der Ablauf der Operation ExternalAuthenticate ist in Tabelle TAB\_KON\_782 beschrieben:



5445 **Tabelle 301: TAB\_KON\_782 Ablauf Operation ExternalAuthenticate**

| Nr. | Aufruf<br>Technischer Use<br>Case oder Interne<br>Operation | Beschreibung   |
|-----|---|--|
| 1.  | checkArguments  | Alle übergebenen Parameterwerte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.   |
| 2.  | TUC_KON_000<br>„Prüfe Zugriffs-<br>berechtigung“            | Die Prüfung erfolgt durch den Aufruf TUC_KON_000 { \$context.mandantId; \$context.clientsystemId; \$context.workplaceId; \$context.userId; \$cardHandle }<br>Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab. |
| 3.  | TUC_KON_026<br>„Liefere<br>CardSession“                     | Ermittle CardSession über TUC_KON_026 { MandantId, CsId, CardHandle, UserId }  |
| 4.  | TUC_KON_218<br>„Signiere“                                   | Signaturberechnung durch Aufruf des TUC_KON_218 {<br>PinRef = PIN.CH bzw. PIN.SMC;<br>KeyRef = PrK.HP.AUT bzw. PrK.HCI.AUT;<br>AlgorithmusID = signPKCS1_V1_5 oder signPSS oder signECDSA;<br>DTBS = Binärstring<br>}                                      |

5446 **Tabelle 302: TAB\_KON\_783 Übersicht Fehler Operation ExternalAuthenticate**

| Fehlercode   | ErrorType | Severity | Fehlertext            |
|--|-----------|----------|-----------------------|
| Neben den Fehlercodes der aufgerufenen TUCs können folgende weitere Fehlercodes auftreten: |           |          |                       |
| 4000   | Technical | Error    | Syntaxfehler          |
| 4058   | Security  | Error    | Aufruf nicht zulässig |

5447 Die folgende Tabelle führt die zulässigen privaten Schlüssel für die Operation  
 5448 ExternalAuthenticate auf:

5449 **Tabelle 303: TAB\_KON\_784 Privater Schlüssel je Karte für ExternalAuthenticate**

| Karte | Schlüssel               |
|-------|-------------------------|
| SM-B  | PrK.HCI.AUT in DF.ESIGN |
| HBAX  | PrK.HP.AUT in DF.ESIGN  |

5450 [**<=**]

#### 5451 **4.1.13.5 Betriebsaspekte**

5452 Keine

#### 4.1.14 Betriebsdatenmeldedienst

##### **A\_21136 - Konnektor:Betriebsdaten - Nutzung der Operation**

###### **I\_Registration\_Service#sendData**

Der Konnektor MUSS die Operation I\_Registration\_Service#sendData benutzen, um die Betriebsdaten täglich zu versenden.

[<=]

##### **A\_21137 - Konnektor:Betriebsdaten - Formatierung der Betriebsdaten**

Der Konnektor MUSS die Betriebsdaten als XML-Dokument gemäß dem Schema „conn/OperatingData.xsd“ mit

- dem MimeType "text/xml" und
- dem Type "OperatingDataConnector"

senden.[<=]

##### **A\_21225 - Konnektor:Betriebsdaten - Annotations im XML-Schema**

Der Konnektor MUSS die XML Elemente der Betriebsdaten gemäß der Annotations im Schema OperatingData.xsd befüllen.[<=]

##### **A\_21138 - Konnektor:Betriebsdaten - Fehlerbehandlung**

Liefert I\_Registration\_Service einen Fehler, MUSS der Konnektor das Senden der Betriebsdaten 3 Mal im Abstand von 5 Minuten erneut versuchen.[<=]

##### **A\_21139 - Konnektor:Betriebsdaten - Soap Operation sendData nicht vorhanden**

Meldet I\_Registration\_Service, dass die Soap Operation sendData nicht vorhanden ist, DARF der Konnektor die Operation NICHT sofort wiederholen. Erst zum nächsten regulären Termin soll wieder gesendet werden.[<=]

##### **A\_21140 - Konnektor:Betriebsdaten - Keine personenbezogenen und medizinischen Daten senden**

Der Konnektor DARF NICHT personenbezogene, personenbeziehbare oder medizinische Daten senden.[<=]

## 4.2 Netzkonnektor

### 4.2.1 Anbindung LAN/WAN

Unter Anbindung LAN/WAN werden die Mechanismen beschrieben, mit denen der Konnektor auf der einen Seite in das lokale Netz der Einsatzumgebung, auf der anderen Seite in die TI bzw. die Bestandsnetze angebunden wird. Diese wesentlichen Aspekte betreffen Routing und Firewall.

Innerhalb des Kapitels Anbindung LAN/WAN werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): „ANLW“
- Konfigurationsparameter: „ANLW\_“

#### 4.2.1.1 Funktionsmerkmalweite Aspekte

##### **TIP1-A\_4723 - Verhalten als IPv4 Router**

Der Konnektor MUSS sich nach den in [RFC1812#1.1.3] definierten Rahmenbedingungen als IP Version 4 (IPv4) Router verhalten.

Hiervon ausgenommen sind die in den folgenden Kapiteln aufgeführten Anforderungen des [RFC1812]:

- 7.2 INTERIOR GATEWAY PROTOCOLS
- 7.3 EXTERIOR GATEWAY PROTOCOLS
- 7.5 FILTERING OF ROUTING INFORMATION
- 7.6 INTER-ROUTING-PROTOCOL INFORMATION EXCHANGE
- 8. APPLICATION LAYER - NETWORK MANAGEMENT PROTOCOLS
- 9. APPLICATION LAYER - MISCELLANEOUS PROTOCOLS
- 10. OPERATIONS AND MAINTENANCE

Die in [RFC2644] geforderten Aktualisierungen zum [RFC1812] müssen vom Konnektor umgesetzt werden.

[<=]

##### **TIP1-A\_5406 - IP-Pakete mit Source Route Option**

Der Konnektor DARF NICHT IP-Pakete mit gesetzter Source Route Option gemäß [RFC791] erzeugen oder weiterleiten.

[<=]

In der folgenden Anforderung wird die Terminologie gemäß [RFC2663] verwendet.

##### **TIP1-A\_5407 - NAT-Umsetzung im Konnektor**

Der Konnektor MUSS für die Kommunikation aus den Adressbereichen NET\_LEKTR-Umgebung mit den Adressbereichen NET\_TI\_OFFENE\_FD und ANLW\_BESTANDSNETZE eine Network Address Port Translation (NAPT) gemäß [RFC3022#2.2, 3, 4.1-4.3] vornehmen.

Für die Umsetzung der Private Local Address aus den Adressbereichen der Einsatzumgebung MUSS die IP-Adresse VPN\_TUNNEL\_TI\_INNER\_IP als Global Address genutzt werden.

Der Konnektor MUSS für die Kommunikation aus den Adressbereichen der NET\_LEKTR-Umgebung mit dem Internet über den VPN-Tunnel SIS eine Network Address Port Translation (NAPT) gemäß RFC3022#2.2, 3, 4.1-4.3 vornehmen. Für die Umsetzung der Local Address MUSS die IP-Adresse VPN\_TUNNEL\_SIS\_INNER\_IP als Global Address genutzt werden.

[<=]

##### **TIP1-A\_4724 - LAN-Adapter**

Der Konnektor MUSS sicherstellen, dass nur über den LAN-Adapter (Adressen aus ANLW\_LAN\_NETWORK\_SEGMENT oder Adressen aus einem der Netzwerksegmente in ANLW\_LEKTR\_INTRANET\_ROUTES) mit den Clientsystemen und den Kartenterminals kommuniziert werden kann.

[<=]

##### **TIP1-A\_4725 - WAN-Adapter**

Für den Betrieb in Reihe (ANLW\_ANBINDUNGS\_MODUS=InReihe) MUSS der Konnektor den WAN-Adapter für den Zugang zum Internet über das IAG der Einsatzumgebung verwenden.

[<=]

**TIP1-A\_4726 - Internet Anbindung nur bei MGM\_LU\_ONLINE**

Der Hersteller des Konnektors MUSS sicherstellen, dass eine Anbindung an das Transportnetz/Internet nur möglich ist, wenn (MGM\_LU\_ONLINE=Enabled) gesetzt ist.  
[<=]

**TIP1-A\_4728 - Nur IPv4. IPv6 nur hardwareseitig vorbereitet**

Der Konnektor MUSS IP Version 4 (IPv4) für alle seine IP-Schnittstellen unterstützen. Die Hardware des Konnektors MUSS für den Einsatz von IPv4 und IPv6 im Dual-Stack-Mode geeignet sein.  
Bis zu einer Migration von IPv4 auf IPv6 MUSS der Konnektor sämtliche empfangene IP-Pakete der Version 6 (IPv6) verwerfen.  
[<=]

**TIP1-A\_4728-01 - IPv4 und IPv6 (Option IPv6)**

Der Konnektor MUSS IP Version 4 (IPv4) und IP Version 6 (IPv6) für alle seine physikalischen Adapter unterstützen.  
Die Hardware des Konnektors MUSS für den Einsatz von IPv4 und IPv6 im Dual-Stack-Mode geeignet sein.  
[<=]

**TIP1-A\_4729 - Es darf kein dynamisches Routing verwendet werden**

Dynamische Routing-Protokolle dürfen vom Konnektor nicht eingesetzt werden. Wird in einem der an den Konnektor angeschlossenen Netzwerke ein dynamisches Routing genutzt, so DÜRFEN Routing Updates vom Konnektor NICHT akzeptiert werden und keine Routen eingetragen werden.  
[<=]

**TIP1-A\_5152 - Aktualisieren der Infrastrukturinformationen aus der TI**

Falls Parameter MGM\_LU\_ONLINE=Enabled, MUSS der Konnektor einmal täglich TUC\_KON\_283 „Infrastruktur Konfiguration aktualisieren“ aufrufen.  
[<=]

**4.2.1.1.1 Netzwerksegmentierung**

In Anlehnung an die in der [gemSpec\_Net#2.3.3] definierten Netzwerksegmente werden in der Konnektorspezifikation die folgenden Bezeichner verwendet:

**Tabelle 304: TAB\_KON\_680 Mapping der Netzwerksegmente**

| ReferenzID im Konnektor | Adressbereich für die TI-Produktivumgebung | Adressbereich für die TI-Testumgebung | Adressbereich für die TI-Referenzumgebung                 |
|-------------------------|--|---------------------------------------|---|
| NET_SIS                 | TI_Dezentral_SIS - Konnektoren             | TI_Test_Dezentral_SIS - Konnektoren   | Ist durch den Testbetriebsverantwortlichen zu definieren. |
| NET_TI_DEZENTRAL        | TI_Dezentral - Konnektoren                 | TI_Test_Dezentral - Konnektoren       | Ist durch den Testbetriebsverantwortlichen zu definieren. |

|                            |  |   |   |
|----------------------------|--|---|---|
| NET_TI_ZENTRAL             | TI_Zentral<br>- Zentrale Dienste   | TI_Test_Zentral<br>- Zentrale Dienste                                     | Ist durch den Testbetriebsverantwortlichen zu definieren. |
| NET_TI_OFFENE_FD           | Anwendungsdienste<br>- Offene Fachdienste<br>- aAdG und aAdG-NetG-TI   | Test_Anwendungsdienste<br>- Offene Fachdienste<br>- aAdG und aAdG-NetG-TI | Ist durch den Testbetriebsverantwortlichen zu definieren. |
| NET_TI_GESICHERTE_FD       | Anwendungsdienste<br>- Gesicherte Fachdienste  | Test_Anwendungsdienste<br>- Gesicherte Fachdienste                        | Ist durch den Testbetriebsverantwortlichen zu definieren. |
| NET_LEKTR                  | Liste der Netzwerke die in der Einsatzumgebung über den Konnektor erreichbar sind. Ein Eintrag der Liste enthält die Netzwerkadresse und den Netzwerkprefix.   |   |   |
| ANLW_BESTANDS_NETZE        | Liste der an die TI angeschlossenen Bestandsnetze (u. a. das Sichere Netz der KVen). Ein Eintrag der Liste enthält die Netzwerkadresse und den Netzwerkprefix. |   |   |
| ANLW_AKTIVE_BESTANDS_NETZE | Liste der an die TI angeschlossenen und aktivierten Bestandsnetze  |   |   |

5568

5569

**Tabelle 305: TAB\_KON\_681 Definition der vom Konnektor verwendeten VPN-Tunnel**

| ReferenzID | Bedeutung/Belegung   |
|------------|--|
| VPN_TI     | Logischer Adapter des VPN-Tunnel zur TI mit dessen VPN_TUNNEL_TI_INNER_IP aus dem Adresssegment NET_TI_DEZENTRAL |
| VPN_SIS    | Logischer Adapter des VPN-Tunnel zur SIS mit dessen VPN_TUNNEL_SIS_INNER_IP aus dem Adresssegment NET_SIS        |

5570

5571

**Tabelle 306: TAB\_KON\_682 Definition der Konnektor IP-Adressen**

| ReferenzID | Bedeutung/Belegung |
|------------|--------------------|
|------------|--------------------|

|                     |  |
|---------------------|--|
| ANLW_LAN_IP_ADDRESS | Dies ist die IP-Adresse des LAN-Adapters.<br>Aus dem Netz der Einsatzumgebung (ANLW_LAN_NETWORK_SEGMENT) die vom Konnektor verwendete IP-Adresse. Unter dieser Adresse werden die Dienste des Konnektor im lokalen Netzwerk bereitgestellt.<br>Diese Adresse entspricht dem in Tabelle TAB_KON_683 LAN-Adapter IP-Konfiguration definierten Parameter ANLW_LAN_IP_ADDRESS. |
| ANLW_WAN_IP_ADDRESS | Dies ist die IP-Adresse des WAN-Adapters.  |

5572 4.2.1.1.2 Routing und Firewall

5573 **Darstellung der Kommunikationsregeln des Konnektors**

5574 Diese Abbildung dient der Veranschaulichung der im Konnektor verwendeten  
5575 Kommunikationsregeln welche in den nachfolgenden Afo definiert werden.

5576

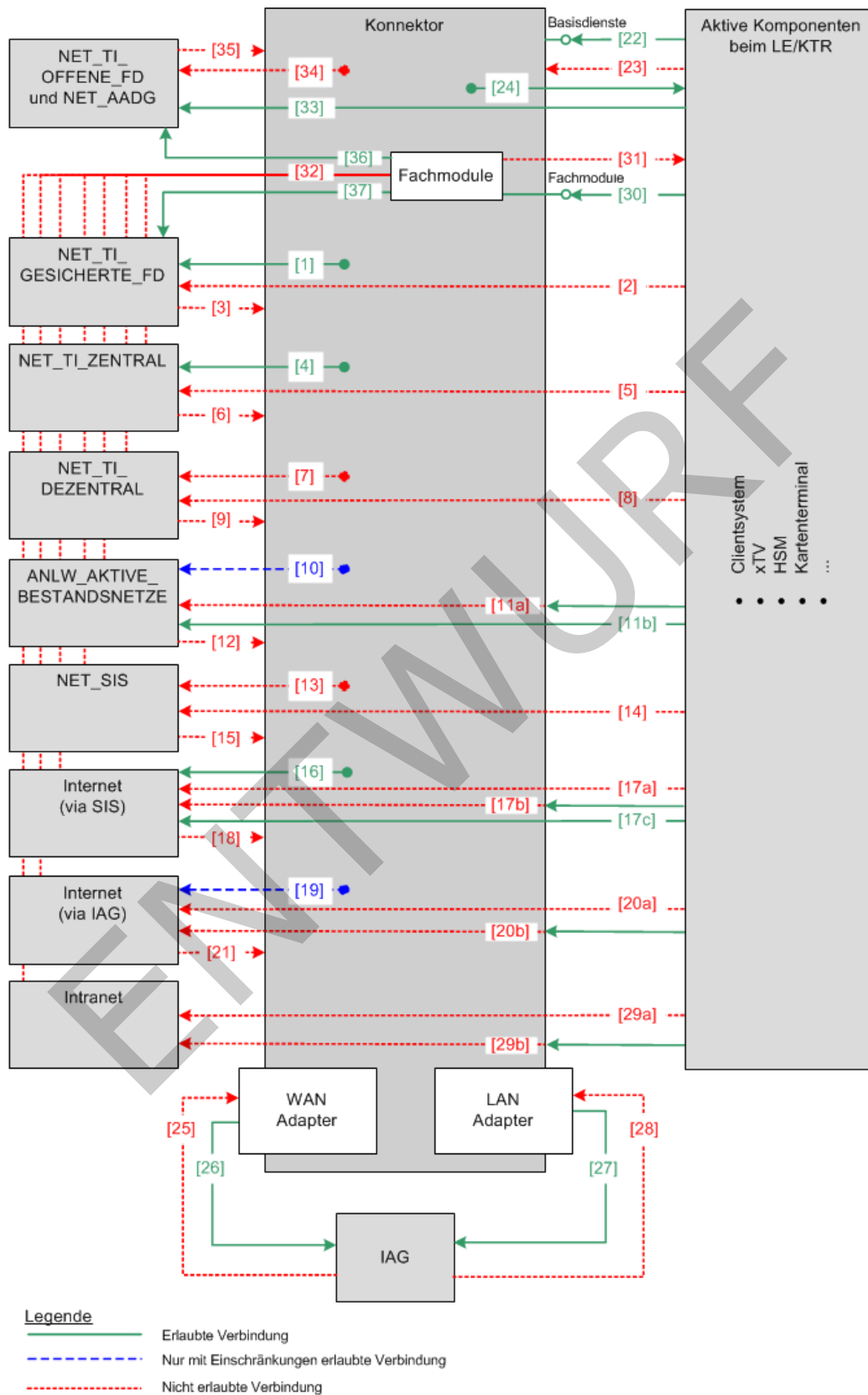


Abbildung 21: PIC\_KON\_115 Kommunikationsregeln Konnektor

5577

5578

**TIP1-A\_4730 - Kommunikation mit NET\_TI\_GESICHERTE\_FD**

Der Konnektor MUSS sicherstellen, dass IP-Pakete mit einer Absenderadresse aus dem Adressbereich NET\_TI\_GESICHERTE\_FD verworfen werden, wenn sie nicht aus dem VPN-Tunnel der TI (VPN\_TI) stammen.

Der Konnektor MUSS die Kommunikation mit Systemen des Netzwerksegments NET\_TI\_GESICHERTE\_FD für folgende Fälle unterstützen:

- [1] vom Konnektor kommend
- [37] wenn (MGM\_LU\_ONLINE=Enabled) vom Fachmodul kommend

Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit Systemen des Netzwerksegments NET\_TI\_GESICHERTE\_FD für folgende Fälle blockieren:

- [2] von „Aktive Komponenten“ kommend
- [3] in Richtung Konnektor gehend

Der Konnektor MUSS sicherstellen, dass die aus einer unterstützten Kommunikation mit Systemen aus dem Netzwerksegment NET\_TI\_GESICHERTE\_FD bestimmten IP-Pakete ausschließlich in den VPN-Tunnel der TI (VPN\_TI) geleitet werden.

[<=]

**TIP1-A\_5530 - Kommunikation mit NET\_TI\_OFFENE\_FD**

Der Konnektor MUSS sicherstellen, dass IP-Pakete mit einer Absenderadresse aus dem Adressbereich NET\_TI\_OFFENE\_FD verworfen werden, wenn sie nicht aus dem VPN-Tunnel der TI (VPN\_TI) stammen.

Der Konnektor MUSS die Kommunikation mit Systemen des Netzwerksegments NET\_TI\_OFFENE\_FD für folgende Fälle unterstützen:

- [33] von „Aktive Komponenten“ kommend
- [36] vom Fachmodul kommend

Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit Systemen des Netzwerksegments NET\_TI\_OFFENE\_FD für folgende Fälle blockieren:

- [34] vom Konnektor kommend
- [35] in Richtung Konnektor gehend

Der Konnektor MUSS sicherstellen, dass die aus einer unterstützten Kommunikation mit Systemen aus dem Netzwerksegment NET\_TI\_OFFENE\_FD bestimmten IP-Pakete ausschließlich in den VPN-Tunnel der TI (VPN\_TI) geleitet werden.

[<=]

**TIP1-A\_4731 - Kommunikation mit NET\_TI\_ZENTRAL**

Der Konnektor MUSS sicherstellen, dass IP-Pakete mit einer Absenderadresse aus dem Adressbereich NET\_TI\_ZENTRAL verworfen werden, wenn sie nicht aus dem VPN-Tunnel der TI (VPN\_TI) stammen.

Der Konnektor MUSS die Kommunikation mit Systemen des Netzwerksegments NET\_TI\_ZENTRAL für folgende Fälle unterstützen:

- [4] vom Konnektor kommend

Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit Systemen des Netzwerksegments NET\_TI\_ZENTRAL für folgende Fälle blockieren:

- [5] von „Aktive Komponenten“ kommend
- [6] in Richtung Konnektor gehend

Der Konnektor MUSS sicherstellen, dass die aus einer unterstützten Kommunikation mit Systemen aus dem Netzwerksegment NET\_TI\_ZENTRAL bestimmten IP-Pakete



5624 ausschließlich in den VPN-Tunnel der TI (VPN\_TI) geleitet werden.

5625 [ $\leq$ ]

5626 **TIP1-A\_4732 - Kommunikation mit NET\_TI\_DEZENTRAL**

5627 Der Konnektor MUSS sicherstellen, dass die Adressen aus dem Adressbereich  
5628 NET\_TI\_DEZENTRAL nur für die Kommunikation mit der TI/den weiteren Anwendungen  
5629 des Gesundheitswesens in Form der inner IP (VPN\_TUNNEL\_TI\_INNER\_IP) des VPN-  
5630 Tunnel der TI (VPN\_TI) verwendet wird.

5631 Der Konnektor MUSS die Kommunikation mit Systemen des Netzwerksegments  
5632 NET\_TI\_DEZENTRAL für folgende Fälle unterstützen:

- 5633
- keine

5634 Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit  
5635 Systemen des Netzwerksegments NET\_TI\_DEZENTRAL für folgende Fälle blockieren:

- 5636
- [7] vom Konnektor kommend (zur Verhinderung des Zugriffs auf fremde  
5637 Konnektoren)

- 5638
- [8] von „Aktive Komponenten“

- 5639
- [9] in Richtung Konnektor gehend

5640 [ $\leq$ ]

5641 **TIP1-A\_4733 - Kommunikation mit ANLW\_AKTIVE\_BESTANDSNETZE**

5642 Der Konnektor MUSS sicherstellen, dass IP-Pakete mit einer Absenderadresse aus dem  
5643 Adressbereich ANLW\_AKTIVE\_BESTANDSNETZE verworfen werden, wenn sie nicht aus  
5644 dem VPN-Tunnel der TI (VPN\_TI) stammen.

5645 Der Konnektor MUSS die Kommunikation mit Systemen des Netzwerksegments  
5646 ANLW\_AKTIVE\_BESTANDSNETZE für folgende Fälle unterstützen:

- 5647
- [10] wenn (MGM\_LU\_ONLINE=Enabled ) vom Konnektor kommend nur für die  
5648 DNS-Namensauflösung mittels DNS\_SERVERS\_BESTANDSNETZE

- 5649
- [11b] wenn (MGM\_LU\_ONLINE=Enabled) von „Aktive Komponenten“ kommend

5650 Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit  
5651 Systemen des Netzwerksegments ANLW\_AKTIVE\_BESTANDSNETZE für folgende Fälle  
5652 blockieren:

5653

- 5654
- [11a] für nicht freigegebene angeschlossene Netze des Gesundheitswesens mit  
5655 aAdG-NetG (ANLW\_BESTANDSNETZE abzüglich ANLW\_AKTIVE\_BESTANDSNETZE)  
5656 von „Aktive Komponenten“ kommend;

- 5657
- [12] in Richtung Konnektor gehend (und den dahinterliegenden „Aktive  
5658 Komponenten“)

5659 Der Konnektor MUSS sicherstellen, dass die aus einer unterstützten Kommunikation mit  
5660 Systemen aus dem Netzwerksegment ANLW\_AKTIVE\_BESTANDSNETZE bestimmten IP-  
5661 Pakete ausschließlich in den VPN-Tunnel der TI (VPN\_TI) geleitet werden.

5662 [ $\leq$ ]

5663 **TIP1-A\_4734 - Kommunikation mit NET\_SIS**

5664 Der Konnektor MUSS sicherstellen, dass eine Adresse aus dem Adressbereich NET\_SIS  
5665 nur für die Kommunikation mit dem Internet (via SIS) in Form der inner IP  
5666 (VPN\_TUNNEL\_SIS\_INNER\_IP) des VPN-Tunnel der SIS (VPN\_SIS) verwendet wird.

5667 Der Konnektor MUSS insbesondere die Kommunikation mit Systemen des  
5668 Netzwerksegments NET\_SIS für folgende Fälle unterstützen:

- 5669
- keine

5670 Der Konnektor MUSS die Kommunikation an seinen Außenschnittstellen mit NET\_SIS für  
5671 folgende Fälle blockieren:

- 5672 • [13] vom Konnektor kommend
- 5673 • [14] von „Aktive Komponenten“ kommend
- 5674 • [15] in Richtung Konnektor gehend (und den dahinterliegenden „Aktiven  
5675 Komponenten“)

5676 [**<=**]

5677 **TIP1-A\_4735 - Kommunikation mit dem Internet (via SIS)**

5678 Der Konnektor MUSS sicherstellen, dass IP-Pakete mit einer Absenderadresse aus dem  
5679 Adressbereich NET\_TI\_ZENTRAL, NET\_TI\_GESICHERTE\_FD; NET\_TI\_OFFENE\_FD,  
5680 NET\_TI\_DEZENTRAL, ANLW\_AKTIVE\_BESTANDSNETZE,  
5681 ANLW\_LAN\_ADDRESS\_SEGMENT, aus einem der Netzwerksegmente in  
5682 ANLW\_LEKTR\_INTRANET\_ROUTES oder ANLW\_WAN\_NETWORK\_SEGMENT verworfen  
5683 werden, wenn sie aus dem VPN-Tunnel der SIS (VPN\_SIS) stammen.  
5684 Der Konnektor MUSS die Kommunikation mit Systemen des Netzwerksegments Internet  
5685 (via SIS) für folgende Fälle unterstützen:

- 5686 • [16] wenn (MGM\_LU\_ONLINE=Enabled und ANLW\_INTERNET\_MODUS=SIS)  
5687 vom Konnektor kommend
- 5688 • [17c] wenn (MGM\_LU\_ONLINE=Enabled und ANLW\_INTERNET\_MODUS=SIS)  
5689 von „Aktive Komponenten“ kommend

5690 Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit  
5691 Internet (via SIS) für folgende Fälle blockieren oder umleiten:

- 5692 • [17a] blockieren, wenn (MGM\_LU\_ONLINE=Enabled und  
5693 ANLW\_INTERNET\_MODUS=KEINER) von „Aktive Komponenten“ kommend
- 5694 • [17b] umleiten, wenn (MGM\_LU\_ONLINE=Enabled und  
5695 ANLW\_INTERNET\_MODUS=IAG) von „Aktive Komponenten“ kommend;  
5696 ➔ Der Konnektor MUSS an Hosts im Internet gerichtete IP-Pakete gemäß  
5697 [RFC792] umleiten (ICMP Redirect).
- 5698 • [18] blockieren, wenn von SIS kommend in Richtung Konnektor (und die  
5699 dahinterliegenden „Aktive Komponenten“)

5700 Der Konnektor MUSS sicherstellen, dass die für die Kommunikation mit dem  
5701 Internet (via SIS) bestimmten IP-Pakete ausschließlich in den VPN-Tunnel des SIS  
5702 (VPN\_SIS) geleitet werden.

5703 [**<=**]

5704 **TIP1-A\_4736-02TIP1-A\_4736 - Kommunikation mit dem Internet (via IAG)**

5705 Der Konnektor MUSS sicherstellen, dass eingehende IP-Pakete von der Kommunikation  
5706 mit dem Internet mit der Empfängeradresse ungleich (ANLW\_LAN\_IP\_ADDRESS oder aus  
5707 einem der Netzwerksegmente in ANLW\_LEKTR\_INTRANET\_ROUTES wenn  
5708 ANLW\_WAN\_ADAPTER\_MODUS=DISABLED) oder (ANLW\_WAN\_IP\_ADDRESS wenn  
5709 ANLW\_WAN\_ADAPTER\_MODUS=ENABLED) verworfen werden.

5710 Der Konnektor MUSS sicherstellen, dass ausgehende IP-Pakete für die Kommunikation  
5711 mit dem Internet mit der Absenderadresse ungleich (ANLW\_LAN\_IP\_ADDRESS oder aus  
5712 einem der Netzwerksegmente in ANLW\_LEKTR\_INTRANET\_ROUTES wenn  
5713 ANLW\_WAN\_ADAPTER\_MODUS=DISABLED) oder (ANLW\_WAN\_IP\_ADDRESS wenn  
5714 ANLW\_WAN\_ADAPTER\_MODUS=ENABLED) verworfen werden.

5715 Der Konnektor MUSS die Kommunikation mit Systemen des Netzwerksegments Internet  
5716 (via IAG) für folgende Fälle unterstützen:

- 5717 • [19] vom Konnektor kommend zu den folgenden Systemen für das Protokoll IPsec
- 5718 • VPN\_KONZENTRATOR\_TI\_IP\_ADDRESS
- 5719 • VPN\_KONZENTRATOR\_SIS\_IP\_ADDRESS
- 5720 • [19] vom Konnektor kommend zu den folgenden Systemen für HTTP und HTTPS
- 5721 • CERT\_CRL\_DOWNLOAD\_ADDRESS
- 5722 • TSL-Download-Punkt des TSL-Dienstes
- 5723 • hash&URL-Server
- 5724 • Registrierungsserver
- 5725 • Remote-Managementserver
- 5726 • DNS\_ROOT\_ANCHOR\_URL (benötigte IP-Adressen um den DNSSEC Trust
- 5727 Anchor im Namensraum Internet zu verifizieren)
- 5728 • [19] vom Konnektor kommend zu den folgenden Systemen für das Protokoll DNS
- 5729 • beliebige Hosts

5730 Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit  
5731 Internet (via IAG) für folgende Fälle blockieren oder umleiten:

- 5732 • [20a] blockieren, wenn (ANLW\_INTERNET\_MODUS=KEINER oder
- 5733 MGM\_LU\_ONLINE=Disabled ) von „Aktive Komponenten“ kommend
- 5734 • [20b] mittels ICMP Redirect gemäß [RFC792] zum Default Gateway umleiten,
- 5735 wenn die Zieladresse des IP-Pakets nicht innerhalb der Adressbereiche
- 5736 (NET\_TI\_ZENTRAL, NET\_TI\_OFFENE\_FD, NET\_TI\_GESICHERTE\_FD und
- 5737 ANLW\_AKTIVE\_BESTANDSNETZE) ist und ANLW\_INTERNET\_MODUS=IAG und von
- 5738 „Aktive Komponenten“ kommend.
- 5739 • [21] blockieren, wenn von IAG kommend in Richtung Konnektor (und die
- 5740 dahinterliegenden „Aktive Komponenten“)

5741 [**<=**]

5742

5743

5744

## 5745 **TIP1-A\_4737 - Kommunikation mit „Aktive Komponenten“**

5746 Der Konnektor MUSS sicherstellen, dass ausgehende IP-Pakete für die Kommunikation  
5747 mit „Aktive Komponenten“ mit einer Absenderadresse ungleich ANLW\_LAN\_IP\_ADDRESS,  
5748 einer Adresse aus einem Netzwerksegment in ANLW\_LEKTR\_INTRANET\_ROUTES oder  
5749 0.0.0.0 verworfen werden.

5750 Der Konnektor MUSS die Kommunikation mit „Aktive Komponenten“ für folgende Fälle  
5751 unterstützen:

- 5752 • [22] auf den Konnektor (mittels der Schnittstelle Basisdienste)
- 5753 • [24] vom Konnektor kommend

5754 Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit  
5755 „Aktive Komponenten“ für folgende Fälle blockieren:

- 5756 • [23] zum Konnektor eingehend (direkt – ohne eine der Schnittstellen Fachmodule
- 5757 oder Basisdienste zu nutzen)

5758 [**<=**]

**TIP1-A\_4738 - Route zum IAG**

Der Konnektor MUSS die Kommunikation mit dem IAG der Einsatzumgebung für folgende Fälle unterstützen:

- [26] wenn (ANLW\_WAN\_ADAPTER\_MODUS=ENABLED) vom WAN-Adapter kommend
- [27] wenn (ANLW\_WAN\_ADAPTER\_MODUS=DISABLED) vom LAN-Adapter kommend

Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit dem IAG der Einsatzumgebung für folgende Fälle blockieren:

- [25] wenn (ANLW\_WAN\_ADAPTER\_MODUS=ENABLED) zum WAN-Adapter eingehend
- [28] wenn (ANLW\_WAN\_ADAPTER\_MODUS=DISABLED) zum LAN-Adapter eingehend

[<=]

**TIP1-A\_4740 - Admin Defined Firewall Rules**

Die Firewall des Konnektor MUSS alle vom Administrator in ANLW\_FW\_SIS\_ADMIN\_RULES definierten Firewall-Regeln als zusätzliche Einschränkung übernehmen.

[<=]

**TIP1-A\_4741 - Kommunikation mit dem Intranet**

Der Konnektor MUSS die Kommunikation mit Systemen aus einem Intranet-VPN (einem der Netzwerksegmente ANLW\_LEKTR\_INTRANET\_ROUTES) für folgende Fälle unterstützen:

- [22] wenn von Aktive Komponenten aus dem Netzwerksegment ANLW\_LEKTR\_INTRANET\_ROUTES kommend zum Konnektor mittels der Schnittstelle Basisdienste
- [24] wenn vom Konnektor kommend zu ANLW\_LEKTR\_INTRANET\_ROUTES
- Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit einem der Intranet Netzwerksegmente für folgende Fälle blockieren bzw. umleiten:
  - [29a] blockieren, wenn (ANLW\_INTRANET\_ROUTES\_MODUS=BLOCK) vom „Aktive Komponenten“ kommend;
  - [29b] umleiten, wenn (ANLW\_INTRANET\_ROUTES\_MODUS=REDIRECT) vom „Aktive Komponenten“ kommend;
    - ➔ Der Konnektor MUSS an ANLW\_LEKTR\_INTRANET\_ROUTES gerichtete IP-Pakete gemäß [RFC792] umleiten (ICMP Redirect).

[<=]

**TIP1-A\_4742 - Kommunikation mit den Fachmodulen**

Der Konnektor MUSS die Kommunikation mit den Fachmodulen für folgende Fälle unterstützen:

- [30] von „Aktive Komponenten“ über Schnittstelle Fachmodule

Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit den Fachmodulen für folgende Fälle blockieren:

- [31] zu „Aktive Komponenten“

- 5803 • [32] zu den Netzwerksegmenten, NET\_TI\_ZENTRAL, NET\_TI\_DEZENTRAL,  
5804 ANLW\_AKTIVE\_BESTANDSNETZE, Internet (via SIS), Internet (via IAG) und  
5805 Intranet

5806 [ $\leq$ ]

5807 **TIP1-A\_4744 - Firewall - Drop statt Reject**

5808 Die Firewall des Konnektor MUSS alle abgelehnten IP-Pakete verwerfen (DROP) ohne ein  
5809 ICMP-Destination-Unreachable (Type 3) zu schicken.

5810 [ $\leq$ ]

5811 **TIP1-A\_4746 - Firewall – Abwehr von IP-Spoofing, DoS/DDoS-Angriffe und**  
5812 **Martian Packets**

5813 Der Konnektor MUSS geeignete technische Funktionen zur Abwehr von IP-Spoofing und  
5814 DoS/DDoS-Angriffen implementieren.

5815 Der Konnektor MUSS Martian Packets (Absender- oder Empfängeradressen aus den von  
5816 der IETF als Special-Purpose definierten Netzbereichen), mindestens jedoch aus  
5817 folgenden Netzbereichen 0.0.0.0/8, 127.0.0.0/8, 169.254.0.0/16, 192.0.0.0/24,  
5818 192.0.2.0/24, 198.18.0.0/15, 198.51.100.0/24, 203.0.113.0/24, 224.0.0.0/4,  
5819 240.0.0.0/4 verwerfen. Die in [RFC1918] und [RFC 6598] definierten Netzbereiche sind  
5820 hiervon ausgenommen.

5821 [ $\leq$ ]

5822 **TIP1-A\_4745 - Eingeschränkte Nutzung von „Ping“**

5823 Die Firewall des Konnektor MUSS TCP-Port-7(Echo)-Pakete verwerfen.

5824 Die Firewall des Konnektor MUSS ICMP-Echo-Request (Typ 8) und ICMP-Echo-Response  
5825 (Typ 0) ausschließlich für die folgenden Kommunikationen zulassen:

- 5826 • vom Konnektor zu den VPN-Konzentratoren für SIS und TI über das Transportnetz  
5827 (via IAG)
- 5828 • vom Konnektor zu dem CRL-Webservern (im Transportnetz) über das Internet  
5829 (via SIS) und das Transportnetz (via IAG)
- 5830 • vom Konnektor zu dem IAG der Einsatzumgebung
- 5831 • vom Konnektor zu NET\_TI\_ZENTRAL
- 5832 • vom Konnektor zu NET\_TI\_GESICHERTE\_FD
- 5833 • vom Konnektor zu NET\_TI\_OFFENE\_FD
- 5834 • vom Konnektor zum lokalen Netzwerk (Adressen aus  
5835 ANLW\_LAN\_NETWORK\_SEGMENT oder Adressen aus einem der  
5836 Netzwerksegmente in ANLW\_LEKTR\_INTRANET\_ROUTES)
- 5837 • vom lokalen Netzwerk (Adressen aus ANLW\_LAN\_NETWORK\_SEGMENT (jedoch  
5838 ohne die ANLW\_LAN\_IP\_ADDRESS) oder Adressen aus einem der  
5839 Netzwerksegmente in ANLW\_LEKTR\_INTRANET\_ROUTES) zum Konnektor
- 5840 • vom lokalen Netzwerk in ANLW\_AKTIVE\_BESTANDSNETZE (die freigegebenen  
5841 angeschlossenen Netze des Gesundheitswesens mit aAdG-NetG)
- 5842 • vom lokalen Netzwerk in das Internet (via SIS)

5843 Die Firewall des Konnektors MUSS für alle anderen Kommunikationen ein ICMP-  
5844 Echo-Request (Typ 8) verwerfen.

5845 [ $\leq$ ]

5846 **TIP1-A\_4747 - Firewall – Einschränkungen der IP-Protokolle**

5847 Der Konnektor MUSS alle IP-Protokolle außer 1 (ICMP), 4 (IP in IP (encapsulation)), 17  
5848 (UDP), 6 (TCP), 50 (ESP) und 108 (IPComp) für alle ein- oder ausgehenden Pakete an



5849 allen seinen Adaptern verwerfen.

5850 [ $\leq$ ]

## 5851 **TIP1-A\_4748 - Firewall – Routing-Regeln**

5852 Der Konnektor DARF seine Routing-Regeln NICHT durch IP-Kommunikation beeinflussen  
5853 lassen, weder mittels eines Routing-Protokolls (wie BGP oder RIP) noch mittels ICMP-  
5854 Kommandos (wie Redirect (5), Router Advertisement (9/10) oder auch Mobile Host  
5855 Redirect (32)) sondern MUSS diese ausschließlich durch TUC\_KON\_304 „Netzwerk-  
5856 Routen einrichten“ setzen.

5857 Die Firewall des Konnektor MUSS alle aus einem der Tunnel (VPN\_TI oder VPN\_SIS)  
5858 kommenden DHCP-Pakete verwerfen.

5859 Die Firewall des Konnektors MUSS an den Konnektor gerichtete IPsec-Pakete (IKE, ESP  
5860 und IPsec NAT-T) verwerfen, sofern sie nicht einer vom Konnektor initiierten IPsec-  
5861 Verbindung (VPN\_TI und VPN\_SIS) zugeordnet werden können.

5862 [ $\leq$ ]

## 5863 **TIP1-A\_4749 - Firewall Restart**

5864 Der Konnektor MUSS gewährleisten, dass unmittelbar nach einer Änderung der  
5865 Parameter eines Adapters (LAN-Adapter, WAN-Adapter, virtueller Adapter VPN\_TI oder  
5866 virtueller Adapter VPN\_SIS) die Firewall des Konnektor neu erstellt und geladen wird.

5867 Wenn der WAN-Adapter verwendet wird (ANLW\_WAN\_ADAPTER\_MODUS=ENABLED)  
5868 DARF die Firewall des Konnektor bei einer Änderung der ANLW\_WAN\_IP\_ADDRESS  
5869 NICHT die Verbindungen über den LAN-Adapter durch einen Restart der Firewall  
5870 beeinflussen.

5871 Wenn der WAN-Adapter verwendet wird (ANLW\_WAN\_ADAPTER\_MODUS=ENABLED),  
5872 DARF die Firewall des Konnektor bei einer Änderung der ANLW\_LAN\_IP\_ADDRESS NICHT  
5873 die Verbindungen über die Adapter WAN, VPN\_TI oder VPN\_SIS durch einen Restart der  
5874 Firewall beeinflussen.

5875 [ $\leq$ ]

5876 Umsetzungshinweis für den Hersteller: Es können zwei getrennten Firewall-Regelsets für  
5877 den LAN- bzw. für den WAN-Adapter verwendet werden.

## 5878 **TIP1-A\_4750 - Firewall-Protokollierung**

5879 Der Konnektor MUSS bei Start und Stopp der Firewall einen Protokolleintrag mit der  
5880 Schwere „Warning“ und dem Typ „Operations“ sowie mindestens folgenden  
5881 Informationen generieren:

- 5882 • Zeitstempel, Aktion (Start/Stop), Ergebnis (Erfolg/Fehler), Auslöser
- 5883 (Prozess/User)

5884 Der Konnektor MUSS bei Konfigurationsänderungen der Firewall einen Protokolleintrag  
5885 mit der Schwere „Warning“ und dem Typ „Operations“ sowie mindestens folgenden  
5886 Informationen generieren:

- 5887 • Zeitstempel, Aktion (Add/Delete/Change), Details (Beschreibung der Änderung),
- 5888 Auslöser (Prozess/User)

5889 Der Konnektor MUSS für alle vom Konnektor ausgehenden, nicht zugelassenen  
5890 Kommunikationsversuche einen Protokolleintrag mit der Schwere „Warning“ und dem Typ  
5891 „Security“ sowie mindestens folgenden Informationen generieren:

- 5892 • Zeitstempel, Aktion (Drop, Reject), Absender-IP-Adresse, Empfänger-IP-Adresse,
- 5893 Protokoll, Absender-Port und Empfänger-Port, Interface über das das Paket
- 5894 empfangen wurde

5895 Der Konnektor MUSS für alle verworfenen IP-Spoofing- und Martian-Packets einen  
5896 Protokolleintrag mit der Schwere „Warning“ und dem Typ „Security“ sowie mindestens  
5897 folgenden Informationen generieren:

5898 • Zeitstempel, Aktion (Drop, Reject), Absender-IP-Adresse, Empfänger-IP-Adresse,  
5899 Protokoll, Absender-Port und Empfänger-Port, Interface über das das Paket  
5900 empfangen wurde

5901 Der Konnektor MUSS für alle von der Firewall verworfenen IP-Pakete einen  
5902 Protokolleintrag mit der Schwere „Info“ und dem Typ „Security“ sowie mindestens  
5903 folgenden Informationen generieren, wobei Layer 3 Broadcasts von der Protokollierung  
5904 ausgenommen werden können:

5905 • Zeitstempel, Aktion (Drop, Reject), Absender-IP-Adresse, Empfänger-IP-Adresse,  
5906 Protokoll, Absender-Port und Empfänger-Port, Interface über das das Paket  
5907 empfangen wurde

5908 Der Konnektor MUSS für die Firewall-Protokollierung den TUC\_KON\_271 „Schreibe  
5909 Protokolleintrag“ nutzen.  
5910 [ $\leq$ ]

## 5911 4.2.1.2 Durch Ereignisse ausgelöste Reaktionen

### 5912 TIP1-A\_4751 - Reagiere auf LAN\_IP\_Changed

5913 Beim Auftreten eines der nachfolgenden Events MUSS der Konnektor den TUC\_KON\_305  
5914 „LAN-Adapter initialisieren“ starten.

- 5915 • Event ANLW/LAN/IP\_CHANGED
- 5916 • Event DHCP/LAN\_CLIENT/RENEW

5917 [ $\leq$ ]

### 5918 TIP1-A\_4752 - Reagiere auf WAN\_IP\_Changed

5919 Beim Auftreten eines der nachfolgenden Events MUSS der Konnektor TUC\_KON\_306  
5920 „WAN-Adapter initialisieren“ starten.

- 5921 • Event ANLW/WAN/IP\_CHANGED
- 5922 • Event DHCP/WAN\_CLIENT/RENEW

5923 [ $\leq$ ]

### 5924 TIP1-A\_4753 - Ereignisbasiert Netzwerkrouen einrichten

5925 Beim Auftreten eines der nachfolgenden Events MUSS der Konnektor den TUC\_KON\_304  
5926 „Netzwerk-Routen einrichten“ aufrufen.

- 5927 • Event NETWORK/VPN\_TI/UP
- 5928 • Event NETWORK/VPN\_TI/DOWN
- 5929 • Event NETWORK/VPN\_SIS/UP
- 5930 • Event NETWORK/VPN\_SIS/DOWN
- 5931 • Event MGM/LU\_CHANGED/LU\_ONLINE

5932 [ $\leq$ ]

5933 **4.2.1.3 Interne TUCs, nicht durch Fachmodule nutzbar**

## 5934 4.2.1.3.1 TUC\_KON\_305 „LAN-Adapter initialisieren“

5935 **TIP1-A\_4754 - TUC\_KON\_305 „LAN-Adapter initialisieren“**

5936 Der Konnektor MUSS den technischen Use Case TUC\_KON\_305 „LAN-Adapter  
5937 initialisieren“ umsetzen.

5938

5939 **Tabelle 307: TAB\_KON\_614 - TUC\_KON\_305 „LAN-Adapter initialisieren“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_305 LAN-Adapter initialisieren  |
| Beschreibung   | Initialisieren der LAN-Netzwerkschnittstelle  |
| Auslöser       | <ul style="list-style-type: none"> <li>• Event ANLW/LAN/IP_CHANGED</li> <li>• Event DHCP/LAN_CLIENT/RENEW; BOOTUP</li> </ul>  |
| Vorbedingungen | <ul style="list-style-type: none"> <li>• Wenn die IP-Konfiguration des LAN-Adapters statisch (DHCP_CLIENT_LAN_STATE=Disabled) gesetzt wird, MUSS der Konnektor gewährleisten, dass alle Konfigurationsparameter gemäß „Tabelle TAB_KON_683 LAN-Adapter IP-Konfiguration,“ vorab über die Managementschnittstelle gesetzt wurden.</li> <li>• Wenn die IP-Konfiguration des LAN-Adapters dynamisch per DHCP (DHCP_CLIENT_LAN_STATE=Enabled) gesetzt wird, MUSS der DHCP-Client diese vorab gesetzt haben.</li> </ul>  |
| Eingangsdaten  | Keine   |
| Komponenten    | Konnektor   |
| Ausgangsdaten  | Keine   |
| Standardablauf | <p>1) Die in „Tabelle TAB_KON_683 LAN-Adapter IP-Konfiguration,“ und „Tabelle TAB_KON_684 LAN-Adapter Erweiterte Parameter „ gesetzten Werte sind zur Konfiguration des LAN-Adapters zu verwenden.</p> <p>2) Rufe TUC_KON_304 „Netzwerk-Routen einrichten“</p> <p>3) Wenn (ANLW_WAN_ADAPTER_MODUS = DISABLED) und MGM_LU_ONLINE = ENABLED:</p> <ul style="list-style-type: none"> <li>• Rufe TUC_KON_321 „Verbindung zu dem VPN-Konzentrator der TI aufbauen“.</li> <li>• Rufe TUC_KON_322 „Verbindung zu dem VPN-Konzentrator des SIS aufbauen“</li> </ul> |



|                                   |  |
|-----------------------------------|--|
|                                   | 4) Firewall-Regeln aktualisieren und aktivieren. Tritt der Fehler 4164 auf, geht der Konnektor in den Betriebszustand EC_Firewall_Not_Reliable über. |
| Varianten/<br>Alternativen        | Keine  |
| Fehlerfälle                       | (→ 1) Fehlerhafte LAN IP-Konfiguration; 4162<br>(→ 4) Beim Aktualisieren oder Aktivieren der Firewall-Regeln ist es zu einem Fehler gekommen; 4164   |
| Nichtfunktionale<br>Anforderungen | Keine  |
| Zugehörige<br>Diagramme           | Keine  |

5940 **Tabelle 308: TAB\_KON\_615 Fehlercodes TUC\_KON\_305 „LAN-Adapter initialisieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4162  | Technical | Error    | Es liegt eine fehlerhafte LAN IP-Konfiguration vor.                                     |
| 4164  | Technical | Fatal    | Beim Aktualisieren oder Aktivieren der Firewall-Regeln ist es zu einem Fehler gekommen. |

5941 [**<=**]

5942 **4.2.1.3.2 TUC\_KON\_306 „WAN-Adapter initialisieren“**

5943 **TIP1-A\_4755 - TUC\_KON\_306 „WAN-Adapter initialisieren“**

5944 Der Konnektor MUSS den technischen Use Case TUC\_KON\_306 „WAN-Adapter  
5945 initialisieren“ umsetzen.

5946

5947 **Tabelle 309: TAB\_KON\_616 - TUC\_KON\_306 „WAN-Adapter initialisieren“**

| Element | Beschreibung                           |
|---------|--|
| Name    | TUC_KON_306 WAN-Adapter initialisieren |

|                                   |  |
|-----------------------------------|--|
| Beschreibung                      | Initialisieren der WAN-Netzwerkschnittstelle   |
| Auslöser                          | <ul style="list-style-type: none"> <li>Event ANLW/WAN/IP_CHANGED</li> <li>Event DHCP/WAN_CLIENT/RENEW; BOOTUP</li> </ul>   |
| Vorbedingungen                    |  |
| Eingangsdaten                     | Keine  |
| Komponenten                       | Konnektor  |
| Ausgangsdaten                     | Keine  |
| Standardablauf                    | <p>1) Wenn ANLW_WAN_ADAPTER_MODUS = DISABLED oder MGM_LU_ONLINE = Disabled:</p> <p>a) Aktive VPN-Tunnel TI oder SIS (VPN_TI oder VPN_SIS) müssen gestoppt werden,</p> <p>2) Wenn ANLW_WAN_ADAPTER_MODUS = ENABLED und MGM_LU_ONLINE = ENABLED:</p> <p>a) Der WAN-Adapter wird abhängig von DHCP_CLIENT_WAN_STATE statisch oder dynamisch über DHCP konfiguriert. Die in „Tabelle TAB_KON_685 WAN-Adapter IP-Konfiguration,“ und „Tabelle TAB_KON_686 WAN-Adapter Erweiterte Parameter,“ gesetzten Werte sind zur Konfiguration des WAN-Adapter zu verwenden.</p> <p>b) Rufe TUC_KON_304 „Netzwerk-Routen einrichten“</p> <p>c) Rufe TUC_KON_321 „Verbindung zu dem VPN-Konzentrator der TI aufbauen“.</p> <p>d) Rufe TUC_KON_322 „Verbindung zu dem VPN-Konzentrator des SIS aufbauen“</p> <p>e) Firewall-Regeln aktualisieren und aktivieren. Tritt der Fehler 4164 auf, geht der Konnektor in den Betriebszustand EC_Firewall_Not_Reliable über.</p> |
| Varianten/<br>Alternativen        | Keine  |
| Fehlerfälle                       | <p>(→ 1) Fehlerhafte WAN IP-Konfiguration; 4163</p> <p>(→ 2) Beim Aktualisieren oder Aktivieren der Firewall-Regeln ist es zu einem Fehler gekommen; 4164</p>  |
| Nichtfunktionale<br>Anforderungen | Keine  |

5948 **Tabelle 310: TAB\_KON\_617 Fehlercodes TUC\_KON\_306 „WAN-Adapter initialisieren“**

| Fehlercode | ErrorType | Severity | Fehlertext |
|------------|-----------|----------|------------|
|------------|-----------|----------|------------|

|   |           |       |   |
|---|-----------|-------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |       |   |
| 4163  | Technical | Error | Es liegt eine fehlerhafte WAN-IP-Konfiguration vor.                                     |
| 4164  | Technical | Fatal | Beim Aktualisieren oder Aktivieren der Firewall-Regeln ist es zu einem Fehler gekommen. |

5949 [ $\leq$ ]

5950 4.2.1.3.3 TUC\_KON\_304 „Netzwerk-Routen einrichten“

5951 **TIP1-A\_4758 - TUC\_KON\_304 „Netzwerk-Routen einrichten“**

5952 Der Konnektor MUSS den technischen Use Case TUC\_KON\_304 „Netzwerk-Routen einrichten“ umsetzen.

5953

5954

5955 **Tabelle 311: TAB\_KON\_622 - TUC\_KON\_304 „Netzwerk-Routen einrichten“**

| Element         | Beschreibung  |
|-----------------|---|
| Name            | TUC_KON_304 Netzwerk-Routen einrichten  |
| Beschreibung    | Anpassen der Routing-Tabelle  |
| Auslöser        | <ul style="list-style-type: none"> <li>TUC_KON_305 „LAN-Adapter initialisieren“</li> <li>TUC_KON_306 „WAN-Adapter initialisieren“</li> <li>Event NETWORK/VPN_TI/UP</li> <li>Event NETWORK/VPN_TI/DOWN</li> <li>Event NETWORK/VPN_SIS/UP</li> <li>Event NETWORK/VPN_SIS/DOWN</li> <li>Event MGM/LU_CHANGED/LU_ONLINE</li> </ul>                  |
| Vorbedingungen  | keine   |
| Eingangsdaten   | <ul style="list-style-type: none"> <li>IP-Konfiguration des LAN-Interface (gemäß Tabelle TAB_KON_683 LAN-Adapter IP-Konfiguration)</li> <li>IP-Konfiguration des WAN-Interface (gemäß Tabelle TAB_KON_685 WAN-Adapter IP-Konfiguration)</li> <li>ANLW_IAG_ADDRESS (IP-Adresse des IAG der Einsatzumgebung )</li> <li>DNS_SERVERS_INT</li> </ul> |
| Komponenten     | Konnektor   |
| Ausgangsdaten   | Keine   |
| Nachbedingungen | <ul style="list-style-type: none"> <li>Die Routing-Einträge im Konnektor wurden gesetzt.</li> </ul>   |
| Standardablauf  | Alle bestehenden Routen MÜSSEN vollständig durch die in diesem TUC ermittelten Routen ersetzt werden.   |

|  |  |
|--|--|
|  | <p><b>1) Wenn (MGM_LU_ONLINE=Enabled)</b><br/> Der Konnektor MUSS die nachfolgenden Routen bereitstellen</p> <p>a)</p> <ul style="list-style-type: none"> <li>i. Ziel: Lokale Netze der Einsatzumgebung gemäß ANLW_LEKTR_INTRANET_ROUTES<br/> Next Hop: gemäß ANLW_LEKTR_INTRANET_ROUTES</li> </ul> <p>b) Wenn die VPN-Tunnel zur TI und zum SIS nicht aufgebaut sind:</p> <ul style="list-style-type: none"> <li>i. Ziel: Default Route<br/> Next Hop: ANLW_IAG_ADDRESSc) Wenn der VPN-Tunnel zur TI aufgebaut und der VPN-Tunnel zum SIS nicht aufgebaut sind:</li> <li>i. Ziel: Default Route<br/> Next Hop: ANLW_IAG_ADDRESS</li> <li>ii. Ziel: TI (NET_TI_OFFENE_FD, NET_TI_GESICHERTE_FD und NET_TI_ZENTRAL)<br/> Next Hop: Innere Tunnel IP-Adresse des VPN-Konzentrators TI</li> <li>iii. Ziel: ANLW_AKTIVE_BESTANDSNETZE<br/> Next Hop: Innere Tunnel IP-Adresse des VPN-Konzentrators TI</li> <li>iv. Ziel: VPN-Konzentrator TI<br/> Next Hop: ANLW_IAG_ADDRESS</li> </ul> <p>d) Wenn die VPN-Tunnel zur TI und zum SIS aufgebaut sind:</p> <ul style="list-style-type: none"> <li>i. Ziel: Default Route<br/> Next Hop: Innere Tunnel IP-Adresse des VPN-Konzentrators SIS</li> <li>ii. Ziel: TI (NET_TI_OFFENE_FD, NET_TI_GESICHERTE_FD und NET_TI_ZENTRAL)<br/> Next Hop: Innere Tunnel IP-Adresse des VPN-Konzentrators TI</li> <li>iii. Ziel: ANLW_AKTIVE_BESTANDSNETZE<br/> Next Hop: Innere Tunnel IP-Adresse des VPN-Konzentrators TI</li> <li>iv. Ziel: VPN-Konzentrator TI<br/> Next Hop: ANLW_IAG_ADDRESS</li> <li>v. Ziel: VPN-Konzentrator SIS<br/> Next Hop: ANLW_IAG_ADDRESSHinweis: Wenn der VPN-Tunnel zur TI nicht existiert, kann auch kein VPN-Tunnel zum SIS existieren, da die Default Route zum IAG zeigen muss, um einen VPN-Tunnel zur TI aufbauen zu können.</li> </ul> <p><b>2) Wenn (MGM_LU_ONLINE=Disabled)</b></p> <p>1. Der Konnektor MUSS die nachfolgenden Routen bereitstellen.</p> <ul style="list-style-type: none"> <li>i. Ziel: Lokale Netze der Einsatzumgebung gemäß ANLW_LEKTR_INTRANET_ROUTES<br/> Next Hop: gemäß ANLW_LEKTR_INTRANET_ROUTES</li> </ul> |
|--|--|

|                                |   |
|--------------------------------|---|
|                                | <b>3) Firewall aktualisieren:</b><br>Die Firewall des Konnektors MUSS die neu eingerichteten Routen berücksichtigen und seine Regeln entsprechend aktualisieren und aktivieren. Tritt der Fehler 4164 auf, geht der Konnektor in den Betriebszustand EC_Firewall_Not_Reliable über. |
| Varianten/Alternativen         | Keine   |
| Fehlerfälle                    | (→ 1-2) Eine oder mehrere Variablen enthalten eine ungültige oder keine IP; 4167<br>(→ 3) Beim Aktualisieren oder Aktivieren der Firewall-Regeln ist es zu einem Fehler gekommen; 4164  |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

5956 **Tabelle 312: TAB\_KON\_623 Fehlercodes TUC\_KON\_304 „Netzwerk-Routen einrichten“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4167  | Technical | Fatal    | CreateRoutes:<br>Ein oder mehrere Adressen sind ungültig.                               |
| 4164  | Technical | Fatal    | Beim Aktualisieren oder Aktivieren der Firewall-Regeln ist es zu einem Fehler gekommen. |

5957  
5958 [**<=**]

#### 5959 **4.2.1.4 Interne TUCs, auch durch Fachmodule nutzbar**

5960 Keine.

#### 5961 **4.2.1.5 Operationen an der Außenschnittstelle**

5962 Keine

#### 5963 **4.2.1.6 Betriebsaspekte**

##### 5964 **TIP1-A\_5414 - Initialisierung „Anbindung LAN/WAN“**

5965 Der Konnektor MUSS in der Bootup-Phase zur Initialisierung des Funktionsmerkmals  
5966 „Anbindung LAN/WAN“:

- 5967 • den LAN-Adapter initialisieren (TUC\_KON\_305)
- 5968 • den WAN-Adapter initialisieren (TUC\_KON\_306)
- 5969 • die Infrastrukturdaten vom KSR einlesen (TUC\_KON\_283)

5970 [**<=**]

##### 5971 **TIP1-A\_4759 - Konfiguration LAN-Interface**

5972 Der Konnektor MUSS gewährleisten, dass die Konfiguration nur dann gespeichert wird,  
5973 wenn alle Parameter der nachfolgenden Tabellen den dazugehörigen Bedingungen

entsprechen, sowie grundsätzlich zulässige Werte darstellen (gemäß RFCs).

Wenn die Konfiguration per Managementschnittstelle geändert wurde, MUSS das folgende Systemereignis ausgelöst werden:

```
TUC_KON_256 {
    topic = "ANLW/LAN/IP_CHANGED";
    eventType = Op;
    severity = Info;
    parameters = („IP=$dieNeueIP“);
    doDisp = false}
```

Wenn (DHCP\_CLIENT\_LAN\_STATE=Disabled) gesetzt ist, MUSS der Administrator des Konnektor die Werte der folgenden Tabelle über die Managementschnittstelle setzen können.

Wenn (DHCP\_CLIENT\_LAN\_STATE=Enabled) gesetzt ist, MUSS der Administrator des Konnektor die Werte der folgenden Tabelle angezeigt bekommen, kann diese jedoch nicht ändern.

**Tabelle 313: TAB\_KON\_683 LAN-Adapter IP-Konfiguration**

| ReferenzID               | Belegung                  | Bedeutung und Administrator-Interaktion  |
|--------------------------|---------------------------|--|
| ANLW_LAN_IP_ADDRESS      | IP-Adresse                | Dies ist die IP-Adresse des LAN-Adapters. Nur wenn DHCP_CLIENT_LAN_STATE=Disabled MUSS der Administrator die LAN-seitige IP-Adresse des Konnektors setzen können. Diese IP-Adresse MUSS innerhalb des ANLW_LAN_NETWORK_SEGMENT liegen.   |
| ANLW_LAN_SUBNETMASK      | Subnetzmaske              | Dies ist die zu ANLW_LAN_IP_ADDRESS gehörende Subnetzmaske. Der Administrator MUSS die Subnetzmaske setzen können. Der Konnektor MUSS gewährleisten das nur eine gültige Subnetzmaske gespeichert werden kann.   |
| ANLW_LAN_NETWORK_SEGMENT | IP-Adresse / Subnetzmaske | ANLW_LAN_NETWORK_SEGMENT ist ein Zustandswert, der sich aus den Werten von ANLW_LAN_IP_ADDRESS und ANLW_LAN_SUBNETMASK ergibt. Der Wert bezeichnet ein lokales Netzwerk in der Umgebung des Anwenders an das der LAN-Adapter des Konnektors angeschlossen ist. Der Konnektor MUSS gewährleisten, das das Netzwerksegment NICHT mit einem der folgenden Netzwerksegmente überlappt:<br>1. NET_TI_DEZENTRAL<br>2. NET_TI_ZENTRAL<br>3. NET_TI_OFFENE_FD<br>4. NET_TI_GESICHERTE_FD |

|  |  |   |
|--|--|---|
|  |  | 5. NET_SIS<br>6. ANLW_BESTANDSNETZE<br>7. ANLW_AKTIVE_BESTANDSNETZE<br>8. ANLW_WAN_NETWORK_SEGMENT<br>9. ANLW_LEKTR_INTRANET_ROUTES |
|--|--|---|

5992 Der Administrator des Konnektor MUSS die Werte der folgenden Tabelle über die  
5993 Managementschnittstelle setzen können.  
5994

5995 **Tabelle 314: TAB\_KON\_684 LAN-Adapter Erweiterte Parameter**

| ReferenzID         | Belegung                                  | Bedeutung und Administrator-Interaktion   |
|--------------------|---|---|
| ANLW_LAN_MTU       | Nummer                                    | Der Administrator MUSS Maximum Transmission Unit (MTU) setzen können. Der Konnektor MUSS sicherstellen, dass der konfigurierte Wert in den Grenzen von 576 bis 9000 liegt. Default-Wert: 1400 |
| ANLW_LAN_PARAMETER | Liste von IP, UDP und/oder TCP Parametern | Der Administrator SOLL weitere Konfigurationsparameter gemäß [gemSpec_Net#2.2.2.1,2.5] konfigurieren können.  |

5996  
5997 [ $\leq$ ]

5998 **TIP1-A\_4760 - Konfiguration WAN-Interface**

5999 Der Konnektor MUSS gewährleisten, dass die Konfiguration nur dann gespeichert wird,  
6000 wenn alle Parameter der nachfolgenden Tabellen den dazugehörigen Bedingungen  
6001 entsprechen.

6002 Wenn die Konfiguration per Managementschnittstelle geändert wurde, MUSS das  
6003 folgende Systemereignis ausgelöst werden:

6004 TUC\_KON\_256 {  
6005 topic = "ANLW/WAN/IP\_CHANGED";  
6006 eventType = Op;  
6007 severity = Info;  
6008 parameters = („IP=\$dieNeueIP“);  
6009 doDisp = false}

6010 Wenn (DHCP\_CLIENT\_WAN\_STATE=Disabled) gesetzt ist, MUSS der Administrator des  
6011 Konnektors die Werte der folgenden Tabelle über die Managementschnittstelle setzen  
6012 können.

6013 Wenn (DHCP\_CLIENT\_WAN\_STATE=Enabled) gesetzt ist, MUSS der Administrator des  
6014 Konnektors die Werte der folgenden Tabelle angezeigt bekommen, kann diese jedoch  
6015 nicht ändern.  
6016

6017 **Tabelle 315: TAB\_KON\_685 WAN-Adapter IP-Konfiguration**

| ReferenzID          | Belegung   | Bedeutung und Administrator-Interaktion   |
|---------------------|------------|---|
| ANLW_WAN_IP_ADDRESS | IP-Adresse | Dies ist die IP-Adresse des WAN-Adapters.<br>Nur wenn<br>DHCP_CLIENT_WAN_STATE=Disabled |

|                          |                           |  |
|--------------------------|---------------------------|--|
|                          |                           | und<br>ANLW_WAN_ADAPTER_MODUS=ENABLED<br>MUSS der Administrator die WAN-seitige IP-Adresse des Konnektors setzen können.<br>Diese IP-Adresse MUSS innerhalb des ANLW_WAN_NETWORK_SEGMENT liegen.   |
| ANLW_WAN_SUBNETMASK      | Subnetzmaske              | Dies ist die zu ANLW_WAN_IP_ADDRESS gehörende Subnetzmaske.<br>Der Administrator MUSS die Subnetzmaske setzen können.<br>Der Konnektor MUSS gewährleisten, dass nur eine gültige Subnetzmaske gespeichert werden kann.   |
| ANLW_WAN_NETWORK_SEGMENT | IP-Adresse / Subnetzmaske | ANLW_WAN_NETWORK_SEGMENT ist ein Zustandswert, der sich aus den Werten von ANLW_WAN_IP_ADDRESS und ANLW_WAN_SUBNETMASK ergibt.<br>Der Wert bezeichnet ein lokales Netzwerk in der Umgebung des Anwenders an das der WAN-Adapter des Konnektors angeschlossen ist.<br>Der Konnektor MUSS gewährleisten, dass das Netzwerksegment nicht mit einem der folgenden Netzwerksegmente überlappt:<br>1. NET_TI_DEZENTRAL<br>2. NET_TI_ZENTRAL<br>3. NET_TI_OFFENE_FD<br>4. NET_TI_GESICHERTE_FD<br>5. NET_SIS<br>6. ANLW_BESTANDSNETZE<br>7. ANLW_AKTIVE_BESTANDSNETZE<br>8. ANLW_LAN_NETWORK_SEGMENT<br>9. ANLW_LEKTR_INTRANET_ROUTES |

6018 Der Administrator des Konnektor MUSS die Werte der folgenden Tabelle über die  
6019 Managementschnittstelle setzen können.  
6020

6021 **Tabelle 316: TAB\_KON\_686 WAN-Adapter Erweiterte Parameter**

| ReferenzID         | Belegung                   | Bedeutung und Administrator-Interaktion   |
|--------------------|----------------------------|---|
| ANLW_WAN_MTU       | Nummer                     | Der Administrator MUSS Maximum Transmission Unit (MTU) setzen können.<br>Der Konnektor MUSS sicherstellen, dass der konfigurierte Wert in den Grenzen von 576 bis 9000 liegt.<br>Default-Wert: 1400 |
| ANLW_WAN_PARAMETER | Liste von IP, UDP und/oder | Der Administrator SOLL weitere Konfigurationsparameter gemäß  |



|  |                   |  |
|--|-------------------|--|
|  | TCP<br>Parametern | [gemSpec_Net#2.2.2.1,2.5]<br>konfigurieren können. |
|--|-------------------|--|

[<=]

#### TIP1-A\_4761 - Konfiguration Anbindung LAN/WAN

Die Managementschnittstelle MUSS es einem Administrator ermöglichen Konfigurationsänderungen gemäß Tabelle TAB\_KON\_624 – „Konfigurationsparameter der Anbindung LAN/WAN vorzunehmen.

Wenn (ANLW\_INTRANET\_ROUTES\_MODUS = REDIRECT) gesetzt ist, MUSS der Konnektor jedes Paket aus einem konfigurierten Intranet mit einem ICMP-Redirect mit dem hinterlegten Next Hop beantworten und der Konnektor MUSS gewährleisten, dass keine IP-Pakete in eines oder mehrere der konfigurierten Intranet geroutet werden. Wenn (ANLW\_INTRANET\_ROUTES\_MODUS = BLOCK) gesetzt ist, MUSS der Konnektor alle IP-Pakete für ein Intranet (gemäß ANLW\_LEKTR\_INTRANET\_ROUTES) ablehnen.

**Tabelle 317: TAB\_KON\_624 – „Konfigurationsparameter der Anbindung LAN/WAN“**

| ReferenzID              | Belegung | Bedeutung und Administrator-Interaktion  |
|-------------------------|----------|--|
| ANLW_ ANBINDUNGS_ MODUS | InReihe  | Der Konnektor ist in Reihe zu dem IAG der Einsatzumgebung geschaltet.<br>Wenn ANLW_WAN_ADAPTER_MODUS= ENABLED befindet sich der Konnektor in diesem Anbindungsmodus.<br>Der Administrator MUSS diesen Wert einsehen und DARF ihn NICHT ändern können.                                    |
|                         | Parallel | Der Konnektor ist parallel (zu allen bestehenden Systemen) ins Netzwerk der Einsatzumgebung angebunden.<br>Wenn ANLW_WAN_ADAPTER_MODUS= DISABLED befindet sich der Konnektor in diesem Anbindungsmodus.<br>Der Administrator MUSS diesen Wert einsehen und DARF ihn NICHT ändern können. |
| ANLW_ INTERNET_ MODUS   | SIS      | Der (am Konnektor LAN-seitig ankommende) Internet-Traffic wird per VPN an den SIS geschickt.   |
|                         | IAG      | Bei Anfragen ins Internet wird der Aufrufer per ICMP-Redirect (Type 5) auf die Route zum IAG verwiesen.<br>Wenn (ANLW_ANBINDUNGS_MODUS = InReihe) DARF dieser Wert NICHT auswählbar sein - statt dessen MUSS dann der Wert SIS verwendet werden.   |
|                         | KEINER   | Es wird kein Traffic ins Internet geroutet   |

|  |  |  |
|--|--|--|
| ANLW_<br>INTRANET_<br>ROUTES_<br>MODUS | REDIRECT   | Der Konnektor MUSS sicherstellen, dass dieser Wert nur gesetzt werden kann, wenn der Administrator zuvor ein oder mehrere Intranet (ANLW_LEKTR_INTRANET_ROUTES) definiert hat.   |
|  | BLOCK  | Der Konnektor MUSS alle IP-Pakete für ein Intranet (gemäß ANLW_LEKTR_INTRANET_ROUTES) ablehnen.  |
| ANLW_WAN_<br>ADAPTER_<br>MODUS         | ENABLED  | Dieser Parameter ändert den Interface-Status des WAN-Adapters.<br>Der Administrator MUSS diesen Wert einsehen können.<br>Der Administrator MUSS diesen Wert ändern können.   |
|  | DISABLED   | Dieser Parameter ändert den Interface-Status des WAN-Adapters.<br>Der Administrator MUSS diesen Wert einsehen können.<br>Der Administrator MUSS diesen Wert ändern können.   |
| ANLW_<br>LEKTR_<br>INTRANET_<br>ROUTES | Tupel aus Netzwerksegment und dazugehörigem Next-Hop | Der Administrator MUSS in diese Liste Einträge hinzufügen, editieren und löschen können.<br>Liste von Routen zur Erreichung der Clientsysteme und Kartenterminals vom Konnektor; jeweils mit IP-Netzwerk dazugehörigem Next Hop.<br>Die Netzwerksegmente DÜRFEN NICHT mit den Netzbereichen <ul style="list-style-type: none"> <li>• NET_SIS</li> <li>• NET_TI_DEZENTRAL</li> </ul> NET_TI_ZENTRAL <ul style="list-style-type: none"> <li>• NET_TI_OFFENE_FD</li> <li>• NET_TI_GESICHERTE_FD</li> <li>• ANLW_BESTANDSNETZE</li> </ul> kollidieren. |

|                           |                                |   |
|---------------------------|--------------------------------|---|
| ANLW_IAG_ADDRESS          | IP Adresse                     | ANLW_IAG_ADDRESS ist die Adresse des Default Gateways. Diese IP-Adresse MUSS innerhalb des ANLW_WAN_NETWORK_SEGMENT liegen. Die Adresse wird entweder über DHCP automatisch (DHCP_CLIENT_WAN_STATE=ENABLED oder DHCP_CLIENT_LAN_STATE=ENABLED) oder anderenfalls manuell durch den Administrator konfiguriert. Bei automatischer Konfiguration per DHCP MUSS der Administrator den Wert von ANLW_IAG_ADDRESS ausschließlich einsehen können.  |
| ANLW_AKTIVE_BESTANDSNETZE | Liste von IP-Address-Segmenten | Der Administrator MUSS manuell aus der empfangenen Liste der zur Verfügung stehenden angeschlossene Netze des Gesundheitswesens mit aAdG-NetG (gemäß TUC_KON_283 „Infrastruktur Konfiguration aktualisieren“) einzelne deaktivieren, bzw. nach vorheriger Deaktivierung, freischalten können. Nur die freigegeben Netze werden in dieser Variablen erfasst und sind aus den Netzwerken der Einsatzumgebung erreichbar. Wird eine Änderung an der Liste der freigegebenen Netze vorgenommen, so MUSS der Konnektor für jedes dieser freigegebenen Netz in DNS_SERVERS_BESTANDSNETZE ein DNS-Referer-Eintrag für jede der dazugehörigen Domains mit allen zugehörigen DNS-Servern im Konnektor hinterlegen. Die Werte hierzu werden der via TUC_KON_283 aktualisierten Bestandsnetze.xml entnommen. Für hier „nicht freigegebene“ oder zwischenzeitlich gelöschte Netze DARF der Konnektor NICHT Referer-Einträge in DNS_SERVERS_BESTANDSNETZE enthalten. Die Einträge in DHCP_AKTIVE_BESTANDSNETZE_ROUTES sind entsprechend zu aktualisieren. Der Konnektor MUSS nach jeder Änderung dieser Variablen durch den Administrator den TUC_KON_304 „Netzwerk-Routen einrichten“ aufrufen. |
| ANLW_IA_BESTANDSNETZE     | AN                             | Der Konnektor MUSS alle über TUC_KON_283 übermittelten angeschlossenen Netze des Gesundheitswesens mit aAdG-NetG aktivieren. Eine spätere manuelle Deaktivierung über das Management-Interface durch den Administrator ist möglich. Dieses Verhalten ist als Standardverhalten zu konfigurieren.  |

|  |     |  |
|--|-----|--|
|  | AUS | Der Konnektor MUSS alle über TUC_KON_283 übermittelten angeschlossenen Netze des Gesundheitswesens mit aAdG-NetG anbieten, diese aber nicht aktivieren. Eine spätere manuelle Aktivierung erfolgt über das Management-Interface durch den Administrator. |
|--|-----|--|

[&lt;=]

**TIP1-A\_5537 - Anzeige IP-Routinginformationen**

Der Konnektor MUSS über die Managementschnittstelle die konfigurierten IP-Routen und die aktuelle IP-Routingtabelle mit mindestens folgenden Informationen anzeigen:

- Forwarding Status
- Zieladresse/Prefix
- Gateway (Next-Hop)
- Routing Typ
- Routing Protocol
- Routing Preference.

[&lt;=]

**TIP1-A\_4762 - Konfigurationsparameter Firewall-Schnittstelle**

Im Anschluss an eine Anpassung der ANLW\_FW\_SIS\_ADMIN\_RULES MUSS der Konnektor die Firewall neu erstellen und laden.

**Tabelle 318: TAB\_KON\_625 - Konfigurationsparameter Firewall-Schnittstelle**

| ReferenzID              | Belegung          | Bedeutung und Administrator-Interaktion   |
|-------------------------|-------------------|---|
| ANLW_FW_SIS_ADMIN_RULES | Firewall Regelset | Der Administrator MUSS Firewall-Regeln (für den einschränkenden Zugriff auf die SIS), auf Grundlage der Parameter Absender-IP-Adresse, Empfänger-IP-Adresse, Protokoll (ggf. mit Absender-Port und Empfänger-Port) und Verbindungsrichtung, einfügen, editieren und löschen können. |

[&lt;=]

**4.2.2 DHCP-Server**

Innerhalb des Kapitels DHCP-Servers werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): „DHCP“
- Konfigurationsparameter: „DHCP\_SERVER\_“

6060 **4.2.2.1 Funktionsmerkmalweite Aspekte**

6061 **TIP1-A\_4763 - DHCP-Server des Konnektors**

6062 Der Konnektor MUSS an seiner LAN-Schnittstelle einen DHCP-Server gemäß [RFC2131]  
 6063 und [RFC2132] anbieten.  
 6064 [ $\leq$ ]

6065 **4.2.2.2 Durch Ereignisse ausgelöste Reaktionen**

6066 Keine.

6067 **4.2.2.3 Interne TUCs, nicht durch Fachmodule nutzbar**

6068 Keine.

6069 **4.2.2.4 Interne TUCs, auch durch Fachmodule nutzbar**

6070 Keine.

6071 **4.2.2.5 Operationen an der Außenschnittstelle**

6072 *4.2.2.5.1 Liefere Netzwerkinformationen über DHCP*

6073 **TIP1-A\_4765 - Liefere Netzwerkinformationen über DHCP**

6074 Der DHCP-Server des Konnektors MUSS an der Client-Schnittstelle eine Operation zur  
 6075 Lieferung von Netzwerkinformationen über DHCP anbieten.  
 6076

6077 **Tabelle 319: TAB\_KON\_626 „Liefere Netzwerkinformationen über DHCP“**

| Name            | Liefere Netzwerkinformationen über DHCP   |
|-----------------|---|
| Beschreibung    | Der Konnektor MUSS anfragenden Clients per DHCP die konfigurierten Netzwerkinformationen liefern (siehe Tabelle TAB_KON_628 und Tabelle TAB_KON_629).   |
| Aufrufparameter | gemäß [RFC2131], [RFC2132]  |
| Rückgabe        | gemäß [RFC2131], [RFC2132]  |
| Standardablauf  | <p>Die an den aufrufenden Client zu übergebenden Parameter ergeben sich aus Tabelle TAB_KON_628 und Tabelle TAB_KON_629:</p> <p>Falls DHCP_SERVER_STATE = Enabled:</p> <ul style="list-style-type: none"> <li>Anhand der MAC-Adresse des anfragenden Client wird die Clientgruppe aus DHCP_SERVER_CLIENTGROUPS bzw. DHCP_SERVER_DEFAULT_CLIENTGROUP ausgewählt.</li> <li>DHCP_OWNDNS_ENABLED               <ul style="list-style-type: none"> <li>Enabled: DNS-Server = &lt;konnektoreigene Adresse&gt;</li> <li>Disabled: DNS-Server = DHCP_DNS_ADDR</li> </ul> </li> <li>DHCP_NTP               <ul style="list-style-type: none"> <li>Enabled: NTP-Server = &lt;konnektoreigene Adresse&gt;</li> <li>Disabled: Keine Wertübermittlung</li> </ul> </li> </ul> |

|                 |   |
|-----------------|---|
|                 | <ul style="list-style-type: none"> <li>• DHCP_OWNDGW_ENABLED <ul style="list-style-type: none"> <li>• Enabled: DGW = &lt;konnektoreigene Adresse&gt;</li> <li>• Disabled: DGW = DHCP_DGW_ADDR</li> </ul> </li> <li>• Falls Client-MAC-Adresse in DHCP_STATIC_LEASE <ul style="list-style-type: none"> <li>• IP_Address = die in der Static Lease konfigurierte Adresse.</li> </ul> </li> <li>• Falls Client IP-Adresse = 0.0.0.0 oder innerhalb DHCP_SERVER_DYNAMIC_RANGE <ul style="list-style-type: none"> <li>• IP_Address = IP_Address aus DHCP_SERVER_DYNAMIC_RANGE</li> <li>• Sonst: keine Zuweisung (Empfehlung: DHCPNAK an den Client)</li> </ul> </li> <li>• Netzmaske = DHCP_IP_NETMASK</li> <li>• Domainname = DHCP_DOMAINNAME</li> <li>• Hostname = DHCP_HOSTNAME</li> <li>• Lease Dauer = DHCP_LEASE_TTL</li> <li>• Routen bestehend aus <ul style="list-style-type: none"> <li>• DHCP_AKTIVE_BESTANDSNETZE_ROUTES</li> <li>• DHCP_INTRANET_ROUTES</li> <li>• DHCP_ROUTES</li> </ul> </li> <li>• Weitere DHCP-Optionen = DHCP_OPTIONS</li> <li>• MTU = ANLW_LAN_MTU</li> </ul> |
| Fehlercodes     | Vgl. [RFC2131], [RFC2132]   |
| Vorbedingungen  | Der DHCP-Server des Konnektors MUSS aktiviert und konfiguriert sein.  |
| Nachbedingungen | Der DHCP-Server MUSS die DHCP-Antwort geliefert haben. Die Statusinformationen (z.B. Client Lease) müssen gemäß [RFC2131] gespeichert werden.   |
| Hinweise        | Keine   |

6078

6079 [ $\leq$ ]6080 **4.2.2.6 Betriebsaspekte**6081 **TIP1-A\_4766 - Deaktivierbarkeit des DHCP-Servers**

6082 Der DHCP- Server des Konnektors MUSS durch den Administrator über die  
6083 Managementschnittstelle aktivierbar und deaktivierbar sein (gemäß TAB\_KON\_627). Der  
6084 DHCP-Server MUSS bei der Auslieferung deaktiviert sein.

6085 Bei der Aktivierung MUSS der Konnektor den TUC\_KON\_343 "Initialisierung DHCP-  
6086 Server" durchlaufen.

6087 Sobald DHCP\_SERVER\_STATE geändert wurde, muss  
6088 TUC\_KON\_256{"DHCP/SERVER/STATECHANGED"; Op; Info;

6089 "STATE=\$DHCP\_SERVER\_STATE "} aufgerufen werden.

6090

6091 **Tabelle 320: TAB\_KON\_627 „Aktivierung des DHCP-Servers“**

| Referenz ID       | Belegung           | Bedeutung  |
|-------------------|--------------------|--|
| DHCP_SERVER_STATE | Enabled / Disabled | Der DHCP-Server MUSS durch den Administrator aktivierbar und deaktivierbar sein. |

6092

6093 [ $\leq$ ]

6094 **TIP1-A\_4767 - Konfiguration des DHCP-Servers**

6095 Der Konnektor MUSS die Möglichkeit bieten die in Tabelle TAB\_KON\_628 und Tabelle  
6096 TAB\_KON\_629 beschriebenen Parameter des DHCP-Servers über die  
6097 Managementschnittstelle zu konfigurieren.

6098

6099 **Tabelle 321: TAB\_KON\_628 „Basiskonfiguration des DHCP-Servers“**

| Referenz ID                     | Belegung  | Bedeutung   |
|---------------------------------|---|---|
| DHCP_SERVER_NETWORK             | IP-Adresse                                      | IP-Netzwerk der Einsatzumgebung.  |
| DHCP_SERVER_BROADCAST           | IP-Adresse                                      | Die Broadcast-Adresse des Konnektors am LAN-Interface   |
| DHCP_SERVER_DYNAMIC_RANGE       | von – bis IP-Adresse                            | Adressbereich für Adressen die dynamisch vergeben werden dürfen.  |
| DHCP_SERVER_CLIENTGROUPS        | Name der Clientgruppe;<br>Liste an MAC-Adressen | Der Konnektor MUSS dem Administrator über die Managementschnittstelle die Möglichkeit bieten mindestens zwei Client-Gruppen zu verwalten. |
| DHCP_SERVER_DEFAULT_CLIENTGROUP | Client-Gruppe                                   | Standardmäßig eingestellte Client-Gruppe. Wird verwendet falls DHCP-Anfrage keiner anderen Client-Gruppe zugeordnet werden kann.          |

6100

6101 **Tabelle 322: TAB\_KON\_629 „Client-Gruppenspezifische Konfigurationsoptionen des Konnektor-DHCP-Servers“**

| ReferenzID  | Belegung         | Bedeutung  |
|---|------------------|--|
| Die gesamte Parameterliste ist für jede Client-Gruppe getrennt konfigurierbar |                  |  |
| DHCP_OWNDNS_ENABLED   | Enabled/Disabled | Der Administrator MUSS konfigurieren können, ob der konnektoreigene DNS-Server als Parameter übergeben wird.<br>Default-Wert: Disabled |

|                     |  |   |
|---------------------|--|---|
| DHCP_DNS_ADDR       | IP-Adressen der DNS-Server                   | Falls der konnektoreigene DNS-Server nicht übergeben werden soll, müssen die Adressen externer aus dem Netz der Einsatzumgebung erreichbaren DNS-Server als Parameter übergeben werden. Der Administrator MUSS diese Adressen konfigurieren können. |
| DHCP_NTP            | Enabled/Disabled                             | Der Administrator MUSS konfigurieren können, ob der Konnektor die Adresse des Konnektor internen NTP-Servers per DHCP an die Clients sendet. Default-Wert: Enabled  |
| DHCP_OWNDGW_ENABLED | Enabled/Disabled                             | Der Administrator MUSS konfigurieren können, ob der Konnektor beim Client als Default-Gateway gesetzt werden soll. Default-Wert: Disabled   |
| DHCP_DGW_ADDR       | IP-Adresse des DGW                           | Falls der Konnektor nicht als Default Gateway gesetzt werden soll, muss die Adresse des zu verwendenden DGW als Parameter übergeben werden. Der Administrator MUSS die Adresse des DGW konfigurieren können.  |
| DHCP_IP_NETMASK     | Netzmaske                                    | Der Administrator MUSS die Netmask des Clients konfigurieren können.  |
| DHCP_DOMAINNAME     | Domainname                                   | Der Administrator MUSS den Domainnamen des Clients konfigurieren können.  |
| DHCP_HOSTNAME       | Liste von Tupel aus Hostname und Mac-Adresse | Der Administrator MUSS eine Liste von Hostname der Clients konfigurieren können (Einträge einfügen, ändern, löschen).   |
| DHCP_STATIC_LEASE   | Liste von Tupel aus IP- und Mac-Adresse      | Der Administrator MUSS für jede MAC-Adresse Static Lease konfigurieren können.  |



|  |   |  |
|--|---|--|
| DHCP_<br>LEASE_TTL                               | X Minuten   | Der Administrator MUSS Lease-Dauer der dynamischen Adressen konfigurieren können.  |
| DHCP_<br>AKTIVE_<br>BESTANDS<br>NETZE_<br>ROUTES | Liste von Tupel:<br>Netzwerksegment<br>je INTRANET und<br>Adresse für Next<br>Hop je<br>freigegebenem<br>angeschlossenen<br>Netze des<br>Gesundheitswesens<br>mit aAdG-NetG | Der Administrator MUSS je freigegebenem angeschlossenen Netze des Gesundheitswesens mit aAdG-NetG (aus ANLW_AKTIVE_BESTANDSNETZE) den an den Client zu übermittelnden Routen-Eintrag aktivieren oder deaktivieren können.  |
| DHCP_<br>INTRANET_<br>ROUTES                     | Liste von Tupel:<br>Netzwerksegment<br>je INTRANET und<br>Adresse für Next<br>Hop in die<br>definierten<br>Intranets  | Der Administrator MUSS je Intranet-Tupel (aus ANLW_LEKTR_INTRANET_ROUTES) den an den Client zu übermittelnden Routen-Eintrag aktivieren oder deaktivieren können.  |
| DHCP_<br>ROUTES                                  | Tupel<br>Netzwerksegment<br>und Adresse für<br>Next Hop   | Der Administrator MUSS Routen zur Verteilung an die Clients frei konfigurieren können. Der Konnektor MUSS sicherstellen, diese Listeneinträge keine Überschneidungen mit folgenden Netzsegmenten haben:<br>- dem Netzwerksegment ANLW_LAN_NETWORK_SEGMENT<br>- dem Netzwerksegment ANLW_WAN_NETWORK_SEGMENT<br>- jedes Netzsegmente in ANLW_BESTANDSNETZE<br>ANLW_AKTIVE_BESTANDSNETZE<br>ANLW_LEKTR_INTRANET_ROUTES<br>Die Routen SOLLEN über DHCP Option 121 (Windows Vista oder höher) bzw.<br>DHCP Option 249 (Windows XP und darunter) verteilt werden. |
| DHCP_<br>OPTIONS                                 | Liste an weiteren<br>DHCP-Optionen.   | Vom Administrator konfigurierbare Liste an weiteren DHCP-Options gemäß [RFC2132].<br>Die Umsetzung dieser Konfigurationsmöglichkeit KANN entfallen.  |

6102  
6103 [ $\leq$ ]

6104 4.2.2.6.1 TUC\_KON\_343 „Initialisierung DHCP-Server“

6105 **TIP1-A\_4768 - TUC\_KON\_343 „Initialisierung DHCP-Server“**

6106 Der Konnektor MUSS in der Bootup-Phase TUC\_KON\_343 "Initialisierung DHCP-Server"  
6107 durchlaufen.

6108

6109 **Tabelle 323: TAB\_KON\_630 - TUC\_KON\_343 „Initialisierung DHCP-Server“**

| Element                        | Beschreibung  |
|--------------------------------|---|
| Name                           | TUC_KON_343 "Initialisierung DHCP-Server"   |
| Beschreibung                   | Falls DHCP-Server Konfiguration aktiv ist, muss der Konnektor in der Bootup-Phase oder bei einer Aktivierung des Servers den DHCP-Server starten. |
| Anwendungsumfeld               | Bereitstellen der Netzwerkkonfiguration für den Betrieb   |
| Eingangsanforderung            | Keine   |
| Auslöser und Vorbedingungen    | Bootup oder Ereignis DHCP/SERVER/STATECHANGED   |
| Eingangsdaten                  | Keine   |
| Komponenten                    | Konnektor   |
| Ausgangsdaten                  | Keine   |
| Standardablauf                 | Falls DHCP_SERVER_STATE = enabled<br>- den DHCP-Server starten<br>Falls DHCP_SERVER_STATE = disabled<br>- den DHCP-Server stoppen                 |
| Varianten/Alternativen         | Keine   |
| Fehlerfälle                    | 4168: DHCP-Server konnte nicht gestartet werden   |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

6110 **Tabelle 324: TAB\_KON\_631 Fehlercodes TUC\_KON\_343 „Initialisierung DHCP-Server“**

| Fehlercode | ErrorType | Severity | Fehlertext  |
|------------|-----------|----------|---|
| 4168       | Technical | Error    | Der DHCP-Server des Konnektors konnte nicht gestartet werden. |

6111  
6112 [ $\leq$ ]

### 4.2.3 DHCP-Client

Innerhalb des Kapitels DHCP-Client werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): „DHCP“
- Konfigurationsparameter: „DHCP\_CLIENT\_“

#### 4.2.3.1 Funktionsmerkmalweite Aspekte

##### TIP1-A\_4769 - DHCP Client Funktionalität des Konnektors

Der Konnektor MUSS an seiner LAN- und WAN-Schnittstelle die Möglichkeit bieten jeweils DHCP zu nutzen.

Der DHCP-Client des Konnektors MUSS die empfangenen Parameter wie folgt verwenden:

- Die IP-Adresse und Subnetzmaske müssen dem Interface zugewiesen und in den Variablen ANLW\_LAN\_IP\_ADDRESS bzw. ANLW\_WAN\_IP\_ADDRESS und ANLW\_LAN\_SUBNETMASK gespeichert werden.
- Der für das Interface, auf Anfrage, gelieferte Wert der MTU Size KANN übernommen werden.
- Das Default Gateway (DGW) muss in der Variable ANLW\_IAG\_ADDRESS gespeichert werden.
- DNS-Server muss in der Variable DNS\_SERVERS\_INT gespeichert werden.

Weitere DHCP-Parameter DÜRFEN nicht übernommen werden.

[<=]

#### 4.2.3.2 Durch Ereignisse ausgelöste Reaktionen

##### TIP1-A\_4771 - Reagieren auf DHCP/LAN\_CLIENT/ STATECHANGED- und DHCP/WAN\_CLIENT/ STATECHANGED-Ereignisse

Wenn das Ereignis DHCP/LAN\_CLIENT/STATECHANGED oder DHCP/WAN\_CLIENT/STATECHANGED empfangen wird, MUSS TUC\_KON\_341 „DHCP-Informationen beziehen“ aufgerufen werden.

[<=]

#### 4.2.3.3 Interne TUCs, nicht durch Fachmodule nutzbar

##### 4.2.3.3.1 TUC\_KON\_341 „DHCP-Informationen beziehen“

##### TIP1-A\_4772 - TUC\_KON\_341 „DHCP-Informationen beziehen“

Der Konnektor MUSS den technischen Use Case TUC\_KON\_341 „DHCP-Informationen beziehen“ umsetzen.

**Tabelle 325: TAB\_KON\_632 – TUC\_KON\_341 „DHCP Informationen beziehen“**

| Element          | Beschreibung  |
|------------------|---|
| Name             | TUC_KON_341 DHCP-Informationen beziehen   |
| Beschreibung     | Der Konnektor muss seine WAN- und/oder LAN-Schnittstelle individuell über einen DHCP-Server aus dem Netz der Einsatzumgebung beziehen können. |
| Anwendungsumfeld | Netzwerkconfiguration für den Betrieb des Konnektors  |

|                                |  |
|--------------------------------|--|
| Eingangsanforderung            | Der Konnektor muss zur Netzwerk-Interface-Konfiguration DHCP nutzen sofern keine statischen Informationen vorhanden sind.  |
| Auslöser                       | Bootup, Ablauf einer DHCP-Lease, manuell angestoßenes DHCP-Renew, Aktivierung der DHCP-Client-Funktionalität.  |
| Vorbedingung                   | aktivierte DHCP-Client Funktion über die Variablen DHCP_CLIENT_LAN_STATE bzw. DHCP_CLIENT_WAN_STATE  |
| Eingangsdaten                  | Netzwerk-Adapter (LAN oder WAN) für den DHCP-Informationen bezogen werden sollen   |
| Komponenten                    | Konnektor  |
| Ausgangsdaten                  | DHCP-Informationen vom DHCP-Server der Einsatzumgebung   |
| Standardablauf                 | <ul style="list-style-type: none"> <li>Ermitteln von DHCP-Informationen (DHCPDISCOVER und DHCPREQUEST) gemäß [RFC2131], [RFC2132]</li> <li>Übernahme der ermittelten Werte, ausschließlich für die in Tabelle TAB_KON_683 LAN-Adapter IP-Konfiguration bzw. Tabelle TAB_KON_685 WAN-Adapter IP-Konfiguration aufgeführten Variablen</li> <li>Wenn DHCP Client LAN-Adapter, nur bei IP-Adressen-Wechsel: Erzeugen eines Events durch den Aufruf von TUC_KON_256{"DHCP/LAN_CLIENT/RENEW"; Op; Info; "IP_ADDRESS=\$Belegung"}</li> <li>Wenn DHCP Client WAN-Adapter, nur bei IP-Adressen-Wechsel: Erzeugen eines Events durch den Aufruf von TUC_KON_256{"DHCP/WAN_CLIENT/RENEW"; Op; Info; "IP_ADDRESS=\$Belegung"}</li> </ul> |
| Varianten/Alternativen         | Keine  |
| Fehlerfälle                    | 4169: Konnektor erhält keine DHCP-Informationen<br>4170: Konnektor besitzt identische IP-Adressen am WAN- und LAN-Interface  |
| Nichtfunktionale Anforderungen | Keine  |
| Zugehörige Diagramme           | Keine  |

6146

6147 **Tabelle 326: TAB\_KON\_633 Fehlercodes TUC\_KON\_341 „DHCP-Informationen beziehen“**

| Fehlercode | ErrorType | Severity | Fehlertext  |
|------------|-----------|----------|---|
| 4169       | Technical | Error    | Konnektor erhält keine DHCP-Informationen.              |
| 4170       | Technical | Error    | Konnektor besitzt identische IP-Adressen am WAN und LAN |

6148  
6149 [ $\leq$ ]

#### 6150 **4.2.3.4 Interne TUCs, auch durch Fachmodule nutzbar**

6151 Keine.

#### 6152 **4.2.3.5 Operationen an der Außenschnittstelle**

6153 Keine.

#### 6154 **4.2.3.6 Betriebsaspekte**

##### 6155 **TIP1-A\_4773 - Konfiguration des DHCP-Clients**

6156 Die DHCP-Client Funktionalität MUSS für LAN- und WAN-Interface vom Administrator  
6157 getrennt aktivierbar und deaktivierbar sein (gemäß TAB\_KON\_634). Falls der DHCP-  
6158 Client nicht verwendet wird MUSS sichergestellt werden, dass eine statische  
6159 Konfiguration, für den LAN-Adapter gemäß Tabelle TAB\_KON\_683 LAN-Adapter IP-  
6160 Konfiguration bzw. für den WAN-Adapter gemäß Tabelle TAB\_KON\_685 WAN-Adapter IP-  
6161 Konfiguration, existiert bevor die Netzwerkeinstellungen übernommen werden.  
6162 Sobald Parameter geändert wurden, MUSS TUC\_KON\_256 „Systemereignis absetzen“ je  
6163 nachdem auf welchem Interface der Client aktiviert oder deaktiviert wurde mit folgenden  
6164 Parameter aufgerufen werden:

6165  
6166 TUC\_KON\_256{"DHCP/LAN\_CLIENT/STATECHANGED"; Op; Info;  
6167 "STATE=\$DHCP\_CLIENT\_LAN\_STATE"; doDisp = false}  
6168 oder  
6169 TUC\_KON\_256{"DHCP/WAN\_CLIENT/STATECHANGED "; Op; Info;  
6170 "STATE=\$DHCP\_CLIENT\_WAN\_STATE "; doDisp = false}

6171  
6172

6173 **Tabelle 327: TAB\_KON\_634 „Konfiguration des DHCP-Clients“**

| ReferenzID            | Belegung         | Bedeutung  |
|-----------------------|------------------|--|
| DHCP_CLIENT_LAN_STATE | Enabled/Disabled | Der Administrator muss den DHCP-Client an der LAN-Schnittstelle aktivieren oder deaktivieren können. |
| DHCP_CLIENT_WAN_STATE | Enabled/Disabled | Der Administrator muss den DHCP-Client an der WAN-Schnittstelle aktivieren oder deaktivieren können. |

6174  
6175 [ $\leq$ ]

##### 6176 **TIP1-A\_4774 - Manuelles anstoßen eines DHCP-Lease-Renew**

6177 Der Administrator MUSS die Möglichkeit haben die DHCP-Lease des Konnektors für jedes  
6178 Interface getrennt zu erneuern.

6179 [ $\leq$ ]

##### 6180 **TIP1-A\_4776 - Setzen der IP-Adresse nach Timeout**

6181 Falls der DHCP-Client auf der LAN-Seite nach einem Timeout von 30s keine IP-Adresse  
6182 bezogen hat, MUSS gemäß [RFC3927] eine Default-Adresse aus 169.254/16 vergeben  
6183 werden.

6184 [ $\leq$ ]

## 4.2.4 VPN-Client

Der VPN-Client beschreibt die Absicherung der Anbindung des Konnektors an die TI und die Bestandsnetze. Während der technische Kern dieser Funktion, der Aufbau der VPN-Kanäle zu den Konzentratoren, in [gemSpec\_VPN\_ZugD#TUC\_VPN-ZD\_0001] und [gemSpec\_VPN\_ZugD#TUC\_VPN-ZD\_0002] beschrieben wird, regelt dieses Kapitel die Interaktion, sowie die Konfiguration des VPN-Clients innerhalb des Konnektors.

Innerhalb des Kapitels VPN-Client werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): „NETWORK“
- Konfigurationsparameter: „VPN\_“

### 4.2.4.1 Funktionsmerkmalweite Aspekte

#### TIP1-A\_4778 - Anforderungen an den VPN-Client

Der Konnektor MUSS sich im Rahmen des IPsec-Verbindungsaufbaus gegenüber den VPN-Konzentratoren mit seiner Identität ID.NK.VPN ausweisen.

Der VPN-Client im Konnektor MUSS das folgende Event generieren, sobald der VPN-Tunnel zur TI nicht mehr zur Verfügung steht:

Rufe TUC\_KON\_256 {"NETWORK/VPN\_TI/DOWN"; Op; Warning;}

Der VPN-Client im Konnektor MUSS das folgende Event generieren, sobald der VPN-Tunnel zum SIS nicht mehr zur Verfügung steht:

Rufe TUC\_KON\_256 {"NETWORKVPN\_SIS/DOWN"; Op; Warning;}

Der Hersteller des Konnektor MUSS sicherstellen, dass eine Anbindung an einen Konzentrador ausschließlich dann möglich ist, wenn (MGM\_LU\_ONLINE = Enabled) gesetzt ist.

Der Administrator des Konnektor MUSS durch die Managementschnittstelle manuell einen Verbindungsaufbau und einen Verbindungsabbau eines VPN-Tunnel zur TI (VPN\_TI) oder zu den SIS (VPN\_SIS) initiieren können.

[<=]

#### TIP1-A\_4779 - Wiederholte Fehler beim VPN-Verbindungsaufbau

Der Konnektor MUSS gewährleisten, dass nach einem Fehler beim VPN-Verbindungsaufbau nicht unmittelbar ein weiterer Versuch des Verbindungsaufbaus durchgeführt wird.

Hierzu MUSS der Hersteller ein inkrementelles (schrittweise anwachsend) Verfahren wählen, welcher den zeitlichen Abstand zwischen einzelnen Versuchen des VPN-Verbindungsaufbau definiert. Dieser Abstand MUSS maximal fünf Minuten betragen. (Diese Pause soll es dem Konnektor ermöglichen, noch ausreichend Ressourcen für die verbleibenden Services zur Verfügung zu stellen).

[<=]

### 4.2.4.2 Durch Ereignisse ausgelöste Reaktionen

#### TIP1-A\_4780 - TI VPN-Client Start Events

Beim Auftreten einer der nachfolgenden Events MUSS der Konnektor den TUC\_KON\_321 „Verbindung zu dem VPN-Konzentrador der TI aufbauen“ starten, sofern auch MGM\_LU\_ONLINE = Enabled.

- Event NETWORKVPN\_TI/DOWN
- Event MGM/LU\_CHANGED/LU\_ONLINE

[<=]

**TIP1-A\_4781 - SIS VPN-Client Start Events**

Beim Auftreten einer der nachfolgenden Events MUSS der Konnektor den TUC\_KON\_322 „Verbindung zu dem VPN-Konzentrator des SIS aufbauen“ starten, sofern ANLW\_INTERNET\_MODUS = SIS, MGM\_LU\_ONLINE = Enabled und die Verbindung VPN-Konzentrator TI aufgebaut ist:

- Event NETWORKVPN\_SIS/DOWN

[&lt;=]

**TIP1-A\_5417 - TI VPN-Client Stop Events**

Beim Auftreten einer der nachfolgenden Events MUSS der Konnektor den VPN-Tunnel zur TI beenden:

- MGM/LU\_CHANGED/LU\_ONLINE mit (Active=Disabled)

[&lt;=]

**TIP1-A\_4782 - SIS VPN-Client Stop Events**

Beim Auftreten einer der nachfolgenden Events MUSS der Konnektor den VPN-Tunnel zum SIS beenden:

- MGM/LU\_CHANGED/LU\_ONLINE mit (Active=Disabled)

[&lt;=]

Hinweis: Wenn der IPsec-Tunnel VPN\_SIS aufgebaut ist, zeigt die Default Route im Konnektor auf die innere Tunnel-IP-Adresse des VPN-Konzentrators SIS. Dies ist bei einer Trennung und dem Wiederaufbau der Verbindung VPN\_TI zu beachten.

**4.2.4.3 Interne TUCs, nicht durch Fachmodule nutzbar**

4.2.4.3.1 TUC\_KON\_321 „Verbindung zu dem VPN-Konzentrator der TI aufbauen“

**TIP1-A\_4783 - TUC\_KON\_321 „Verbindung zu dem VPN-Konzentrator der TI aufbauen“**

Der Konnektor MUSS den technischen Use Case TUC\_KON\_321 „Verbindung zu dem VPN-Konzentrator der TI aufbauen“ umsetzen.

**Tabelle 328: TAB\_KON\_635 – TUC\_KON\_321 „Verbindung zu dem VPN-Konzentrator der TI aufbauen“**

| Element      | Beschreibung   |
|--------------|--|
| Name         | TUC_KON_321 Verbindung zu dem VPN-Konzentrator der TI aufbauen   |
| Beschreibung | Es wird ein IPsec-Tunnel zum VPN-Konzentrator der TI aufgebaut werden. Über den erfolgreichen Aufbau wird per Event informiert.  |
| Auslöser     | Bootup-Phase<br>TUC_KON_305 „LAN-Adapter initialisieren“<br>TUC_KON_306 „WAN-Adapter initialisieren“<br>Event MGM/LU_CHANGED/LU_ONLINE<br>Event NETWORK/VPN/CONFIG_CHANGED<br>Optional: Änderungen |



|                |  |
|----------------|--|
|                | ANLW_AKTIVE_BESTANDSNETZE<br>Manueller Aufruf über Managementschnittstelle   |
| Vorbedingungen | Der TUC_KON_304 „Netzwerk-Routen einrichten“ MUSS fehlerfrei durchgelaufen sein.   |
| Eingangsdaten  |  |
| Komponenten    | Konnektor  |
| Ausgangsdaten  | Der virtuelle Adapter VPN_TI mit der IP-Adresse VPN_TUNNEL_TI_INNER_IP des Konnektors wurde zur Verfügung gestellt. <ul style="list-style-type: none"> <li>Innere Tunnel IP-Adresse des VPN-Konzentrators TI</li> <li>DNS_SERVERS_TI</li> <li>VPN_KONZENTRATOR_TI_IP_ADDRESS</li> <li>DOMAIN_SRVZONE_TI</li> </ul>   |
| Standardablauf | <p>1) Wenn der Auslöser = Event NETWORK/VPN/CONFIG_CHANGED oder eine Änderung von ANLW_AKTIVE_BESTANDSNETZE ist, muss der VPN-Tunnel TI abgebaut werden.</p> <p>2) Wenn der VPN-Tunnel TI noch aktiv ist, ist der Ablauf abgeschlossen. Anderenfalls ist mit Ablaufschritt 3) fortzufahren.</p> <p>3) Prüfen, MGM_LU_ONLINE = Enabled, falls nicht ist der TUC ohne Ausgabe einer Fehlermeldung zu beenden.</p> <p>4) Prüfen, ob die im Konnektor hinterlegte CRL noch gültig ist.<br/>falls nicht, muss der TUC_KON_040 „CRL aktualisieren“ aufgerufen werden, Anschließend erneut prüfen, ob die im Konnektor hinterlegte CRL nun gültig ist.<br/>Falls die CRL nicht gültig ist, ist der TUC mit Fehler zu beenden.</p> <p>5) Aufrufen von TUC_VPN-ZD_0001 „IPsec Tunnel TI aufbauen“<br/>Die folgenden Rückgabewerte des TUC_VPN-ZD_0001 „IPsec Tunnel TI aufbauen“ sind in die laufende Konfiguration des Konnektors zu übernehmen:</p> <ul style="list-style-type: none"> <li>VPN_TUNNEL_TI_INNER_IP</li> <li>DNS_SERVERS_TI</li> </ul> <p>6) Aufrufen von TUC_KON_304 „Netzwerk-Routen einrichten“<br/>Sobald der Tunnel erfolgreich aufgebaut wurde, ist der folgende Event zu generieren:<br/>TUC_KON_256 {"NETWORK/VPN_TI/UP"; Op;</p> |



|                                |   |
|--------------------------------|---|
|                                | Info;IP= \$VPN_TUNNEL_TI_INNER_IP}  |
| Varianten/Alternativen         | Keine   |
| Fehlerfälle                    | (→4) CRL ist abgelaufen (outdated);<br>Herstellerspezifisch kann entweder (4a) oder (4b) umgesetzt werden:<br>(4a) Kritischer Fehlerzustand EC_CRL_Out_Of_Date wurde noch nicht festgestellt: 4173<br>(4b) kritischer Fehlerzustand EC_CRL_Out_Of_Date wurde bereits festgestellt: 4002<br>(->4) Wenn Fehler 4173 bzw. 4002 nicht zutreffen, ist ein herstellerspezifischer Fehler zu verwenden.<br>(→5) VPN-Tunnel konnte nicht aufgebaut werden; Fehlercode: 4174 |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

**Tabelle 329: TAB\_KON\_636 Fehlercodes TUC\_KON\_321 „Verbindung zu dem VPN-Konzentrator der TI aufbauen“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4002  | Security  | Fatal    | Der Konnektor befindet sich in einem kritischen Betriebszustand |
| 4172  | Technical | Fatal    | Es ist keine Online-Verbindung zulässig.                        |
| 4173  | Technical | Fatal    | Die CRL ist nicht mehr gültig (outdated).                       |
| 4174  | Technical | Fatal    | TI-VPN-Tunnel:<br>Verbindung konnte nicht aufgebaut werden      |

[<=]

6262 4.2.4.3.2 TUC\_KON\_322 „Verbindung zu dem VPN-Konzentrator des SIS aufbauen“

6263 **TIP1-A\_4784 - TUC\_KON\_322 „Verbindung zu dem VPN-Konzentrator des SIS**  
6264 **aufbauen“**

6265 Der Konnektor MUSS den technischen Use Case TUC\_KON\_322 „Verbindung zu dem  
6266 VPN-Konzentrator der SIS aufbauen“ umsetzen.

6267

6268 **Tabelle 330: TAB\_KON\_637 – TUC\_KON\_322 „Verbindung zu dem VPN-Konzentrator der**  
6269 **SIS aufbauen“**

| Element        | Beschreibung  |
|----------------|---|
| Name           | TUC_KON_322 Verbindung zu dem VPN-Konzentrator der SIS aufbauen   |
| Beschreibung   | Es muss ein IPsec-Tunnel zum VPN-Konzentrator der SIS aufgebaut werden  |
| Auslöser       | Bootup-Phase<br>TUC_KON_305 „LAN-Adapter initialisieren<br>TUC_KON_306 „WAN-Adapter initialisieren<br>Event NETWORK/VPN/CONFIG_CHANGED<br>Optional: Event MGM/LU_CHANGED/LU_ONLINE<br>Manueller Aufruf über Managementschnittstelle   |
| Vorbedingungen | ANLW_INTERNET_MODUS = SIS<br>Die Verbindung VPN-Konzentrator TI ist aufgebaut.<br>Der TUC_KON_304 „Netzwerk-Routen einrichten“ muss erfolgreich durchgeführt worden sein.   |
| Eingangsdaten  | Keine   |
| Komponenten    | Konnektor   |
| Ausgangsdaten  | Der virtuelle Adapter VPN_SIS mit der IP-Adresse VPN_TUNNEL_SIS_INNER_IP wurde zur Verfügung gestellt. <ul style="list-style-type: none"> <li>Innere Tunnel-IP-Adresse des VPN-Konzentrators SIS</li> <li>VPN_KONZENTRATOR_SIS_IP_ADDRESS</li> <li>DNS_SERVER_SIS</li> </ul>  |
| Standardablauf | 1) Wenn der Auslöser Event NETWORK/VPN/CONFIG_CHANGED ist, muss der VPN-Tunnel SIS abgebaut werden.<br>2) Wenn der VPN-Tunnel SIS noch aktiv ist, ist der Ablauf abgeschlossen. Anderenfalls ist mit Ablaufschritt 3) fortzufahren.<br>3) Prüfen, ob (MGM_LU_ONLINE=Enabled). falls nicht ist der TUC ohne Ausgabe einer Fehlermeldung zu |

|                                |  |
|--------------------------------|--|
|                                | <p>beenden.<br/> 4) entfällt<br/> 5) Prüfen, ob die im Konnektor hinterlegte CRL noch gültig ist.<br/> falls nicht, MUSS der TUC_KON_040 „CRL aktualisieren“ aufgerufen werden, Anschließend erneut prüfen, ob die im Konnektor hinterlegte CRL nun gültig ist.<br/> Falls die CRL nicht gültig ist, ist der TUC mit Fehler zu beenden.<br/> 6) Aufrufen von TUC_VPN-ZD_0002 „IPsec Tunnel SIS aufbauen“<br/> 7) Aufrufen von TUC_KON_304 „Netzwerk-Routen einrichten“<br/> Sobald der Tunnel erfolgreich aufgebaut wurde, ist der folgende Event zu generieren:<br/> TUC_KON_256 {"NETWORK/VPN_SIS/UP"; Op; Info;IP= \$VPN_TUNNEL_SIS_INNER_IP}</p> |
| Varianten/Alternativen         | Keine  |
| Fehlerfälle                    | <p>(→3) Keine Online-Verbindung zulässig; 4172<br/> (→5) CRL ist abgelaufen (outdated);</p> <p>Herstellerspezifisch kann entweder (5a) oder (5b) umgesetzt werden:<br/> (5a) Kritischer Fehlerzustand EC_CRL_Out_Of_Date wurde noch nicht festgestellt: 4173<br/> (5b) kritischer Fehlerzustand EC_CRL_Out_Of_Date wurde bereits festgestellt: 4002</p> <p>(-&gt;5) Wenn Fehler 4173 bzw. 4002 nicht zutreffen, ist ein herstellerspezifischer Fehler zu verwenden.<br/> (→6) VPN Tunnel konnte nicht aufgebaut werden; Fehlercode: 4176</p>   |
| Nichtfunktionale Anforderungen | Keine  |
| Zugehörige Diagramme           | Keine  |

6270  
6271

**Tabelle 331: TAB\_KON\_638 Fehlercodes TUC\_KON\_322 „Verbindung zu dem VPN-Konzentrator der SIS aufbauen“**

| Fehlercode  | ErrorType | Severity | Fehlertext |
|---|-----------|----------|------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |            |

|      |           |       |   |
|------|-----------|-------|---|
| 4002 | Security  | Fatal | Der Konnektor befindet sich in einem kritischen Betriebszustand |
| 4172 | Technical | Fatal | Es ist keine Online-Verbindung zulässig.                        |
| 4173 | Technical | Fatal | Die CRL ist nicht mehr gültig (outdated).                       |
| 4176 | Technical | Fatal | SIS-VPN-Tunnel:<br>Verbindung konnte nicht aufgebaut werden     |

6272

6273 [ $\leq$ ]6274 **4.2.4.4 Interne TUCs, auch durch Fachmodule nutzbar**

6275 Keine

6276 **4.2.4.5 Operationen an der Außenschnittstelle**

6277 Keine

6278 **4.2.4.6 Betriebsaspekte**6279 **TIP1-A\_5415 - Initialisierung „VPN-Client“**

6280 Der Konnektor MUSS in der Bootup-Phase zur Initialisierung des Funktionsmerkmals „VPN-Client“:

- 6282 • die Verbindung zum VPN-Konzentrator TI aufbauen (TUC\_KON\_321)
- 6283 • die Verbindung zum VPN-Konzentrator SIS aufbauen (TUC\_KON\_322)

6284 [ $\leq$ ]6285 **TIP1-A\_4785-03 - Konfigurationsparameter VPN-Client**

6286 Die Managementschnittstelle MUSS es einem Administrator ermöglichen

6287 Konfigurationsänderungen am VPN-Client gemäß Tabelle TAB\_KON\_639 vorzunehmen.

6288 Der Konnektor MUSS bei einer Änderung der Konfigurationswerte den folgenden Event

6289 auslösen:

6290 Rufe TUC\_KON\_256 {"NETWORK/VPN/CONFIG\_CHANGED"; Op; Info;; doDisp = false}

6291

6292 **Tabelle 332: TAB\_KON\_639 – Konfigurationsparameter VPN-Client**

| ReferenzID           | Belegung         | Bedeutung und Administrator-Interaktion  |
|----------------------|------------------|--|
| IKE_KEEPA_LIVE_MODUS | Enabled/Disabled | Der Administrator MUSS einstellen können, ob IKE Keep-Alive-Pakete gesendet werden.<br>Ein Hinweis MUSS ausgegeben werden, dass dies bei Nutzung von Dial-Up-Verbindungen nicht zu empfehlen ist. Dies dient der Vermeidung von Kosten bei Nutzung eines Internetzugangs ohne Flatrate.<br>Default-Wert: Enabled |

|                                   |                    |  |
|-----------------------------------|--------------------|--|
| IKE_KEEPAIVE_INTERVAL             | X Sekunden         | Der Administrator MUSS die Zeit in Sekunden angeben können, nach der ein neues IKE Keep-Alive-Paket gesendet wird.<br>Default-Wert: 30   |
| IKE_KEEPAIVE_RETRY                | X                  | Der Administrator MUSS angeben können, nach wie vielen IKE Keep-Alive-Paketen ohne Acknowledge Message die Verbindung beendet wird.<br>Default-Wert: 3   |
| VPN_IDLE_TIMEOUT_MODUS            | Enabled/Disabled   | Der Administrator MUSS einstellen können, ob nach Inaktivität die VPN-Verbindung automatisch abgebaut werden soll.<br>Ein Hinweis MUSS ausgegeben werden, dass dies insbesondere bei Nutzung von Dial-Up-Verbindungen Enabled werden sollte.<br>Default-Wert: Disabled |
| VPN_IDLE_TIMEOUT                  | X Sekunden         | Der Administrator MUSS die Zeit in Sekunden angeben können, nach der eine inaktive VPN-Verbindung zu einem Abbau der Verbindung führt.<br>Default-Wert: 600  |
| NAT_KEEPAIVE_MODUS                | Enabled/Disabled   | Der Administrator MUSS einstellen können, ob NAT Keep-Alive-Pakete gesendet werden.<br>Ein Hinweis MUSS ausgegeben werden, dass dies insbesondere bei Nutzung von Dial-Up-Verbindungen nicht zu empfehlen ist.<br>Default-Wert: Enabled                                |
| NAT_KEEPAIVE_INTERVAL             | X Sekunden         | Der Administrator MUSS die Zeit in Sekunden angeben können, nach der ein neues NAT Keep-Alive-Paket gesendet wird.<br>Default-Wert: 20   |
| VPN_KONZENTRATOR_TI_IP_ADDRESS    | IP-Adresse         | IP-Adresse des VPN-Konzentrators TI im Transportnetz zu dem der IPsec-Tunnel VPN_TI aufgebaut wird. Der Wert kann vom Administrator nur eingesehen werden.   |
| VPN_KONZENTRATOR_SIS_IP_ADDRESSES | IP-Adresse         | IP-Adresse des VPN-Konzentrators SIS im Transportnetz zu dem der IPsec-Tunnel VPN_SIS aufgebaut wird. Der Wert kann vom Administrator nur eingesehen werden.   |
| VPN_TI_MTU                        | Paketgröße in Byte | Der Administrator MUSS die MTU für ESP-Pakete zur TI (excl. ESP-Header-Size) in den Grenzen von 576 bis 8076 konfigurieren können.<br>Default-Wert: 1318   |
| VPN_SIS_MTU                       | Paketgröße in Byte | Der Administrator MUSS die MTU für ESP Pakete zum SIS (excl. ESP-Header-Size) in den Grenzen von 576 bis 8076 konfigurieren können.<br>Default-Wert: 1318  |

|              |                  |  |
|--------------|------------------|--|
| HASH_AND_URL | Enabled/Disabled | Der Administrator MUSS die Nutzung des hash&URL-Verfahrens zum Zertifikatsaustausch konfigurieren können.<br>Wenn HASH_AND_URL = Enabled gesetzt ist, wird die URL für das hash&URL-Verfahren automatisch durch DNS SRV- und TXT-Anfragen mit Owner „_hashandurl._tcp.<DNS_DOMAIN_VPN_ZUGD_I NT>“ ermittelt.<br>Default-Wert: Disabled |
|--------------|------------------|--|

6293 [ $\leq$ ]

6294

6295 **4.2.5 Zeitdienst**

6296 Der Zeitdienst schafft die Grundlage einer gleichen Systemzeit für alle in der TI  
 6297 einzusetzenden Produkttypen. Grundsätzlich ist ein NTP-Server der Stratum-3-Ebene  
 6298 innerhalb des Konnektors erforderlich, welcher die Zeitangaben eines NTP-Servers  
 6299 Stratum-2-Ebene abfragt (GS-A\_3942). Die in [gemSpec\_Net#5.1] „NTP-Topologie“  
 6300 getroffenen Anforderungen werden durch dieses Kapitel erweitert.

6301 Innerhalb des Zeitdienstes werden folgende Präfixe für Bezeichner verwendet:

- 6302 • Events (Topic Ebene 1): „NTP“
- 6303 • Konfigurationsparameter: „NTP\_“

6304 **4.2.5.1 Funktionsmerkmalweite Aspekte**6305 **TIP1-A\_4786 - Maximale Zeitabweichung**

6306 Falls der Leistungsumfang Online nicht aktiviert ist (MGM\_LU\_ONLINE=Disabled), MUSS  
 6307 sichergestellt werden, dass der maximale zulässige Fehler von +/- 20ppm (part per  
 6308 million) gegenüber einer Referenzuhr nicht überschritten wird. Dies entspricht einer  
 6309 maximalen Abweichung im Freilauf von +/- 34,56 Sekunden über 20 Tage.

6310 [ $\leq$ ]6311 **TIP1-A\_4787 - Konfigurationsabhängige Funktionsweise**

6312 Der NTP-Server des Konnektors MUSS deaktiviert sein, falls der Konnektor  
 6313 Leistungsumfang Online nicht aktiviert ist (MGM\_LU\_ONLINE=Disabled).

6314 [ $\leq$ ]

6315 Falls die Systemzeit des Konnektors zu stark von der Zeit der zentralen TI-Plattform  
 6316 abweicht, deutet dies auf ein schwerwiegendes Problem im Konnektor oder der  
 6317 Umgebung hin, da dies im ordnungsgemäßen Betrieb nicht auftreten sollte.

6318 **TIP1-A\_4788 - Verhalten bei Abweichung zwischen lokaler Zeit und erhaltenen Zeit**

6319 Der Konnektor DARF die im Konnektor vorgehaltene Systemzeit im Rahmen einer  
 6320 automatisierten Synchronisation NICHT aktualisieren, wenn die lokale Zeit von der im  
 6321 Rahmen der Synchronisation erhaltenen Zeit um mehr als NTP\_MAX\_TIMEDIFFERENCE  
 6322 abweicht. Dies betrifft NICHT Änderungen in der Darstellung der Systemzeit, die  
 6323 zeitzonenbedingt sind (MEZ -> MESZ -> MEZ), da die Zeitsynchronisation grundsätzlich  
 6324 UTC berücksichtigt. Bei einer erstmaligen Synchronisierung nach dem Boot-Vorgang oder  
 6325 bei einer erstmaligen Synchronisierung bei der Inbetriebnahme des Konnektors darf eine  
 6326 Synchronisation trotz einer Zeitabweichung größer einer Stunde durchgeführt werden.  
 6327 Daher MUSS der Konnektor bei einer Abweichung von mehr als einer Stunde in den  
 6328 kritischen Betriebszustand EC\_TIME\_DIFFERENCE\_INTOLERABLE übergehen, ein weiterer  
 6329

6330 fachlicher Betrieb des Konnektors DARF NICHT mehr erfolgen.

6331 [ $\leq$ ]

6332 Der kritische Betriebszustand kann anschließend über einen manuellen Eingriff (z. B.  
6333 Reboot) behoben werden (siehe 3.3 Betriebszustand).

6334 **TIP1-A\_4789 - Zustandsvariablen des Konnektor Zeitdiensts**

6335 TAB\_KON\_640 listet die zu verwendenden Zustandsvariablen des Konnektor NTP-  
6336 Servers. Diese Werte DÜRFEN NICHT durch den Administrator geändert werden.

6337

6338 **Tabelle 333: TAB\_KON\_640 Zustandswerte für Konnektor NTP-Server**

| ReferenzID             | Belegung | Zustandswerte  |
|------------------------|----------|--|
| NTP_WARN_PERIOD        | 30       | Anzahl an Tagen nach der ersten erfolglosen Zeitsynchronisierung nach der eine Warnung an den Betreiber erfolgen soll  |
| NTP_GRACE_PERIOD       | 50       | Anzahl an Tagen nach der ersten erfolglosen Zeitsynchronisierung nach welcher der Konnektor in einen kritischen Betriebszustand übergehen muss.<br>Dieser Parameter wirkt nur bei MGM_LU_ONLINE = Enabled. |
| NTP_MAX_TIMEDIFFERENCE | 3600     | Maximale Zeitabweichung in Sekunden zwischen Systemzeit und Zeit des Stratum-2-Zeitserver zum Zeitpunkt der Zeitsynchronisierung.<br>Dieser Parameter wirkt nur bei MGM_LU_ONLINE = Enabled.               |

6339

6340 [ $\leq$ ]

6341 **4.2.5.2 Durch Ereignisse ausgelöste Reaktionen**

6342 Keine.

6343 **4.2.5.3 Interne TUCs, nicht durch Fachmodule nutzbar**

6344 Keine.

6345 **4.2.5.4 Interne TUCs, auch durch Fachmodule nutzbar**

6346 **4.2.5.4.1 TUC\_KON\_351 "Liefere Systemzeit"**

6347 **TIP1-A\_4790 - TUC\_KON\_351 „Liefere Systemzeit“**

6348 Der Konnektor MUSS den technischen Use Case TUC\_KON\_351 „Liefere Systemzeit“  
6349 umsetzen.

6350

6351 **Tabelle 334: TAB\_KON\_776 TUC\_KON\_351 „Liefere Systemzeit“**

| Element | Beschreibung                     |
|---------|----------------------------------|
| Name    | TUC_KON_351 „Liefere Systemzeit“ |

|                                |   |
|--------------------------------|---|
| Beschreibung                   | Der Konnektor MUSS die Systemzeit auf Anforderung an Fachmodule liefern können.   |
| Anwendungsumfeld               | Den Fachanwendungen ist die Systemzeit zu liefern.  |
| Eingangsanforderung            | Die Echtzeituhr des Konnektors wurde gemäß den geforderten Synchronisationsintervallen aktualisiert (bei MGM_LU_ONLINE=Enabled) oder manuell gesetzt (bei MGM_LU_ONLINE=Disabled) |
| Auslöser und Vorbedingungen    | Fachmodule benötigen die aktuelle Systemzeit des Konnektors.  |
| Eingangsdaten                  | Echtzeituhr des Konnektors  |
| Komponenten                    | Konnektor, Fachmodule   |
| Ausgangsdaten                  | Systemzeit des Konnektors   |
| Standardablauf                 | Siehe [gemSpec_Net]   |
| Varianten/Alternativen         | Keine   |
| Fehlerfälle                    | 4178: Konnektor retourniert keine Systemzeit  |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

6352 **Tabelle 335: TAB\_KON\_641 Fehlercodes TUC\_KON\_351 „Liefere Systemzeit“**

| Fehlercode | ErrorType | Severity | Fehlertext  |
|------------|-----------|----------|---|
| 4178       | Technical | Error    | Das Fachmodul konnte die aktuelle Systemzeit des Konnektors nicht abrufen |

6353

6354 [ $\leq$ ]

#### 6355 4.2.5.5 Operationen an der Außenschnittstelle

##### 6356 4.2.5.5.1 Sync\_Time

##### 6357 **TIP1-A\_4791 - Operation sync\_Time**

6358 Der NTP-Server des Konnektors MUSS an der Client-Schnittstelle eine Operation  
6359 sync\_Time anbieten.

6360

6361 **Tabelle 336: TAB\_KON\_642 Operation sync\_Time**

| Name            | I_NTP_Time_Information:sync_Time  |
|-----------------|---|
| Beschreibung    | Der Konnektor MUSS anfragenden Clients (z.B. Arztarbeitsplatz) per NTP-Version 4 die Systemzeit liefern |
| Aufrufparameter | Vgl. [NTPv4]  |
| Rückgabe        | Vgl. [NTPv4]  |



|                 |   |
|-----------------|---|
| Vorbedingungen  | MGM_LU_ONLINE=Enabled   |
| Nachbedingungen | Der anfragende Client hat die korrekte Zeit geliefert bekommen.             |
| Hinweise        | Keine   |
| Fehler          | Der Aufruf schlägt fehl (bleibt unbeantwortet), wenn MGM_LU_ONLINE=Disabled |

[&lt;=]

#### 4.2.5.6 Betriebsaspekte

##### TIP1-A\_4792 - Explizites Anstoßen der Zeitsynchronisierung

Der Konnektor MUSS dem Administrator die Möglichkeit bieten, eine Synchronisation mit dem zentralen Zeitdienst explizit anzustoßen.

[&lt;=]

##### TIP1-A\_4793 - Konfigurierbarkeit des Konnektor NTP-Servers

Der Administrator MUSS die in TAB\_KON\_643 aufgelisteten Parameter über die Managementschnittstelle konfigurieren und die in TAB\_KON\_730 aufgelisteten Parameter ausschließlich einsehen können.

**Tabelle 337: TAB\_KON\_643 Konfiguration des Konnektor NTP-Servers**

| ReferenzID   | Belegung | Bedeutung   |
|--------------|----------|---|
| NTP_TIMEZONE | Zeitzone | Der Administrator MUSS die Zeitzone des Konnektors einstellen können.<br>Default-Wert: Central European Time/Mitteuropäische Zeit (CET/MEZ) |
| NTP_TIME     | Zeit     | Der Administrator MUSS die Zeit des Konnektors (NTP_TIME) über die Managementschnittstelle manuell einstellen können.                       |

**Tabelle 338: TAB\_KON\_730 Einsehbare Konfigurationsparameter des Konnektor NTP-Servers**

| ReferenzID      | Belegung    | Bedeutung   |
|-----------------|-------------|---|
| NTP_SERVER_ADDR | IP-Adressen | Die Adressen des primären und sekundären Stratum-2-Zeitserver der zentralen TI-Plattform für die Synchronisation mit dem NTP-Server des Konnektors. |

[&lt;=]

##### TIP1-A\_4794 - Warnung und Übergang in kritischen Betriebszustand bei nichterfolgter Zeitsynchronisierung

Befindet sich der Konnektor im Zustand EC\_TIME\_SYNC\_PENDING\_CRITICAL oder EC\_Time\_Difference\_Intolerable, MUSS der Administrator eine Korrektur oder Bestätigung der Systemzeit vornehmen können. Anschließend MUSS der Konnektor wie nach einer erfolgreichen Zeitsynchronisation verfahren, d. h. der Tagezähler wird auf 0 zurückgesetzt.

[&lt;=]

## 6387 4.2.5.6.1 TUC\_KON\_352 Initialisierung Zeitdienst

6388 **TIP1-A\_4795 - TUC\_KON\_352 „Initialisierung Zeitdienst“**

6389 Der Konnektor MUSS in der Bootup-Phase TUC\_KON\_352 "Initialisierung Zeitdienst"

6390 durchlaufen.

6391

6392 **Tabelle 339: TAB\_KON\_644 – TUC\_KON\_352 „Initialisierung Zeitdienst“**

| Element                | Beschreibung   |
|------------------------|--|
| Name                   | TUC_KON_352 „Initialisierung Zeitdienst“   |
| Beschreibung           | Der Konnektor muss zum Bootup den konnektoreigenen NTP-Server mit einem NTP-Server der zentralen TI-Plattform synchronisieren falls MGM_LU_ONLINE=Enabled.   |
| Anwendungsumfeld       | Synchronisierung der Systemzeit zur Startzeit  |
| Eingangsanforderung    | Keine  |
| Auslöser               | <ul style="list-style-type: none"> <li>• Bootup</li> <li>• Event NETWORK/VPN_TI/UP</li> </ul>  |
| Vorbedingungen         | Verbindung zum VPN-Konzentrator TI muss aufgebaut sein   |
| Eingangsdaten          | NTP-Server der zentralen TI-Plattform  |
| Komponenten            | Konnektor  |
| Ausgangsdaten          | Keine  |
| Standardablauf         | Falls MGM_LU_ONLINE=Enabled: <ul style="list-style-type: none"> <li>• Durch eine DNS-Anfrage an den DNS-Forwarder zur Auflösung des SRV-RR mit dem Bezeichner "_ntp._udp.&lt;DOMAIN_SRVZONE_TI&gt;„ erhält der Konnektor Adressen der NTP-Server der zentralen TI-Plattform.</li> <li>• gemäß [NTPv4]</li> <li>• Falls keine Antwort erfolgt ist oder falls der Zeitserver nicht erreichbar ist, wird Fehler 4177 ausgelöst. Zur Feststellung werden die NTPv4 eigenen Timeoutwerte berücksichtigt.</li> </ul> |
| Varianten/Alternativen | Keine  |

|                                |   |
|--------------------------------|---|
| Fehlerfälle                    | 4177: Der NTP-Server des Konnektors empfängt keine Systemzeit |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

6393 **Tabelle 340: TAB\_KON\_645 Fehlercodes TUC\_KON\_352 „Initialisierung Zeitdienst“**

| Fehlercode | ErrorType | Severity | Fehlertext  |
|------------|-----------|----------|---|
| 4177       | Technical | Warning  | Der NTP-Server des Konnektors konnte nicht synchronisiert werden. |

6394

6395 [**<=**]

## 6396 4.2.6 Namensdienst und Dienstlokalisierung

6397 Innerhalb des Namensdienstes werden folgende Präfixe für Bezeichner verwendet:

6398 • Events (Topic Ebene 1): keine Events vorhanden

6399 • Konfigurationsparameter: „DNS\_“

### 6400 4.2.6.1 Funktionsmerkmalweite Aspekte

#### 6401 **TIP1-A\_4796 - Grundlagen des Namensdienstes**

6402 Der Konnektor MUSS einen Recursive Caching Nameserver zur Auflösung von DNS-Anfragen sowie einen autoritativen Nameserver zur Verwaltung der Zone „konlan.“ bereitstellen.

6405 Der Caching-Nameserver des Konnektors MUSS für Clientsysteme aus dem lokalen Netzwerk (ANLW\_LAN\_NETWORK\_SEGMENT oder ANLW\_LEKTR\_INTRANET\_ROUTES) erreichbar sein.

6408 Der Caching-Nameserver des Konnektors MUSS einen Timeout für die Bearbeitung von DNS-Abfragen beachten. Konnte eine DNS-Abfrage nicht durchgeführt werden, MUSS die Bearbeitung abgebrochen werden.

6411 [**<=**]

#### 6412 **TIP1-A\_6480 - Resource Records der Zone konlan.**

6413 Der Konnektor MUSS in der Zone „konlan.“ die folgenden Resource Records bereitstellen:

- 6414 • label: „konnektor.konlan.“, ttl: <Time To Live>, class: IN, type: A, rdata:
- 6415 <LAN-seitige IP-Adresse des Konnektors>

6416 Die in spitzen Klammern angegebenen Werte müssen implementierungs- und konfigurationsabhängig vergeben werden.

6418 [**<=**]

#### 6419 **TIP1-A\_4797 - DNS-Forwards des DNS-Servers**

6420 Der DNS-Server des Konnektors MUSS die folgenden DNS-Forwards durchführen:

6421 **Tabelle 341: TAB\_KON\_687 DNS-Forwards des DNS-Servers**

| Domain   | Forwarders   | Bemerkungen  |
|--|--|--|
| Namensraum TI, *.DNS_TOP_LEVEL_DOMAIN_TI   | DNS_SERVERS_TI   | DNS Forward Rule zur Auflösung aller DNS-Namen innerhalb des Namensraums der TI mit der Top Level Domain telematik (für die PU) und telematik-test (für die RU und TU).            |
| Namensraum TI, Top Level Domain ti-wa (PU) und ti-wa-test (RU und TU).   | DNS_SERVERS_TI   | DNS Forward Rule zur Auflösung aller DNS-Namen innerhalb des Namensraums der TI mit der Top Level Domain ti-wa (für die PU) und ti-wa-test (für die RU und TU).                    |
| Namensraum angeschlossene Netze des Gesundheitswesens mit aAdG-NetG (Domainnamen von angeschlossenen Netzen des Gesundheitswesens mit aAdG-NetG gemäß Bestandsnetze.xml) | DNS_SERVERS_BESTANDSNETZE<br>(Je Domainnamen eines angeschlossenen Netzes des Gesundheitswesens mit aAdG-NetG alle zugehörigen DNS-Server IP-Adressen gemäß Bestandsnetze.xml) | Je angeschlossenes Netz des Gesundheitswesens mit aAdG-NetG in ANLW_AKTIVE_BESTANDSNETZE wird eine DNS Forward Rule zur Auflösung von DNS-Namen innerhalb dieses Netzes verwendet. |
| Namensraum lokale Einsatzumgebung (DNS_DOMAIN_LEKTR)   | DNS_SERVERS_LEKTR  | DNS Forward Rule zur Auflösung aller DNS-Namen innerhalb der DNS-Domain DNS_DOMAIN_LEKTR   |
| Namensraum Internet  | DNS_SERVERS_SIS  | Wenn der VPN-Tunnel SIS aktiv ist, muss eine Forward Rule für den Namensraum Internet über die DNS_SERVERS_SIS existieren.   |
| Namensraum Internet  | DNS_SERVERS_INT  | Wenn der VPN-Tunnel SIS nicht aktiv ist, muss eine Forward Rule für den Namensraum Internet über die DNS_SERVERS_INT existieren  |
| Lokale Zone „konlan.“  | autoritativer Nameserver des Konnektors  | DNS Forward Rule zur Auflösung aller DNS-Namen innerhalb der Zone „konlan.“  |

6422

6423 [**<=**]

6424 **TIP1-A\_4798 - DNS Stub-Resolver**

6425 Der Stub-Resolver im Konnektor MUSS von allen internen Diensten zur Namensauflösung  
6426 genutzt werden.

6427 Der Stub-Resolver im Konnektor MUSS immer den Caching-Nameserver im Konnektor  
6428 anfragen.

6429 [ $\leq$ ]

6430 **TIP1-A\_4799 - Aktualität der DNS-Vertrauensanker sicherstellen**

6431 Der Konnektor, der einen Caching Nameserver als Validating Resolver umsetzt, MUSS  
6432 den DNSSEC-Vertrauensanker der TI aus dem Zertifikatspeicher in den Caching-  
6433 Nameserver übernehmen, wenn ein Fehler bei der Validierung der Namensauflösung der  
6434 TI aufgetreten ist. [ $\leq$ ]

6435

6436 **4.2.6.2 Durch Ereignisse ausgelöste Reaktionen**

6437 Keine.

6438 **4.2.6.3 Interne TUCs, nicht durch Fachmodule nutzbar**

6439 Keine.

6440 **4.2.6.4 Interne TUCs, auch durch Fachmodule nutzbar**

6441 *4.2.6.4.1 TUC\_KON\_361 „DNS-Namen auflösen“*

6442 **TIP1-A\_4801 - TUC\_KON\_361 „DNS-Namen auflösen“**

6443 Der Konnektor MUSS den technischen Use Case TUC\_KON\_361 „DNS-Namen auflösen“  
6444 umsetzen.

6445

6446 **Tabelle 342: TAB\_KON\_646 – TUC\_KON\_361 „DNS-Namen auflösen“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_361 „DNS-Namen auflösen“   |
| Beschreibung   | Ein FQDN wird in ein oder mehrere IPs aufgelöst  |
| Auslöser       | interne Anfrage (Basisdienst oder Fachmodul)   |
| Vorbedingungen | Die vom Konnektor zu verwendenden DNS-Server (DNS_SERVERS_INT, DNS_SERVERS_TI, DNS_SERVERS_SIS, DNS_SERVERS_BESTANDSNETZE) müssen konfiguriert sein. |
| Eingangsdaten  | FQDN (Name, für den die IP-Adressen ermittelt werden sollen)   |

|                                |  |
|--------------------------------|--|
| Komponenten                    | Konnektor  |
| Ausgangsdaten                  | LIST_OF_IP_ADDRESSES   |
| Standardablauf                 | 1) Mit dem FQDN wird eine Anfrage an den Stub-Resolver des Konnektors (Typ A und AAAA) durchgeführt.<br>Für alle ermittelten IPv4-Adressen und IPv6-Adressen werden als LIST_OF_IP_ADDRESSES zurückgeliefert.<br>Da IPv6 nicht produktiv eingesetzt wird muss die aufrufende Instanz die IPv6-Adressen ignorieren.<br>Falls keine IP-Adressen ermittelt werden konnten, wird eine leere Liste zurückgeliefert. |
| Varianten/Alternativen         | Keine  |
| Fehlerfälle                    | (→ 1) Timeout der Anfrage; Fehlercode 4179<br>(→ 1) DNS-Fehler; Fehlercode 4180  |
| Nichtfunktionale Anforderungen | Keine  |
| Zugehörige Diagramme           | Keine  |

6447 **Tabelle 343: TAB\_KON\_647 Fehlercodes TUC\_KON\_361 „DNS Namen auflösen“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4179  | Technical | Error    | „DNS: Anfrage wurde wegen Timeout abgebrochen.“   |
| 4180  | Technical | Fatal    | „DNS: Es ist ein Fehler bei der Namensauflösung aufgetreten“<br>Die Fehlerdetails sind gemäß DNS-Protokoll zu ergänzen. |

6448 [**<=**]

6449 **TIP1-A\_4801-02 - ab PTV4: TUC\_KON\_361 „DNS-Namen auflösen“**

6450 Der Konnektor MUSS den technischen Use Case TUC\_KON\_361 „DNS-Namen auflösen“  
6451 umsetzen.

6452

6453 **Tabelle 344: TAB\_KON\_646 – TUC\_KON\_361 „DNS-Namen auflösen“**

| Element      | Beschreibung                                    |
|--------------|---|
| Name         | TUC_KON_361 „DNS-Namen auflösen“                |
| Beschreibung | Ein FQDN wird in ein oder mehrere IPs aufgelöst |

|                                |  |
|--------------------------------|--|
| Auslöser                       | interne Anfrage (Basisdienst oder Fachmodul)   |
| Vorbedingungen                 | Die vom Konnektor zu verwendenden DNS-Server (DNS_SERVERS_INT, DNS_SERVERS_TI, DNS_SERVERS_SIS, DNS_SERVERS_BESTANDSNETZE) müssen konfiguriert sein.   |
| Eingangsdaten                  | FQDN (Name, für den die IP-Adressen ermittelt werden sollen)   |
| Komponenten                    | Konnektor  |
| Ausgangsdaten                  | LIST_OF_IP_ADDRESSES   |
| Standardablauf                 | 1) Mit dem FQDN wird eine Anfrage an den Stub-Resolver des Konnektors (Typ A und AAAA) durchgeführt.<br>Für alle ermittelten IPv4-Adressen und IPv6-Adressen werden als LIST_OF_IP_ADDRESSES zurückgeliefert.<br>Wird IPv6 nicht produktiv eingesetzt, muss die aufrufende Instanz die IPv6-Adressen ignorieren.<br>Falls keine IP-Adressen ermittelt werden konnten, wird eine leere Liste zurückgeliefert. |
| Varianten/Alternativen         | Keine  |
| Fehlerfälle                    | (→ 1) Timeout der Anfrage; Fehlercode 4179<br>(→ 1) DNS-Fehler; Fehlercode 4180  |
| Nichtfunktionale Anforderungen | Keine  |
| Zugehörige Diagramme           | Keine  |

6454 **Tabelle 345: TAB\_KON\_647 Fehlercodes TUC\_KON\_361 „DNS Namen auflösen“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4179  | Technical | Error    | „DNS: Anfrage wurde wegen Timeout abgebrochen.“   |
| 4180  | Technical | Fatal    | „DNS: Es ist ein Fehler bei der Namensauflösung aufgetreten“<br>Die Fehlerdetails sind gemäß DNS-Protokoll zu ergänzen. |

6455  
6456 **[<=]**

6457 4.2.6.4.2 TUC\_KON\_362 „Liste der Dienste abrufen“

6458 **TIP1-A\_4802 - TUC\_KON\_362 „Liste der Dienste abrufen“**

6459 Der Konnektor MUSS den technischen Use Case TUC\_KON\_362 „Liste der Dienste  
6460 abrufen“ umsetzen.

6461 **Tabelle 346: TAB\_KON\_648 – TUC\_KON\_362 „Liste der Dienste abrufen“**

| Element                        | Beschreibung  |
|--------------------------------|---|
| Name                           | TUC_KON_362 „Liste der Dienste abrufen“   |
| Beschreibung                   | Ermittlung aller zu einer DNS-SD-Gruppe gehörenden DNS-Namen.                         |
| Auslöser                       | interne Anfrage (Basisdienst oder Fachmodul)  |
| Vorbedingungen                 | Die vom Konnektor zu verwendenden DNS-Server müssen konfiguriert sein.                |
| Eingangsdaten                  | FQDN des PTR Resource Records   |
| Komponenten                    | Konnektor   |
| Ausgangsdaten                  | LIST_OF_SRV_ENTITIES  |
| Standardablauf                 | Mit dem FQDN wird eine Typ „PTR“ Anfrage an den Stub-Resolver des Konnektor gestellt. |
| Varianten/Alternativen         | Keine   |
| Fehlerfälle                    | (→ 1) Timeout der Anfrage; Fehlercode 4179<br>(→ 1) DNS-Fehler; Fehlercode 4180       |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

6462

6463

6464 **Tabelle 347: TAB\_KON\_649 Fehlercodes TUC\_KON\_362 „Liste der Dienste abrufen“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4179  | Technical | Error    | „DNS: Anfrage wurde wegen Timeout abgebrochen.“   |
| 4180  | Technical | Fatal    | „DNS: Es ist ein Fehler bei der Namensauflösung aufgetreten“<br>Die Fehlerdetails sind gemäß [gemSpec_Net] zu ergänzen. |

6465

6466 [**<=**]



6467 4.2.6.4.3 TUC\_KON\_363 „Dienstdetails abrufen“

6468 **TIP1-A\_4803 - TUC\_KON\_363 „Dienstdetails abrufen“**

6469 Der Konnektor MUSS den technischen Use Case TUC\_KON\_363 „Dienstdetails abrufen“  
6470 umsetzen.

6471

6472 **Tabelle 348: TAB\_KON\_650 - TUC\_KON\_363 „Dienstdetails abrufen“**

| Element                        | Beschreibung  |
|--------------------------------|---|
| Name                           | TUC_KON_363 Dienstdetails abrufen   |
| Beschreibung                   | Ermitteln aller DNS-SD-Details zu einem vollqualifizierten DNS-Namen.   |
| Auslöser                       | interne Anfrage (Basisdienst oder Fachmodul)  |
| Vorbedingungen                 | Die vom Konnektor zu verwendenden DNS-Server müssen konfiguriert sein.  |
| Eingangsdaten                  | FQDN (der Name eines DNS-SD-Elements)   |
| Komponenten                    | Konnektor   |
| Ausgangsdaten                  | LIST_OF_SRV_ENTRIES<br>LIST_OF_SRV_DETAILS  |
| Standardablauf                 | <p>1) Mit dem FQDN wird eine Typ-„SRV“-Anfrage an den Stub-Resolver des Konnektors gestellt. Die vom DNS-Server zurück gelieferten SRV-Einträge werden als LIST_OF_SRV_ENTRIES (bestehend aus TTL, Priority, Weight, Port, Target) zurückgeliefert. Wenn kein Eintrag gefunden werden konnte, wird eine leere Liste LIST_OF_SRV_ENTRIES zurückgeliefert.</p> <p>2) Mit dem FQDN wird zusätzlich eine Typ-„TXT“-Anfrage an den Stub-Resolver des Konnektors gestellt. Wenn ein oder mehrere entsprechende Einträge gefunden werden konnten, werden diese in einer gemeinsamen Liste LIST_OF_SRV_DETAILS (bestehend aus TTL und TXT) zusammengefasst. Wenn kein Eintrag gefunden werden konnte, wird eine leere Liste LIST_OF_SRV_DETAILS zurückgeliefert. Falls keine FQDN ermittelt werden konnten, wird je eine leere Liste LIST_OF_SRV_ENTRIES und LIST_OF_SRV_DETAILS zurückgeliefert.</p> |
| Varianten/Alternativen         | Keine   |
| Fehlerfälle                    | (→ 1-2) Timeout der Anfrage; Fehlercode 4179<br>(→ 1-2) DNS Fehler; Fehlercode 4180   |
| Nichtfunktionale Anforderungen | Keine   |
| Zugehörige Diagramme           | Keine   |

6473 **Tabelle 349: TAB\_KON\_651 Fehlercodes TUC\_KON\_363 „Dienstdetails abrufen“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4179  | Technical | Error    | „DNS: Anfrage wurde wegen Timeout abgebrochen.“   |
| 4180  | Technical | Fatal    | „DNS: Es ist ein Fehler bei der Namensauflösung aufgetreten“<br>Die Fehlerdetails sind gemäß [gemSpec_Net] zu ergänzen. |

6474  
6475 [ $\leq$ ]

6476 **4.2.6.5 Operationen an der Außenschnittstelle**6477 **TIP1-A\_4804 - Basisanwendung Namensdienst**

6478 Der Konnektor MUSS für Clients eine Basisanwendung Namensdienst anbieten.

6479 **Tabelle 350: TAB\_KON\_652 Basisanwendung Namensdienst**

|                          |  |   |
|--------------------------|--|---|
| <b>Name</b>              | Namendienst  |   |
| <b>Version</b>           | wird im Produktsteckbrief des Konnektors definiert |   |
| <b>Namensraum</b>        | Keiner   |   |
| <b>Namensraum-Kürzel</b> | Keiner   |   |
| <b>Operationen</b>       | Name   | Kurzbeschreibung  |
|                          | GetIPAddress                                       | Diese Operation ermöglicht die Auflösung von FQDNs in IP-Adressen |
| <b>WSDL</b>              | Keines   |   |
| <b>Schema</b>            | Keines   |   |

6480  
6481 [ $\leq$ ]

6482 **4.2.6.5.1 GetIPAddress**6483 **TIP1-A\_5035 - Operation GetIPAddress**

6484 Der Namensdienst des Konnektors MUSS an der Client-Schnittstelle eine Operation  
6485 GetIPAddress anbieten.

6486

6487 **Tabelle 351: TAB\_KON\_653 Operation GetIPAddress**

|             |              |
|-------------|--------------|
| <b>Name</b> | GetIPAddress |
|-------------|--------------|

|                        |   |
|------------------------|---|
| <b>Beschreibung</b>    | Diese Operation ermöglicht die Auflösung von FQDN in IP-Adressen.<br>(DNS-Forwarder Abfrage ohne Cache)   |
| <b>Aufrufparameter</b> | Address (FQDN)<br>DNSSECValidation (Boolean)  |
| <b>Rückgabe</b>        | IPAddr (IPAddress)<br>DNSSECValidated (Boolean)   |
| <b>Vorbedingungen</b>  | Der DNS-Server im Konnektor muss aktiv sein.<br>Die Forward Nameserver (DNS_SERVERS_TI, DNS_SERVERS_SIS, DNS_SERVERS_BESTANDSNETZE) müssen konfiguriert sein. |
| <b>Nachbedingungen</b> | Keine   |
| <b>Standardablauf</b>  | Für Details zu DNS Namensauflösung wird auf [gemSpec_Net] verwiesen.  |

6488

6489 [**<=**]6490 **4.2.6.6 Betriebsaspekte**6491 **TIP1-A\_5416 - Initialisierung „Namensdienst und Dienstlokalisierung“**

6492 Der Konnektor MUSS in der Bootup-Phase zur Initialisierung des Funktionsmerkmals  
6493 „Namensdienst und Dienstlokalisierung“:

- 6494
  - den autoritativen Nameserver starten
- 6495
  - den Caching-Nameserver starten.

6496 [**<=**]6497 **TIP1-A\_4805 - Konfigurationsparameter Namensdienst und Dienstlokalisierung**

6498 Der Administrator MUSS die in TAB\_KON\_654 aufgelisteten Parameter über die  
6499 Managementschnittstelle konfigurieren und die in TAB\_KON\_731 aufgelisteten Parameter  
6500 ausschließlich einsehen können.

6501 Nach jeder Änderung MUSS sichergestellt werden, dass die Änderungen sofort am  
6502 autoritativen bzw. am Caching-Nameserver zur Verfügung stehen.  
6503

6504 **Tabelle 352: TAB\_KON\_654 - Konfigurationsparameter Namensdienst**

| ReferenzID              | Belegung                             | Bedeutung und Administrator-Interaktion  |
|-------------------------|--------------------------------------|--|
| DNS_SERVERS_INT         | Liste von IP-Adressen der DNS-Server | Liste von DNS-Servern für das Transportnetz.<br>Die IP-Adressen KÖNNEN auf einen öffentlich zugänglichen Adressbereich eingeschränkt sein. |
| DNS_DOMAIN_VPN_ZUGD_INT | DNS Domainname                       | DNS-Domainname für die Service Discovery der VPN-Konzentratoren des VPN-Zugangsdienstes  |

|                   |  |  |
|-------------------|--|--|
| DNS_SERVERS_LEKTR | Liste von IP-Adressen der DNS-Server     | Liste von DNS-Servern, die zur Namensauflösung von Namensräumen in der Einsatzumgebung verwendet werden.<br>Der Administrator MUSS die Liste von DNS-Servern, die die DNS_DOMAIN_LEKTR auflösen, bearbeiten können.<br>Die IP-Adressen der DNS-Server KÖNNEN auf den Adressbereich der ANLW_LAN_IP_ADDRESS eingeschränkt sein. |
| DNS_DOMAIN_LEKTR  | DNS Domainname                           | DNS Domainname, der von einem DNS-Server der Einsatzumgebung aufgelöst wird. Der Name DARF NICHT mit einem „.“ beginnen und nicht mit einem „.“ enden.   |
|                   |  |  |
| DNS_TA_CONFIG     | Ist abhängig von der gewählten Umsetzung | Wenn der Konnektor als Validating Resolver für den Namensraum Internet implementiert ist gilt:<br>Der Administrator MUSS die aktuellen DNSSEC Trustanchor für den Namensraum Internet auf geeignetem Weg in den Konnektor übernehmen können.   |

6505 **Tabelle 353: TAB\_KON\_731 Einsehbare Konfigurationsparameter Namensdienst**

| ReferenzID                | Belegung  | Bedeutung  |
|---------------------------|---|--|
| DNS_SERVERS_TI            | Liste von IP-Adressen der DNS-Server  | Liste von DNS-Servern, die zur Namensauflösung des Namensraums der TI verwendet werden                       |
| DNS_SERVERS_SIS           | Liste von IP-Adressen der DNS-Server  | Liste von DNS-Servern, die zur Namensauflösung des Namensraums Internet bei Nutzung des SIS verwendet werden |
| DNS_SERVERS_BESTANDSNETZE | Liste von IP-Adressen der DNS-Servern je Domäne je freigegebenem angeschlossenen Netz des Gesundheitswesens mit aAdG-NetG | Liste von DNS-Servern je Domain eines dieser freigegebenen Netze.  |
| DNS_TOP_LEVEL_DOMAIN_TI   | DNS Domainname  | Top Level Domain des Namensraumes TI   |

6506  
6507  
6508

[&lt;=]

## 4.2.7 Optionale Verwendung von IPv6

Der Konnektor kann zusätzlich eine IPv6-Adresse an den Netzwerkschnittstellen zum Transportnetz implementieren. Entscheidet sich der Hersteller für den parallelen Einsatz von IPv4 und IPv6 (Dual-Stack-Mode), sind die nachfolgenden Anforderungen dieses Kapitels umzusetzen. Einhergehend mit der Entscheidung, IPv6 an diesem Interface zu konfigurieren, ist der spätere VPN-Tunnelaufbau zur TI und SIS über das IPv6 Interface möglich. Die durch den jeweiligen IPv6-Tunnel zu transportierenden IP-Pakete sind IPv4 adressierte Pakete.

### **A\_17199 - IPv6 - Adressierung der Schnittstelle zum Internet (Option IPv6)**

Der Konnektor MUSS bei Verwendung von IPv6 für den VPN-Tunnelaufbau zur TI und SIS auf geeignete Weise (z.B. DHCP vom IAG) mit einer IPv6-Adresse auf dem physikalischen Interface in Richtung Internet konfiguriert werden (Dual-Stack-Mode).[<=]

### **A\_17200 - IPv6 - Fragmentierung der IKEv2-Nachrichten (Option IPv6)**

Der Konnektor MUSS bei Verwendung von IPv6 für den VPN-Tunnelaufbau zur TI und SIS die Fragmentierung von IKEv2 Nachrichten gemäß [RFC7383] unterstützen.[<=]

### **A\_17201 - IPv6 - Verhalten als IPv6 Router (Option IPv6)**

Der Konnektor MUSS bei Verwendung von IPv6 für den VPN-Tunnelaufbau zur TI und SIS die notwendige Route für das Erreichen des Internets bereitstellen.[<=]

## 4.3 Konnektormanagement

Das Konnektormanagement dient ausschließlich Betriebsaspekten des Konnektors. Daher wird in diesem Kapitel weitestgehend auf die übliche Strukturierung nach TUCs (intern/für Fachmodule), Außenoperationen und Betriebsaspekten verzichtet. Lediglich der KSR-Client verwendet diese Kapitelstruktur.

Innerhalb des Konnektormanagements werden vorrangig folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): „MGM“
- Konfigurationsparameter: „MGM\_“

Eine Ausnahme hiervon bildet der Anteil der Software-Aktualisierung (KSR-Client). Dieser verwendet folgende Präfixe für Bezeichner:

- Events (Topic Ebene 1): „KSR“
- Konfigurationsparameter: „MGM\_“

### **TIP1-A\_4806 - Verpflichtende Managementschnittstelle**

Der Konnektor MUSS LAN-seitig über eine Managementschnittstelle für Konfiguration und Diagnose verfügen.

Die Ausführung der Schnittstelle ist herstellerspezifisch, MUSS aber entweder als Konfigurations-Frontend im Sinne einer eigenständigen Client-Applikation oder als Web-Oberfläche ausgeprägt sein.

Wenn die Schnittstelle als Web-Oberfläche ausgeprägt ist, MUSS im Handbuch beschrieben sein, wo angegeben ist, welche Browser-Versionen für welche Betriebssysteme unterstützt werden (bspw. im Handbuch selbst oder über einen Link auf eine Web-Seite des Herstellers), und wo diese als installierbares Softwarepaket oder direkt ausführbare Datei bezogen werden können.

Die Verbindung zur Managementschnittstelle MUSS zur Sicherung der Vertraulichkeit,

6553 Integrität und Authentizität durch Nutzung eines kryptographischen Verfahrens gemäß  
 6554 [gemSpec\_Krypt] abgesichert werden, falls die Sicherheit der übertragenen Daten nicht  
 6555 auf andere Weise erreicht wird. Die Absicherung der Daten kann z. B. durch Nutzung von  
 6556 TLS unter Berücksichtigung der in [gemSpec\_Krypt] angegebenen Algorithmen und  
 6557 Schlüssellängen geschehen.

6558 Die Managementschnittstelle MUSS in thematisch gegliederte Konfigurationsbereiche  
 6559 unterteilt sein. Die konkrete Gliederung selbst ist herstellerspezifisch.

6560 Die Managementschnittstelle KANN einen Managementbereich aufweisen, der nur für  
 6561 autorisierte Techniker des Herstellers zugänglich ist. Ein Zugriff auf diesen Bereich MUSS  
 6562 durch eine eigene Authentisierungsfunktion geschützt werden (z. B. durch  
 6563 Passwortschutz).

6564 [ $\leq$ ]

6565 Die über die Managementschnittstelle zu erreichenden und zu verändernden Inhalte  
 6566 werden erhoben in:

- 6567 • diesem Kapitel
- 6568 • in allen Betriebsaspektkapiteln der Funktionsmerkmale, sowie der
- 6569 Übergreifenden Festlegungen
- 6570 • den Fachmodulspezifikationen der Fachanwendungen (siehe Kapitel 4.3.4).
- 6571 • Den übergreifenden Spezifikationen [gemSpec\_Net] und [gemSpec\_PKI]

6572 Eine Ergänzung um weitere, herstellerspezifische Konfigurationseinhalte ist möglich.

#### 6573 **TIP1-A\_5661 - Automatisierung Managementschnittstelle**

6574 Der Konnektor MUSS für die Automatisierung von Konnektor-Tests alle Funktionen, die  
 6575 über die Managementschnittstelle bereitgestellt werden, über eine LAN-seitige  
 6576 Schnittstelle ohne graphische Benutzerführung bereitstellen.

6577 Der Konnektorhersteller MUSS eine Dokumentation der Schnittstelle bereitstellen, welche  
 6578 die Nutzung so beschreibt, dass die Schnittstelle von der gematik in vollem Umfang  
 6579 genutzt werden kann. Die Dokumentation MUSS der gematik im Regelfall zwei Wochen  
 6580 vor Einreichung des Zulassungsobjekts bereitgestellt werden. Von diesem Regelfall KANN  
 6581 in Abstimmung mit der gematik abgewichen werden.

6582 Die Schnittstelle SOLL mittels JSON [RFC7159] bereitgestellt werden. Wenn die  
 6583 Bereitstellung nicht mittels JSON erfolgt, MUSS sie über eine vergleichbare Technologie  
 6584 erfolgen.

6585 Der Zugriff auf die Schnittstelle MUSS in RU/TU erlaubt sein. Falls der Zugriff in der PU  
 6586 erlaubt ist, MUSS er dort ebenso wie die Managementschnittstelle abgesichert sein:

- 6587 • Die Verbindung zu dieser Schnittstelle MUSS zur Sicherung der Vertraulichkeit,  
 6588 Integrität und Authentizität durch Nutzung eines kryptographischen  
 6589 Verfahrens gemäß [gemSpec\_Krypt] abgesichert werden, falls die Sicherheit  
 6590 der übertragenen Daten nicht auf andere Weise erreicht wird. Die Absicherung  
 6591 der Daten kann z. B. durch Nutzung von TLS unter Berücksichtigung der in  
 6592 [gemSpec\_Krypt] angegebenen Algorithmen und Schlüssellängen geschehen.
- 6593 • Der Konnektor MUSS die Schnittstelle mittels Benutzername und Passwort  
 6594 oder einem mindestens gleich starken Mechanismus vor unberechtigtem  
 6595 Zugang schützen.

6596 Ansonsten DARF der Zugriff in der PU NICHT möglich sein.

6597 [ $\leq$ ]

#### 6598 **TIP1-A\_4807 - Mandantenübergreifende Managementschnittstelle**

6599 Das Management des Konnektors MUSS über die Managementschnittstelle  
 6600 mandantenübergreifend erfolgen. Dies bedeutet insbesondere, dass ein Administrator  
 6601 (gemäß seiner Zugriffsberechtigungen) in einer Management-Session alle Einstellungen

6602 einsehen und verändern können MUSS, egal welchem Mandanten diese Werte zugeordnet  
6603 sind.

6604 [ $\leq$ ]

6605 **TIP1-A\_5658 - Konnektor, rollenspezifische Endpunkte der**  
6606 **Managementschnittstelle**

6607 Der Konnektor MUSS die Managementschnittstelle mit zwei getrennten Endpunkten  
6608 implementieren. Der Konnektor MUSS sicherstellen, dass auf den einen Endpunkt nur  
6609 Nutzer mit der Rolle Lokaler-Administrator oder Super-Administrator zugreifen können,  
6610 und auf den anderen Endpunkt nur Nutzer mit der Rolle Remote-Administrator.

6611 [ $\leq$ ]

6612 **TIP1-A\_5005 - Protokollierung in der Managementschnittstelle**

6613 Jede Änderung, die ein Administrator vornimmt, MUSS protokolliert werden durch

6614 TUC\_KON\_271 „Schreibe Protokolleintrag“ {

6615     topic=„MGM/ADMINCHANGES“;

6616     eventType=Op;

6617     severity=Info;

6618     parameters =(„User=\$AdminUsername,

6619                     RefID=\$ReferenzID,

6620                     NewVal=\$NeuEingestellterWert“)}  
6621

6622 Der hier geforderte Logging-Level gilt, wenn nicht an anderer Stelle eine abweichende  
6623 Regelung spezifiziert ist.

6624 Wenn die Änderung über ein Remote-Management-System durchgeführt wird, ohne dass  
6625 ein Remote-Administrator im Konnektor konfiguriert ist, so MUSS als User eine Referenz  
6626 auf das Remote-Management-System verwendet werden.

6627 Passwörter DÜRFEN NICHT in den Protokolleinträgen geschrieben werden.

6628 [ $\leq$ ]

6629 **4.3.1 Zugang und Benutzerverwaltung des**  
6630 **Konnektormanagements**

6631 Der Konnektor verfügt über keine Verwaltung der fachlichen Nutzer, wohl aber über eine  
6632 Verwaltung der Nutzer, die in der Rolle eines Administrators den Konnektor konfigurieren  
6633 und die Protokolle einsehen dürfen. Dabei werden drei Administrator-Rollen  
6634 unterschieden:

6635     1. Lokaler-Administrator: zur Konfiguration des Konnektors über die lokale  
6636         Managementschnittstelle

6637     2. Remote-Administrator: zur Konfiguration des Konnektors über die remote  
6638         Managementschnittstelle.

6639     3. Super-Administrator: zur Verwaltung von Benutzerkonten und zur Konfiguration  
6640         des Konnektors über die lokale Managementschnittstelle

6641 **TIP1-A\_4808 - Zugangsschutz der Managementschnittstelle**

6642 Der Konnektor MUSS sicherstellen, dass die Managementschnittstelle vor unberechtigtem  
6643 Zugang geschützt ist. Die Managementschnittstelle MUSS durch eine Kombination aus  
6644 Benutzername und Passwort oder einen mindestens gleich starken Mechanismus vor  
6645 unberechtigtem Zugang geschützt sein.

6646 Für die Erstellung und Verarbeitung von Passwörtern der Managementschnittstelle  
6647 MÜSSEN die Empfehlungen der Grundsatz-Kataloge des BSI beachtet werden (siehe  
6648 Maßnahme „M 2.11 Regelung des Passwortgebrauchs“ in [BSI\_GK]).  
6649



6650 Für die Passworterstellung MUSS der Konnektor mindestens folgende Aspekte  
6651 berücksichtigen:

- 6652 • dem Benutzer muss es möglich sein, die Zeichen eines Passworts aus den  
6653 Zeichenklassen Großbuchstaben, Kleinbuchstaben, Sonderzeichen und Ziffern  
6654 zu wählen. Ein Passwort muss Zeichen aus mindestens drei dieser  
6655 Zeichenklassen enthalten.
- 6656 • ein Passwort muss mindestens 8 Zeichen lang sein
- 6657 • ein Passwort darf nicht die zugehörige Benutzerkennung enthalten (weder  
6658 vorwärts noch rückwärts, bei Vergleich unter Ignorierung der Groß- und  
6659 Kleinschreibung)
- 6660 • die Wiederholung alter Passwörter beim Passwortwechsel durch den Benutzer  
6661 selbst muss vom Konnektor verhindert werden (Passworthistorie). Dazu muss  
6662 der Konnektor mindestens die letzten drei Passwörter eines Benutzers bei der  
6663 Passwortneuvergabe erkennen und als neues Passwort ablehnen.

6664 Für die Passwortverarbeitung MUSS der Konnektor mindestens folgende Aspekte  
6665 berücksichtigen:

- 6666 • für die Erstanmeldung neuer Benutzer müssen Einmalpasswörter vergeben  
6667 werden, also Passwörter, die nach einmaligem Gebrauch gewechselt werden  
6668 müssen. Gleiches gilt, wenn ein Passwort eines Benutzers vom Super-Admin  
6669 zurückgesetzt wird.
- 6670 • jeder Benutzer muss sein eigenes Passwort jederzeit ändern können
- 6671 • bei der Eingabe darf das Passwort nicht im Klartext auf dem Bildschirm  
6672 angezeigt werden
- 6673 • die Passwörter müssen im Konnektor zugriffssicher gespeichert werden
- 6674 • der Konnektor muss nach einem durch den Super-Admin konfigurierbaren  
6675 Zeitraum (Voreinstellung: 120 Tage) einen Passwortwechsel beim nächsten  
6676 Login initiieren
- 6677 • erfolglose Anmeldeversuche müssen mit einer kurzen Fehlermeldung ohne  
6678 Angabe von näheren Einzelheiten abgelehnt werden. Insbesondere darf bei  
6679 erfolglosen Anmeldeversuchen nicht erkennbar sein, ob der eingegebene  
6680 Benutzername oder das eingegebene Passwort (oder beides) falsch ist.
- 6681 • Nach einer Fehleingabe des Passworts muss eine Verzögerung bis zur  
6682 nächsten Eingabemöglichkeit des Passworts für dieselbe Benutzerkennung  
6683 erfolgen. Die Verzögerung soll 3 Sekunden betragen.

6684 [ $\leq$ ]

6685 Näheres hierzu regeln die Schutzprofile des Konnektors.

## 6686 **TIP1-A\_4810 - Benutzerverwaltung der Managementschnittstelle**

6687 Der Konnektor MUSS eine Benutzerverwaltung für die Managementschnittstelle  
6688 enthalten, in der anmeldeberechtigte Administratoren-Benutzer definiert werden können.  
6689 Die Benutzerverwaltung MUSS die Administrator-Rollen Lokaler-Administrator, Remote-  
6690 Administrator und Super-Administrator unterstützen.

6691 Den Administrator-Rollen MÜSSEN folgende Rechte zugewiesen sein:

- 6692 • Lokaler-Administrator:
- 6693 • ausschließlicher Zugriff über lokalen Endpunkt der Managementschnittstelle



- 6694 • Verwaltung aller Konfigurationsdaten und Durchführung aller
- 6695 Administratoraktionen mit Ausnahme von:
- 6696 • Benutzerverwaltung gemäß Tabelle TAB\_KON\_655
- 6697 • Remote-Administrator:
- 6698 • ausschließlicher Zugriff über remote-Endpunkt der Managementschnittstelle
- 6699 • Verwaltung aller Konfigurationsdaten und Durchführung aller
- 6700 Administratoraktionen mit Ausnahme von:
- 6701 • Benutzerverwaltung gemäß Tabelle TAB\_KON\_655
- 6702 • Konfigurationseinstellungen und Administratoraktionen gemäß Tabelle
- 6703 TAB\_KON\_851
- 6704 • Super-Administrator:
- 6705 • ausschließlicher Zugriff über lokalen Endpunkt der Managementschnittstelle
- 6706 • Benutzerverwaltung gemäß Tabelle TAB\_KON\_655
- 6707 • Verwaltung aller Konfigurationsdaten und Durchführung aller
- 6708 Administratoraktionen

**Tabelle 354: TAB\_KON\_655 Konfigurationen der Benutzerverwaltung (Super-Administrator)**

| ReferenzID       | Belegung                                       | Bedeutung und Administrator-Interaktion  |
|------------------|--|--|
| MGM_USER_LIST    | Liste von Benutzernamen und deren Kontaktdaten | Liste von Benutzern und deren Kontaktdaten.<br>Benutzerkonten MÜSSEN angelegt, geändert und gelöscht werden können.<br>Das Passwort eines Benutzerkontos MUSS neu gesetzt werden können.   |
| MGM_ADMIN_RIGHTS | Liste von Zugriffsrechten eines Benutzers      | i. Eindeutige Zuordnung eines Benutzerkontos zu einer Rolle.<br>Die Benutzerverwaltung MUSS sicherstellen, dass zu jeder Zeit mindestens ein Benutzerkonto mit der Rolle Super-Administrator vorhanden ist.<br>Gewähren/Entziehen von Rechten für Benutzerkonten:<br>ii. Zugriffsrechte bezüglich der Konfigurationsbereiche.<br>iii. Recht zum Aufbau einer Remote-Management-Session und/oder zur Konfiguration des Remote-Management gemäß TAB_KON_663 (USER_INIT_REMOTESESSION).<br>iv. Recht für einen Werksreset (USER_RESET_PERMISSION) |

|  |  |  |
|--|--|--|
|  |  |  |
|--|--|--|

Die Benutzerverwaltung MUSS es jedem Benutzer ermöglichen Konfigurationsänderungen gemäß Tabelle TAB\_KON\_656 vorzunehmen:

**Tabelle 355: TAB\_KON\_656 Konfigurationen der Benutzerverwaltung**

| ReferenzID    | Belegung     | Bedeutung und Administrator-Interaktion  |
|---------------|--------------|--|
| MGM_USER_INFO | Kontaktdaten | Der angemeldete Benutzer MUSS seine Kontaktdaten ändern können. Der Benutzername DARF NICHT änderbar sein. |

[<=]

### 4.3.2 Konnektornamen und Versionsinformationen

#### **TIP1-A\_4811 - Festlegung des Konnektornamens**

Der Konnektor MUSS die Konfiguration und Nutzung eines sprechenden Konnektornamens unterstützen, der identisch zum Hostnamen des Konnektors ist. Der Konnektornamen MUSS dauerhaft an der Managementschnittstelle angezeigt werden. Die Managementschnittstelle MUSS es einem Administrator ermöglichen Konfigurationsänderungen gemäß Tabelle TAB\_KON\_657 vorzunehmen:

**Tabelle 356: TAB\_KON\_657 Konfigurationsparameter des Konnektornamens**

| ReferenzID        | Belegung   | Bedeutung und Administrator-Interaktion   |
|-------------------|------------|---|
| MGM_KONN_HOSTNAME | 12 Zeichen | Der Konnektornamen MUSS folgende Anforderungen erfüllen (in Anlehnung an die Definition eines „Labels“ in [RFC1034]):<br><ul style="list-style-type: none"> <li>• Verwendung der Buchstaben „A bis Z“ und „a bis z“,</li> <li>• Verwendung der Ziffern „0 bis 9“,</li> <li>• als Sonderzeichen „-“ (Minus), sowie</li> <li>• eine maximale Länge von 12 Zeichen,</li> </ul> Die Verwendung weiterer Sonderzeichen sowie des Leerzeichens DARF NICHT möglich sein. |

Optional KANN ein Hersteller zusätzlich zum Konnektor- bzw. Hostnamen die Konfiguration eines DNS-Suffixes vorsehen. Der DNS-Suffix DARF NICHT Bestandteil des Konnektornamens sein.

[<=]

#### **TIP1-A\_4812 - Anzeige der Versionsinformationen (Selbstauskunft)**

Der Administrator MUSS die Versionsinformationen des Konnektors einsehen können. Dabei MÜSSEN alle über ProductInformation.xsd definierten Elemente verständlich angezeigt werden.

Ferner MUSS der Administrator dabei die aktuelle Firmware-Gruppenversion des Konnektors einsehen können.

[<=]

6737 **A\_18929 - Sichtbarkeit der ECC-Vorbereitung an der Managementschnittstelle**  
 6738 Der Hersteller MUSS die ECC-Vorbereitung der gSMC-K durch die Bezeichnung „ECC-  
 6739 Vorbereitet“ zusammen mit den Versionsinformationen des Konnektors an der  
 6740 Managementschnittstelle sichtbar machen.  
 6741 [ $\leq$ ]

6742 **TIP1-A\_7255 - Anzeige von Fachmodulversionen**  
 6743 Der Administrator MUSS die Versionen der in der Firmware des Konnektors enthaltenen  
 6744 Fachmodule einsehen können.  
 6745 [ $\leq$ ]

6746 Fachmodulversionsinformationen sind nicht Bestandteil der Selbstauskunft gemäß  
 6747 ProductInformation.xsd.

### 6748 **4.3.3 Konfigurationsdaten: Persistieren sowie Export-Import**

6749 **TIP1-A\_4813 - Persistieren der Konfigurationsdaten**  
 6750 Der Konnektor MUSS die Konfigurationsdaten nach Änderung persistieren. Dabei  
 6751 MÜSSEN Integrität, Authentizität und Vertraulichkeit der Konfigurationsdaten gewährt  
 6752 sein. Der Mechanismus hierfür ist herstellerspezifisch.  
 6753 Der Konnektor MUSS sicherstellen, dass immer ein integerer Konfigurationssatz  
 6754 persistiert ist.  
 6755 Bei der Konnektorinitialisierung MÜSSEN die persistierten Konfigurationsdaten eingelesen  
 6756 werden.  
 6757 Die Verpflichtung zur Persistierung gilt für alle innerhalb der Konnektor- und Fachmodul-  
 6758 Spezifikationen erhobenen Konfigurationsdaten.  
 6759 [ $\leq$ ]

6760 **TIP1-A\_4814 - Export- Import von Konfigurationsdaten**  
 6761 Der Administrator MUSS die gesamten Konfigurationsdaten des Konnektors ex- und  
 6762 importieren können. Dazu gehören die Konfigurationsparameter des Konnektors, die  
 6763 persistenten Daten wie im Informationsmodell des Konnektors (Tabelle TAB\_KON\_507  
 6764 Informationsmodell Entitäten) definiert und die Pairing Informationen der  
 6765 Kartenterminals.  
 6766 Die Konfigurationsdaten des Anwendungs- und Netzkonnektors KÖNNEN gemeinsam oder  
 6767 getrennt exportiert bzw. importiert werden. Das Format der Konfigurationsdaten ist  
 6768 herstellerspezifisch.  
 6769 Auf hardwareseitig baugleichen Geräten:

- 6770 • MUSS der Import von Konfigurationsdateien möglich sein, die unter der gleichen  
 6771 oder einer früheren Firmwareversion exportiert wurden
- 6772 • SOLL der Import von Konfigurationsdateien möglich sein, die unter einer neueren  
 6773 Firmwareversion exportiert wurden

6774 Der Import von Konfigurationsdateien, die von einem Konnektor mit anderer  
 6775 Hardwareversion exportiert wurden, KANN ermöglicht werden.

6776  
 6777 (für Fachmodule siehe Kapitel 4.3.4)  
 6778 Der Konnektor MUSS sicherstellen, dass der Exportvorgang nur von einem am Konnektor  
 6779 angemeldeten User mit mindestens der Rolle Administrator ausgelöst werden kann.  
 6780 Der Konnektor MUSS sicherstellen, dass der Importvorgang nur von einem am Konnektor  
 6781 angemeldeten User mit der Rolle Super-Administrator ausgelöst werden kann.  
 6782 Sowohl Ex- als auch Import MÜSSEN protokolliert werden durch TUC\_KON\_271 „Schreibe  
 6783 Protokolleintrag“ {  
 6784 topic = „MGM/CONFIG\_EXIMPORT“;  
 6785 eventType = Op;

6786 severity = Info;  
6787 parameters = („User=\$AdminUsername,  
6788 Mode=[Export/Import]“)}.

6789  
6790 [**<=**]

#### 6791 **A\_19738 - Optionaler Import von Konfigurationsdaten durch lokalen** 6792 **Administrator**

6793 Der Konnektor KANN einem am Konnektor angemeldeten User mit der Rolle Lokaler-  
6794 Administrator erlauben, den Importvorgang von Konfigurationsdateien auszulösen, wenn  
6795 in den Konfigurationsdaten keine Benutzerdaten gemäß Tabelle TAB\_KON\_655 enthalten  
6796 sind. [**<=**]

6797 Nähere Vorgaben zum Ablauf des Imports der Kartenterminalinformationen finden sich  
6798 im Kapitel 4.1.4.6.3.

#### 6799 **TIP1-A\_4815 - Export: Schutz der Integrität, Authentizität und** 6800 **Nichtabstreitbarkeit**

6801 Die **Integrität, Authentizität und Nichtabstreitbarkeit** der exportierten Daten MUSS  
6802 sichergestellt werden. Dies MUSS durch eine Signatur mit der OSIG-Identität der SM-B  
6803 oder mit einem herstellerspezifischen Schlüsselpaar realisiert werden. In die zu  
6804 signierenden Daten MUSS eine Zeitangabe zum Signaturzeitpunkt integriert werden.  
6805 Beim Import MUSS die Signatur vor der Übernahme der Daten erfolgreich verifiziert  
6806 worden sein. Im Laufe des Importvorgangs MUSS dem Administrator das zur Signatur  
6807 zugehörige Zertifikat (oder der herstellerspezifische öffentliche Schlüssel) sowie die  
6808 Zeitangabe zum Signaturzeitpunkt der exportierten Konfiguration angezeigt werden, und  
6809 der Administrator MUSS explizit bestätigen, dass er die zu dem angezeigten Zeitpunkt  
6810 gehörige Konfiguration importieren will.  
6811 Wird die SM-B zur Signatur eingesetzt, so MUSS die Prüfung des genutzten  
6812 Signaturzertifikats anhand von TUC\_KON\_037 erfolgen. Das Zertifikat der OSIG-  
6813 Identität, mit dem die Daten signiert wurden, MUSS zusammen mit den exportierten  
6814 Daten gespeichert werden, um eine Verifikation der Signatur auf neuen Konnektoren  
6815 auch ohne Zugriff auf die entsprechende SM-B zu ermöglichen.  
6816 Da Konfigurationsdaten mit einem Schutzbedarf von mindestens „Hoch“ für Authentizität  
6817 und Nichtabstreitbarkeit exportiert werden (z. B. Pairing-Geheimnisse (ShS.KT.AUT) der  
6818 Kartenterminals), MUSS durch geeignete Maßnahmen sichergestellt werden, dass der  
6819 Zugriff auf die Daten auf eine natürliche Person rückführbar ist. Dies kann  
6820 organisatorisch (durch Einträge des Administrators in ein Betriebsführungs-Handbuch  
6821 beim Nutzer) technisch (durch eine personenbezogene Administratorenverwaltung) oder  
6822 äquivalent herstellerspezifisch erreicht werden.

6823 [**<=**]

#### 6824 **TIP1-A\_4816 - Export: Schutz der Vertraulichkeit**

6825 Zum Schutz der **Vertraulichkeit** der exportierten Daten MÜSSEN die Daten vor dem  
6826 Export verschlüsselt werden. Dies kann durch asymmetrische oder symmetrische  
6827 Verschlüsselungsverfahren nach [gemSpec\_Krypt] realisiert werden.  
6828 Wird ein rein symmetrisches Verfahren eingesetzt, so MUSS als Mindestanforderung eine  
6829 Passphrase einer Mindestlänge von 16 Zeichen (Groß- und Kleinbuchstaben, Ziffern und  
6830 Sonderzeichen) zur Verschlüsselung der Daten eingesetzt werden. Diese Passphrase  
6831 MUSS dabei vom Konnektor zufällig generiert werden und aus einer Kombination von  
6832 Buchstaben und Ziffern bestehen. Diese Passphrase MUSS dem Administrator  
6833 anschließend angezeigt werden.

6834 [**<=**]

#### 4.3.4 Administration von Fachmodulen

Die Konfiguration von Fachmodulen ist innerhalb der Managementschnittstelle des Konnektors von der Konfiguration der Plattformanteile des Konnektors logisch entkoppelt. Die Festlegungen, welche Konfigurationsparameter und welche zusätzlichen administrativen Funktionen für ein Fachmodul benötigt werden, werden in den jeweiligen Fachmodulspezifikationen getroffen. Der Konnektor muss aber für jedes Fachmodul hinsichtlich der Administrierbarkeit des Fachmoduls die folgende Basisfunktionalität zur Verfügung stellen:

##### **TIP1-A\_4818 - Konfigurieren von Fachmodulen**

Neben den Konfigurationsbereichen der Plattformanteile des Konnektors, MUSS die Managementschnittstelle auch die Konfiguration der im Konnektor enthaltenen Fachmodule unterstützen. Ein Administrator MUSS die in den Fachmodulspezifikationen enthaltenen Konfigurationsparameter ändern und die dort definierten Informationen einsehen können. Der Konnektor MUSS die Konfigurationsdaten von Fachmodulen nach deren Änderung persistieren, sowie bei einem Neustart eines Fachmoduls die Fachmodul-Konfigurationsdaten vor der Initialisierung des Fachmoduls einlesen. Die persistierten Fachmodulkonfigurationsdaten MÜSSEN ebenso wie die plattformeigenen Konfigurationsdaten hinsichtlich ihrer Integrität und Authentizität sowie ihrer Vertraulichkeit geschützt werden. Der Ex- und Import von Fachmodulkonfigurationen MUSS äquivalent zum Ex- und Import der Plattformanteile für den Administrator möglich sein (siehe 4.3.3). Die Konfigurationsdaten der Fachmodule KÖNNEN dabei in der Gesamt Export-Datei des Konnektors enthalten sein oder separat exportiert und importiert werden.

[<=]

##### **TIP1-A\_5484 - Persistente Speicherung von Konfigurationsdaten der Fachmodule**

Der Konnektor MUSS den Fachmodulen die Möglichkeit bereitstellen, die in den Fachmodulspezifikationen gekennzeichneten Konfigurationsdaten persistent zu speichern, auszulesen und zu löschen. Je Fachmodul muss ein exklusiv durch das Fachmodul nutzbarer Speicherbereich verwendet werden. Namenskonvention zur Kennzeichnung der Konfigurationsdaten der Fachmodule für persistent zu speichernde Daten:  
FM\_<fmName>\_<fmDataName>

**Tabelle 357: TAB\_KON\_833 Bezeichner für persistente Konfigurationsdaten für Fachmodule**

| Bezeichner   | Bedeutung   |
|--------------|---|
| FM           | fester Namensbestandteil zur Kennzeichnung von persistenten fachmodulspezifischen Konfigurationsdaten |
| —            | Trennzeichen  |
| <fmName>     | Name des Fachmoduls (innerhalb eines Fachmoduls konstanter Bezeichner)                                |
| —            | Trennzeichen  |
| <fmDataName> | Name der persistent zu speichernden fachmodulspezifischen Konfigurationsdaten                         |

[<=]

#### 4.3.5 Neustart und Werksreset

##### **TIP1-A\_4819 - Auslösen eines Konnektorneustarts**

Der Administrator MUSS einen Neustart des Konnektors auslösen können.

[<=]

##### **TIP1-A\_4820 - Werksreset des Konnektors**

Ein Administrator mit USER\_RESET\_PERMISSION MUSS einen Werksreset des Konnektors auslösen können.

Zur Durchführung des Werksreset MUSS der Administrator nach Funktionsaufruf per Sicherheitsabfrage zur Bestätigung des Werksresets aufgefordert werden. Nach bestätigter Sicherheitsabfrage MUSS der Konnektor die gesamte Konfiguration des Konnektors und alle internen Speicher, mit Ausnahme des aktuellen Vertrauensraums sowie der Sicherheitsprotokolle und der installierten Firmware, auf den Auslieferungszustand zurücksetzen. Die in CERT\_IMPORTED\_CA\_LIST enthaltenen Zertifikate MÜSSEN aus dem aktuellen Vertrauensraum gelöscht werden.

Die Durchführung des Werksresets MUSS protokolliert werden durch TUC\_KON\_271

„Schreibe Protokolleintrag“ {

topic = „MGM/FACTORYSETTINGS“;

eventType = Op;

severity = Info;

parameters = „User=\$AdminUsername“}.

Dieser Protokolleintrag DARF NICHT durch einen erfolgreichen Werksreset verloren gehen.

Der Hersteller MUSS ferner einen alternativen, herstellerspezifischen Weg zum Auslösen des Werksresets vorsehen, welcher die Arbeitsabläufe beim Nutzer nur minimal unterbricht. Auch für diesen zusätzlichen Weg MUSS zuvor eine Authentisierung durch eine Kombination aus Benutzernamen und Passwort oder einem mindestens gleich starken Mechanismus erfolgen. Der Mechanismus MUSS auch dann funktionieren, wenn sich keiner der in der Benutzerverwaltung definierten Administratoren mehr erfolgreich anmelden kann.

[<=]

#### 4.3.6 Leistungsumfänge und Standalone-Szenarios

Obgleich der Konnektor in seinem Auslieferungszustand alle Leistungsmerkmale aufweisen muss, die gemäß Produkttypsteckbrief gefordert werden, so soll es dem Administrator doch ermöglicht werden grundsätzliche Leistungsumfänge gezielt deaktivieren zu können, um den Konnektor so besser in die organisatorische/technische Struktur der Betriebsstätte eingliedern zu können.

##### **TIP1-A\_4821-02 - Aktivieren/Deaktivieren von Leistungsumfängen**

Die Managementschnittstelle MUSS es einem Administrator ermöglichen Konfigurationsänderungen gemäß Tabelle TAB\_KON\_658 vorzunehmen:

**Tabelle 358: TAB\_KON\_658 Aktivieren/Deaktivieren von Leistungsumfängen**

| ReferenzID    | Belegung             | Bedeutung und Administrator-Interaktion   |
|---------------|----------------------|---|
| MGM_LU_ONLINE | Enabled/<br>Disabled | Der Administrator MUSS den „Leistungsumfang Online“ aktivieren und deaktivieren können. Bei Veränderung MUSS TUC_KON_256 gerufen werden { |



|            |                      |   |
|------------|----------------------|---|
|            |                      | topic = „MGM/LU_CHANGED/LU_ONLINE“;<br>eventType = Op;<br>severity = Info;<br>parameters = „Active=\$MGM_LU_ONLINE“}  |
| MGM_LU_SAK | Enabled/<br>Disabled | Der Administrator MUSS den „Leistungsumfang Signaturanwendungskomponente“ aktivieren und deaktivieren können.<br>Default-Wert: Enabled<br>Bei Veränderung MUSS TUC_KON_256 gerufen werden {<br>topic = „MGM/LU_CHANGED/LU_SAK“;<br>eventType = Op;<br>severity = Info;<br>parameters = „Active=\$MGM_LU_SAK“} |

6916 [**<=**]

6917

6918 Der Konfigurationsparameter MGM\_LU\_SAK wirkt hauptsächlich in dem

6919 Funktionsmerkmal „Signaturdienst“ (siehe Kapitel 4.1.8).

6920 Ist MGM\_LU\_ONLINE Disabled, so baut der Konnektor grundsätzlich keine Online-

6921 Verbindungen auf (weder zur TI, noch zum SIS). Der Parameter wirkt hauptsächlich in

6922 den Funktionsmerkmalen:

- 6923 • „Zertifikatsdienst“ (Kapitel 4.1.9)
- 6924 • „TLS-Dienst“ (Kapitel 4.1.11)
- 6925 • „Anbindung LAN/WAN“ (Kapitel 4.2.1)
- 6926 • „VPN-Client“ (Kapitel 4.2.4)
- 6927 • „Zeitdienst“ (Kapitel 4.2.5)
- 6928 • „Software-Aktualisierungsdienst (KSR-Client)“ (Kapitel 4.3.9)
- 6929 • „LDAP-Proxy“ (Kapitel 4.1.12)

6930 Ob es sich bei einem Konnektor um den losgelöst (stand alone) vom Netz der

6931 Einsatzumgebung betriebenen handelt, also einen Konnektor, auf welchen kein

6932 Clientsystem zugreift, muss diesem mitgeteilt werden:

#### 6933 **TIP1-A\_4822 - Konnektor Standalone einsetzen**

6934 Die Managementschnittstelle MUSS es einem Administrator ermöglichen

6935 Konfigurationsänderungen gemäß Tabelle TAB\_KON\_659 vorzunehmen:

6936

6937 **Tabelle 359: TAB\_KON\_659 Konnektor Standalone einsetzen**

| ReferenzID         | Belegung             | Bedeutung und Administrator-Interaktion  |
|--------------------|----------------------|--|
| MGM_STANDALONE_KON | Enabled/<br>Disabled | Der Administrator MUSS den Konnektor als alleinstehend konfigurieren können.<br>Default-Wert: Disabled<br>Bei Veränderung MUSS TUC_KON_256 gerufen werden {<br>topic = „MGM/STANDALONE_CHANGED“; |

|  |  |   |
|--|--|---|
|  |  | eventType = Op;<br>severity = Info;<br>parameters =<br>„Active=\$MGM_STANDALONE_KON“} |
|--|--|---|

6938  
6939  
6940

[<=]

6941 Das Setzen von MGM\_STANDALONE\_KON auf Enabled dient dem Konnektor als Anzeige,  
6942 dass dieser ohne angeschlossenes Clientsystem (Primärsystem) betrieben wird. Diese  
6943 Information kann seitens der Fachmodule verwendet werden, damit diese sich im  
6944 Standalone-Fall anders als im Normalfall verhalten.

#### 6945 4.3.7 Online-Anbindung verwalten

6946 Um Zugang zur TI erlangen zu können, muss der Betriebsstättenverantwortliche einen  
6947 Vertrag mit einem Zugangsdienstprovider (ZGDP) abgeschlossen haben. Von diesem  
6948 erhält er eine ContractID. Der Administrator muss den Konnektor (genauer das NK-  
6949 Zertifikat C.NK.VPN) mit dieser Information unter Nutzung einer SM-B über den  
6950 Registrierungsdienst des ZGDP bei diesem freischalten.

6951 Die Berechtigung zur Einwahl in die TI ist von der Gültigkeit der **beiden** bei der  
6952 Freischaltung übermittelten Zertifikate abhängig (C.NK.VPN und C.HCI.OSIG). Die  
6953 Berechtigung zur Einwahl in die TI wird verweigert, bzw. eine bestehende Verbindung zur  
6954 TI wird beendet, wenn ein Zertifikat abgelaufen oder gesperrt ist. Aus diesem Grund  
6955 muss der Administrator vor Ablauf eines der beiden Zertifikate eine wiederholte  
6956 Registrierung mit neuem Netzkonnektorzertifikat bzw. neuer SM-B beim ZGDP  
6957 durchführen. (Hinweis: neue NK-Zertifikate werden erst mit Etablierung der  
6958 Nachladefunktionalität für gSMC-K verfügbar sein.)

6959 Soll ein Konnektor außer Betrieb genommen werden oder wird der Vertrag mit einem  
6960 ZGDP gekündigt, muss der Administrator den Konnektor über den Registrierungsdienst  
6961 des ZGDP abmelden.

#### 6962 TIP1-A\_4824 - Freischaltdaten des Konnektors bearbeiten

6963 Der Administrator MUSS die in TAB\_KON\_661 aufgelisteten Parameter über die  
6964 Managementschnittstelle konfigurieren und die in TAB\_KON\_732 aufgelisteten Parameter  
6965 ausschließlich einsehen können.  
6966

6967 **Tabelle 360: TAB\_KON\_661 Konfigurationsparameter der Konnektorfreischaltung**

| ReferenzID          | Belegung | Bedeutung und Administrator-Interaktion   |
|---------------------|----------|---|
| MGM_ZGDP_CONTRACTID | String   | Der Administrator MUSS die vom Zugangsdienstprovider für die Freischaltung des Konnektors erhaltene ContractID eingeben können. |
| MGM_ZGDP_SMCB       | ICCSN    | Der Administrator MUSS die zur Freischaltung zu verwendende SM-B aus der Liste der verwalteten SM-Bs auswählen können.          |



**Tabelle 361: TAB\_KON\_732 Einsehbare Konfigurationsparameter der Konnektorfreischaltung**

| ReferenzID         | Belegung | Bedeutung und Administrator-Interaktion                  |
|--------------------|----------|--|
| MGM_ZGDP_REGSERVER | URI      | URI des Registrierungsservers des Zugangsdienstproviders |

Den Zustand der Freischaltung verwaltet der Konnektor gemäß Tabelle TAB\_KON\_662 Zustandswerte der Konnektorfreischaltung.  
Im Auslieferungszustand MUSS MGM\_TI\_ACCESS\_GRANTED=Disabled belegt sein.

**Tabelle 362: TAB\_KON\_662 Zustandswerte der Konnektorfreischaltung**

| ReferenzID            | Belegung             | Zustandswerte  |
|-----------------------|----------------------|--|
| MGM_TI_ACCESS_GRANTED | Enabled/<br>Disabled | Status der Freischaltung des Konnektors:<br>- Enabled: Konnektor wurde erfolgreich beim Zugangsdienstprovider freigeschaltet<br>- Disabled: Freischaltung noch nicht erfolgt |

[<=]

**TIP1-A\_4825 - Konnektor zur Nutzung (wiederholt) freischalten**

Der Administrator MUSS den Konnektor über folgenden Mechanismus zur Nutzung freischalten bzw. eine vorhandene Freischaltung mit einer neuen SM-B aktualisieren können (Voraussetzung ist eine korrekte Konfiguration aller für einen Online-Zugang erforderlicher Parameter):

1. Der Konnektor MUSS eine registerKonnektorRequest-Struktur gemäß ProvisioningService.xsd [gemSpec\_VPN\_ZugD] erstellen und mit den entsprechenden Parametern befüllen (aktuelles Datum/Uhrzeit, C.NK.VPN, MGM\_ZGDP\_CONTRACTID). Der Konnektor MUSS die Request-Nachricht mittels der ausgewählten SM-B (ID.HCI.OSIG von MGM\_ZGDP\_SMCB) im Element registerKonnektorRequest/Signature signieren und das SM-B-Zertifikat im Element X509Data ablegen.  
Ist der nötige Sicherheitszustand für den privaten Schlüssel der SM-B nicht gesetzt, MUSS der Konnektor zur PIN-Verifikation an dem Kartenterminal auffordern, in dem die SM-B steckt.
2. Der Konnektor ermittelt die URI des Registrierungsservers (MGM\_ZGDP\_REGSERVER) durch eine DNS-Anfrage nach dem SRV und TXT Resource Record „\_regserver.\_tcp.<DNS\_DOMAIN\_VPN\_ZUGD\_INT>“
3. Der Konnektor ruft unter Verwendung der erzeugten Request-Nachricht die in [gemSpec\_VPN\_ZugD#Tab\_ZD\_registerKonnektor] definierte Operation I\_Registration\_Service::registerKonnektor mit der Zieladresse MGM\_ZGDP\_REGSERVER auf.
4. Der Konnektor zeigt dem Administrator den Inhalt von registerKonnektorResponse/AdditionalInformation und /Status an
5. Der Response der Operation wird verarbeitet:
  - a. Setze MGM\_TI\_ACCESS\_GRANTED auf
    - Enabled, wenn /RegistrationStatus = „Registriert“
    - Disabled, wenn /RegistrationStatus = „Nicht registriert“

- 7006 b. Persistiere diese Zustandsinformation zusammen mit dem  
7007 VPN:ContractStatus
- 7008 c. Verteile das folgende interne Ereignis über TUC\_KON\_256 {  
7009 topic = "MGM/TI\_ACCESS\_GRANTED";  
7010 eventType = Op;  
7011 severity = Info;  
7012 parameters = „Active=\$MGM\_TI\_ACCESS\_GRANTED“;  
7013 doDisp = false }
- 7014 Tritt während der Verarbeitungskette ein Fehler auf, so bricht die weitere Verarbeitung  
7015 ab und der Administrator MUSS darüber geeignet informiert werden (u.a. Klartextanzeige  
7016 des vom Registrierungsdienst gemeldeten Fehlers).  
7017 Wenn eine Reregistrierung mit einer neuen SMC-B fehlschlägt (Request wird mit einem  
7018 SOAP-Error beantwortet ) dann ist der Konnektor nicht registriert  
7019 (MGM\_TI\_ACCESS\_GRANTED = Disabled).  
7020 [ $\leq$ ]
- 7021 **TIP1-A\_4826 - Status Konnektorfreischaltung einsehen**  
7022 Der Administrator MUSS über die Managementschnittstelle den aktuellen Freischaltstatus  
7023 einsehen können (MGM\_TI\_ACCESS\_GRANTED). Ist der Konnektor aktuell freigeschaltet,  
7024 so MUSS ihm dies zusammen mit dem VPN:ContractStatus angezeigt werden.  
7025 [ $\leq$ ]
- 7026 Möchte ein Konnektoreigentümer das Gerät weiterveräußern oder vollständig außer  
7027 Betrieb nehmen, so sollte er eine vorhandene Freischaltung zuvor rückgängig machen.
- 7028 **TIP1-A\_4827 - Konnektorfreischaltung zurücknehmen**  
7029 Ist MGM\_TI\_ACCESS\_GRANTED=Enabled, dann MUSS der Administrator über die  
7030 Managementschnittstelle des Konnektors die Freischaltung über den folgenden  
7031 Mechanismus zurücknehmen können:
- 7032 1. Der Administrator MUSS eine Sicherheitsabfrage zur Zurücknahme der  
7033 Freischaltung bestätigen
  - 7034 2. Der Konnektor MUSS eine deRegisterKonnektorRequest-Struktur gemäß  
7035 [gemSpec\_VPN\_ZugD] erstellen und mit den entsprechenden Parametern befüllen  
7036 (aktuelles Datum/Uhrzeit, C.NK.VPN, MGM\_ZGDP\_CONTRACTID)
  - 7037 3. Der Konnektor MUSS die Request-Nachricht mittels einer verfügbaren SM-B  
7038 (ID.HCI.OSIG) im Element deRegisterKonnektorRequest/Signature signieren.  
7039 (MGM\_ZGDP\_SMCB ist zu bevorzugen, es kann aber auch jede andere SM-B  
7040 verwendet werden)  
7041 Ist der nötige Sicherheitszustand für den privaten Schlüssel der SM-B nicht  
7042 gesetzt, MUSS der Konnektor zur PIN-Verifikation an dem Kartenterminal  
7043 auffordern, in dem die SM-B steckt.
  - 7044 4. Der Konnektor ermittelt die URI des Registrierungsservers  
7045 (MGM\_ZGDP\_REGSERVER) durch eine DNS-Anfrage nach dem SRV und TXT  
7046 Resource Record „\_regserver.\_tcp.<DNS\_DOMAIN\_VPN\_ZUGD\_INT>“
  - 7047 5. Der Konnektor ruft unter Verwendung der erzeugten Request-Nachricht die in  
7048 [gemSpec\_VPN\_ZugD#Tab\_ZD\_deregisterKonnektor] definierte Operation  
7049 I\_Registration\_Service::deRegisterKonnektor mit der Zieladresse  
7050 MGM\_ZGDP\_REGSERVER auf.
  - 7051 6. Der Konnektor zeigt dem Administrator den Inhalt von  
7052 deregisterKonnektorResponse/AdditionalInformation /ContractStatus und  
7053 /RegistrationStatus an

- 7054 7. Der Response der Operation wird verarbeitet:
- 7055 a. Setze MGM\_TI\_ACCESS\_GRANTED auf
- 7056 - Enabled, wenn /RegistrationStatus = „Registriert“
- 7057 - Disabled, wenn /RegistrationStatus = „Nicht registriert“
- 7058 b. Persistiere diese Zustandsinformation zusammen mit dem Zeitpunkt
- 7059 c. Verteile das folgende interne Ereignis über TUC\_KON\_256: {
- 7060 topic = "MGM/TI\_ACCESS\_GRANTED";
- 7061 eventType = Op;
- 7062 severity = Info;
- 7063 parameters = „Active=\$MGM\_TI\_ACCESS\_GRANTED“;
- 7064 doDisp=false }
- 7065 Tritt während der Verarbeitungskette ein Fehler auf, so bricht die weitere Verarbeitung
- 7066 ab und der Administrator MUSS darüber geeignet informiert werden (u.a. Klartextanzeige
- 7067 des vom Registrierungsdienst gemeldeten Fehlers).
- 7068 Wenn eine Deregistrierung mit einer neuen SMC-B fehlschlägt (Request wird mit einem
- 7069 SOAP-Error beantwortet) dann ist der Konnektor weiterhin registriert
- 7070 (MGM\_TI\_ACCESS\_GRANTED = Enabled).
- 7071 [ $\leq$ ]

7072 **TIP1-A\_5655 - Deregistrierung bei Außerbetriebnahme**

7073 Der Hersteller des Konnektors MUSS im Handbuch den Administrator darüber

7074 informieren, dass der Konnektor bei dauerhafter Außerbetriebnahme (z. B. Verkauf,

7075 Schenkung, Entsorgung) beim Zugangsdienstprovider deregistriert werden muss.

7076 [ $\leq$ ]

7077 **4.3.8 Remote Management (Optional)**

7078 Im Betreibermodell der TI wird unter Remote Management ein delegierter Betrieb

7079 dezentraler Produkte durch einen durch den Anwender beauftragten Servicepartner

7080 verstanden. Der Servicepartner stellt als Vertragsbestandteil bevollmächtigte Personen

7081 zur Verfügung, die sich ständig um die Betriebs- und Datensicherheit der dezentralen

7082 Produkte im Rahmen eines Remote Managements kümmern.

7083 Voraussetzung für die Etablierung dieses Bestandteils des Betreibermodells der TI ist,

7084 dass ein dezentrales Produkt ein Remote Management technisch unterstützt. Die

7085 nachfolgend aufgeführten Anforderungen bilden die Grundlage für die Nutzung von

7086 Remote Management am Konnektor.

7087 Zum Remote Management gehören die Verwaltung von Konfigurationsdaten und die

7088 Durchführung weiterer Administratoraktionen wie z. B. die Aktualisierung der Software

7089 des Konnektors. Im Rahmen des Remote Managements kann der Konnektor Remote

7090 Monitoring unterstützen. Dazu übermittelt der Konnektor Betriebszustandsdaten an das

7091 Remote- Management-System.

7092 **TIP1-A\_7276-01 - Remote Management Konnektor**

7093 Der Konnektor KANN Remote Management technisch unterstützen.

7094 Falls der Konnektor das Remote Management technisch unterstützt, MUSS der Konnektor

7095 alle Anforderungen, die das Remote Management (z.B. auch Remote-Administrator)

7096 betreffen, umsetzen.

7097 Andernfalls sind die Anforderungen, die das Remote Management (z.B. auch Remote-

7098 Administrator) betreffen, für den Konnektor nicht relevant.[ $\leq$ ]

7099

- 7100 **TIP1-A\_5647 - Remote Management Konnektor: Personenbezogene Daten**  
 7101 Der Konnektor DARF über die Remote-Managementschnittstelle KEINE  
 7102 personenbezogenen Daten übertragen oder darstellen.  
 7103 [ $\leq$ ]
- 7104 **TIP1-A\_5648 - Remote Management Konnektor: Offene Schnittstelle**  
 7105 Der Hersteller des Konnektors MUSS die zur Nutzung der Remote-  
 7106 Managementschnittstelle notwendigen Informationen offenlegen. Der Hersteller des  
 7107 Konnektors MUSS die Remote-Managementschnittstelle so spezifizieren und  
 7108 implementieren, dass diese auch für Dritte (z.B. einen durch den Anwender beauftragten  
 7109 Servicepartner) nutzbar ist.  
 7110 [ $\leq$ ]
- 7111 **TIP1-A\_5649 - Remote Management Konnektor: Standardbasierte Protokolle**  
 7112 Der Hersteller des Konnektors SOLL für die Implementierung der Remote-  
 7113 Managementschnittstelle standardbasierte Verfahren und Protokolle einsetzen.  
 7114 [ $\leq$ ]
- 7115 **TIP1-A\_5650 - Remote Management Konnektor: Aufbau der Verbindung**  
 7116 Der Konnektor MUSS sicherstellen, dass die Initiierung einer Remote-  
 7117 Managementverbindung im Sinne des Verbindungsaufbaus immer vom Konnektor  
 7118 ausgeht.  
 7119 [ $\leq$ ]
- 7120
- 7121 **TIP1-A\_5651 - Remote Management Konnektor: Absicherung der Verbindung**  
 7122 Der Konnektor MUSS die Remote-Management-Verbindung durch Nutzung eines  
 7123 kryptographischen Verfahrens gemäß [gemSpec\_Krypt] hinsichtlich Vertraulichkeit,  
 7124 Integrität und Authentizität absichern.  
 7125 [ $\leq$ ]
- 7126 Das Remote-Management-System authentisiert sich auf Transportebene zertifikatsbasiert  
 7127 gegenüber dem Konnektor.
- 7128 **TIP1-A\_7277 - Authentifizierung des Remote-Management-Systems**  
 7129 Der Konnektor MUSS eine zertifikatsbasierte Authentifizierung des Remote-Management-  
 7130 Systems auf Transportebene durchführen.[ $\leq$ ]
- 7131 **TIP1-A\_7278 - Authentisierung des Konnektors gegenüber Remote-  
 Management-System**  
 7132 Der Konnektor MUSS sich gegenüber dem Remote-Management-System zertifikatsbasiert  
 7133 oder mittels Username/Password authentisieren.[ $\leq$ ]
- 7134
- 7135 **TIP1-A\_7281 - Authentifizierung des Konnektors durch das Remote-  
 Management-System**  
 7136 Das Remote-Management-System MUSS eine Authentifizierung des Konnektors  
 7137 durchführen.[ $\leq$ ]
- 7138
- 7139 Die Authentifizierung des Remote-Management-Systems durch den Konnektor auf  
 7140 Transportebene ist verpflichtend gefordert.
- 7141 Darüber hinaus können optional Remote-Administratoren in der Benutzerverwaltung des  
 7142 Konnektors konfiguriert werden. Wenn Remote-Administratoren in der  
 7143 Benutzerverwaltung konfiguriert sind, muss der Konnektor diese verpflichtend auf  
 7144 Anwendungsebene authentifizieren.
- 7145 Wenn kein Remote-Administrator konfiguriert ist, vertraut der Konnektor der  
 7146 Benutzerverwaltung des Remote-Management-Systems. Auch wenn die Verwaltung von  
 7147 Remote-Administratoren an das Remote-Management-System delegiert ist, werden alle

7148 Zugriffe über das Remote-Management-System auf den Konnektor mit der Rolle Remote-  
 7149 Administrator ausgeführt. Das Remote-Management-System muss die Authentisierung  
 7150 der Remote-Administratoren und die Nachvollziehbarkeit der Zugriffe sicherstellen.

7151 **TIP1-A\_7279 - Authentifizierung des Remote-Administrators**

7152 Wenn in der Benutzerverwaltung des Konnektors Administratoren mit der Administrator-  
 7153 Rolle Remote-Administrator konfiguriert sind, MUSS der Konnektor diese gemäß TIP1-  
 7154 A\_4808 authentifizieren. [≤]

7155 **TIP1-A\_7280 - Einschränkung der Rechte des Remote-Administrators**

7156 Der Konnektor DARF Remote-Administratoren Rechte gemäß TAB\_KON\_851 und  
 7157 TAB\_KON\_655 NICHT gewähren. [≤]

7158

7159 **Tabelle 363: TAB\_KON\_851 Einschränkung der Rechte des Remote-Administrators**  
 7160 **(Blacklist)**

| Fachliche Anbindung der Clientsysteme |  |   |
|---------------------------------------|--|---|
| TIP1-A_4517                           | Schlüssel und X.509-Zertifikate für die Authentisierung des Clientsystems erzeugen und exportieren sowie X.509-Zertifikate importieren |   |
| TIP1-A_4518                           | Konfiguration der Anbindung Clientsysteme  |   |
| Kartendienst                          |  |   |
| TIP1-A_5110                           | Übersicht über alle verfügbaren Karten   | Karten vom Typ eGK und HBA DÜRFEN dem Remote-Administrator NICHT angezeigt werden                 |
| TIP1-A_5111                           | PIN-Management der SM-Bs für den Administrator   |   |
| Zertifikatsdienst                     |  |   |
| TIP1-A_4704                           | Zertifikatsablauf anzeigen   | Zertifikate von Karten vom Typ eGK und HBA DÜRFEN dem Remote-Administrator NICHT angezeigt werden |
| Protokollierungsdienst                |  |   |

|                                |  |   |
|--------------------------------|--|---|
| TIP1-A_4716                    | Einsichtnahme und Veränderung der Protokolle | Personenbezogene Daten DARF der Remote-Administrator NICHT einsehen und exportieren. Fachmodulprotokolle müssen daher entweder gesperrt, oder die personenbezogenen Daten aus diesen für den Remote-Administrator gefiltert werden. |
| TIP1-A_4814                    | Export- Import von Konfigurationsdaten       |   |
| <b>Neustart und Werksreset</b> |  |   |
| TIP1-A_4820                    | Werksreset des Konnektors                    |   |

7161

7162

7163

7164

7165

7166

7167

7168

### **TIP1-A\_5652 - Remote Management Konnektor: Konfiguration Remote Management**

Der Konnektor MUSS sicherstellen, dass es ausschließlich einem Administrator mit einer der Rollen {Lokaler Administrator; Super-Administrator} und dem Recht USER\_INIT\_REMOTESESSION möglich ist, Konfigurationsänderungen gemäß TAB\_KON\_663 vorzunehmen.

**Tabelle 364: TAB\_KON\_663 Konfigurationen des Remote Managements**

| ReferenzID                    | Belegung              | Bedeutung und Administrator-Interaktion   |
|-------------------------------|-----------------------|---|
| MGM_REMOTE_ALLOWED            | Enabled/<br>Disabled  | Der Administrator MUSS einstellen können, ob der Konnektor eine Remote-Management-Verbindung aufbauen kann, über die Konfigurationen vorgenommen werden können.<br>Enabled: Der Konnektor kann eine Remote-Management-Verbindung aufbauen und erlaubt Konfiguration über das Remote-Management System.<br>Disabled: Der Konnektor erlaubt keine Konfiguration über das Remote Management-System<br>Default-Wert: Disabled |
| MGM_REMOTE_MONITORING_ALLOWED | Enabled /<br>Disabled | Der Konnektor KANN Remote-Monitoring unterstützen.<br>In diesem Fall MUSS der Konnektor dem Administrator die Aktivierung und Deaktivierung des Remote-Monitoring ermöglichen.<br>Enabled: Der Konnektor baut eine Remote-Managementverbindung  |



|   |  |  |
|---|--|--|
|   |  | <p>auf.</p> <p>Der Konnektor übermittelt Betriebszustände gemäß TAB_KON_503 an das Remote-Management-System.</p> <p>Disabled: Remote-Monitoring ist deaktiviert. Der Konnektor übermittelt keine Betriebszustände an das Remote-Management-System.</p> <p>Default-Wert: Disabled</p> |
| <p>Der Konnektor SOLL die Konfiguration der URL des Remote-Management-Systems, der Zertifikatsinformationen zur Authentisierung des Remote-Management-Systems und der Credentials für die Authentisierung des Konnektors beim Remote-Management-System ermöglichen.</p> |  |  |

7169  
7170  
7171

[<=]

7172 **TIP1-A\_5653 - Remote Management Konnektor: Protokollierung Remote**  
7173 **Management**

7174 Der Konnektor MUSS im Rahmen des Remote-Managements folgende Aktionen  
7175 protokollieren:

- 7176
- 7177 • Beginn einer Remote-Session durch  
7178 TUC\_KON\_271 „Schreibe Protokolleintrag“ {  
7179 topic = „MGM/REMOTE\_SESSION“;  
7180 eventType = Op;  
7181 severity = Info;  
7182 parameters = („InitUser=\$AdminUsername,  
7183 RemoteID=<Kennung der Gegenstelle>,  
Mode=[InitSuccess/InitFail]“)}  
7184
  - 7185 • Verbindungsabbau Remote-Session durch  
7186 TUC\_KON\_271 „Schreibe Protokolleintrag“ {  
7187 topic = „MGM/REMOTE\_SESSION“;  
7188 eventType = Op;  
7189 severity = Info;  
7190 parameters = („InitUser=\$AdminUsername,  
7191 RemoteID=<Kennung der Gegenstelle>,  
7192 Mode=Exit“}  
7193 Die Protokollierungspflicht gilt nicht für das Remote Monitoring.  
7194 Wenn ein remote-Zugriff erfolgt, ohne dass ein Remote-Administrator im  
7195 Konnektor konfiguriert ist, so MUSS als InitUser eine Referenz auf das Remote-  
Management-System verwendet werden.

7196 [≤]

7197 Ein Softwareupdate gemäß TIP1-A\_5657 kann auch über das Remote Management  
7198 initiiert, aktiviert und freigeschaltet werden.

#### 4.3.9 Software- und Konfigurationsaktualisierung (KSR-Client)

Die Umsetzung des KSR-Clients bezüglich des Mechanismus zur Durchführung der Aktualisierungen, sowie die Art der Darstellung an der Managementschnittstelle sind herstellerspezifisch.

Innerhalb der Software- und Konfigurationsaktualisierung (KSR-Client) werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): „KSR“
- Konfigurationsparameter: „MGM\_“

##### 4.3.9.1 Funktionsmerkmalweite Aspekte

Der Konnektor muss einen KSR-Client bereitstellen, über den der Administrator sowohl den Konnektor selbst als auch die vom Konnektor verwalteten Kartenterminals (CT-Objects in CTM\_CT\_LIST mit CT.CORRELATION>=„gepairt“ und CT.VALID\_VERSION=True und CT.IS\_PHYSICAL = Ja) softwareseitig aktualisieren kann.

Weiterhin muss über den KSR-Client eine Aktualisierung von ausgewählten Konfigurationsdaten möglich sein.

##### **TIP1-A\_4829 - Vollständige Aktualisierbarkeit des Konnektors**

Die Software-Aktualisierung des Konnektor SOLL sicherstellen, dass alle Software-Bestandteile des Konnektors aktualisiert werden können, damit eine ungehinderte Nachnutzung der Hardware-Basis im Feld mit neuen Funktionalitäten nicht durch nichtaktualisierbare Software-Bestandteile gefährdet wird. Weicht ein Hersteller für sein Konnektormodell von dieser Forderung in Teilen ab, so MUSS er im Rahmen der Zulassung nachweisen, dass dies auf Grund von Sicherheitsaspekten für sein eingereichtes Konnektormodell zwingend erforderlich ist.

[<=]

##### **TIP1-A\_5657-02 - Freischaltung von Softwareupdates**

Der Konnektor MUSS die Möglichkeit bieten, dass Softwareupdates durch den Nutzer bzw. einen von ihm beauftragten Administrator einzeln freigeschaltet werden.

[<=]

##### **A\_18387 - Automatische Softwareupdates**

Der Konnektor MUSS die Möglichkeit bieten, die automatische Installation von Softwareupdates pro Gerät (Konnektor und Kartenterminals) ein- und auszuschalten.[<=]

##### **A\_18389 - Nur Nutzung von zugelassenen Versionen**

Der Hersteller des Konnektors MUSS in seinem Handbuch den Nutzer darauf hinweisen, dass er sich bei der Arbeit mit dem Konnektor vergewissern muss, dass er mit einer zugelassenen Version arbeitet und beschreiben, wie der Nutzer diese Information mittels seines Primärsystems erhalten kann.

[<=]

##### **TIP1-A\_6476 - Lieferung von Softwareupdates**

Der Hersteller des Konnektors MUSS jede zugelassene Firmware-Version umgehend als Update-Paket über die in [gemSpec\_KSR] definierte Schnittstelle P\_KSRS\_Upload im Konfigurationsdienst (KSR) ablegen.

Der Hersteller des Konnektors MUSS in den jeweiligen UpdateInformation/Firmware/FirmwareReleaseNotes eine Internet-URL zum Download des FW-Updates bereitstellen.

[<=]



7246 **TIP1-A\_6026 - Anzeige URL zum Download des FW-Updates an der**  
 7247 **Managementschnittstelle**  
 7248 Das Managementinterface des Konnektors MUSS einem authentisierten Administrator die  
 7249 Internet-URL zum Download des FW-Updates anzeigen.  
 7250 [ $\leq$ ]

#### 7251 4.3.9.2 Durch Ereignisse ausgelöste Reaktionen

7252 **TIP1-A\_4831 - KT-Update nach Wiedererreichbarkeit erneut anstoßen**  
 7253 Wenn aus (TIP1-A\_4840 Auslösen der durchzuführenden Updates) heraus für ein  
 7254 Kartenterminal noch ein ausstehendes Updates vorhanden ist, dessen  
 7255 Ausführungszeitpunkt nicht gesetzt oder überschritten ist, und für dieses Kartenterminal  
 7256 das Ereignis „CT/CONNECTED“ eintritt, so MUSS TUC\_KON\_281  
 7257 „Kartenterminalaktualisierung anstoßen“ für dieses KT gerufen werden.  
 7258 [ $\leq$ ]

#### 7259 4.3.9.3 Interne TUCs, nicht durch Fachmodule nutzbar

7260 4.3.9.3.1 TUC\_KON\_280 „Konnektoraktualisierung durchführen“

7261 **TIP1-A\_4832-02 - TUC\_KON\_280 „Konnektoraktualisierung durchführen“**  
 7262 Der Konnektor MUSS den technischen Use Case TUC\_KON\_280 „Konnektoraktualisierung  
 7263 durchführen“ umsetzen.  
 7264

7265 **Tabelle 365: TAB\_KON\_664 – TUC\_KON\_280 „Konnektoraktualisierung durchführen“**

| Element         | Beschreibung   |
|-----------------|--|
| Name            | TUC_KON_280 „Konnektoraktualisierung durchführen“  |
| Beschreibung    | Dieser TUC aktualisiert den Konnektor mit einem Update, dessen Update-Dateien entweder direkt übergeben oder per UpdateInformation (vom KSRS bezogen) referenziert werden  |
| Auslöser        | <ul style="list-style-type: none"> <li>Der Administrator hat UpdateInformation zur Anwendung ausgewählt und bestätigt bzw. ein lokales Updatepaket bezogen und zur Anwendung übergeben.</li> <li>automatisches Softwareupdate [A_18387]</li> </ul> |
| Vorbedingungen  |  |
| Eingangsdaten   | <ul style="list-style-type: none"> <li>UpdateInformation (gemäß [gemSpec_KSR#5.2])</li> </ul> oder <ul style="list-style-type: none"> <li>Updatepaket (herstellerspezifisch, von lokaler Datenquelle, mit enthaltenen FirmwareFiles)</li> </ul>    |
| Komponenten     | Konnektor, Konfigurationsdienst  |
| Ausgangsdaten   | Keine  |
| Nachbedingungen | Der Konnektor arbeitet basierend auf der übergebenen, im Updatepaket enthaltenen neuen Version.  |
| Standardablauf  | Der Konnektor MUSS die zur Anwendung übergebene UpdateInformation wie folgt anwenden:  |

|  |  |
|--|--|
|  | <ol style="list-style-type: none"> <li>1. Integrität und Authentizität der UpdateInformation prüfen (Mechanismus ist herstellerspezifisch)</li> <li>2. Download aller in UpdateInformation.FirmwareFiles gelisteten Dateien. Dabei wird die Komprimierung des File Transfers vom Konfigurationsdienst über http „Content Coding“ [RFC2616] „gzip“ genutzt.</li> <li>3. Integrität und Authentizität jeder der via UpdateInformation/FirmwareFiles heruntergeladenen Dateien prüfen (Mechanismus ist herstellerspezifisch)</li> <li>4. Prüfen auf Zulässigkeit des Updates basierend auf der Firmware-Gruppe (siehe [gemSpec_OM#2.5])</li> <li>5. Anwenden der zur Verfügung stehenden FirmwareFiles <ol style="list-style-type: none"> <li>a. TUC_KON_256{ <pre> topic = „KSR/UPDATE/START“; eventType = Sec; severity = Info; parameters = („Target=Konnektor, Name=\$MGM_KONN_HOSTNAME“ )} (betroffene Fachmodule und Basisdienste reagieren und stoppen sich) </pre> </li> <li>b. Herstellerspezifischer Mechanismus zur Aktualisierung der internen Konnektorsoftware durch die FirmwareFiles inklusive anschließender Prüfung auf Erfolg.</li> <li>c. Bestehende Konfigurationsdaten des Konnektors MÜSSEN erhalten bleiben und sofern erforderlich und möglich automatisch auf die Definitionen der neuen Firmware angepasst werden.</li> <li>d. Ist ein händischer Anpassungs- oder Ergänzungsbedarf der Konfigurationsdaten erforderlich, so MUSS der Administrator hierüber geeignet informiert werden</li> <li>e. TUC_KON_256 { <pre> topic = „KSR/UPDATE/SUCCESS“; eventType = Sec; severity = Info; parameters = („Target=Konnektor, Name= \$MGM_KONN_HOSTNAME, NewFirmwareversion = UpdateInformation.FirmwareVersio n, ConfigurationChanged=&lt;Ja/Nein&gt;, ManualInputNeeded=&lt;Ja/Nein&gt;„) } </pre> </li> </ol> <p>Der TUC endet in jedem Fall mit:</p> <pre> TUC_KON_256 { topic = „KSR/UPDATE/END“; eventType = Sec; severity = Info; </pre> </li> </ol> |
|--|--|

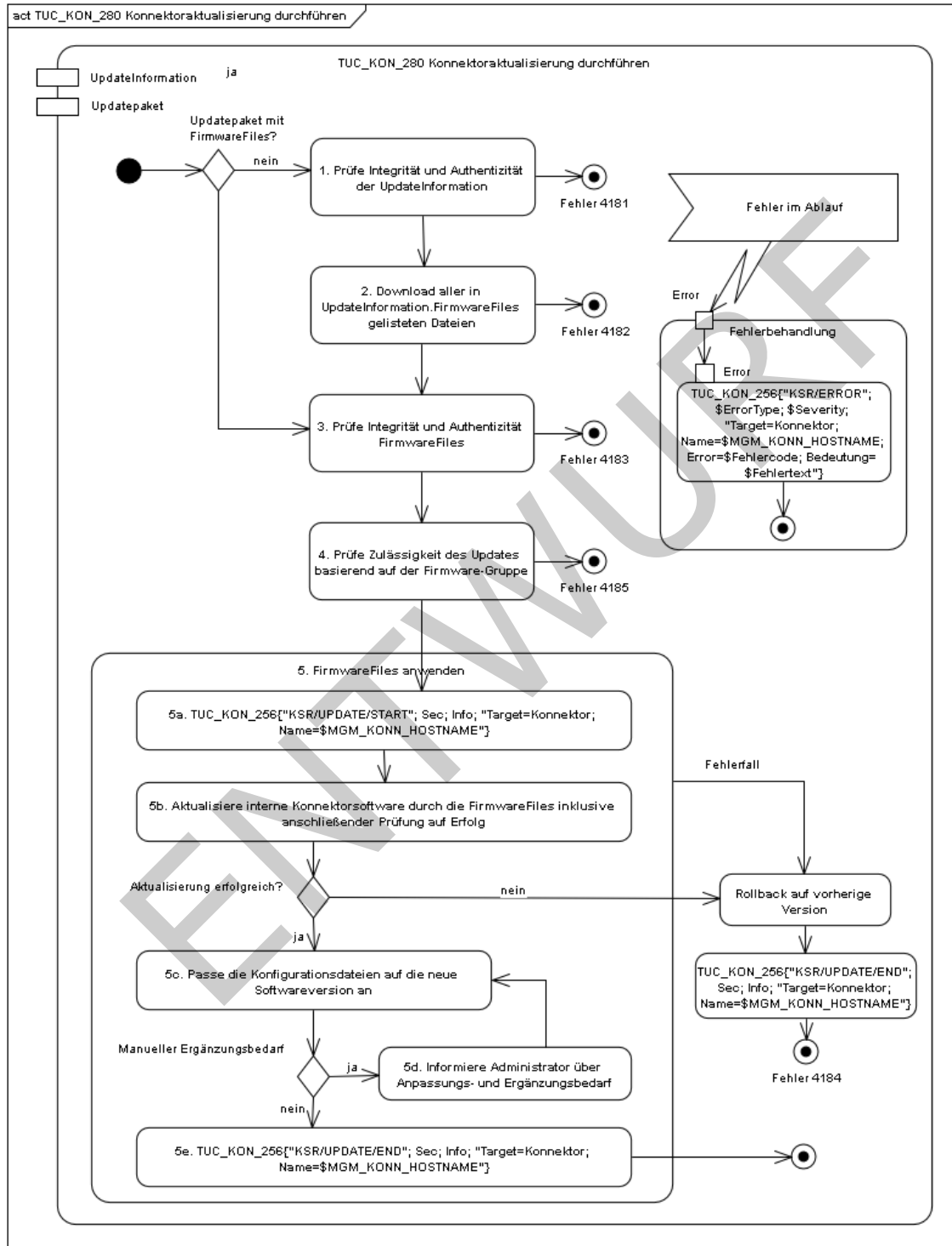
|                                   |  |
|-----------------------------------|--|
|                                   | parameters = („Target=Konnektor,<br>Name=\$MGM_KONN_HOSTNAME“) }<br>(betroffene Fachmodule und Basisdienste reagieren und<br>starten sich)   |
| Varianten/Alternative<br>n        | Sofern direkt ein Updatepaket (mit enthaltenen FirmwareFiles)<br>übergeben wurde beginnt der Ablauf ab Nummer 4 mit der<br>Integritätsprüfung des Updatepakets   |
| Fehlerfälle                       | <p>Fehler in den folgenden Schritten des Ablaufs führen zu:</p> <p>a) Aufruf von TUC_KON_256 {<br/>topic = „KSR/ERROR“;<br/>eventType = \$ErrorType;<br/>severity = \$Severity;<br/>parameters = („Target=Konnektor,<br/>Name= \$MGM_KONN_HOSTNAME,<br/>Error=\$Fehlercode,<br/>Bedeutung=\$Fehlertext“) }</p> <p>b) Abbruch der Verarbeitung mit den ausgewiesenen<br/>Fehlercodes</p> <p>(→1) Integritätsprüfung UpdateInformation fehlgeschlagen,<br/>Fehlercode: 4181<br/>(→2) Fehler bei der Downloaddurchführung, Fehlercode: 4182<br/>(→3) Integritätsprüfung eines FirmwareFiles fehlgeschlagen,<br/>Fehlercode: 4183<br/>(→ 4) Firmwaregruppenprüfung fehlgeschlagen, Fehlercode:<br/>4185<br/>(→5b) Interne Aktualisierung fehlgeschlagen, dann:<br/>1. Rollback auf vorherige Version<br/>2. Abbruch mit Fehlercode: 4184</p> |
| Nichtfunktionale<br>Anforderungen | Der laufende Updatevorgang MUSS in der<br>Managementschnittstelle ausgewiesen und der Fortschritt<br>mindestens für die Schritte 1-5b dargestellt werden.  |
| Zugehörige<br>Diagramme           | Abbildung PIC_KON_105 Aktivitätsdiagramm<br>Konnektoraktualisierung durchführen  |

7266  
7267**Tabelle 366: TAB\_KON\_665 Fehlercodes TUC\_KON\_280 „Konnektoraktualisierung durchführen“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4181  | Security  | Error    | Integritätsprüfung UpdateInformation fehlgeschlagen.  |
| 4182  | Security  | Error    | Download nicht aller UpdateFiles möglich.             |
| 4183  | Security  | Error    | Integritätsprüfung UpdateFiles fehlgeschlagen.        |
| 4184  | Security  | Error    | Anwendung der UpdateFiles fehlgeschlagen (<Details>). |

|      |          |       |   |
|------|----------|-------|---|
| 4185 | Security | Error | Firmware-Version liegt außerhalb der gültigen Firmware-Gruppe |
|------|----------|-------|---|

7268  
7269



7270

7271

7272

7273

**Abbildung 22: PIC\_KON\_105 Aktivitätsdiagramm Konnektoraktualisierung durchführen**

[<=]

## 7274 4.3.9.3.2 TUC\_KON\_281 „Kartenterminalaktualisierung anstoßen“

7275 Im Vergleich zur Durchführung des Konnektor-Update (TUC\_KON\_280), werden die  
7276 Updates der Kartenterminals nur durch den Konnektor initiiert. Der Konnektor liefert dem  
7277 Kartenterminal das Updatefile, der eigentliche Updatevorgang (inklusive der Prüfung des  
7278 Updatepakets auf Integrität und Authentizität) erfolgt ausschließlich und  
7279 eigenverantwortlich auf Seiten des Kartenterminals.

7280 **TIP1-A\_4833-02 - TUC\_KON\_281 „Kartenterminalaktualisierung anstoßen“**

7281 Der Konnektor MUSS den technischen Use Case TUC\_KON\_281

7282 „Kartenterminalaktualisierung anstoßen“ umsetzen.

7283

7284 **Tabelle 367: TAB\_KON\_666 – TUC\_KON\_281 „Kartenterminalaktualisierung anstoßen“**

| Element         | Beschreibung  |
|-----------------|---|
| Name            | TUC_KON_281 „Kartenterminalaktualisierung anstoßen“   |
| Beschreibung    | Dieser TUC fordert ein Kartenterminal auf einen Update durchzuführen, dessen Update-Dateien entweder direkt übergeben oder per UpdateInformation (vom KSRS bezogen) referenziert werden   |
| Auslöser        | <ul style="list-style-type: none"> <li>Der Administrator hat UpdateInformation für ein Kartenterminal zur Anwendung ausgewählt und bestätigt bzw. ein lokales Updatepaket für ein Kartenterminal bezogen und zur Anwendung übergeben.</li> <li>automatisches Softwareupdate [A_18387]</li> </ul>  |
| Vorbedingungen  | <ul style="list-style-type: none"> <li>CT(ctId).IS_PHYSICAL=Ja</li> <li>CT(ctId).CORRELATION&gt;="gepairt"</li> </ul>   |
| Eingangsdaten   | <ul style="list-style-type: none"> <li>ctId (ID des Ziel-KTs)</li> <li>UpdateInformation (gemäß [gemSpec_KSR]) oder</li> <li>Updatepaket (herstellerspezifisch, von lokaler Datenquelle, mit enthaltenen FirmwareFiles)</li> </ul>  |
| Komponenten     | Konnektor, Kartenterminal   |
| Ausgangsdaten   | Keine   |
| Nachbedingungen | Das Kartenterminal arbeitet basierend auf der übergebenen, im Updatepaket enthaltenen neuen Version.  |
| Standardablauf  | <p>Der Konnektor MUSS die zur Anwendung übergebene UpdateInformation wie folgt anwenden:</p> <ol style="list-style-type: none"> <li>Download der in UpdateInformation/FirmwareFiles gelisteten Datei (für KT-Updates darf nur genau ein FirmwareFile angegeben werden)</li> <li>TUC_KON_256{             <ul style="list-style-type: none"> <li>topic = „KSR/UPDATE/START“;</li> <li>eventType = Sec;</li> <li>severity = Info;</li> <li>parameters = („Target=KT, CtID=\$ctId“) }</li> </ul> </li> </ol> |

|                                |   |
|--------------------------------|---|
|                                | <p>3. Durchführen des KT-Updates durch:</p> <p>a) Wechsel in eine Admin-Session durch TUC_KON_050<br/>„Beginne Kartenterminalsitzung“{role=„Admin“; ctId}</p> <p>b) Senden der SICCT Kommandos: SICCT CT Download INIT,<br/>SICCT CT Download DATA (Übermittlung des UpdateFiles) und<br/>SICCT CT Download FINISH an ctId</p> <p>c) TUC_KON_256{<br/>topic = „KSR/UPDATE/SUCCESS“;<br/>eventType = Sec;<br/>severity = Info;<br/>parameters = („Target=KT,<br/>Name= \$CT.HOSTNAME, CtID = \$ctId,<br/>NewFirmwareversion =<br/>&lt;UpdateInformation.FirmwareVersion&gt;„,}</p> <p>Der TUC endet in jedem Fall mit:</p> <ul style="list-style-type: none"> <li>TUC_KON_256 {<br/>topic = „KSR/UPDATE/END“;<br/>eventType = Sec;<br/>severity = Info;<br/>parameters = („Target=KT, CtID = \$ctId“) }</li> </ul> |
| Varianten/Alternativen         | Sofern direkt ein Updatepaket (mit enthaltenem FirmwareFile) übergeben wurde beginnt der Ablauf ab Nummer 2 mit Signalisierung des Beginns des KT-Updates   |
| Fehlerfälle                    | <p>Fehler in den folgenden Schritten des Ablaufs führen zu:</p> <p>a) Aufruf von TUC_KON_256 {<br/>topic = „KSR/ERROR“;<br/>eventType = \$ErrorType;<br/>severity = \$Severity;<br/>parameters = („Target=KT, Name=\$CT.HOSTNAME,<br/>CtID = \$ctId, Error=\$Fehlercode,<br/>Bedeutung=\$Fehlertext“) }</p> <p>b) Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes<br/>(→1) Download fehlgeschlagen, Fehlercode: 4186<br/>(→3b) SICCT-Download fehlgeschlagen, Fehlercode: 4187</p>   |
| Nichtfunktionale Anforderungen | <p>Die Durchführung eines KT-Updates DARF die weitere Operation des Konnektors NICHT behindern (weder auf Schnittstellenebene noch in der Managementschnittstelle). Der laufende Updatevorgang eines KT MUSS in der Managementschnittstelle ausgewiesen und der Fortschritt dargestellt werden.</p> <p>Der Konnektor MUSS mindestens 5 Kartenterminal-Updates parallel durchführen können.</p>  |
| Zugehörige Diagramme           | keine   |

7285 **Tabelle 368: TAB\_KON\_667 Fehlercodes TUC\_KON\_281 „Kartenterminalaktualisierung**  
 7286 **anstoßen“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4186  | Security  | Error    | Download nicht aller UpdateFiles möglich.           |
| 4187  | Security  | Error    | KT-Update fehlgeschlagen (<Fehlerinfo gemäß SICCT>) |

7287  
 7288 [ $\leq$ ]

7289

7290 4.3.9.3.3 TUC\_KON\_282 „UpdateInformationen beziehen“

7291 **TIP1-A\_4834 - TUC\_KON\_282 „UpdateInformationen beziehen“**

7292 Der Konnektor MUSS den technischen Use Case TUC\_KON\_282 „UpdateInformationen  
 7293 beziehen“ umsetzen.

7294

7295 **Tabelle 369: TAB\_KON\_668 – TUC\_KON\_282 „UpdateInformationen beziehen“**

| Element         | Beschreibung  |
|-----------------|---|
| Name            | TUC_KON_282 „UpdateInformationen beziehen“  |
| Beschreibung    | Dieser TUC ermittelt vom zentralen Konfigurationsdienst sowohl für den Konnektor als auch für alle durch ihn verwalteten Kartenterminals die verfügbaren UpdateInformationen  |
| Auslöser        | <ul style="list-style-type: none"> <li>Manuell durch den Administrator</li> <li>Automatisch</li> </ul>  |
| Vorbedingungen  | Keine   |
| Eingangsdaten   | Keine   |
| Komponenten     | Konnektor, Konfigurationsdienst   |
| Ausgangsdaten   | Keine   |
| Nachbedingungen | Der Konnektor verfügt über alle aktuellen UpdateInformationen   |
| Standardablauf  | Der Konnektor MUSS folgende Schritte durchlaufen:<br>1. Der Konnektor MUSS die TLS-Verbindungen zum Konfigurationsdienst anhand des in MGM_KSR_FIRMWARE_URL angegebenen Wertes aufbauen. Dabei MUSS er das durch den Server präsentierte Zertifikat mittels TUC_KON_037 „Zertifikat prüfen“ {<br>certificate = C.ZD.TSL-S;<br>qualifiedCheck = not_required;<br>offlineAllowNoCheck = true;<br>policyList = oid_zd_tls_s; |

|                        |  |
|------------------------|--|
|                        | <p>intendedKeyUsage= intendedKeyUsage(C.ZD.TLS-S);<br/> intendedExtendedKeyUsage = id-kp-serverAuth;<br/> validationMode = OCSP}<br/> auf Gültigkeit prüfen.<br/> Das Server-Zertifikat C.ZD.TLS-S MUSS für den Konfigurationsdienst ausgestellt sein.</p> <p>2. Der Konnektor MUSS sowohl für sich wie auch für jedes Kartenterminal (CT) aus CTM_CT_LIST mit CT.IS_PHYSICAL=Ja und CT.CORRELATION&gt;=„gepairt“ folgende Schritte durchlaufen:</p> <p>a. Belegen von listUpdatesRequest mit den korrekten Werten für ProductVendorID, ProductCode, HardwareVersion und FirmwareVersion</p> <p>b. Aufruf von I_KSRS_Download::list_Updates</p> <p>Liefert der Aufruf mindestens eine UpdateInformation mit einer UpdateInformation/Firmware/FWVersion &gt; aktuelle Version der Konnektorsoftware, deren UpdateInformation/Firmware/FWPriority = „Kritisch“, dann geht der Konnektor über in den Betriebszustand EC_Connector_Software_Out_Of_Date.</p> <p>Liefert der Aufruf mindestens eine UpdateInformation mit einer UpdateInformation/Firmware/FWVersion &gt; aktuelle Version der Kartenterminalsoftware, deren UpdateInformation/Firmware/FWPriority = „Kritisch“, dann geht der Konnektor über in den Betriebszustand EC_CardTerminal_Software_Out_Of_Date.</p> <p>3. Beenden der TLS-Verbindung</p> |
| Varianten/Alternativen | Keine  |
| Fehlerfälle            | <p>Fehler in den folgenden Schritten des Ablaufs führen zu:</p> <p>a) Aufruf von TUC_KON_256 {<br/> topic = „KSR/ERROR“;<br/> eventType = \$ErrorType;<br/> severity = \$Severity;<br/> parameters = („Error=\$Fehlercode;<br/> Bedeutung=\$Fehlertext“)}<br/> b) Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes<br/> (→1) Konfigurationsdienst nicht erreichbar, Fehlercode: 4188<br/> (→1) Serverzertifikat ist nicht C.ZD.TLS_S, Fehlercode: 4189</p>   |



|                                |  |
|--------------------------------|--|
|                                | (→2b) Fehler beim Beziehen der Updatelisten, Fehlercode: 4190  |
| Nichtfunktionale Anforderungen | Der Konnektor muss die Vorgaben aus [gemSpec_Krypt#3.3.2] für TLS-Verbindungen und hinsichtlich ECC-Migration die Vorgaben aus [gemSpec_Krypt#5] befolgen. |
| Zugehörige Diagramme           | keine  |

7296 **Tabelle 370: TAB\_KON\_669 Fehlercodes TUC\_KON\_282 „UpdateInformationen beziehen“**

| Fehlercode   | ErrorType | Severity | Fehlertext  |
|--|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases, sowie der Fehlercodes von „I_KSRS_Download::listUpdates Response“ können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4188   | Technical | Error    | Konfigurationsdienst nicht erreichbar, konfigurierte Adresse kontrollieren. |
| 4189   | Security  | Fatal    | Konfigurationsdienst liefert falsches Zertifikat                            |
| 4190   | Technical | Error    | Fehler beim Beziehen der Updatelisten                                       |

7297

7298 [ $\leq$ ]

7299 4.3.9.3.4 TUC\_KON\_283 „Infrastruktur Konfiguration aktualisieren“

7300 **TIP1-A\_5153 - TUC\_Kon\_283 „Infrastruktur Konfiguration aktualisieren“**

7301 Der Konnektor MUSS den technischen Use Case TUC\_Kon\_283 „Infrastruktur  
7302 Konfiguration aktualisieren“ umsetzen.

7303

7304 **Tabelle 371: TAB\_KON\_799 – TUC\_KON\_283 „Infrastruktur Konfiguration aktualisieren“**

| Element        | Beschreibung   |
|----------------|--|
| Name           | TUC_KON_283 Infrastruktur Konfiguration aktualisieren  |
| Beschreibung   | Dieser TUC liest die Infrastrukturdaten vom KSR ein.   |
| Auslöser       | Automatisch einmal täglich; BOOTUP, Administrator  |
| Vorbedingungen | Der TUC_KON_304 „Netzwerk-Routen einrichten“ MUSS fehlerfrei durchgelaufen sein.<br>Der TUC_KON_321 „Verbindung zu dem VPN-Konzentrator der TI aufbauen“ MUSS fehlerfrei durchgelaufen sein. |
| Eingangsdaten  | Keine  |
| Komponenten    | Konnektor, Konfigurationsdienst  |

|               |       |
|---------------|-------|
| Ausgangsdaten | Keine |
|---------------|-------|

ENTWURF

|                |  |
|----------------|--|
| Standardablauf | <p>Der Konnektor MUSS folgende Schritte durchlaufen:</p> <ol style="list-style-type: none"> <li>1. „Einlesen des Konfigurations-XML“: <ol style="list-style-type: none"> <li>a. Der Konnektor MUSS eine TLS-Verbindung zum Konfigurationsdienst anhand des in MGM_KSR_KONFIG_URL angegebenen Parameters aufbauen. Dabei MUSS er das durch den Server präsentierte Zertifikat mittels TUC_KON_037 „Zertifikat prüfen“ { <pre> certificate = C.ZD.TSL-S; qualifiedCheck = not_required; offlineAllowNoCheck = true; policyList = oid_zd_tls_s; intendedKeyUsage =     intendedKeyUsage(C.ZD.TSL-S); intendedExtendedKeyUsage = id-kp-serverAuth; validationMode = OCSP} auf Gültigkeit prüfen. Das Server-Zertifikat C.ZD.TLS-S MUSS für den Konfigurationsdienst ausgestellt sein.</pre> </li> <li>b. Herunterladen der Konfigurationsdaten mittels I_KSRS_Download::get_Ext_Net_Config (MGM_KSR_KONFIG_URL, „Bestandsnetze.xml“)</li> </ol> </li> <li>2. Beenden der TLS-Verbindung<br/> „Prüfen der Versionskennung auf Änderungen“:<br/> Wenn das Element /Infrastructure/Version der heruntergeladenen Datei keine höhere Versionsnummer als die aktuell im Konnektor hinterlegte Version trägt, muss der TUC ohne Fehler beendet und ein Protokolleintrag geschrieben werden:<br/> TUC_KON_271 „Schreibe Protokolleintrag“ { <pre> topic = „KSR/UPDATE_KONFIG“; eventType = Op; severity = Info; parameters = („AlteVersion=\$aktuelleVersion,     NeueVersion=/Infrastructure/Version “)} </pre> </li> <li>3. Aktualisieren der Gesamtnetzliste<br/> Alle in der Datei enthaltenen Netzsegmente sind nach ANLW_BESTANDSNETZE zu übernehmen. In Abhängigkeit von ANLW_IA_BESTANDSNETZE sind neue angeschlossene Netze des Gesundheitswesens mit aAdG-NetG nach ANLW_AKTIVE_BESTANDSNETZE zu übernehmen. Identifiziert wird ein Bestandsnetz hierbei an dessen ID in der Bestandsnetze.xml (&lt;ID&gt;). War der Aktivierungsstatus eines dieser Netze bereits durch den Administrator manuell konfiguriert, so muss dieser Status erhalten bleiben.</li> <li>4. „Aktualisieren von Konfigurationsinformationen“<br/> Haben sich Konfigurationsdaten zu einem in</li> </ol> |
|----------------|--|

|  |   |
|--|---|
|  | <p>ANLW_AKTIVE_BESTANDSNETZE gelisteten Netz verändert, so</p> <ol style="list-style-type: none"> <li>a. sind die Änderungen entsprechend zu übernehmen und zu aktivieren (Anpassung ANLW_AKTIVE_BESTANDSNETZE, DHCP_AKTIVE_BESTANDSNETZE_ROUTES, DNS_SERVERS_BESTANDSNETZE).</li> <li>b. alle Statusänderungen an ANLW_AKTIVE_BESTANDSNETZE sind zu protokollieren. Der Protokolleintrag je Änderung enthält den Status, &lt;ID&gt;, &lt;Name&gt; und &lt;NetworkAddress/NetworkPrefix&gt; als topic=KSR/UPDATE_KONFIG,protocolType=OP und protocolSeverity=INFO.</li> <li>c. ist anschließend TUC_KON_304 „Netzwerk-Routen einrichten“ aufzurufen</li> </ol> <p>5. „Entfernen von nicht mehr gültigen angeschlossenen Netzen des Gesundheitswesens mit aAdG-NetG“<br/>Ist ein Netz in der neuen Datei gegenüber der alten Datei nicht mehr vorhanden, so:</p> <ol style="list-style-type: none"> <li>a. a) sind alle diesbezüglichen Daten zu entfernen und die Änderungen direkt aktiv zu schalten (Anpassung ANLW_AKTIVE_BESTANDSNETZE, DHCP_AKTIVE_BESTANDSNETZE_ROUTES, DNS_SERVERS_BESTANDSNETZE).</li> <li>b. b) ist anschließend TUC_KON_304 „Netzwerk-Routen einrichten“ aufzurufen.</li> </ol> <p>6. Protokollierung der heruntergeladenen Version von Bestandsnetze.xml durch Aufruf von TUC_KON_271 „Schreibe Protokolleintrag“ {<br/>             topic = „KSR/UPDATE_KONFIG“;<br/>             eventType = Op;<br/>             severity = Info;<br/>             parameters = („AlteVersion=\$aktuelleVersion,<br/>                             NeueVersion=/Infrastructure/Version<br/>           “)}       </p> |
|--|---|

|                                |  |
|--------------------------------|--|
| Varianten/Alternativen         | Keine  |
| Fehlerfälle                    | (→ 1-5) Es ist ein unerwarteter Fehler aufgetreten; Fehlercode: 4198 |
| Nichtfunktionale Anforderungen | Keine  |
| Zugehörige Diagramme           | Keine  |

**Tabelle 372: Tab\_Kon\_726 Fehlercodes TUC\_KON\_283 „Infrastruktur Konfiguration aktualisieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext  |
|---|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4198  | Technical | Error    | Beim Übernehmen der angeschlossenen Netze des Gesundheitswesens mit aAdG-NetG ist ein Fehler aufgetreten. |

[<=]

#### 4.3.9.4 Interne TUCs, auch durch Fachmodule nutzbar

##### 4.3.9.4.1 TUC\_KON\_285 „UpdateInformationen für Fachmodul beziehen“

#### **TIP1-A\_6018 - TUC\_KON\_285 „UpdateInformationen für Fachmodul beziehen“**

Der Konnektor MUSS den technischen Use Case TUC\_KON\_285 „UpdateInformationen für Fachmodul beziehen“ umsetzen.

**Tabelle 373: TAB\_KON\_833 – TUC\_KON\_285 „UpdateInformationen für Fachmodul beziehen“**

| Element      | Beschreibung   |
|--------------|--|
| Name         | TUC_KON_285 „UpdateInformationen für Fachmodul beziehen“   |
| Beschreibung | Dieser TUC ermittelt vom zentralen Konfigurationsdienst für ein Fachmodul die verfügbaren UpdateInformationen eines angegebenen SW-Pakets. |

|                 |  |
|-----------------|--|
| Auslöser        | <ul style="list-style-type: none"> <li>Aufruf durch Fachmodul</li> </ul>   |
| Vorbedingungen  | <ul style="list-style-type: none"> <li>Verbindung zum VPN-Konzentrator der TI wurde erfolgreich aufgebaut</li> </ul>   |
| Eingangsdaten   | <ul style="list-style-type: none"> <li>productVendorID [String] -<br/>(Identifiziert den Hersteller des Produkts, für welches auf Updates geprüft werden soll.)</li> <li>productCode [String] -<br/>(Identifiziert das Produkt zusammen mit ProductVendorID, für welches auf Updates geprüft werden soll.)</li> <li>hwVersion [String]<br/>(Identifiziert die Hardware zusammen mit ProductCode und ProductVendorID, für welches auf Updates geprüft werden soll. [gemSpec_OM] beschreibt dieses Element ausführlich.)</li> <li>fwVersion [String]<br/>aktuell im Produkt verwendete Firmwareversion</li> </ul> <p>Hinweis: Definition von productVendorID, productCode, hwVersion, fwVersion (entspricht FWVersion) siehe [gemSpec_KSR#TIP1-A_3331]</p> |
| Komponenten     | Konnektor, Konfigurationsdienst  |
| Ausgangsdaten   | <ul style="list-style-type: none"> <li>listOfUpdates [listUpdatesResponse]<br/>Liste von Update Informationen der verfügbaren Pakete für das angegebene Produkt;<br/>Datentyp listUpdatesResponse definiert in Konfigurationsdienst.xsd siehe [gemSpec_KSR]</li> </ul>   |
| Nachbedingungen | keine  |
| Standardablauf  | <p>Der Konnektor MUSS folgende Schritte durchlaufen:</p> <ol style="list-style-type: none"> <li>Der Konnektor MUSS die TLS-Verbindung zum Konfigurationsdienst anhand des in MGM_KSR_FIRMWARE_URL angegebenen Wertes aufbauen. Dabei MUSS er das durch den Server präsentierte Zertifikat mittels TUC_KON_037 „Zertifikat prüfen“ {<br/> certificate = C.ZD.TSL-S;<br/> qualifiedCheck = not_required;<br/> offlineAllowNoCheck = true;<br/> policyList = oid_zd_tls_s;<br/> intendedKeyUsage =<br/> intendedKeyUsage(C.ZD.TSL-S);<br/> intendedExtendedKeyUsage = id-kp-serverAuth;<br/> validationMode = OCSP}<br/> auf Gültigkeit prüfen.</li> </ol>  |

|                                |  |
|--------------------------------|--|
|                                | <p>Das Server-Zertifikat C.ZD.TLS-S MUSS für den Konfigurationsdienst ausgestellt sein.</p> <ol style="list-style-type: none"> <li>2. Belegen von listUpdatesRequest mit den korrekten Werten für ProductVendorID, ProductCode, HardwareVersion und FirmwareVersion = fwVersion</li> <li>3. Aufruf von I_KSRS_Download::list_Updates gemäß [gemSpec_KSR#TIP1-A_3331]</li> <li>4. Beenden der TLS-Verbindung</li> </ol> |
| Varianten/Alternativen         | Keine  |
| Fehlerfälle                    | <p>Fehler in den folgenden Schritten des Ablaufs führen zu:</p> <p>(→1) Konfigurationsdienst nicht erreichbar, Fehlercode: 4188</p> <p>(→1) Serverzertifikat ist nicht C.ZD.TLS_S, Fehlercode: 4189</p> <p>(→3) Fehler beim Beziehen der Updatelisten, Fehlercode: 4190</p>  |
| Nichtfunktionale Anforderungen | Der Konnektor muss die Vorgaben aus [gemSpec_Krypt#3.3.2] für TLS-Verbindungen und hinsichtlich ECC-Migration die Vorgaben aus [gemSpec_Krypt#5] befolgen.   |
| Zugehörige Diagramme           | keine  |

7318 **Tabelle 374: TAB\_KON\_834 Fehlercodes TUC\_KON\_285 „UpdateInformationen für**  
 7319 **Fachmodul beziehen“**

| Fehlercode   | ErrorType | Severity | Fehlertext  |
|--|-----------|----------|---|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases, sowie der Fehlercodes von „I_KSRS_Download::listUpdates Response“ können folgende weitere Fehlercodes auftreten: |           |          |   |
| 4188   | Technical | Error    | Konfigurationsdienst nicht erreichbar, konfigurierte Adresse kontrollieren. |
| 4189   | Security  | Fatal    | Konfigurationsdienst liefert falsches Zertifikat                            |
| 4190   | Technical | Error    | Fehler beim Beziehen der Updatelisten                                       |

7320  
 7321 [**<=**]

7322 4.3.9.4.2 TUC\_KON\_286 „Paket für Fachmodul laden“

7323 **TIP1-A\_6019 - TUC\_KON\_286 „Paket für Fachmodul laden“**

7324 Der Konnektor MUSS den technischen Use Case TUC\_KON\_286 „Paket für Fachmodul  
 7325 laden“ umsetzen.  
 7326

7327

**Tabelle 375: TAB\_KON\_835 – TUC\_KON\_286 „Paket für Fachmodul laden“**

| Element                | Beschreibung  |
|------------------------|---|
| Name                   | TUC_KON_286 „Paket für Fachmodul laden“   |
| Beschreibung           | Dieser TUC lädt ein bestimmtes SW-Paket für ein Fachmodul vom zentralen Konfigurationsdienst.   |
| Auslöser               | Aufruf durch Fachmodul  |
| Vorbedingungen         | <ul style="list-style-type: none"> <li>Verbindung zum VPN-Konzentrator der TI wurde erfolgreich aufgebaut</li> </ul>  |
| Eingangsdaten          | <ul style="list-style-type: none"> <li>filename<br/>(Filename des SW-Pakets, welches vom KSR geladen werden soll)</li> </ul>  |
| Komponenten            | Konnektor, Konfigurationsdienst   |
| Ausgangsdaten          | <ul style="list-style-type: none"> <li>swPackage<br/>(das durch filename am KSR identifizierte SW-Paket wurde heruntergeladen)</li> </ul>   |
| Nachbedingungen        | keine   |
| Standardablauf         | <ol style="list-style-type: none"> <li>Der Konnektor MUSS die TLS-Verbindung zum Konfigurationsdienst anhand des in MGM_KSR_FIRMWARE_URL angegebenen Wertes aufbauen. Dabei MUSS er das durch den Server präsentierte Zertifikat mittels TUC_KON_037 „Zertifikat prüfen“ {<br/> certificate = C.ZD.TLS-S;<br/> qualifiedCheck = not_required;<br/> offlineAllowNoCheck = true;<br/> policyList = oid_zd_tls_s;<br/> intendedKeyUsage =<br/> intendedKeyUsage(C.ZD.TLS-S);<br/> intendedExtendedKeyUsage = id-kp-serverAuth;<br/> validationMode = OCSP}<br/> auf Gültigkeit prüfen.<br/><br/> Das Server-Zertifikat C.ZD.TLS-S MUSS für den Konfigurationsdienst ausgestellt sein.</li> <li>Herunterladen der Softwarepakets swPackage mittels I_KSRS_Download::get_File<br/>(MGM_KSR_FIRMWARE_URL /\$filename)</li> <li>Beenden der TLS-Verbindung</li> <li>swPackage an Aufrufer zurückgeben</li> </ol> |
| Varianten/Alternativen | keine   |
| Fehlerfälle            | (→ 1) Verbindung zum KSR konnte nicht aufgebaut werden;<br>Fehlercode: 4188<br>(→ 1) Serverzertifikat ist nicht C.ZD.TLS_S, Fehlercode:<br>4189<br>(→ 2) Wenn Größe des Pakets größer als 25MB, Fehlercode:<br>4242   |



|                                |  |
|--------------------------------|--|
|                                | (→ 2) Sonstige Fehler beim Download: Das Paket konnte nicht geladen werden, Fehlercode: 4238 |
| Nichtfunktionale Anforderungen | Keine  |
| Zugehörige Diagramme           | Keine  |

7328 **Tabelle 376: TAB\_KON\_836 Fehlercodes TUC\_KON\_286 „Paket für Fachmodul laden“**

| Fehlercode  | ErrorType | Severity | Fehlertext   |
|---|-----------|----------|--|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten: |           |          |  |
| 4188  | Technical | Error    | Konfigurationsdienst nicht erreichbar, konfigurierte Adresse kontrollieren.        |
| 4189  | Security  | Fatal    | Konfigurationsdienst liefert falsches Zertifikat                                   |
| 4238  | Technical | Error    | Der Download des Pakets vom KSR ist fehlgeschlagen.                                |
| 4242  | Technical | Error    | Der Download des Pakets vom KSR ist fehlgeschlagen. Das Paket ist größer als 25MB. |

7329

7330 [ $\leq$ ]

#### 7331 4.3.9.5 Operationen an der Außenschnittstelle

7332 Keine.

#### 7333 4.3.9.6 Betriebsaspekte

##### 7334 4.3.9.6.1 TUC\_KON\_284 KSR-Client initialisieren

#### 7335 TIP1-A\_5938 - TUC\_KON\_284 „KSR-Client initialisieren“

7336 Der Konnektor MUSS in der Bootup-Phase TUC\_KON\_284 „KSR-Client initialisieren“  
7337 durchlaufen.

7338

7339 **Tabelle 377: TAB\_KON\_864 – TUC\_KON\_284 „KSR-Client initialisieren“**

| Element                     | Beschreibung  |
|-----------------------------|---|
| Name                        | TUC_KON_284 "KSR-Client initialisieren"   |
| Beschreibung                | Der Konnektor muss während des Bootups die Downloadpunkte für Konfigurationsdaten und Firmware ermitteln. |
| Eingangsanforderung         | Keine   |
| Auslöser und Vorbedingungen | Bootup<br>Verbindung zum VPN-Konzentrator TI muss aufgebaut sein  |
| Eingangsdaten               | Keine   |

|                                |  |
|--------------------------------|--|
| Komponenten                    | Konnektor  |
| Ausgangsdaten                  | <ul style="list-style-type: none"> <li>• MGM_KSR_KONFIG_URL</li> <li>• MGM_KSR_FIRMWARE_URL</li> </ul>   |
| Standardablauf                 | <p>- Falls MGM_LU_ONLINE=Enabled:</p> <p>- Durch DNS-Anfragen an den DNS-Forwarder zur Auflösung der SRV-RR und TXT-RR mit den Bezeichnern „_ksrkongfig._tcp.ksr.&lt;TOP_LEVEL_DOMAIN_TI&gt;“ und „_ksrfirmware._tcp.ksr.&lt;TOP_LEVEL_DOMAIN_TI&gt;“ erhält der Konnektor URLs der Downloadpunkte des KSR für Konfigurationsdaten (MGM_KSR_KONFIG_URL) und für Firmware (MGM_KSR_FIRMWARE_URL).</p> |
| Varianten/Alternativen         | Keine  |
| Fehlerfälle                    | Keine  |
| Nichtfunktionale Anforderungen | Keine  |
| Zugehörige Diagramme           | Keine  |

7340 **Tabelle 378: TAB\_KON\_822 Fehlercodes TUC\_KON\_284 „KSR-Client initialisieren“**

| Fehlercode  | ErrorType | Severity | Fehlertext |
|---|-----------|----------|------------|
| Neben den Fehlercodes der aufgerufenen technischen Use Cases können keine weiteren Fehlercodes auftreten. |           |          |            |

7341

7342 [**<=**]

7343 **TIP1-A\_4835-02 - Konfigurationswerte des KSR-Client**

7344 Der Administrator MUSS die in TAB\_KON\_670 aufgelisteten Parameter über die  
 7345 Managementschnittstelle konfigurieren und die in TAB\_KON\_820 aufgelisteten Parameter  
 7346 ausschließlich einsehen können.

7347 **Tabelle 379: TAB\_KON\_670 Konfigurationsparameter der Software-Aktualisierung**

| ReferenzID           | Belegung             | Bedeutung und Administrator-Interaktion   |
|----------------------|----------------------|---|
| MGM_KSR_AUTODOWNLOAD | Enabled/<br>Disabled | Der Administrator MUSS den automatischen Download verfügbarer Update-Pakete über den Konfigurationsparameter MGM_KSR_AUTODOWNLOAD an- und abschalten können.<br>Default-Wert: Enabled |

|                            |  |  |
|----------------------------|--|--|
| MGM_KSR_SHOW_TRIAL_UPDATES | Enabled / Disabled                               | Der Administrator MUSS einschalten können, dass zusätzlich zur Anzeige von Update-Paketen für den Online-Produktivbetrieb auch die Anzeige von Erprobungs-Update-Paketen erfolgt. Wenn MGM_KSR_SHOW_TRIAL_UPDATES von Disabled auf Enabled gesetzt wird, muss ein Warnhinweis angezeigt werden, dass die Installation von Erprobungs-Update-Paketen nur für Teilnehmer der Erprobungen vorgesehen ist.<br>Default-Wert: Disabled |
| MGM_KSR_AUTO_UPDATE        | Enabled / Disabled                               | Der Administrator MUSS pro Gerät (Konnektor und Kartenterminals) das automatische Softwareupdate ein- und ausschalten können.<br>Default-Wert: Enabled<br>Falls MGM_KSR_AUTO_UPDATE=Enabled wird MGM_KSR_AUTODOWNLOAD=Enabled gesetzt.   |
| MGM_KSR_AUTO_UPDATE_TIME   | Wochentag / Uhrzeit<br>Oder<br>täglich / Uhrzeit | Der Administrator MUSS den Wochentag und die Uhrzeit einstellen können, wann automatische Softwareupdates durchgeführt werden. Als Wochentag MUSS es neben den einzelnen Wochentagen auch einen Wert für eine tägliche Prüfung auf Aktualität und gegebenenfalls Durchführung von Softwareupdates geben.<br>Default-Wert: Montag / 1:00 Uhr  |

7348 **Tabelle 380: TAB\_KON\_820 Einsehbare Konfigurationsparameter der Software-**  
7349 **Aktualisierung**

| ReferenzID           | Belegung | Bedeutung und Administrator-Interaktion                                       |
|----------------------|----------|---|
| MGM_KSR_KONFIG_URL   | URL      | SOAP-Endpunkt des Konfigurationsdienstes zum Download von Konfigurationsdaten |
| MGM_KSR_FIRMWARE_URL | URL      | SOAP-Endpunkt des Konfigurationsdienstes zum Download der Firmware            |

7350 [**<=**]

7351 *Hinweis: Die Adressen des Konfigurationsdienstes werden im Rahmen des VPN-*  
7352 *Verbindungsaufbaus ermittelt (siehe [gemSpec\_VPN\_ZugD#5.1.1.2 TUC\_VPN-ZD\_0001])*

7353

7354 **TIP1-A\_6025 - Zugang zur TI sperren, wenn Deadline für kritische FW-Updates**  
7355 **erreicht**

7356 Der Konnektor MUSS täglich überprüfen, ob unter den auf die aktuelle Konnektor-  
7357 Firmware anwendbaren Updates ein Update mit FWPriority = „Kritisch“ ist, dessen  
7358 Deadline (entspricht UpdateInformation/DeploymentInformation/Deadline)  
7359 abgelaufen ist, d.h. Deadline <= Systemzeit. In diesem Fall MUSS der Konnektor den  
7360 Verbindungsaufbau zur TI Plattform verhindern, bestehende Verbindungen in die TI  
7361 abbauen und den kritischen Betriebszustand EC\_FW\_Not\_Valid\_Status\_Blocked

7362 annehmen.

7363 [ $\leq$ ]

7364 **TIP1-A\_4836 - Automatische Prüfung und Download von Update-Paketen**

7365 Der Konnektor MUSS täglich die folgenden Schritte durchführen:

- 7366 1. TUC\_KON\_282 „UpdateInformationen beziehen“ aufrufen.
- 7367 2. pro zurück geliefertem Listeneintrag prüfen, ob eine neuere Version enthalten ist,
- 7368 als auf dem zugehörigen Gerät (Konnektor selbst oder Kartenterminal) vorhanden
- 7369 3. Ist für wenigstens ein Gerät eine neuere Version vorhanden, MUSS der Konnektor
- 7370 darüber via
- 7371 TUC\_KON\_256 „Systemereignis absetzen“ {
- 7372 topic = „KSR/UPDATES\_AVAILABLE“;
- 7373 eventType = Op;
- 7374 severity = Info;
- 7375 parameters = (<Param>);
- 7376 doLog=false }
- 7377 informieren. Je gefundenem Update MUSS <Param> mit folgenden Werten belegt
- 7378 sein:
- 7379 <Param> = „ProductVendorID= \$UpdateInformation/ProductVendorID;
- 7380 ProductCode= \$UpdateInformation/ProductCode;
- 7381 ProductName=\$UpdateInformation/ProductName;
- 7382 FirmwareVersion=\$UpdateInformation/FirmwareVersion;
- 7383 Deadline=\$UpdateInformation/DeploymentInformation/Deadline;
- 7384 FWPriority=\$UpdateInformation/Firmware/FWPriority;
- 7385 FirmwareReleaseNotes=
- 7386 \$UpdateInformation/Firmware/FirmwareReleaseNotes“
- 7387 4. Die listUpdateResponse mit neueren Firmwareversionen MÜSSEN für eine spätere
- 7388 Einsichtnahme durch den Administrator bereitgehalten werden (via (TIP1-A\_4837)
- 7389 „Übersichtsseite des KSR-Client). Ein neuerlicher Abruf dieser Informationen DARF
- 7390 NICHT erforderlich sein.
- 7391 5. Sofern ein Update-Paket für den Konnektor vorliegt, MUSS der Konnektor die mit
- 7392 diesem Paket gelieferten Parameter Priority (entspricht
- 7393 UpdateInformation/Firmware/FWPriority) und Deadline (entspricht
- 7394 UpdateInformation/DeploymentInformation/Deadline) auswerten und bei
- 7395 KSR:Priority=Kritisch persistent ablegen.
- 7396 6. Sofern MGM\_KSR\_AUTODOWNLOAD = Enabled, MUSS der Konnektor bei Update-
- 7397 Paketen, die den Konnektor selbst betreffen, das Update-Paket mit der höchsten
- 7398 FirmwareVersion über I\_KSRS\_Download::get\_Updates herunterladen.
- 7399 7. Ist der Download von Update-Paketen für den Konnektor abgeschlossen, MUSS
- 7400 der Konnektor darüber via
- 7401 TUC\_KON\_256 „Systemereignis absetzen“ {
- 7402 topic = „KSR/UPDATE/KONNEKTOR\_DOWNLOAD\_END“;
- 7403 eventType = Op;
- 7404 severity = Info;
- 7405 parameters = (<Param>)}
- 7406 informieren. Je heruntergeladenem FW-Paket MUSS <Param> mit folgenden
- 7407 Werten belegt sein:
- 7408 <Param> = „ProductVendorID= \$UpdateInformation/ProductVendorID;
- 7409 ProductCode= \$UpdateInformation/ProductCode;
- 7410 ProductName=\$UpdateInformation/ProductName;

```

7411      FirmwareVersion=$UpdateInformation/Firmware/FWVersion;
7412      Deadline=$UpdateInformation/DeploymentInformation/Deadline;
7413      FWPriority=$UpdateInformation/Firmware/FWPriority;
7414      FirmwareReleaseNotes
7415      =$UpdateInformation/Firmware/FirmwareReleaseNotes"

```

8. Sofern MGM\_KSR\_AUTODOWNLOAD = Enabled, SOLL der Konnektor bei Update-Paketen, die Kartenterminals betreffen, pro KT-Modell das Update-Paket mit der höchsten FirmwareVersion über I\_KSRS\_Download::get\_Updates herunterladen.

Der Konnektor MUSS immer nur die neusten Update-Pakete für eine Nutzung vorhalten. Eventuell vorhandene ältere, nicht genutzte Update-Pakete KÖNNEN überschrieben werden.[<=]

#### **TIP1-A\_4836-02 - ab PTV4: Automatische Prüfung und Download von Update-Paketen**

Der Konnektor MUSS täglich die folgenden Schritte durchführen:

1. TUC\_KON\_282 „UpdateInformationen beziehen“ aufrufen.
2. pro zurück geliefertem Listeneintrag prüfen, ob eine neuere Version enthalten ist, als auf dem zugehörigen Gerät (Konnektor selbst oder Kartenterminal) vorhanden
3. Ist für wenigstens ein Gerät eine neuere Version vorhanden, MUSS der Konnektor darüber via  
TUC\_KON\_256 „Systemereignis absetzen“ {  
    topic = „KSR/UPDATES\_AVAILABLE“;  
    eventType = Op;  
    severity = Info;  
    parameters = (<Param>);  
    doLog=false }  
informieren. Je gefundenem Update MUSS <Param> mit folgenden Werten belegt sein:  
    <Param> = „ProductVendorID= \$UpdateInformation/ProductVendorID;  
    ProductCode= \$UpdateInformation/ProductCode;  
    ProductName=\$UpdateInformation/ProductName;  
    FirmwareVersion=\$UpdateInformation/FirmwareVersion;  
    Deadline=\$UpdateInformation/DeploymentInformation/Deadline;  
    FWPriority=\$UpdateInformation/Firmware/FWPriority;  
    FirmwareReleaseNotes=  
    \$UpdateInformation/Firmware/FirmwareReleaseNotes“
4. Ist für wenigstens ein Gerät eine neuere Version vorhanden, MUSS der Konnektor in den Betriebszustand EC\_FW\_Update\_Available übergehen.
5. Die listUpdateResponse mit neueren Firmwareversionen MÜSSEN für eine spätere Einsichtnahme durch den Administrator bereitgehalten werden (via (TIP1-A\_4837) „Übersichtsseite des KSR-Client). Ein neuerlicher Abruf dieser Informationen DARF NICHT erforderlich sein.
6. Sofern ein Update-Paket für den Konnektor selbst vorliegt, MUSS der Konnektor die mit diesem Paket gelieferten Parameter Priority (entspricht UpdateInformation/Firmware/FWPriority) und Deadline (entspricht UpdateInformation/DeploymentInformation/Deadline) auswerten und bei KSR:Priority=Kritisch persistent ablegen.

7. Sofern `MGM_KSR_AUTODOWNLOAD = Enabled`, MUSS der Konnektor bei Update-Paketen, die den Konnektor selbst betreffen, das Updatepaket mit der höchsten FirmwareVersion über `I_KSRS_Download::get_Updates` herunterladen, falls das Update-Paket nicht bereits von einem vorherigen Download auf dem Konnektor vorhanden ist.
8. Sofern `I_KSRS_Download::get_Updates` den http Status Code 503 Server Unavailable zurückgibt, MUSS der Konnektor die Informationen aus dem zurückgegebenen Retry-After Header nutzen, um den Zeitpunkt des Retry zu bestimmen.
9. Ist der Download von Update-Paketen für den Konnektor abgeschlossen, MUSS der Konnektor darüber via  
TUC\_KON\_256 „Systemereignis absetzen“ {  
    topic = „KSR/UPDATE/KONNEKTOR\_DOWNLOAD\_END“;  
    eventType = Op;  
    severity = Info;  
    parameters = (<Param>)}  
informieren. Je heruntergeladenem FW-Paket MUSS <Param> mit folgenden Werten belegt sein:  
    <Param> = „ProductVendorID= \$UpdateInformation/ProductVendorID;  
    ProductCode= \$UpdateInformation/ProductCode;  
    ProductName=\$UpdateInformation/ProductName;  
    FirmwareVersion=\$UpdateInformation/Firmware/FWVersion;  
    Deadline=\$UpdateInformation/DeploymentInformation/Deadline;  
    FWPriority=\$UpdateInformation/Firmware/FWPriority;  
    FirmwareReleaseNotes  
    =\$UpdateInformation/Firmware/FirmwareReleaseNotes“
10. Sofern `MGM_KSR_AUTODOWNLOAD = Enabled`, SOLL der Konnektor bei Update-Paketen, die Kartenterminals betreffen, pro KT-Modell das Updatepaket mit der höchsten FirmwareVersion über `I_KSRS_Download::get_Updates` herunterladen, falls das Update-Paket nicht bereits von einem vorherigen Download auf dem Konnektor vorhanden ist.
11. Sofern `I_KSRS_Download::get_Updates` den http Status Code 503 Server Unavailable zurückgibt, MUSS der Konnektor die Informationen aus dem zurückgegebenen Retry-After Header nutzen, um den Zeitpunkt des Retry zu bestimmen.

Der Konnektor MUSS immer nur die neusten Update-Pakete für eine Nutzung vorhalten. Eventuell vorhandene ältere, nicht genutzte Update-Pakete KÖNNEN überschrieben werden.

Nach einem erfolgreichen Download DÜRFEN die Namen der Dateien eines Update-Paketes beim Abspeichern NICHT verändert werden. [≤]

#### **TIP1-A\_7220 - Konnektoraktualisierung File Transfer Ranges**

Der Konnektor KANN für den Download von Update-Paketen über `I_KSRS_Download::get_Updates` die Option Range Requests [RFC7233#3.1] zur Fortsetzung von unterbrochenen Transfers nutzen. [≤]

#### **TIP1-A\_4837 - Übersichtsseite des KSR-Client**

Die Administrationsoberfläche des KSR-Clients MUSS dem Administrator eine Übersichtsseite anbieten, die einen Geräteeintrag für den Konnektor selbst, sowie eine Liste von Geräteeinträgen für jedes Kartenterminal (CT) aus `CTM_CT_LIST` mit



7507 CT.IS\_PHYSICAL=Ja und CT.CORRELATION>=„gepairt“ enthält.  
 7508 Der Administrator MUSS die Liste der Kartenterminals nach Kartenterminalmodellen  
 7509 gruppieren können (gleiche Werte für ProductVendorID, ProductCode, HardwareVersion  
 7510 und FirmwareVersion).  
 7511 Je Geräteeintrag MÜSSEN die über „Automatische Prüfung und Download von Update-  
 7512 Paketen“ ermittelten listUpdatesResponse bereitstehen.  
 7513 Je Geräteeintrag MUSS die Version der aktuell installierten Software dargestellt werden.  
 7514 Sind Bestandteile der installierten Software unabhängig aktualisierbar, so MUSS für jedes  
 7515 der Bestandteile die Version angezeigt werden.  
 7516 Der Administrator MUSS eine Aktualisierung aller listUpdatesResponse über  
 7517 TUC\_KON\_282 „UpdateInformationen beziehen“ auslösen können.  
 7518 Geräteeinträge, die über listUpdatesResponse mit neuerer Firmwareversion als das  
 7519 zugehörige Gerät verfügen, MÜSSEN hervorgehoben werden.  
 7520 Je Geräteeintrag MUSS die Zugehörigkeit der installierten Software und der Software-  
 7521 Updates zum Online-Produktivbetrieb oder zu einer Erprobung (inklusive Name der  
 7522 Erprobung) dargestellt werden.

7523 [ $\leq$ ]

#### 7524 **TIP1-A\_4838 - Einsichtnahme in Update-Informationen**

7525 Für alle Geräteeinträge MUSS der Administrator zu den listUpdatesResponse sowohl die  
 7526 FirmwareGroupReleaseNotes als auch jedes enthaltene UpdateInformation-Element  
 7527 einsehen können. Dazu MUSS der Konnektor

- 7528 • alle Felder der Struktur verständlich umsetzen und strukturiert anzeigen (inkl. der
- 7529 Notes für jedes Firmwarefiles- und Documentationsfiles-Element)
- 7530 • jedes über das Documentationfiles-Element erreichbare Dokument auf
- 7531 Anforderung des Administrator herunterladen und anzeigen. Es MÜSSEN dabei
- 7532 mindestens die folgenden Dokumentenformate zur Anzeige gebracht werden
- 7533 können: Text, PDF, JPEG, TIFF

7534 [ $\leq$ ]

#### 7535 **TIP1-A\_4839-01 - Festlegung der durchzuführenden Updates**

7536 Der Administrator MUSS in der Übersichtsliste einzelne Geräteeinträge bzw. Gruppen mit  
 7537 der jeweils anzuwendenden UpdateInformation für die Durchführung eines Updates  
 7538 markieren können.

7539 Alternativ MUSS der Administrator neben der Markierung je Geräteeintrag bzw. Gruppe  
 7540 Update-Pakete lokal einspielen können (etwa durch ein Upload- bzw. Download-Interface  
 7541 in der Administrationsoberfläche).

7542 Je Geräteeintrag MUSS der Administrator einen individuellen Ausführungszeitpunkt für  
 7543 die Durchführung des Updates einstellen können.

7544 Der Administrator MUSS für den Geräteeintrag Konnektor festlegen können, ob dieses  
 7545 Update erst gestartet werden darf, wenn zuvor alle festgelegten KT-Updates erfolgreich  
 7546 durchlaufen wurden.

7547 Der Administrator MUSS zu jeder Zeit die gerätebezogene Festlegung für ein Update  
 7548 ändern bzw. löschen können, sofern dieses konkrete Update noch nicht begonnen wurde.

7549 Je Geräteeintrag MUSS der Administrator automatische Softwareupdates aktivieren und  
 7550 deaktivieren können.

7551 [ $\leq$ ]

#### 7552 **TIP1-A\_4840-01 - Manuelles Auslösen der durchzuführenden Updates**

7553 Der Administrator MUSS für die Liste der markierten Geräteeinträge ein gesammeltes  
 7554 Update auslösen können. Dieses MUSS nach folgendem Muster ablaufen:

- 7555 1. Alle Kartenterminaleinträge abarbeiten. Pro markiertem Kartenterminal:

- 7556 • Wenn Ausführungszeitpunkt nicht gesetzt:  
7557 Anwenden des definierten Updates mittels TUC\_KON\_281  
7558 „Kartenterminalaktualisierung anstoßen“
- 7559 • Wenn Ausführungszeitpunkt gesetzt:  
7560 Anwenden des definierten Updates mittels TUC\_KON\_281 sobald der  
7561 Ausführungszeitpunkt erreicht ist oder, sofern der Konnektor zum  
7562 Ausführungszeitpunkt nicht in Betrieb war, überschritten wurde. Konnte das  
7563 Kartenterminal nicht erreicht werden, so MUSS das gesetzte Update im KSR-Client  
7564 für eine spätere Anwendung erhalten bleiben (wird ereignisgesteuert neu  
7565 ausgelöst).
- 7566 2. Sofern die KonnektorUpdate-Abhängigkeit von KT-Updates nicht gesetzt wurde  
7567 oder wenn alle vorgesehenen Kartenterminal-Updates durchgeführt wurden,  
7568 MUSS das Konnektor-Updates mittels TUC\_KON\_280 „Konnektoraktualisierung  
7569 durchführen“ wie folgt begonnen werden:
- 7570 • wenn Ausführungszeitpunkt nicht gesetzt: TUC-Aufruf direkt
- 7571 • wenn Ausführungszeitpunkt gesetzt: TUC-Aufruf direkt sobald der  
7572 Ausführungszeitpunkt erreicht ist oder, sofern der Konnektor zum  
7573 Ausführungszeitpunkt nicht in Betrieb war, überschritten wurde
- 7574 Wenn der Administrator ein Erprobungs-Update zur Installation auswählt, MUSS er über  
7575 einen Warnhinweis darüber informiert werden,
- 7576 • dass es sich um ein Erprobungs-Update handelt,  
7577 • für welche Erprobung es vorgesehen ist,  
7578 • dass das Update-Paket nur installiert werden sollte, wenn die Institution oder  
7579 Organisation des Gesundheitswesens an der Erprobung teilnimmt,
- 7580 dass, falls die Institution oder Organisation des Gesundheitswesens nicht an der  
7581 Erprobung teilnimmt und dennoch das Update installiert wird, es zu funktionalen  
7582 Einschränkungen des Konnektors kommen kann. [ $\leq$ ]
- 7583 Wurde die ECC-Migration durchgeführt, so muss sichergestellt werden, dass der  
7584 Konnektor auch wieder in den ursprünglichen Zustand, d.h. den Zustand vor der ECC-  
7585 Migration (TI-Vertrauensanker für RSA und Firmware vor der ECC-Migration),  
7586 zurückgesetzt werden kann.
- 7587 **A\_18390 - Automatisches Auslösen der durchzuführenden Updates**  
7588 Wenn für mindestens ein Gerät das automatische Softwareupdate aktiviert ist, MUSS der  
7589 Konnektor zur MGM\_KSR\_AUTO\_UPDATE\_TIME die Updates nach folgendem Muster  
7590 durchführen:
- 7591 • Alle Geräte (Kartenterminals und Konnektor), für die  
7592 MGM\_KSR\_AUTO\_UPDATE=Enabled ist, werden markiert
- 7593 • Alle Kartenterminaleinträge abarbeiten
- 7594 • Pro markiertem Kartenterminal: Anwenden des automatischen Updates mittels  
7595 TUC\_KON\_281 „Kartenterminalaktualisierung anstoßen“
- 7596 • Sofern die Konnektorupdate-Abhängigkeit von KT-Updates nicht gesetzt wurde  
7597 oder wenn alle vorgesehenen Kartenterminal-Updates durchgeführt wurden,  
7598 MUSS für einen markierten Konnektor das Konnektor-Update mittels  
7599 TUC\_KON\_280 „Konnektoraktualisierung durchführen“ begonnen werden.
- 7600 [ $\leq$ ]



**7601 A\_18391 - Automatisches Updates nicht nachholen**

7602 Sofern der Konnektor zu MGM\_KSR\_AUTO\_UPDATE\_TIME nicht in Betrieb war, DÜRFEN  
7603 die automatischen Updates später NICHT nachgeholt werden. [ <= ]

**7604 A\_18779 - Hinweise in KSR Update Paket zu Auto-Update**

7605 Wenn mit einem Update erstmalig MGM\_KSR\_AUTO\_UPDATE=Enabled aktiv wird, MUSS  
7606 der Konnektorhersteller über das entsprechende KSR-Paket den Admin an der Konnektor  
7607 Oberfläche darauf hinweisen, dass mit diesem Update der automatische Softwareupdate  
7608 aktiv wird.  
7609 [ <= ]

**7610 A\_20531 - Größe der Bestandsnetze.xml**

7611 Der Konnektor MUSS eine Bestandsnetze.xml mit einer Größe von mindestens 3 MByte  
7612 und 2000 Netzen (XML Element <Network>) verarbeiten können. [ <= ]

**7613 4.3.10 Konnektorstatus****7614 TIP1-A\_5542 - Konnektor, Funktion zur Prüfung der Erreichbarkeit von  
7615 Systemen**

7616 Der Konnektor MUSS an der Managementschnittstelle eine Funktion anbieten, die es  
7617 ermöglicht die Erreichbarkeit von Systemen durch Eingabe der IP-Adresse oder des FQDN  
7618 zu prüfen. Das Ergebnis des Tests MUSS angezeigt werden.  
7619 [ <= ]

**7620 4.4 Hardware-Merkmale des Konnektors****7621 TIP1-A\_4841 - Hardware für Dauerbetrieb**

7622 Der Konnektor MUSS sowohl in seiner Stromversorgung als auch in seinen restlichen  
7623 Hardwarekomponenten auf einen 24x7-Dauerbetrieb ausgelegt sein.  
7624 Der Hersteller DARF NICHT davon ausgehen oder gar in seiner Guidance darauf  
7625 verweisen, dass der Konnektor mehrere Stunden am Tag nicht betrieben wird.  
7626 [ <= ]

7627 Diese Anforderung verlangt keinen Schutz gegen Stromausfall in den Betriebsräumen.

**7628 TIP1-A\_4842 - Gehäuseversiegelung**

7629 Jeder Konnektor, der als Appliance (dezidierte, geschlossene Kombination aus  
7630 spezifischer Hard- und Software) ausgeprägt ist, MUSS über eine fälschungssichere  
7631 Gehäuseversiegelung verfügen. Die Versiegelung MUSS so angebracht werden, dass eine  
7632 Öffnung des Gehäuses nicht ohne Beschädigung des Siegels erfolgen kann.  
7633 Der Konnektor MUSS die Umsetzung entsprechend der Festlegungen für das  
7634 Kartenterminal nach der TR-03120 [BSI TR-03120], Kapitel bzgl. Gehäuseversiegelung 9  
7635 vornehmen.  
7636 Die optische Gestaltung der Siegel ist herstellerspezifisch.  
7637 [ <= ]

7638 Die Prüfung auf Einhaltung der Versiegelungsvorgaben erfolgt nicht im Rahmen der CC-  
7639 Evaluierung, sondern im Zuge der Prüfung auf funktionale Eignung.

**7640 TIP1-A\_4843 - Zustandsanzeige**

7641 Im Betrieb MUSS der Zustand des Konnektors erkennbar sein. Zur Anzeige des  
7642 Betriebszustandes des Konnektors SOLL es eine Signaleinrichtung (z. B. über Status-  
7643 LEDs) am Konnektor geben. Falls keine Signalvorrichtung am Konnektorgehäuse  
7644 verwendet wird MUSS es eine softwareseitige Lösung über das Managementinterface  
7645 geben. Bei verbauter Hardware-Signalgebung KANN eine softwareseitige Lösung

7646 zusätzlich angeboten werden.

7647 Es MÜSSEN mindestens folgende angezeigt werden:

- 7648 • Power ON,
- 7649 • Link Status pro physischer Netzwerkschnittstelle
- 7650 • Fehler/Kritischer Betriebszustand gemäß Kapitel 3.3

7651 Es SOLLEN folgende Zustände angezeigt werden:

- 7652 • Status pro IPsec-Verbindung

7653 [ $\leq$ ]

#### 7654 **TIP1-A\_4844-02 - Ethernet-Schnittstellen**

7655 Der Konnektor MUSS mindestens zwei Ethernetinterfaces nach [IEEE802.3] als  
7656 physikalische Schnittstellen zur Verfügung stellen.

7657 [ $\leq$ ]

7658

#### 7659 **TIP1-A\_4845 - Verwendungsumgebung - Klima**

7660 Als normaler Einsatzort wird für den Konnektor ein Büroraum angenommen. Der  
7661 Konnektor MUSS die in Tabelle TAB\_KON\_671 aufgeführten Anforderungen erfüllen,  
7662 welche unter der Annahme des normalen Einsatzortes erhoben werden.

7663

#### 7664 **Tabelle 381: TAB\_KON\_671 Anforderungen Klima**

| Prüfung Klima   |
|---|
| Trockene Wärme (Dry Heat) nach DIN EN 60068-2-2 Methode Bb wird für die Bedingungen als obere Lagertemperatur von 55°C und einer Beanspruchungsdauer von 16 h geprüft und die Funktionsfähigkeit MUSS bestätigt werden.                                 |
| Kälte (Cold) nach DIN EN 60068-2-1 Methode Ab wird für die Bedingungen als untere Lagertemperatur von -10°C und einer Beanspruchungsdauer von 16 h geprüft und die Funktionsfähigkeit MUSS bestätigt werden.  |
| Nach den beiden oben genannten Belastungen durch extreme Lagertemperaturen und der Nachbehandlungsdauer von 1 h MUSS die Funktionsfähigkeit des Konnektors gewährleistet sein, was durch Funktionsprüfungen nachzuweisen ist.                           |
| Die Funktionsfähigkeit im Betrieb MUSS bei einer oberen Temperatur von 40°C über eine Dauer von 24 h gewährleistet sein. Dies wird für den Konnektor durch Prüfung nach DIN EN 60068-2-2 Methode Bb bei gleichzeitigen Funktionsprüfungen nachgewiesen. |

7665

7666 [ $\leq$ ]

#### 7667 **TIP1-A\_4846 - Verwendungsumgebung – Vibration**

7668 Die durch Vibrationen und mechanische Schockbelastungen auftretenden Belastungen  
7669 MÜSSEN vom Konnektor schadensfrei gemäß IEC 68-2 Methode nach den Anforderungen  
7670 aus TAB\_KON\_672 absolviert, geprüft und nachgewiesen werden.

7671

#### 7672 **Tabelle 381: TAB\_KON\_672 Anforderungen Vibration** **Tabelle 382: TAB\_KON\_672** 7673 **Anforderungen Vibration**

| Prüfung Vibration |
|-------------------|
|-------------------|

Sinusförmige Schwingungstests (Vibration, sinusoidal) nach DIN EN 60068-2-6 Methode Fc in drei senkrecht stehenden Achsen in einem Frequenzbereich von 2 Hz bis 200 Hz üblicherweise 1 h je Achse MÜSSEN erfolgreich nachgewiesen werden. Bis zu einer Frequenz von 9 Hz wird dabei mit einer konstanten Amplitude von 1,5 mm belastet, darüber bis zur oberen Frequenz wird mit einer konstanten Beschleunigung von  $5 \text{ m/s}^2$  (0,5 g) belastet.

Es MÜSSEN mechanische Schockprüfungen (Shock) nach DIN EN 60068-2-27 Methode Ea in drei senkrecht stehenden Achsen (sechs Richtungen) erfolgreich nachgewiesen werden. Es wird dabei für jede Achse mit 3 positiven und 3 negativen Schocks mit  $150 \text{ m/s}^2$  (15 g) Amplitude und einer Dauer von 11 ms belastet.

Dauerschocktests (Bump) nach DIN EN 60068-2-29 Methode Eb in drei senkrecht stehenden Achsen mit halbsinusförmigen Schocks MÜSSEN erfolgreich nachgewiesen werden. Es wird dabei für jede Achse mit 1000 positiven und 1000 negativen Schocks mit  $100 \text{ m/s}^2$  (10 g) Amplitude und einer Dauer von 16 ms belastet.

[<=>]

#### Prüfung Vibration

Sinusförmige Schwingungstests (Vibration, sinusoidal) nach DIN EN 60068-2-6 Methode Fc in drei senkrecht stehenden Achsen in einem Frequenzbereich von 2 Hz bis 200 Hz üblicherweise 1 h je Achse MÜSSEN erfolgreich nachgewiesen werden. Bis zu einer Frequenz von 9 Hz wird dabei mit einer konstanten Amplitude von 1,5 mm belastet, darüber bis zur oberen Frequenz wird mit einer konstanten Beschleunigung von  $5 \text{ m/s}^2$  (0,5 g) belastet.

Es MÜSSEN mechanische Schockprüfungen (Shock) nach DIN EN 60068-2-27 Methode Ea in drei senkrecht stehenden Achsen (sechs Richtungen) erfolgreich nachgewiesen werden. Es wird dabei für jede Achse mit 3 positiven und 3 negativen Schocks mit  $150 \text{ m/s}^2$  (15 g) Amplitude und einer Dauer von 11 ms belastet.

Dauerschocktests (Bump) nach DIN EN 60068-2-29 Methode Eb in drei senkrecht stehenden Achsen mit halbsinusförmigen Schocks MÜSSEN erfolgreich nachgewiesen werden. Es wird dabei für jede Achse mit 1000 positiven und 1000 negativen Schocks mit  $100 \text{ m/s}^2$  (10 g) Amplitude und einer Dauer von 16 ms belastet.

[<=]

#### TIP1-A\_4846-02 - ab PTV4: Verwendungsumgebung – Vibration

Die durch Vibrationen und mechanische Schockbelastungen auftretenden Belastungen MÜSSEN vom Konnektor schadensfrei gemäß IEC 68-2 Methode nach den Anforderungen aus TAB\_KON\_672 absolviert, geprüft und nachgewiesen werden.

**Tabelle 383: TAB\_KON\_672 Anforderungen Vibration**

#### Prüfung Vibration

Sinusförmige Schwingungstests (Vibration, sinusoidal) nach DIN EN 60068-2-6 Methode Fc in drei senkrecht stehenden Achsen in einem Frequenzbereich von 2 Hz bis 200 Hz üblicherweise 1 h je Achse MÜSSEN erfolgreich nachgewiesen werden. Bis zu einer Frequenz von 9 Hz wird dabei mit einer konstanten Amplitude von 1,5 mm belastet, darüber bis zur oberen Frequenz wird mit einer konstanten Beschleunigung von  $5 \text{ m/s}^2$  (0,5 g) belastet.

Es MÜSSEN mechanische Schockprüfungen (Shock) nach DIN EN 60068-2-27 Methode Ea in drei senkrecht stehenden Achsen (sechs Richtungen) erfolgreich nachgewiesen werden. Es wird dabei für jede Achse mit 3 positiven und 3 negativen Schocks mit  $150 \text{ m/s}^2$  (15 g) Amplitude und einer Dauer von 11 ms belastet.

#### **TIP1 A\_4846-02 ab PTV4: Verwendungsumgebung—Vibration**

Die durch Vibrationen und mechanische Schockbelastungen auftretenden Belastungen MÜSSEN vom Konnektor schadensfrei gemäß IEC 68-2 Methode nach den Anforderungen aus TAB\_KON\_672 absolviert, geprüft und nachgewiesen werden.

#### **Tabelle 382: TAB\_KON\_672 Anforderungen Vibration**

##### **Prüfung Vibration**

Sinusförmige Schwingungstests (Vibration, sinusoidal) nach DIN EN 60068-2-6 Methode Fc in drei senkrecht stehenden Achsen in einem Frequenzbereich von 2 Hz bis 200 Hz üblicherweise 1 h je Achse MÜSSEN erfolgreich nachgewiesen werden. Bis zu einer Frequenz von 9 Hz wird dabei mit einer konstanten Amplitude von 1,5 mm belastet, darüber bis zur oberen Frequenz wird mit einer konstanten Beschleunigung von  $5 \text{ m/s}^2$  (0,5 g) belastet.

Es MÜSSEN mechanische Schockprüfungen (Shock) nach DIN EN 60068-2-27 Methode Ea in drei senkrecht stehenden Achsen (sechs Richtungen) erfolgreich nachgewiesen werden. Es wird dabei für jede Achse mit 3 positiven und 3 negativen Schocks mit  $150 \text{ m/s}^2$  (15 g) Amplitude und einer Dauer von 11 ms belastet.

Dauerschocktests (Bump) nach DIN EN 60068-2-27 Methode Eb in drei senkrecht stehenden Achsen mit halbsinusförmigen Schocks MÜSSEN erfolgreich nachgewiesen werden. Es wird dabei für jede Achse mit 1000 positiven und 1000 negativen Schocks mit  $100 \text{ m/s}^2$  (10 g) Amplitude und einer Dauer von 16 ms belastet.

[<=]

7693

## 5 Anhang A – Verzeichnisse

7694

### 5.1 Abkürzungen

| Kürzel  | Erläuterung                                  |
|---------|--|
| AMTS    | Arzneimitteltherapiesicherheit               |
| APPL DO | Application Label Data Object                |
| CC      | Common Criteria                              |
| DHCP    | Dynamic Host Configuration Protocol          |
| DNS     | Domain Name Service                          |
| DO      | Datenobjekt                                  |
| DSL     | Digital Subscriber Line                      |
| ECC     | Elliptic Curve Cryptography                  |
| EVG     | Evaluiierungsgegenstand                      |
| gSMC-K  | Security Module Card Typ K (Konnektor)       |
| gSMC-KT | Security Module Card Typ KT (Kartenterminal) |
| HBA     | Heilberufsausweis                            |
| HSM-B   | Hardware Security Module Typ B               |
| IAG     | Internet Access Gateway                      |
| ID      | Identifizier                                 |
| ISP     | Internet Service Provider                    |
| KT      | Kartenterminal                               |
| KVK     | Krankenversichertenkarte                     |
| LAN     | Local Area Network                           |
| MTOM    | Message Transmission Optimization Mechanism  |

|             |   |
|-------------|---|
| NFDM        | Notfalldatenmanagement                                    |
| NK          | Netzkonnektor   |
| NTP         | Network Time Protokoll                                    |
| OCSP        | Online Certificate Status Protocol                        |
| OID         | Object Identifier   |
| PIN         | Personal  |
| PKI         | Public Key Infrastructure                                 |
| PP          | Protection Profile  |
| PU          | Produktivumgebung   |
| PUK         | Personal Unblocking Key                                   |
| QES         | Qualifizierte elektronische Signatur                      |
| RU          | Referenzumgebung  |
| SIS         | Secure Internet Service                                   |
| SMC-B       | Security Module Card Typ B                                |
| SMTBD<br>DO | SICCT Message-To-Be-Displayed Data Object                 |
| SOAP        | Standard für die Kommunikation innerhalb der WEB-Services |
| TI          | Telematikinfrastruktur                                    |
| TLS         | Transport Layer Security                                  |
| TSF         | TOE Security Functionality                                |
| TU          | Testumgebung  |
| TUC         | Technischer Use Case                                      |
| URL         | Uniform Resource Locator                                  |
| VPN         | Virtual Private Network                                   |

|         |                                  |
|---------|----------------------------------|
| VSDM    | Versichertenstammdatenmanagement |
| VZD     | Verzeichnisdienst                |
| WAN     | Wide Area Network                |
| XML     | Extensible Markup Language       |
| ZD      | Zertifizierungsdienst            |
| ZOD 2.0 | Zahnärzte Online Deutschland 2.0 |

## 7695 5.2 Glossar

| Begriff          | Erläuterung   |
|------------------|---|
| Funktionsmerkmal | Der Begriff beschreibt eine Funktion oder auch einzelne, eine logische Einheit bildende Teilfunktionen der TI im Rahmen der funktionalen Zerlegung des Systems. |

7696 Das Glossar wird als eigenständiges Dokument, vgl. [gemGlossar] zur Verfügung gestellt.

## 7697 5.3 Abbildungsverzeichnis

|      |  |     |
|------|--|-----|
| 7698 | Abbildung 1: PIC_KON_116 Schnittstellen des Konnektors von und zu anderen        |     |
| 7699 | Produkttypen.....  | 31  |
| 7700 | Abbildung 2: PIC_KON_117 Logische Zerlegung des Konnektors in Anwendungs- und    |     |
| 7701 | Netzkonnektor.....   | 33  |
| 7702 | Abbildung 3: PIC_KON_107 XML Struktur des Status-Elements einer SOAP-Antwort.... | 67  |
| 7703 | Abbildung 4: PIC_Kon_100 Informationsmodell des Konnektors.....                  | 74  |
| 7704 | Abbildung 5: PIC_KON_101 Aufrufkontext der Operation.....                        | 84  |
| 7705 | Abbildung 6: PIC_KON_118 Aktivitätsdiagramm zu „TUC_KON_000 Prüfe                |     |
| 7706 | Zugriffsberechtigung“.....   | 88  |
| 7707 | Abbildung 7: PIC_KON_071 Korrelationszustände eines eHealth-KT.....              | 110 |
| 7708 | Abbildung 8: PIC_KON_110 Aktivitätsdiagramm zu „Beginne Kartenterminalsitzung“.. | 119 |
| 7709 | Abbildung 9: PIC_KON_057 Aktivitätsdiagramm zu „PaireKartenterminal“.....        | 126 |
| 7710 | Abbildung 10: PIC_KON_111 Aktivitätsdiagramm zu „PIN-verifizieren“.....          | 165 |
| 7711 | Abbildung 11: PIC_KON_022 Grundsätzlicher Aufbau der Ereignisnachricht.....      | 238 |
| 7712 | Abbildung 12: PIC_KON_112 Aktivitätsdiagramm zu „Systemereignis absetzen“.....   | 245 |
| 7713 | Abbildung 13: PIC_KON_058 Aktivitätsdiagramm „Daten hybrid verschlüsseln“.....   | 283 |
| 7714 | Abbildung 14: PIC_KON_059 Aktivitätsdiagramm „Daten hybrid entschlüsseln“.....   | 289 |
| 7715 | Abbildung 15: PIC_KON_103 Use Case Diagramm Signaturdienst (nonQES).....         | 314 |



|      |  |     |
|------|--|-----|
| 7716 | Abbildung 16: PIC_KON_104 Use Case Diagramm Signaturdienst (QES) .....               | 315 |
| 7717 | Abbildung 17: PIC_KON_102 Use Case Diagramm Signaturdienst (Komfortsignatur) ..      | 315 |
| 7718 | Abbildung 18: PIC_KON_113 Aktivitätsdiagramm zu „QES-Signaturen erstellen“ .....     | 326 |
| 7719 | Abbildung 19: PIC_KON_114 Aktivitätsdiagramm zu „Dokument QES signieren“ .....       | 349 |
| 7720 | Abbildung 20: PIC_KON_118 Aufbau und Struktur der Protokolldateien für Plattform und |     |
| 7721 | Fachmodule .....   | 448 |
| 7722 | Abbildung 21: PIC_KON_115 Kommunikationsregeln Konnektor .....                       | 471 |
| 7723 | Abbildung 22: PIC_KON_105 Aktivitätsdiagramm Konnektoraktualisierung durchführen     |     |
| 7724 | .....  | 548 |
| 7725 | Abbildung 23: PIC_KON_120 Abbildung von CardSessions auf logische Kanäle .....       | 660 |
| 7726 | Abbildung 24: PIC_KON_007 Übersicht Zeichensatz ISO646DE/DIN66003 .....              | 662 |
| 7727 | Abbildung 25: Szenario einer einfachen Installation .....                            | 664 |
| 7728 | Abbildung 26: Szenario einer Installation mit mehreren Behandlungsräumen .....       | 666 |
| 7729 | Abbildung 27: Szenario einer Integration der TI Produkte in eine bestehende          |     |
| 7730 | Infrastruktur .....  | 667 |
| 7731 | Abbildung 28: Szenario einer Integration der TI Produkte in eine bestehende          |     |
| 7732 | Infrastruktur mit existierendem Router .....   | 669 |
| 7733 | Abbildung 29: Szenario mit zentral gesteckten HBA und SMC-B .....                    | 670 |
| 7734 | Abbildung 30: Szenario mit zentralem Primärsystem als Clientsystem .....             | 672 |
| 7735 | Abbildung 31: Szenario für den Zugriff .....   | 674 |
| 7736 | Abbildung 32: Standalone-Szenario mit physischer Trennung im Konnektor .....         | 675 |
| 7737 | Abbildung 1: PIC_KON_116 Schnittstellen des Konnektors von und zu anderen            |     |
| 7738 | Produkttypen .....   | 31  |
| 7739 | Abbildung 2: PIC_KON_117 Logische Zerlegung des Konnektors in Anwendungs- und        |     |
| 7740 | Netzkonnektor .....  | 33  |
| 7741 | Abbildung 3: PIC_KON_107 XML-Struktur des Status-Elements einer SOAP-Antwort ....    | 67  |
| 7742 | Abbildung 4: PIC_Kon_100 Informationsmodell des Konnektors .....                     | 74  |
| 7743 | Abbildung 5: PIC_KON_101 Aufrufkontext der Operation .....                           | 84  |
| 7744 | Abbildung 6: PIC_KON_118 Aktivitätsdiagramm zu „TUC_KON_000 Prüfe                    |     |
| 7745 | Zugriffsberechtigung“ .....  | 88  |
| 7746 | Abbildung 7: PIC_KON_071 Korrelationszustände eines eHealth-KT .....                 | 110 |
| 7747 | Abbildung 8: PIC_KON_110 Aktivitätsdiagramm zu „Beginne Kartenterminalsitzung“ .     | 119 |
| 7748 | Abbildung 9: PIC_KON_057 Aktivitätsdiagramm zu „Paire Kartenterminal“ .....          | 126 |
| 7749 | Abbildung 10: PIC_KON_111 Aktivitätsdiagramm zu „PIN verifizieren“ .....             | 165 |
| 7750 | Abbildung 11: PIC_KON_022 Grundsätzlicher Aufbau der Ereignisnachricht .....         | 238 |
| 7751 | Abbildung 12: PIC_KON_112 Aktivitätsdiagramm zu „Systemereignis absetzen“ .....      | 245 |
| 7752 | Abbildung 13: PIC_KON_058 Aktivitätsdiagramm „Daten hybrid verschlüsseln“ .....      | 283 |
| 7753 | Abbildung 14: PIC_KON_059 Aktivitätsdiagramm „Daten hybrid entschlüsseln“ .....      | 289 |
| 7754 | Abbildung 15: PIC_KON_103 Use Case Diagramm Signaturdienst (nonQES) .....            | 314 |



|      |  |     |
|------|--|-----|
| 7755 | Abbildung 16: PIC_KON_104 Use Case Diagramm Signaturdienst (QES) .....               | 315 |
| 7756 | Abbildung 17: PIC_KON_102 Use Case Diagramm Signaturdienst (Komfortsignatur) ..      | 315 |
| 7757 | Abbildung 18: PIC_KON_113 Aktivitätsdiagramm zu „QES Signaturen erstellen“ .....     | 326 |
| 7758 | Abbildung 19: PIC_KON_114 Aktivitätsdiagramm zu „Dokument QES signieren“ .....       | 349 |
| 7759 | Abbildung 20: PIC_KON_118 Aufbau und Struktur der Protokolldateien für Plattform und |     |
| 7760 | Fachmodule .....   | 448 |
| 7761 | Abbildung 21: PIC_KON_115 Kommunikationsregeln Konnektor .....                       | 471 |
| 7762 | Abbildung 22: PIC_KON_105 Aktivitätsdiagramm Konnektoraktualisierung durchführen     |     |
| 7763 | .....  | 548 |
| 7764 | Abbildung 23: PIC_KON_120 Abbildung von CardSessions auf logische Kanäle .....       | 660 |
| 7765 | Abbildung 24: PIC_KON_007 Übersicht Zeichensatz ISO646DE/DIN66003 .....              | 662 |
| 7766 | Abbildung 25: Szenario einer einfachen Installation .....                            | 664 |
| 7767 | Abbildung 26: Szenario einer Installation mit mehreren Behandlungsräumen .....       | 666 |
| 7768 | Abbildung 27: Szenario einer Integration der TI Produkte in eine bestehende          |     |
| 7769 | Infrastruktur .....  | 667 |
| 7770 | Abbildung 28: Szenario einer Integration der TI Produkte in eine bestehende          |     |
| 7771 | Infrastruktur mit existierendem Router .....   | 669 |
| 7772 | Abbildung 29: Szenario mit zentral gesteckten HBA und SMC-B .....                    | 670 |
| 7773 | Abbildung 30: Szenario mit zentralem Primärsystem als Clientsystem .....             | 672 |
| 7774 | Abbildung 31: Szenario für den Zugriff .....   | 674 |
| 7775 | Abbildung 32: Standalone-Szenario mit physischer Trennung im Konnektor .....         | 675 |
| 7776 |  |     |

## 7777 5.4 Tabellenverzeichnis

|      |   |    |
|------|---|----|
| 7778 | Tabelle 1: TAB_KON_500 Wertetabelle Kartentypen .....                                 | 39 |
| 7779 | Tabelle 2: TAB_KON_856: Identitäten des Konnektors auf der gSMC-K .....               | 41 |
| 7780 | Tabelle 3: TAB_KON_503 Betriebszustand_Fehlerzustandsliste .....                      | 46 |
| 7781 | Tabelle 4: TAB_KON_504 Ausführungserlaubnis für Dienste in kritischen Fehlerzuständen |    |
| 7782 | .....   | 52 |
| 7783 | Tabelle 5: TAB_KON_502 Fehlercodes „Betriebszustand“ .....                            | 58 |
| 7784 | Tabelle 6: TAB_KON_505 Konfigurationswerte Missbrauchserkennung .....                 | 58 |
| 7785 | Tabelle 7: TAB_KON_852 Konfigurationsvarianten der Verbindungen zwischen Konnektor    |    |
| 7786 | und Clientsystemen .....  | 61 |
| 7787 | Tabelle 8: TAB_KON_506 Konfigurationsparameter der Clientsystem Authentisierung ..    | 64 |
| 7788 | Tabelle 9: TAB_KON_812 Umgebungsabhängige Konfigurationsparameter .....               | 70 |
| 7789 | Tabelle 10: TAB_KON_507 Informationsmodell Entitäten .....                            | 74 |
| 7790 | Tabelle 11: TAB_KON_508 Informationsmodell Attribute .....                            | 78 |
| 7791 | Tabelle 12: TAB_KON_509 Informationsmodell Entitätenbeziehungen .....                 | 79 |

|      |   |     |
|------|---|-----|
| 7792 | Tabelle 13: TAB_KON_510 Informationsmodell Constraints .....                      | 81  |
| 7793 | Tabelle 14: TAB_KON_511 — TUC_KON_000 „Prüfe Zugriffsberechtigung“ .....          | 84  |
| 7794 | Tabelle 15: TAB_KON_512 Zugriffsregeln-Beschreibung .....                         | 87  |
| 7795 | Tabelle 16: TAB_KON_513 Zugriffsregeln-Regelzuordnung .....                       | 89  |
| 7796 | Tabelle 17: TAB_KON_514-01 Zugriffsregeln-Definition .....                        | 90  |
| 7797 | Tabelle 18: TAB_KON_515 Fehlercodes TUC_KON_000 „Prüfe Zugriffsberechtigung“ .... | 94  |
| 7798 | Tabelle 19: TAB_KON_143 — TUC_KON_080 „Dokument validieren“ .....                 | 96  |
| 7799 | Tabelle 20: TAB_KON_144 Fehlercodes TUC_KON_080 „Dokument validieren“ .....       | 98  |
| 7800 | Tabelle 21: TAB_KON_516 Basisanwendung Dienstverzeichnisdienst .....              | 100 |
| 7801 | Tabelle 22: TAB_KON_517 Schemabeschreibung Produktinformation                     |     |
| 7802 | (ProductInformation.xsd) .....  | 101 |
| 7803 | Tabelle 23: TAB_KON_518 Schemabeschreibung Serviceinformation                     |     |
| 7804 | (Serviceinformation.xsd) .....  | 102 |
| 7805 | Tabelle 24: TAB_KON_519 — TUC_KON_041 „Einbringen der Endpunktinformationen       |     |
| 7806 | während der Bootup-Phase“ .....   | 103 |
| 7807 | Tabelle 25: TAB_KON_520 Fehlercodes TUC_KON_041 „Einbringen der                   |     |
| 7808 | Endpunktinformationen während der Bootup-Phase“ .....                             | 104 |
| 7809 | Tabelle 26: TAB_KON_521 Schnittstelle der Basisanwendung Dienstverzeichnisdienst  | 104 |
| 7810 | Tabelle 27: TAB_KON_522 Parameterübersicht des Kartenterminaldienstes .....       | 106 |
| 7811 | Tabelle 28: TAB_KON_785 Erlaubte SICCT-Kommandos bei CT.CONNECTED=Nein ....       | 111 |
| 7812 | Tabelle 29: TAB_KON_727 Terminalanzeigen beim Anfordern und Auswerfen von Karten  |     |
| 7813 | .....   | 112 |
| 7814 | Tabelle 30: TAB_KON_039 — TUC_KON_050 „Beginne Kartenterminalsitzung“ .....       | 114 |
| 7815 | Tabelle 31: TAB_KON_523 Fehlercodes TUC_KON_050 „Beginne Kartenterminalsitzung“   |     |
| 7816 | .....   | 120 |
| 7817 | Tabelle 32: TAB_KON_524 — TUC_KON_054 „Kartenterminal hinzufügen“ .....           | 120 |
| 7818 | Tabelle 33: TAB_KON_525 Fehlercodes TUC_KON_054 „Kartenterminal hinzufügen“ ..    | 122 |
| 7819 | Tabelle 34: TAB_KON_041 — TUC_KON_053 „Paare Kartenterminal“ .....                | 122 |
| 7820 | Tabelle 35: TAB_KON_113 Fehlercodes TUC_KON_053 „Paare Kartenterminal“ .....      | 125 |
| 7821 | Tabelle 36: TAB_KON_526 — TUC_KON_055 „Befülle CT-Object“ .....                   | 127 |
| 7822 | Tabelle 37: TAB_KON_112 — TUC_KON_051 „Mit Anwender über Kartenterminal           |     |
| 7823 | interagieren“ .....   | 128 |
| 7824 | Tabelle 38: TAB_KON_114 Fehlercodes TUC_KON_051 „Mit Anwender über                |     |
| 7825 | Kartenterminal interagieren“ .....  | 130 |
| 7826 | Tabelle 39: TAB_KON_723 — TUC_KON_056 „Karte anfordern“ .....                     | 131 |
| 7827 | Tabelle 40: TAB_KON_724 Fehlercodes TUC_KON_056 „Karte anfordern“ .....           | 133 |
| 7828 | Tabelle 41: TAB_KON_725 — TUC_KON_057 „Karte auswerfen“ .....                     | 133 |
| 7829 | Tabelle 42: TAB_KON_796 Fehlercodes TUC_KON_057 „Karte auswerfen“ .....           | 135 |
| 7830 | Tabelle 43: TAB_KON_854 — TUC_KON_058 „Displaygröße ermitteln“ .....              | 136 |

|      |   |     |
|------|---|-----|
| 7831 | Tabelle 44: TAB_KON_855 Fehlercodes TUC_KON_058 „Displaygröße ermitteln“ .....      | 137 |
| 7832 | Tabelle 45: TAB_KON_722 Basisdienst Kartenterminaldienst .....                      | 137 |
| 7833 | Tabelle 46: TAB_KON_716 Operation RequestCard .....                                 | 137 |
| 7834 | Tabelle 47: TAB_KON_717 Ablauf RequestCard .....                                    | 139 |
| 7835 | Tabelle 48: TAB_KON_718 Fehlercodes „RequestCard“ .....                             | 139 |
| 7836 | Tabelle 49: TAB_KON_719 Operation EjectCard .....                                   | 140 |
| 7837 | Tabelle 50: TAB_KON_720 Ablauf EjectCard .....                                      | 141 |
| 7838 | Tabelle 51: TAB_KON_721 Fehlercodes Operation „EjectCard“ .....                     | 142 |
| 7839 | Tabelle 52: TAB_KON_527 Konfigurationswerte eines Kartenterminalobjekts .....       | 142 |
| 7840 | Tabelle 53: TAB_KON_528 Informationsparameter des Kartenterminaldienstes .....      | 143 |
| 7841 | Tabelle 54: TAB_KON_529 Anzeigewerte zu einem Kartenterminalobjekt .....            | 144 |
| 7842 | Tabelle 55: TAB_KON_530 Konfigurationswerte eines Kartenterminalobjekts .....       | 146 |
| 7843 | Tabelle 56: TAB_KON_531 Parameterübersicht des Kartendienstes .....                 | 149 |
| 7844 | Tabelle 57: TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal    |     |
| 7845 | .....   | 152 |
| 7846 | Tabelle 58: TAB_KON_734 — TUC_KON_001 „Karte öffnen“ .....                          | 157 |
| 7847 | Tabelle 59: TAB_KON_735 — TUC_KON_026 .....   | 160 |
| 7848 | Tabelle 60: TAB_KON_824 Fehlercodes TUC_KON_026 „Liefere CardSession“ .....         | 161 |
| 7849 | Tabelle 61: TAB_KON_087 — TUC_KON_012 „PIN verifizieren“ .....                      | 161 |
| 7850 | Tabelle 62: TAB_KON_089 Fehlercodes TUC_KON_012 „PIN verifizieren“ .....            | 165 |
| 7851 | Tabelle 63: TAB_KON_736 — TUC_KON_019 „PIN ändern“ .....                            | 166 |
| 7852 | Tabelle 64: TAB_KON_093 Fehlercodes TUC_KON_019 „PIN ändern“ .....                  | 169 |
| 7853 | Tabelle 65: TAB_KON_236 — TUC_KON_021 „PIN entsperren“ .....                        | 170 |
| 7854 | Tabelle 66: TAB_KON_193 Fehlercodes TUC_KON_021 „PIN entsperren“ .....              | 173 |
| 7855 | Tabelle 67: TAB_KON_532 — TUC_KON_022 „Liefere PIN-Status“ .....                    | 174 |
| 7856 | Tabelle 68: TAB_KON_091 Fehlercodes TUC_KON_022 „Liefere PIN-Status“ .....          | 176 |
| 7857 | Tabelle 69: TAB_KON_240 — TUC_KON_027 „PIN-Schutz ein-/ausschalten“ .....           | 176 |
| 7858 | Tabelle 70: TAB_KON_838 Mapping von pinRef auf ANW .....                            | 179 |
| 7859 | Tabelle 71: TAB_KON_241 Fehlercodes TUC_KON_027 „PIN-Schutz ein-/ausschalten“ ..... | 179 |
| 7860 | Tabelle 72: TAB_KON_533 — TUC_KON_023 „Karte reservieren“ .....                     | 180 |
| 7861 | Tabelle 73: TAB_KON_534 Fehlercodes TUC_KON_023 „Karte reservieren“ .....           | 181 |
| 7862 | Tabelle 74: TAB_KON_096 — TUC_KON_005 „Card-to-Card authentisieren“ .....           | 182 |
| 7863 | Tabelle 75: TAB_KON_673 AuthMode für C2C .....                                      | 184 |
| 7864 | Tabelle 76: TAB_KON_674 Erlaubte Parameterkombinationen und resultierende CV-       |     |
| 7865 | Zertifikate für C2C .....   | 185 |
| 7866 | Tabelle 77: TAB_KON_535 Fehlercodes TUC_KON_005 „Card-to-Card authentisieren“ ..... | 185 |
| 7867 | Tabelle 78: TAB_KON_218 — TUC_KON_202 „LeseDatei“ .....                             | 186 |

|      |  |     |
|------|--|-----|
| 7868 | Tabelle 79: TAB_KON_536 Fehlercodes TUC_KON_202 „LeseDatei“ .....              | 187 |
| 7869 | Tabelle 80: TAB_KON_219 — TUC_KON_203 „SchreibeDatei“ .....                    | 188 |
| 7870 | Tabelle 81: TAB_KON_537 Fehlercodes TUC_KON_203 „Schreibe Datei“ .....         | 189 |
| 7871 | Tabelle 82: TAB_KON_204 — TUC_KON_204 „LöscheDateiInhalt“ .....                | 190 |
| 7872 | Tabelle 83: TAB_KON_785 Fehlercodes TUC_KON_204 „LöscheDateiInhalt“ .....      | 191 |
| 7873 | Tabelle 84: TAB_KON_538 — TUC_KON_209 „LeseRecord“ .....                       | 192 |
| 7874 | Tabelle 85: TAB_KON_539 Fehlercodes TUC_KON_209 „LeseRecord“ .....             | 193 |
| 7875 | Tabelle 86: TAB_KON_224 — TUC_KON_210 „SchreibeRecord“ .....                   | 194 |
| 7876 | Tabelle 87: TAB_KON_540 Fehlercodes TUC_KON_210 „SchreibeRecord“ .....         | 195 |
| 7877 | Tabelle 88: TAB_KON_211 — TUC_KON_211 „LöscheRecordInhalt“ .....               | 196 |
| 7878 | Tabelle 89: TAB_KON_786 Fehlercodes TUC_KON_211 „LöscheRecordInhalt“ .....     | 197 |
| 7879 | Tabelle 90: TAB_KON_228 — TUC_KON_214 „FügeHinzuRecord“ .....                  | 198 |
| 7880 | Tabelle 91: TAB_KON_541 Fehlercodes TUC_KON_214 „FügeHinzuRecord“ .....        | 199 |
| 7881 | Tabelle 92: TAB_KON_229 — TUC_KON_215 „SucheRecord“ .....                      | 200 |
| 7882 | Tabelle 93: TAB_KON_542 Fehlercodes TUC_KON_215 „SucheRecord“ .....            | 201 |
| 7883 | Tabelle 94: TAB_KON_110 — TUC_KON_018 „eGK Sperrung prüfen“ .....              | 202 |
| 7884 | Tabelle 95: TAB_KON_239 Fehlercodes TUC_KON_018 „eGK Sperrung prüfen“ .....    | 203 |
| 7885 | Tabelle 96: TAB_KON_108 — TUC_KON_006 „Datenzugriffsaudit eGK schreiben“ ..... | 204 |
| 7886 | Tabelle 97: TAB_KON_238 Fehlercodes TUC_KON_006 „Datenzugriffsaudit eGK        |     |
| 7887 | schreiben“ .....   | 205 |
| 7888 | Tabelle 98: TAB_KON_231 — TUC_KON_218 „Signiere“ .....                         | 205 |
| 7889 | Tabelle 99: TAB_KON_543 Fehlercodes TUC_KON_218 „Signiere“ .....               | 207 |
| 7890 | Tabelle 100: TAB_KON_232 — TUC_KON_219 „Entschlüssele“ .....                   | 207 |
| 7891 | Tabelle 101: TAB_KON_210 Fehlercodes TUC_KON_219 „Entschlüssele“ .....         | 208 |
| 7892 | Tabelle 102: TAB_KON_215 TUC_KON_200 „SendeAPDU“ .....                         | 209 |
| 7893 | Tabelle 103: TAB_KON_216 Fehlercodes TUC_KON_200 „SendeAPDU“ .....             | 210 |
| 7894 | Tabelle 104: TAB_KON_737 — TUC_KON_024 „Karte zurücksetzen“ .....              | 210 |
| 7895 | Tabelle 105: TAB_KON_544 Fehlercodes TUC_KON_024 „Karte zurücksetzen“ .....    | 211 |
| 7896 | Tabelle 106: TAB_KON_230 — TUC_KON_216 „LeseZertifikat“ .....                  | 212 |
| 7897 | Tabelle 107: TAB_KON_209 Fehlercodes TUC_KON_216 „LeseZertifikat“ .....        | 213 |
| 7898 | Tabelle 108: TAB_KON_827 TUC_KON_036 „LiefereFachlicheRolle“ .....             | 214 |
| 7899 | Tabelle 109: TAB_KON_829 Fehlercodes TUC_KON_036 „LiefereFachlicheRolle“ ..... | 215 |
| 7900 | Tabelle 110: TAB_KON_038 Basisanwendung Karten- und Kartenterminaldienst ..... | 215 |
| 7901 | Tabelle 111: TAB_KON_047 Operation VerifyPin .....                             | 216 |
| 7902 | Tabelle 112: TAB_KON_738 Ablauf VerifyPin .....                                | 218 |
| 7903 | Tabelle 113: TAB_KON_545 Fehlercodes „VerifyPin“ .....                         | 219 |
| 7904 | Tabelle 114: TAB_KON_049 Operation ChangePin .....                             | 219 |

|      |   |     |
|------|---|-----|
| 7905 | Tabelle 115: TAB_KON_546 Ablauf ChangePin.....                                  | 221 |
| 7906 | Tabelle 116: TAB_KON_547 Fehlercodes „ChangePin“.....                           | 222 |
| 7907 | Tabelle 117: TAB_KON_051 Operation GetPinStatus.....                            | 222 |
| 7908 | Tabelle 118: TAB_KON_548 Ablauf GetPinStatus .....                              | 224 |
| 7909 | Tabelle 119: TAB_KON_549 Fehlercodes „GetPinStatus“ .....                       | 225 |
| 7910 | Tabelle 120: TAB_KON_053 Operation UnblockPin .....                             | 225 |
| 7911 | Tabelle 121: TAB_KON_550 Ablauf UnblockPIN .....                                | 227 |
| 7912 | Tabelle 122: TAB_KON_551 Fehlercodes „UnblockPin“ .....                         | 228 |
| 7913 | Tabelle 123: TAB_KON_242 Operation EnablePin .....                              | 228 |
| 7914 | Tabelle 124: TAB_KON_243 Ablauf EnablePin .....                                 | 230 |
| 7915 | Tabelle 125: TAB_KON_244 Fehlercodes „EnablePin“ .....                          | 231 |
| 7916 | Tabelle 126: TAB_KON_245 Operation DisablePin .....                             | 231 |
| 7917 | Tabelle 127: TAB_KON_246 Ablauf DisablePin .....                                | 233 |
| 7918 | Tabelle 128: TAB_KON_247 Fehlercodes „DisablePin“ .....                         | 233 |
| 7919 | Tabelle 129: TAB_KON_554 Konfiguration des Kartendienstes .....                 | 234 |
| 7920 | Tabelle 130: TAB_KON_555 TUC_KON_025 „Initialisierung Kartendienst“.....        | 234 |
| 7921 | Tabelle 131: TAB_KON_030 Ereignisnachricht .....                                | 238 |
| 7922 | Tabelle 132: TAB_KON_556 TUC_KON_256 „Systemereignis absetzen“.....             | 240 |
| 7923 | Tabelle 133: TAB_KON_557 Fehlercodes TUC_KON_256 „Systemereignis absetzen“ ..   | 245 |
| 7924 | Tabelle 134: TAB_KON_558 TUC_KON_252 „Liefere KT_Liste“ .....                   | 245 |
| 7925 | Tabelle 135: TAB_KON_559 TUC_KON_253 „Liefere Karten_Liste“ .....               | 246 |
| 7926 | Tabelle 136: TAB_KON_560 Fehlercodes TUC_KON_253 „Liefere Karten_Liste“ .....   | 248 |
| 7927 | Tabelle 137: TAB_KON_561 TUC_KON_254 „Liefere Ressourcendetails“ .....          | 248 |
| 7928 | Tabelle 138: TAB_KON_562 Fehlercodes TUC_KON_254 „Liefere Ressourcendetails“ .. | 250 |
| 7929 | Tabelle 139: TAB_KON_029 Basisanwendung Systeminformationsdienst .....          | 250 |
| 7930 | Tabelle 140: TAB_KON_563 Operation GetCardTerminals.....                        | 251 |
| 7931 | Tabelle 141: TAB_KON_564 Ablauf GetCardTerminals.....                           | 253 |
| 7932 | Tabelle 142: TAB_KON_823 Fehlercodes „GetCardTerminals“ .....                   | 254 |
| 7933 | Tabelle 143: TAB_KON_565 Operation GetCards.....                                | 254 |
| 7934 | Tabelle 144: TAB_KON_566 Ablauf GetCards.....                                   | 258 |
| 7935 | Tabelle 145: TAB_KON_567 Fehlercodes „GetCards“ .....                           | 259 |
| 7936 | Tabelle 146: TAB_KON_568 Operation GetResourceInformation .....                 | 259 |
| 7937 | Tabelle 147: TAB_KON_569 Ablauf GetResourceInformation.....                     | 262 |
| 7938 | Tabelle 148: TAB_KON_570 Fehlercodes „GetResourceInformation“ .....             | 263 |
| 7939 | Tabelle 149: TAB_KON_571 Operation Subscribe .....                              | 263 |
| 7940 | Tabelle 150: TAB_KON_572 Ablauf Subscribe .....                                 | 265 |

|      |  |     |
|------|--|-----|
| 7941 | Tabelle 151: TAB_KON_573 Fehlercodes „Subscribe“ .....                         | 266 |
| 7942 | Tabelle 152: TAB_KON_574 Operation Unsubscribe .....                           | 266 |
| 7943 | Tabelle 153: TAB_KON_575 Ablauf Unsubscribe .....                              | 267 |
| 7944 | Tabelle 154: TAB_KON_576 Fehlercodes „Unsubscribe“ .....                       | 267 |
| 7945 | Tabelle 155: TAB_KON_792 Operation RenewSubscriptions .....                    | 268 |
| 7946 | Tabelle 156: TAB_KON_793 Ablauf RenewSubscriptions .....                       | 269 |
| 7947 | Tabelle 157: TAB_KON_794 Fehlercodes „RenewSubscriptions“ .....                | 269 |
| 7948 | Tabelle 158: TAB_KON_577 Operation GetSubscription .....                       | 270 |
| 7949 | Tabelle 159: TAB_KON_578 Ablauf GetSubscription .....                          | 271 |
| 7950 | Tabelle 160: TAB_KON_579 Fehlercodes „GetSubscription“ .....                   | 272 |
| 7951 | Tabelle 161: TAB_KON_580 Konfigurationswerte des Systeminformationsdienstes    |     |
| 7952 | (Administrator) .....  | 272 |
| 7953 | Tabelle 162: TAB_KON_581 Verschlüsselungsdienst Operationen für                |     |
| 7954 | EVT_MONITOR_OPERATIONS .....   | 273 |
| 7955 | Tabelle 163: TAB_KON_747 KeyReference für Encrypt /DecryptDocument .....       | 274 |
| 7956 | Tabelle 164: TAB_KON_859 Werteliste und Defaultwert des Parameters crypt bei   |     |
| 7957 | hybrider Verschlüsselung .....   | 275 |
| 7958 | Tabelle 165: TAB_KON_739 – TUC_KON_070 „Daten hybrid verschlüsseln“ .....      | 275 |
| 7959 | Tabelle 166: TAB_KON_073 Vorgaben zum Format verschlüsselter XML-Dokumente ..  | 283 |
| 7960 | Tabelle 167: TAB_KON_740 Fehlercodes TUC_KON_070 „Daten hybrid verschlüsseln“  | 284 |
| 7961 | Tabelle 168: TAB_KON_140 – TUC_KON_071 „Daten hybrid entschlüsseln“ .....      | 285 |
| 7962 | Tabelle 169: TAB_KON_142 Fehlercodes TUC_KON_071 „Daten hybrid entschlüsseln“  | 289 |
| 7963 | Tabelle 170: TAB_KON_741 – TUC_KON_072 „Daten symmetrisch verschlüsseln“ ..... | 289 |
| 7964 | Tabelle 171: TAB_KON_742 Fehlercodes TUC_KON_072 „Daten symmetrisch            |     |
| 7965 | verschlüsseln“ .....   | 290 |
| 7966 | Tabelle 172: TAB_KON_743 – TUC_KON_073 „Daten symmetrisch entschlüsseln“ ..... | 290 |
| 7967 | Tabelle 173: TAB_KON_744 Fehlercodes TUC_KON_073 „Daten symmetrisch            |     |
| 7968 | entschlüsseln“ .....   | 291 |
| 7969 | Tabelle 174: TAB_KON_860 – TUC_KON_075 „Symmetrisch verschlüsseln“ .....       | 291 |
| 7970 | Tabelle 175: TAB_KON_861 – TUC_KON_076 „Symmetrisch entschlüsseln“ .....       | 293 |
| 7971 | Tabelle 176: TAB_KON_745 Basisdienst Verschlüsselungsdienst .....              | 294 |
| 7972 | Tabelle 177: TAB_KON_071 Operation EncryptDocument .....                       | 295 |
| 7973 | Tabelle 178: TAB_KON_746 Ablauf EncryptDocument .....                          | 299 |
| 7974 | Tabelle 179: TAB_KON_141 Fehlercodes „EncryptDocument“ .....                   | 299 |
| 7975 | Tabelle 180: TAB_KON_075 Operation DecryptDocument .....                       | 300 |
| 7976 | Tabelle 181: TAB_KON_076 Ablauf DecryptDocument .....                          | 302 |
| 7977 | Tabelle 182: TAB_KON_145 Fehlercodes „DecryptDocument“ .....                   | 302 |
| 7978 | Tabelle 183: TAB_KON_582 – Signaturverfahren Dokumentensignatur .....          | 303 |



|      |   |     |
|------|---|-----|
| 7979 | Tabelle 184: TAB_KON_778 — Einsatzbereich der Signaturvarianten für XAdES, CAdES      |     |
| 7980 | und PAdES .....   | 304 |
| 7981 | Tabelle 185: TAB_KON_583 — Default-Signaturverfahren .....                            | 307 |
| 7982 | Tabelle 186: TAB_KON_584 nonQES-Operationen für EVT_MONITOR_OPERATIONS...             | 307 |
| 7983 | Tabelle 187: TAB_KON_900 Zertifikate und private-Schlüssel für Signaturerstellung und |     |
| 7984 | Signaturprüfung (QES und nonQES).....   | 308 |
| 7985 | Tabelle 188: TAB_KON_862 Werteliste und Defaultwert des Parameters crypt bei QES-     |     |
| 7986 | Erzeugung.....  | 309 |
| 7987 | Tabelle 189: TAB_KON_863 Werteliste und Defaultwert des Parameters crypt bei          |     |
| 7988 | nonQES-Erzeugung .....  | 309 |
| 7989 | Tabelle 190: TAB_KON_748 — TUC_KON_155 „Dokumente zur Signatur vorbereiten“ ..        | 316 |
| 7990 | Tabelle 191: TAB_KON_586 Fehlercodes TUC_KON_155 „Dokumente zur Signatur              |     |
| 7991 | vorbereiten“ .....  | 319 |
| 7992 | Tabelle 192: TAB_KON_749 — TUC_KON_165 „Signaturvoraussetzungen für nonQES            |     |
| 7993 | prüfen“ .....   | 319 |
| 7994 | Tabelle 193: TAB_KON_587 Fehlercodes TUC_KON_165 „Signaturvoraussetzungen für         |     |
| 7995 | nonQES prüfen“ .....  | 320 |
| 7996 | Tabelle 194: TAB_KON_750 — TUC_KON_166 „nonQES Signaturen erstellen“ .....            | 320 |
| 7997 | Tabelle 195: TAB_KON_120 Fehlercodes TUC_KON_166 „nonQES Signaturen erstellen“        |     |
| 7998 | .....   | 321 |
| 7999 | Tabelle 196: TAB_KON_751 — TUC_KON_152 „Signaturvoraussetzungen für QES prüfen“       |     |
| 8000 | .....   | 322 |
| 8001 | Tabelle 197: TAB_KON_588 Fehlercodes TUC_KON_152 „Signaturvoraussetzungen für         |     |
| 8002 | QES prüfen“ .....   | 323 |
| 8003 | Tabelle 198: TAB_KON_752 — TUC_KON_154 „QES Signaturen erstellen“ .....               | 323 |
| 8004 | Tabelle 199: TAB_KON_126 Fehlercodes TUC_KON_154 „QES Signaturen erstellen“ ..        | 327 |
| 8005 | Tabelle 200: TAB_KON_293 — TUC_KON_168 „Einzelsignatur QES erstellen“ .....           | 327 |
| 8006 | Tabelle 201: TAB_KON_590 Fehlercodes TUC_KON_168 „Einzelsignatur QES erstellen“       |     |
| 8007 | .....   | 328 |
| 8008 | Tabelle 202: TAB_KON_870 — TUC_KON_158 „Komfortsignaturen erstellen“ .....            | 328 |
| 8009 | Tabelle 203: TAB_KON_873 Fehlercodes TUC_KON_158 „Komfortsignaturen erstellen“        |     |
| 8010 | .....   | 331 |
| 8011 | Tabelle 204: TAB_KON_753 — TUC_KON_160 „Dokumente nonQES signieren“ .....             | 331 |
| 8012 | Tabelle 205: TAB_KON_127 Fehlercodes TUC_KON_160 „Dokumente nonQES signieren“         |     |
| 8013 | .....   | 334 |
| 8014 | Tabelle 206: TAB_KON_753 — TUC_KON_160 „Dokumente nonQES signieren“ .....             | 334 |
| 8015 | Tabelle 207: TAB_KON_127 Fehlercodes TUC_KON_160 „Dokumente nonQES signieren“         |     |
| 8016 | .....   | 337 |
| 8017 | Tabelle 208: TAB_KON_121 — TUC_KON_161 „nonQES Dokumentsignatur prüfen“ .....         | 337 |
| 8018 | Tabelle 209: TAB_KON_124 Fehlercodes TUC_KON_161 „nonQES Dokumentensignatur           |     |
| 8019 | prüfen“ .....   | 341 |

|      |   |     |
|------|---|-----|
| 8020 | Tabelle 210: TAB_KON_754 Übersicht Status für Prüfung einer Dokumentensignatur      | 342 |
| 8021 | Tabelle 211: TAB_KON_430 – TUC_KON_162 „Kryptographische Prüfung der XML-           |     |
| 8022 | Dokumentensignatur“ .....   | 344 |
| 8023 | Tabelle 212: TAB_KON_431 Fehlercodes TUC_KON_162 „Kryptographische Prüfung der      |     |
| 8024 | XML-Dokumentensignatur“ .....   | 345 |
| 8025 | Tabelle 213: TAB_KON_755 – TUC_KON_150 „Dokumente QES signieren“ .....              | 345 |
| 8026 | Tabelle 214: TAB_KON_128 Fehlercodes TUC_KON_150 „Dokument QES signieren“ ..        | 350 |
| 8027 | Tabelle 215: TAB_KON_192 Verhalten des Konnektors beim Abbruch einer Stapelsignatur |     |
| 8028 | .....   | 351 |
| 8029 | Tabelle 216: TAB_KON_591 – TUC_KON_151 „QES Dokumentensignatur prüfen“ .....        | 352 |
| 8030 | Tabelle 217: TAB_KON_592 Fehlercodes TUC_KON_151 „QES Dokumentensignatur            |     |
| 8031 | prüfen“ .....   | 356 |
| 8032 | Tabelle 218: TAB_KON_593 Übersicht Status für Prüfung einer Dokumentensignatur      | 357 |
| 8033 | Tabelle 219: TAB_KON_871 – TUC_KON_170 „Dokumente mit Komfort signieren“ .....      | 358 |
| 8034 | Tabelle 220: TAB_KON_872 Fehlercodes TUC_KON_170 „Dokumente mit Komfort             |     |
| 8035 | signieren“ .....  | 361 |
| 8036 | Tabelle 221: TAB_KON_883 – TUC_KON_171 „Komfortsignatur einschalten“ .....          | 361 |
| 8037 | Tabelle 222: TAB_KON_886 Fehlercodes TUC_KON_171 „Komfortsignatur einschalten“      |     |
| 8038 | .....   | 363 |
| 8039 | Tabelle 223: TAB_KON_884 – TUC_KON_172 „Komfortsignatur ausschalten“ .....          | 363 |
| 8040 | Tabelle 224: TAB_KON_887 Fehlercodes TUC_KON_172 „Komfortsignatur ausschalten“      |     |
| 8041 | .....   | 364 |
| 8042 | Tabelle 225: TAB_KON_885 – TUC_KON_173 „Liefere Signaturmodus“ .....                | 365 |
| 8043 | Tabelle 226: TAB_KON_888 Fehlercodes TUC_KON_173 „Liefere Signaturmodus“ .....      | 366 |
| 8044 | Tabelle 227: TAB_KON_197 Basisdienst Signaturdienst (nonQES und QES) .....          | 367 |
| 8045 | Tabelle 228: TAB_KON_065 Operation SignDocument (nonQES und QES) .....              | 368 |
| 8046 | Tabelle 229: TAB_KON_756 Ablauf Operation SignDocument (nonQES und QES) .....       | 380 |
| 8047 | Tabelle 230: TAB_KON_757 Fehlercodes „SignDocument (nonQES und QES)“ .....          | 381 |
| 8048 | Tabelle 231: TAB_KON_066 Operation VerifyDocument (nonQES und QES) .....            | 381 |
| 8049 | Tabelle 232: TAB_KON_760 Ablauf Operation VerifyDocument (nonQES und QES) .....     | 386 |
| 8050 | Tabelle 233: TAB_KON_761 Fehlercodes „VerifyDocument (nonQES und QES)“ .....        | 386 |
| 8051 | Tabelle 234: TAB_KON_840 Operation StopSignature .....                              | 387 |
| 8052 | Tabelle 235: TAB_KON_841 Ablauf Operation StopSignature .....                       | 388 |
| 8053 | Tabelle 236: TAB_KON_842 Fehlercodes „StopSignature“ .....                          | 388 |
| 8054 | Tabelle 237: TAB_KON_843 Operation GetJobNumber .....                               | 388 |
| 8055 | Tabelle 238: TAB_KON_844 Ablauf Operation GetJobNumber .....                        | 389 |
| 8056 | Tabelle 239: TAB_KON_845 Fehlercodes „GetJobNumber“ .....                           | 389 |
| 8057 | Tabelle 240: TAB_KON_874 ActivateComfortSignature .....                             | 389 |
| 8058 | Tabelle 241: TAB_KON_877 Ablauf ActivateComfortSignature .....                      | 390 |



|      |   |     |
|------|---|-----|
| 8059 | Tabelle 242: TAB_KON_879 Fehlercodes ActivateComfortSignature .....               | 391 |
| 8060 | Tabelle 243: TAB_KON_875 DeactivateComfortSignature .....                         | 391 |
| 8061 | Tabelle 244: TAB_KON_878 Ablauf DeactivateComfortSignature .....                  | 392 |
| 8062 | Tabelle 245: TAB_KON_880 Fehlercodes DeactivateComfortSignature .....             | 392 |
| 8063 | Tabelle 246: TAB_KON_876 GetSignatureMode .....                                   | 392 |
| 8064 | Tabelle 247: TAB_KON_882 Ablauf GetSignatureMode .....                            | 394 |
| 8065 | Tabelle 248: TAB_KON_881 Fehlercodes GetSignatureMode .....                       | 394 |
| 8066 | Tabelle 249: TAB_KON_596 Konfigurationswerte des Signatordienstes (Administrator) |     |
| 8067 | .....   | 395 |
| 8068 | Tabelle 250: TAB_KON_853 intendedKeyUsage bei Zertifikatsprüfung .....            | 396 |
| 8069 | Tabelle 251: TAB_KON_858 Kartenobjekt in Abhängigkeit vom kryptographischen       |     |
| 8070 | Verfahren .....   | 398 |
| 8071 | Tabelle 252: TAB_KON_825 Fehlercodes „TLS Verbindungsaufbau zum TSL-Dienst“ ..    | 400 |
| 8072 | Tabelle 253: TAB_KON_826 Fehlercodes „TLS Verbindungsaufbau zum TSL-Dienst bei    |     |
| 8073 | Prüfung der technischen Rolle“ .....  | 400 |
| 8074 | Tabelle 254: TAB_KON_597 Operationen in EVT_MONITOR_OPERATIONS .....              | 402 |
| 8075 | Tabelle 255: TAB_KON_766 TUC_KON_032 „TSL aktualisieren“ .....                    | 402 |
| 8076 | Tabelle 256: TAB_KON_598 Fehlercodes TUC_KON_032 „TSL aktualisieren“ .....        | 405 |
| 8077 | Tabelle 257: TAB_KON_618 TUC_KON_031 „BNetzA-VL aktualisieren“ .....              | 407 |
| 8078 | Tabelle 258: TAB_KON_619 Fehlercodes TUC_KON_031 „BNetzA-VL aktualisieren“ ....   | 408 |
| 8079 | Tabelle 259: TAB_KON_767 TUC_KON_040 „CRL aktualisieren“ .....                    | 408 |
| 8080 | Tabelle 260: TAB_KON_599 Fehlercodes TUC_KON_040 „CRL aktualisieren“ .....        | 410 |
| 8081 | Tabelle 261: TAB_KON_768 TUC_KON_033 „Zertifikatsablauf prüfen“ .....             | 410 |
| 8082 | Tabelle 262: TAB_KON_600 Fehlercodes TUC_KON_033 „Zertifikatsablauf prüfen“ ....  | 413 |
| 8083 | Tabelle 263: TAB_KON_769 TUC_KON_037 „Zertifikat prüfen“ .....                    | 413 |
| 8084 | Tabelle 264: TAB_KON_601 Fehlercodes TUC_KON_037 „Zertifikat prüfen“ .....        | 418 |
| 8085 | Tabelle 265: TAB_KON_818 TUC_KON_042 „CV-Zertifikat prüfen“ .....                 | 418 |
| 8086 | Tabelle 266: TAB_KON_819 Fehlercodes TUC_KON_042 „CV-Zertifikat prüfen“ .....     | 420 |
| 8087 | Tabelle 267: TAB_KON_770 TUC_KON_034 „Zertifikatsinformationen extrahieren“ ....  | 420 |
| 8088 | Tabelle 268: TAB_KON_602 Fehlercodes TUC_KON_034 „Zertifikatsinformationen        |     |
| 8089 | extrahieren“ .....  | 423 |
| 8090 | Tabelle 269: TAB_KON_771 Basisanwendung Zertifikatsdienst .....                   | 423 |
| 8091 | Tabelle 270: TAB_KON_676 Operation CheckCertificateExpiration .....               | 424 |
| 8092 | Tabelle 271: TAB_KON_677 Ablauf CheckCertificateExpiration .....                  | 426 |
| 8093 | Tabelle 272: TAB_KON_603 Fehlercodes „CheckCertificateExpiration“ .....           | 427 |
| 8094 | Tabelle 273: TAB_KON_678 Operation ReadCardCertificate .....                      | 427 |
| 8095 | Tabelle 274: TAB_KON_679 Ablauf ReadCardCertificate .....                         | 430 |
| 8096 | Tabelle 275: TAB_KON_604 Fehlercodes „ReadCardCertificate“ .....                  | 431 |

|      |   |     |
|------|---|-----|
| 8097 | Tabelle 276: TAB_KON_795 Operation VerifyCertificate .....  | 431 |
| 8098 | Tabelle 277: TAB_KON_797 Ablauf VerifyCertificate .....   | 432 |
| 8099 | Tabelle 278: TAB_KON_800 Fehlercodes „VerifyCertificate“ .....  | 433 |
| 8100 | Tabelle 279: TAB_KON_772 TUC_KON_035 „Zertifikatsdienst initialisieren“ .....                           | 434 |
| 8101 | Tabelle 280: TAB_KON_605 Fehlercodes TUC_KON_035 „Zertifikatsdienst initialisieren“ .....               | 435 |
| 8102 | .....   | 435 |
| 8103 | Tabelle 281: TAB_KON_606 Konfiguration des Zertifikatsdienstes .....                                    | 435 |
| 8104 | Tabelle 282: TAB_KON_733 Einsehbare Konfigurationsparameter des Zertifikatsdienstes .....               | 437 |
| 8105 | .....   | 437 |
| 8106 | Tabelle 283: TAB_KON_857 Fehlercodes beim Import des Cross-Zertifikats für TI-Vertrauensanker ECC ..... | 441 |
| 8107 | .....   | 441 |
| 8108 | Tabelle 284: TAB_KON_607 TUC_KON_271 „Schreibe Protokolleintrag“ .....                                  | 444 |
| 8109 | Tabelle 285: TAB_KON_608 Fehlercodes TUC_KON_271 „Schreibe Protokolleintrag“ ..                         | 447 |
| 8110 | Tabelle 286: TAB_KON_609 Konfigurationswerte des Protokollierungsdienstes (Administrator) .....         | 449 |
| 8111 | .....   | 449 |
| 8112 | Tabelle 287: TAB_KON_610 TUC_KON_272 „Initialisierung Protokollierungsdienst“ ..                        | 450 |
| 8113 | Tabelle 288: TAB_KON_611 Fehlercodes TUC_KON_272 „Initialisiere Protokollierungsdienst“ .....           | 451 |
| 8114 | .....   | 451 |
| 8115 | Tabelle 289: TAB_KON_773 TUC_KON_110 „Kartenbasierte TLS-Verbindung aufbauen“ .....                     | 452 |
| 8116 | .....   | 452 |
| 8117 | Tabelle 290: TAB_KON_612 Fehlercodes TUC_KON_110 „Kartenbasierte TLS-Verbindung aufbauen“ .....         | 453 |
| 8118 | .....   | 453 |
| 8119 | Tabelle 291: TAB_KON_774 TUC_KON_111 „Kartenbasierte TLS-Verbindung abbauen“ .....                      | 454 |
| 8120 | .....   | 454 |
| 8121 | Tabelle 292: TAB_KON_613 Fehlercodes TUC_KON_111 „Kartenbasierte TLS-Verbindung abbauen“ .....          | 455 |
| 8122 | .....   | 455 |
| 8123 | Tabelle 293: TAB_KON_805 TUC_KON_290 „LDAP-Verbindung aufbauen“ .....                                   | 456 |
| 8124 | Tabelle 294: TAB_KON_815 TUC_KON_291 „Verzeichnis abfragen“ .....                                       | 457 |
| 8125 | Tabelle 295: TAB_KON_816 TUC_KON_292 „LDAP-Verbindung trennen“ .....                                    | 458 |
| 8126 | Tabelle 296: TAB_KON_817 TUC_KON_293 „Verzeichnisabfrage abbrechen“ .....                               | 459 |
| 8127 | Tabelle 297: TAB_KON_780 Signaturverfahren Externe Authentisierung .....                                | 460 |
| 8128 | Tabelle 298: TAB_KON_839 Basisdienst Authentifizierungsdienst .....                                     | 461 |
| 8129 | Tabelle 299: TAB_KON_781 Operation ExternalAuthenticate .....   | 462 |
| 8130 | Tabelle 300: TAB_KON_782 Ablauf Operation ExternalAuthenticate .....                                    | 465 |
| 8131 | Tabelle 301: TAB_KON_783 Übersicht Fehler Operation ExternalAuthenticate .....                          | 465 |
| 8132 | Tabelle 302: TAB_KON_784 Privater Schlüssel je Karte für ExternalAuthenticate .....                     | 465 |
| 8133 | Tabelle 303: TAB_KON_680 Mapping der Netzwerksegmente .....   | 468 |
| 8134 | Tabelle 304: TAB_KON_681 Definition der vom Konnektor verwendeten VPN-Tunnel ..                         | 469 |
| 8135 | Tabelle 305: TAB_KON_682 Definition der Konnektor-IP-Adressen .....                                     | 469 |

|      |  |     |
|------|--|-----|
| 8136 | Tabelle 306: TAB_KON_614 – TUC_KON_305 „LAN-Adapter initialisieren“ .....            | 480 |
| 8137 | Tabelle 307: TAB_KON_615 Fehlercodes TUC_KON_305 „LAN-Adapter initialisieren“ ..     | 481 |
| 8138 | Tabelle 308: TAB_KON_616 – TUC_KON_306 „WAN-Adapter initialisieren“ .....            | 481 |
| 8139 | Tabelle 309: TAB_KON_617 Fehlercodes TUC_KON_306 „WAN-Adapter initialisieren“ ..     | 482 |
| 8140 | Tabelle 310: TAB_KON_622 – TUC_KON_304 „Netzwerk-Routen einrichten“ .....            | 483 |
| 8141 | Tabelle 311: TAB_KON_623 Fehlercodes TUC_KON_304 „Netzwerk-Routen einrichten“ .....  | 485 |
| 8142 | .....  |     |
| 8143 | Tabelle 312: TAB_KON_683 LAN-Adapter IP-Konfiguration .....                          | 486 |
| 8144 | Tabelle 313: TAB_KON_684 LAN-Adapter Erweiterte Parameter .....                      | 487 |
| 8145 | Tabelle 314: TAB_KON_685 WAN-Adapter IP-Konfiguration .....                          | 487 |
| 8146 | Tabelle 315: TAB_KON_686 WAN-Adapter Erweiterte Parameter .....                      | 488 |
| 8147 | Tabelle 316: TAB_KON_624 – „Konfigurationsparameter der Anbindung LAN/WAN“ ...       | 489 |
| 8148 | Tabelle 317: TAB_KON_625 – Konfigurationsparameter Firewall-Schnittstelle .....      | 492 |
| 8149 | Tabelle 318: TAB_KON_626 „Liefere Netzwerkinformationen über DHCP“ .....             | 493 |
| 8150 | Tabelle 319: TAB_KON_627 „Aktivierung des DHCP-Servers“ .....                        | 495 |
| 8151 | Tabelle 320: TAB_KON_628 „Basiskonfiguration des DHCP-Servers“ .....                 | 495 |
| 8152 | Tabelle 321: TAB_KON_629 „Client-Gruppenspezifische Konfigurationsoptionen des       |     |
| 8153 | Konnektor-DHCP-Servers“ .....  | 495 |
| 8154 | Tabelle 322: TAB_KON_630 – TUC_KON_343 „Initialisierung DHCP-Server“ .....           | 498 |
| 8155 | Tabelle 323: TAB_KON_631 Fehlercodes TUC_KON_343 „Initialisierung DHCP-Server“ ..    | 498 |
| 8156 | Tabelle 324: TAB_KON_632 – TUC_KON_341 „DHCP-Informationen beziehen“ .....           | 499 |
| 8157 | Tabelle 325: TAB_KON_633 Fehlercodes TUC_KON_341 „DHCP-Informationen beziehen“ ..... | 500 |
| 8158 | .....  |     |
| 8159 | Tabelle 326: TAB_KON_634 „Konfiguration des DHCP-Clients“ .....                      | 501 |
| 8160 | Tabelle 327: TAB_KON_635 – TUC_KON_321 „Verbindung zu dem VPN-Konzentrator der       |     |
| 8161 | TI aufbauen“ .....   | 503 |
| 8162 | Tabelle 328: TAB_KON_636 Fehlercodes TUC_KON_321 „Verbindung zu dem VPN-             |     |
| 8163 | Konzentrator der TI aufbauen“ .....  | 505 |
| 8164 | Tabelle 329: TAB_KON_637 – TUC_KON_322 „Verbindung zu dem VPN-Konzentrator der       |     |
| 8165 | SIS aufbauen“ .....  | 506 |
| 8166 | Tabelle 330: TAB_KON_638 Fehlercodes TUC_KON_322 „Verbindung zu dem VPN-             |     |
| 8167 | Konzentrator der SIS aufbauen“ .....   | 507 |
| 8168 | Tabelle 331: TAB_KON_639 – Konfigurationsparameter VPN-Client .....                  | 508 |
| 8169 | Tabelle 332: TAB_KON_640 Zustandswerte für Konnektor-NTP-Server .....                | 511 |
| 8170 | Tabelle 333: TAB_KON_776 TUC_KON_351 „Liefere Systemzeit“ .....                      | 511 |
| 8171 | Tabelle 334: TAB_KON_641 Fehlercodes TUC_KON_351 „Liefere Systemzeit“ .....          | 512 |
| 8172 | Tabelle 335: TAB_KON_642 Operation sync_Time .....                                   | 512 |
| 8173 | Tabelle 336: TAB_KON_643 Konfiguration des Konnektor-NTP-Servers .....               | 513 |

|      |   |     |
|------|---|-----|
| 8174 | Tabelle 337: TAB_KON_730-Einschbare Konfigurationsparameter des Konnektor NTP-    |     |
| 8175 | Servers.....  | 513 |
| 8176 | Tabelle 338: TAB_KON_644 – TUC_KON_352 „Initialisierung Zeitdienst“.....          | 514 |
| 8177 | Tabelle 339: TAB_KON_645 Fehlercodes TUC_KON_352 „Initialisierung Zeitdienst“ ... | 515 |
| 8178 | Tabelle 340: TAB_KON_687 DNS-Forwards des DNS-Servers.....                        | 516 |
| 8179 | Tabelle 341: TAB_KON_646 – TUC_KON_361 „DNS-Namen auflösen“.....                  | 517 |
| 8180 | Tabelle 342: TAB_KON_647 Fehlercodes TUC_KON_361 „DNS-Namen auflösen“.....        | 518 |
| 8181 | Tabelle 343: TAB_KON_646 – TUC_KON_361 „DNS-Namen auflösen“.....                  | 518 |
| 8182 | Tabelle 344: TAB_KON_647 Fehlercodes TUC_KON_361 „DNS-Namen auflösen“.....        | 519 |
| 8183 | Tabelle 345: TAB_KON_648 – TUC_KON_362 „Liste der Dienste abrufen“.....           | 520 |
| 8184 | Tabelle 346: TAB_KON_649 Fehlercodes TUC_KON_362 „Liste der Dienste abrufen“..    | 520 |
| 8185 | Tabelle 347: TAB_KON_650 – TUC_KON_363 „Dienstdetails abrufen“.....               | 521 |
| 8186 | Tabelle 348: TAB_KON_651 Fehlercodes TUC_KON_363 „Dienstdetails abrufen“.....     | 522 |
| 8187 | Tabelle 349: TAB_KON_652 Basisanwendung Namensdienst.....                         | 522 |
| 8188 | Tabelle 350: TAB_KON_653 Operation GetIPAddress.....                              | 522 |
| 8189 | Tabelle 351: TAB_KON_654 – Konfigurationsparameter Namensdienst.....              | 523 |
| 8190 | Tabelle 352: TAB_KON_731-Einschbare Konfigurationsparameter Namensdienst .....    | 524 |
| 8191 | Tabelle 353: TAB_KON_655 Konfigurationen der Benutzerverwaltung (Super-           |     |
| 8192 | Administrator).....   | 529 |
| 8193 | Tabelle 354: TAB_KON_656 Konfigurationen der Benutzerverwaltung .....             | 530 |
| 8194 | Tabelle 355: TAB_KON_657 Konfigurationsparameter des Konnektornamens .....        | 530 |
| 8195 | Tabelle 356: TAB_KON_833 Bezeichner für persistente Konfigurationsdaten für       |     |
| 8196 | Fachmodule.....   | 533 |
| 8197 | Tabelle 357: TAB_KON_658 Aktivieren/Deaktivieren von Leistungsumfängen.....       | 534 |
| 8198 | Tabelle 358: TAB_KON_659 Konnektor Standalone einsetzen .....                     | 535 |
| 8199 | Tabelle 359: TAB_KON_661 Konfigurationsparameter der Konnektorfreischaltung ..... | 536 |
| 8200 | Tabelle 360: TAB_KON_732-Einschbare Konfigurationsparameter der                   |     |
| 8201 | Konnektorfreischaltung .....  | 537 |
| 8202 | Tabelle 361: TAB_KON_662 Zustandswerte der Konnektorfreischaltung .....           | 537 |
| 8203 | Tabelle 362: TAB_KON_851 Einschränkung der Rechte des Remote-Administrators       |     |
| 8204 | (Blacklist) .....   | 541 |
| 8205 | Tabelle 363: TAB_KON_663 Konfigurationen des Remote-Managements.....              | 542 |
| 8206 | Tabelle 364: TAB_KON_664 – TUC_KON_280 „Konnektoraktualisierung durchführen“      | 545 |
| 8207 | Tabelle 365: TAB_KON_665 Fehlercodes TUC_KON_280 „Konnektoraktualisierung         |     |
| 8208 | durchführen“ .....  | 547 |
| 8209 | Tabelle 366: TAB_KON_666 – TUC_KON_281 „Kartenterminalaktualisierung anstoßen“    |     |
| 8210 | .....   | 549 |
| 8211 | Tabelle 367: TAB_KON_667 Fehlercodes TUC_KON_281 „Kartenterminalaktualisierung    |     |
| 8212 | anstoßen“ .....   | 551 |

|      |   |     |
|------|---|-----|
| 8213 | Tabelle 368: TAB_KON_668 – TUC_KON_282 „UpdateInformationen beziehen“ .....                         | 551 |
| 8214 | Tabelle 369: TAB_KON_669 Fehlercodes TUC_KON_282 „UpdateInformationen beziehen“ .....               | 553 |
| 8215 | .....   | 553 |
| 8216 | Tabelle 370: TAB_KON_799 – TUC_KON_283 „Infrastruktur Konfiguration aktualisieren“ .....            | 553 |
| 8217 | .....   | 553 |
| 8218 | Tabelle 371: Tab_Kon_726 Fehlercodes TUC_KON_283 „Infrastruktur Konfiguration aktualisieren“ .....  | 557 |
| 8219 | .....   | 557 |
| 8220 | Tabelle 372: TAB_KON_833 – TUC_KON_285 „UpdateInformationen für Fachmodul beziehen“ .....           | 557 |
| 8221 | .....   | 557 |
| 8222 | Tabelle 373: TAB_KON_834 Fehlercodes TUC_KON_285 „UpdateInformationen für Fachmodul beziehen“ ..... | 559 |
| 8223 | .....   | 559 |
| 8224 | Tabelle 374: TAB_KON_835 – TUC_KON_286 „Paket für Fachmodul laden“ .....                            | 560 |
| 8225 | Tabelle 375: TAB_KON_836 Fehlercodes TUC_KON_286 „Paket für Fachmodul laden“ .....                  | 561 |
| 8226 | Tabelle 376: TAB_KON_864 – TUC_KON_284 „KSR-Client initialisieren“ .....                            | 561 |
| 8227 | Tabelle 377: TAB_KON_822 Fehlercodes TUC_KON_284 „KSR-Client initialisieren“ .....                  | 562 |
| 8228 | Tabelle 378: TAB_KON_670 Konfigurationsparameter der Software-Aktualisierung .....                  | 562 |
| 8229 | Tabelle 379: TAB_KON_820 Einschbare Konfigurationsparameter der Software-Aktualisierung .....       | 563 |
| 8230 | .....   | 563 |
| 8231 | Tabelle 380: TAB_KON_671 Anforderungen Klima .....  | 570 |
| 8232 | Tabelle 381: TAB_KON_672 Anforderungen Vibration .....  | 570 |
| 8233 | Tabelle 382: TAB_KON_672 Anforderungen Vibration .....  | 572 |
| 8234 | Tabelle 383: TAB_KON_779 „Profilierung der Signaturformate“ .....                                   | 611 |
| 8235 | Tabelle 384: TAB_KON_775 „Profilierung der Dokumentformate und Nachrichten“ .....                   | 618 |
| 8236 | Tabelle 385: TAB_KON_688 Version der Schemas aus dem Namensraum des Konnektors .....                | 620 |
| 8237 | .....   | 620 |
| 8238 | Tabelle 386: TAB_KON_798 Schnittstellenversionen .....  | 623 |
| 8239 | Tabelle 387: TAB_KON_777 Events Interne Mechanismen .....   | 630 |
| 8240 | Tabelle 388: TAB_KON_711 Architektur der TI-Plattform, Berechtigt Fachmodule .....                  | 648 |
| 8241 | Tabelle 389: TAB_KON_712 Architektur der TI-Plattform, Berechtigt Clientsysteme ...                 | 653 |
| 8242 | Tabelle 390: TAB_KON_713 Architektur der TI-Plattform, Berechtigt eHealth-KT .....                  | 655 |
| 8243 | Tabelle 391: TAB_KON_714 Architektur der TI-Plattform, Berechtigt Administrator .....               | 655 |
| 8244 | Tabelle 392: Aufzähltypen .....   | 678 |
| 8245 | Tabelle 1: TAB_KON_500 Wertetabelle Kartentypen .....   | 39  |
| 8246 | Tabelle 2: TAB_KON_856: Identitäten des Konnektors auf der gSMC-K .....                             | 41  |
| 8247 | Tabelle 3: TAB_KON_503 Betriebszustand_Fehlerzustandsliste .....                                    | 46  |
| 8248 | Tabelle 4: TAB_KON_504 Ausführungserlaubnis für Dienste in kritischen Fehlerzuständen .....         | 52  |
| 8249 | .....   | 52  |
| 8250 | Tabelle 5: TAB_KON_502 Fehlercodes „Betriebszustand“ .....  | 58  |
| 8251 | Tabelle 6: TAB_KON_505 Konfigurationswerte Missbrauchserkennung .....                               | 58  |



|      |  |     |
|------|--|-----|
| 8252 | Tabelle 7: TAB_KON_852 Konfigurationsvarianten der Verbindungen zwischen Konnektor |     |
| 8253 | und Clientsystemen .....   | 61  |
| 8254 | Tabelle 8: TAB_KON_860 Konfigurationsvarianten der Verbindungen zwischen Konnektor |     |
| 8255 | und Clientsystemen bei LDAPS .....   | 62  |
| 8256 | Tabelle 9: TAB_KON_506 Konfigurationsparameter der Clientsystem-Authentisierung .. | 64  |
| 8257 | Tabelle 10: TAB_KON_812 Umgebungsabhängige Konfigurationsparameter .....           | 70  |
| 8258 | Tabelle 11: TAB_KON_507 Informationsmodell Entitäten .....                         | 74  |
| 8259 | Tabelle 12: TAB_KON_508 Informationsmodell Attribute .....                         | 78  |
| 8260 | Tabelle 13: TAB_KON_509 Informationsmodell Entitätenbeziehungen .....              | 79  |
| 8261 | Tabelle 14: TAB_KON_510 Informationsmodell Constraints .....                       | 81  |
| 8262 | Tabelle 15: TAB_KON_511 – TUC_KON_000 „Prüfe Zugriffsberechtigung“ .....           | 84  |
| 8263 | Tabelle 16: TAB_KON_512 Zugriffsregeln Beschreibung .....                          | 87  |
| 8264 | Tabelle 17: TAB_KON_513 Zugriffsregeln Regelzuordnung .....                        | 89  |
| 8265 | Tabelle 18: TAB_KON_514-01 Zugriffsregeln Definition.....                          | 90  |
| 8266 | Tabelle 19: TAB_KON_515 Fehlercodes TUC_KON_000 „Prüfe Zugriffsberechtigung“ ....  | 94  |
| 8267 | Tabelle 20: TAB_KON_143 – TUC_KON_080 „Dokument validieren“.....                   | 96  |
| 8268 | Tabelle 21: TAB_KON_144 Fehlercodes TUC_KON_080 „Dokument validieren“ .....        | 98  |
| 8269 | Tabelle 22: TAB_KON_516 Basisanwendung Dienstverzeichnisdienst .....               | 100 |
| 8270 | Tabelle 23: TAB_KON_517 Schemabeschreibung Produktinformation                      |     |
| 8271 | (ProductInformation.xsd) .....   | 101 |
| 8272 | Tabelle 24: TAB_KON_518 Schemabeschreibung Serviceinformation                      |     |
| 8273 | (Serviceinformation.xsd) .....   | 102 |
| 8274 | Tabelle 25: TAB_KON_519 - TUC_KON_041 „Einbringen der Endpunkthinformationen       |     |
| 8275 | während der Bootup-Phase“ .....  | 103 |
| 8276 | Tabelle 26: TAB_KON_520 Fehlercodes TUC_KON_041 „Einbringen der                    |     |
| 8277 | Endpunkthinformationen während der Bootup-Phase“ .....                             | 104 |
| 8278 | Tabelle 27: TAB_KON_521 Schnittstelle der Basisanwendung Dienstverzeichnisdienst   | 104 |
| 8279 | Tabelle 28: TAB_KON_522 Parameterübersicht des Kartenterminaldienstes .....        | 106 |
| 8280 | Tabelle 29: TAB_KON_785 Erlaubte SICCT-Kommandos bei CT.CONNECTED=Nein ....        | 111 |
| 8281 | Tabelle 30: TAB_KON_727 Terminalanzeigen beim Anfordern und Auswerfen von Karten   |     |
| 8282 | .....  | 112 |
| 8283 | Tabelle 31: TAB_KON_039 – TUC_KON_050 „Beginne Kartenterminalsitzung“ .....        | 114 |
| 8284 | Tabelle 32: TAB_KON_523 Fehlercodes TUC_KON_050 „Beginne Kartenterminalsitzung“    |     |
| 8285 | .....  | 120 |
| 8286 | Tabelle 33: TAB_KON_524 – TUC_KON_054 „Kartenterminal hinzufügen“ .....            | 120 |
| 8287 | Tabelle 34: TAB_KON_525 Fehlercodes TUC_KON_054 „Kartenterminal hinzufügen“ ..     | 122 |
| 8288 | Tabelle 35: TAB_KON_041 – TUC_KON_053 „Paire Kartenterminal“ .....                 | 122 |
| 8289 | Tabelle 36: TAB_KON_113 Fehlercodes TUC_KON_053 „Paire Kartenterminal“ .....       | 125 |
| 8290 | Tabelle 37: TAB_KON_526 – TUC_KON_055 „Befülle CT-Object“ .....                    | 127 |

|      |  |     |
|------|--|-----|
| 8291 | Tabelle 38: TAB_KON_112 – TUC_KON_051 „Mit Anwender über Kartenterminal          |     |
| 8292 | interagieren“ .....  | 128 |
| 8293 | Tabelle 39: TAB_KON_114 Fehlercodes TUC_KON_051 „Mit Anwender über               |     |
| 8294 | Kartenterminal interagieren“ .....   | 130 |
| 8295 | Tabelle 40: TAB_KON_723 – TUC_KON_056 „Karte anfordern“ .....                    | 131 |
| 8296 | Tabelle 41: TAB_KON_724 Fehlercodes TUC_KON_056 „Karte anfordern“ .....          | 133 |
| 8297 | Tabelle 42: TAB_KON_725 – TUC_KON_057 „Karte auswerfen“ .....                    | 133 |
| 8298 | Tabelle 43: TAB_KON_796 Fehlercodes TUC_KON_057 „Karte auswerfen“ .....          | 135 |
| 8299 | Tabelle 44: TAB_KON_854 – TUC_KON_058 „Displaygröße ermitteln“ .....             | 136 |
| 8300 | Tabelle 45: TAB_KON_855 Fehlercodes TUC_KON_058 „Displaygröße ermitteln“ .....   | 137 |
| 8301 | Tabelle 46: TAB_KON_722 Basisdienst Kartenterminaldienst .....                   | 137 |
| 8302 | Tabelle 47: TAB_KON_716 Operation RequestCard .....                              | 137 |
| 8303 | Tabelle 48: TAB_KON_717 Ablauf RequestCard .....                                 | 139 |
| 8304 | Tabelle 49: TAB_KON_718 Fehlercodes „RequestCard“ .....                          | 139 |
| 8305 | Tabelle 50: TAB_KON_719 Operation EjectCard .....                                | 140 |
| 8306 | Tabelle 51: TAB_KON_720 Ablauf EjectCard .....                                   | 141 |
| 8307 | Tabelle 52: TAB_KON_721 Fehlercodes Operation „EjectCard“ .....                  | 142 |
| 8308 | Tabelle 53: TAB_KON_527 Konfigurationswerte eines Kartenterminalobjekts .....    | 142 |
| 8309 | Tabelle 54: TAB_KON_528 Informationsparameter des Kartenterminaldienstes .....   | 143 |
| 8310 | Tabelle 55: TAB_KON_529 Anzeigewerte zu einem Kartenterminalobjekt .....         | 144 |
| 8311 | Tabelle 56: TAB_KON_530 Konfigurationswerte eines Kartenterminalobjekts .....    | 146 |
| 8312 | Tabelle 57: TAB_KON_531 Parameterübersicht des Kartendienstes .....              | 149 |
| 8313 | Tabelle 58: TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal |     |
| 8314 | .....  | 152 |
| 8315 | Tabelle 59: TAB_KON_734 – TUC_KON_001 „Karte öffnen“ .....                       | 157 |
| 8316 | Tabelle 60: TAB_KON_735 – TUC_KON_026 .....                                      | 160 |
| 8317 | Tabelle 61: TAB_KON_824 Fehlercodes TUC_KON_026 „Liefere CardSession“ .....      | 161 |
| 8318 | Tabelle 62: TAB_KON_087 – TUC_KON_012 „PIN verifizieren“ .....                   | 161 |
| 8319 | Tabelle 63: TAB_KON_089 Fehlercodes TUC_KON_012 „PIN verifizieren“ .....         | 165 |
| 8320 | Tabelle 64: TAB_KON_736 – TUC_KON_019 „PIN ändern“ .....                         | 166 |
| 8321 | Tabelle 65: TAB_KON_093 Fehlercodes TUC_KON_019 „PIN ändern“ .....               | 169 |
| 8322 | Tabelle 66: TAB_KON_236 – TUC_KON_021 „PIN entsperren“ .....                     | 170 |
| 8323 | Tabelle 67: TAB_KON_193 Fehlercodes TUC_KON_021 „PIN entsperren“ .....           | 173 |
| 8324 | Tabelle 68: TAB_KON_532 – TUC_KON_022 „Liefere PIN-Status“ .....                 | 174 |
| 8325 | Tabelle 69: TAB_KON_091 Fehlercodes TUC_KON_022 „Liefere PIN-Status“ .....       | 176 |
| 8326 | Tabelle 70: TAB_KON_240 – TUC_KON_027 „PIN-Schutz ein-/ausschalten“ .....        | 176 |
| 8327 | Tabelle 71: TAB_KON_838 Mapping von pinRef auf ANW .....                         | 179 |

|      |  |     |
|------|--|-----|
| 8328 | Tabelle 72: TAB_KON_241 Fehlercodes TUC_KON_027 „PIN-Schutz ein/ausschalten“ . | 179 |
| 8329 | Tabelle 73: TAB_KON_533 - TUC_KON_023 „Karte reservieren“ .....                | 180 |
| 8330 | Tabelle 74: TAB_KON_534 Fehlercodes TUC_KON_023 „Karte reservieren“ .....      | 181 |
| 8331 | Tabelle 75: TAB_KON_096 – TUC_KON_005 „Card-to-Card authentisieren“ .....      | 182 |
| 8332 | Tabelle 76: TAB_KON_673 AuthMode für C2C.....                                  | 184 |
| 8333 | Tabelle 77: TAB_KON_674 Erlaubte Parameterkombinationen und resultierende CV-  |     |
| 8334 | Zertifikate für C2C.....   | 185 |
| 8335 | Tabelle 78: TAB_KON_535 Fehlercodes TUC_KON_005 „Card-to-Card authentisieren“  | 185 |
| 8336 | Tabelle 79: TAB_KON_218 – TUC_KON_202 „LeseDatei“ .....                        | 186 |
| 8337 | Tabelle 80: TAB_KON_536 Fehlercodes TUC_KON_202 „LeseDatei“ .....              | 187 |
| 8338 | Tabelle 81: TAB_KON_219 – TUC_KON_203 „SchreibeDatei“ .....                    | 188 |
| 8339 | Tabelle 82: TAB_KON_537 Fehlercodes TUC_KON_203 „Schreibe Datei“ .....         | 189 |
| 8340 | Tabelle 83: TAB_KON_204 – TUC_KON_204 „LöscheDateiInhalt“ .....                | 190 |
| 8341 | Tabelle 84: TAB_KON_785 Fehlercodes TUC_KON_204 „LöscheDateiInhalt“ .....      | 191 |
| 8342 | Tabelle 85: TAB_KON_538 – TUC_KON_209 „LeseRecord“ .....                       | 192 |
| 8343 | Tabelle 86: TAB_KON_539 Fehlercodes TUC_KON_209 „LeseRecord“ .....             | 193 |
| 8344 | Tabelle 87: TAB_KON_224 – TUC_KON_210 „SchreibeRecord“ .....                   | 194 |
| 8345 | Tabelle 88: TAB_KON_540 Fehlercodes TUC_KON_210 „SchreibeRecord“ .....         | 195 |
| 8346 | Tabelle 89: TAB_KON_211 – TUC_KON_211 „LöscheRecordInhalt“ .....               | 196 |
| 8347 | Tabelle 90: TAB_KON_786 Fehlercodes TUC_KON_211 „LöscheRecordInhalt“ .....     | 197 |
| 8348 | Tabelle 91: TAB_KON_228 – TUC_KON_214 „FügeHinzuRecord“ .....                  | 198 |
| 8349 | Tabelle 92: TAB_KON_541 Fehlercodes TUC_KON_214 „FügeHinzuRecord“ .....        | 199 |
| 8350 | Tabelle 93: TAB_KON_229 – TUC_KON_215 „SucheRecord“ .....                      | 200 |
| 8351 | Tabelle 94: TAB_KON_542 Fehlercodes TUC_KON_215 „SucheRecord“ .....            | 201 |
| 8352 | Tabelle 95: TAB_KON_110 - TUC_KON_018 „eGK-Sperrung prüfen“ .....              | 202 |
| 8353 | Tabelle 96: TAB_KON_239 Fehlercodes TUC_KON_018 „eGK-Sperrung prüfen“ .....    | 203 |
| 8354 | Tabelle 97: TAB_KON_108 - TUC_KON_006 „Datenzugriffsaudit eGK schreiben“ ..... | 204 |
| 8355 | Tabelle 98: TAB_KON_238 Fehlercodes TUC_KON_006 „Datenzugriffsaudit eGK        |     |
| 8356 | schreiben“ .....   | 205 |
| 8357 | Tabelle 99: TAB_KON_231 – TUC_KON_218 „Signiere“ .....                         | 205 |
| 8358 | Tabelle 100: TAB_KON_543 Fehlercodes TUC_KON_218 „Signiere“ .....              | 207 |
| 8359 | Tabelle 101: TAB_KON_232 – TUC_KON_219 „Entschlüssele“ .....                   | 207 |
| 8360 | Tabelle 102: TAB_KON_210 Fehlercodes TUC_KON_219 „Entschlüssele“ .....         | 208 |
| 8361 | Tabelle 103: TAB_KON_215 TUC_KON_200 „SendeAPDU“ .....                         | 209 |
| 8362 | Tabelle 104: TAB_KON_216 Fehlercodes TUC_KON_200 „SendeAPDU“ .....             | 210 |
| 8363 | Tabelle 105: TAB_KON_737 – TUC_KON_024 „Karte zurücksetzen“ .....              | 210 |
| 8364 | Tabelle 106: TAB_KON_544 Fehlercodes TUC_KON_024 „Karte zurücksetzen“ .....    | 211 |



|      |   |     |
|------|---|-----|
| 8365 | Tabelle 107: TAB_KON_230 – TUC_KON_216 „LeseZertifikat“ .....                   | 212 |
| 8366 | Tabelle 108: TAB_KON_209 Fehlercodes TUC_KON_216 „LeseZertifikat“ .....         | 213 |
| 8367 | Tabelle 109: TAB_KON_827 TUC_KON_036 „LiefereFachlicheRolle“ .....              | 214 |
| 8368 | Tabelle 110: TAB_KON_829 Fehlercodes TUC_KON_036 „LiefereFachlicheRolle“ .....  | 215 |
| 8369 | Tabelle 111: TAB_KON_038 Basisanwendung Karten- und Kartenterminaldienst.....   | 215 |
| 8370 | Tabelle 112: TAB_KON_047 Operation VerifyPin .....                              | 216 |
| 8371 | Tabelle 113: TAB_KON_738 Ablauf VerifyPin .....                                 | 218 |
| 8372 | Tabelle 114: TAB_KON_545 Fehlercodes „VerifyPin“ .....                          | 219 |
| 8373 | Tabelle 115: TAB_KON_049 Operation ChangePin .....                              | 219 |
| 8374 | Tabelle 116: TAB_KON_546 Ablauf ChangePin.....                                  | 221 |
| 8375 | Tabelle 117: TAB_KON_547 Fehlercodes „ChangePin“.....                           | 222 |
| 8376 | Tabelle 118: TAB_KON_051 Operation GetPinStatus.....                            | 222 |
| 8377 | Tabelle 119: TAB_KON_548 Ablauf GetPinStatus .....                              | 224 |
| 8378 | Tabelle 120: TAB_KON_549 Fehlercodes „GetPinStatus“ .....                       | 225 |
| 8379 | Tabelle 121: TAB_KON_053 Operation UnblockPin .....                             | 225 |
| 8380 | Tabelle 122: TAB_KON_550 Ablauf UnblockPIN .....                                | 227 |
| 8381 | Tabelle 123: TAB_KON_551 Fehlercodes „UnblockPin“ .....                         | 228 |
| 8382 | Tabelle 124: TAB_KON_242 Operation EnablePin .....                              | 228 |
| 8383 | Tabelle 125: TAB_KON_243 Ablauf EnablePin .....                                 | 230 |
| 8384 | Tabelle 126: TAB_KON_244 Fehlercodes „EnablePin“ .....                          | 231 |
| 8385 | Tabelle 127: TAB_KON_245 Operation DisablePin .....                             | 231 |
| 8386 | Tabelle 128: TAB_KON_246 Ablauf DisablePin .....                                | 233 |
| 8387 | Tabelle 129: TAB_KON_247 Fehlercodes „DisablePin“ .....                         | 233 |
| 8388 | Tabelle 130: TAB_KON_554 Konfiguration des Kartendienstes .....                 | 234 |
| 8389 | Tabelle 131: TAB_KON_555 - TUC_KON_025 „Initialisierung Kartendienst“ .....     | 234 |
| 8390 | Tabelle 132: TAB_KON_030 Ereignisnachricht .....                                | 238 |
| 8391 | Tabelle 133: TAB_KON_556 - TUC_KON_256 „Systemereignis absetzen“.....           | 240 |
| 8392 | Tabelle 134: TAB_KON_557 Fehlercodes TUC_KON_256 „Systemereignis absetzen“ ..   | 245 |
| 8393 | Tabelle 135: TAB_KON_558 – TUC_KON_252 „Liefere KT_Liste“ .....                 | 245 |
| 8394 | Tabelle 136: TAB_KON_559 – TUC_KON_253 „Liefere Karten_Liste“ .....             | 246 |
| 8395 | Tabelle 137: TAB_KON_560 Fehlercodes TUC_KON_253 „Liefere Karten_Liste“ .....   | 248 |
| 8396 | Tabelle 138: TAB_KON_561 - TUC_KON_254 „Liefere Ressourcendetails“ .....        | 248 |
| 8397 | Tabelle 139: TAB_KON_562 Fehlercodes TUC_KON_254 „Liefere Ressourcendetails“ .. | 250 |
| 8398 | Tabelle 140 TAB_KON_029 Basisanwendung Systeminformationsdienst .....           | 250 |
| 8399 | Tabelle 141: TAB_KON_563 Operation GetCardTerminals.....                        | 251 |
| 8400 | Tabelle 142: TAB_KON_564 Ablauf GetCardTerminals.....                           | 253 |

|      |   |     |
|------|---|-----|
| 8401 | Tabelle 143: TAB_KON_823 Fehlercodes „GetCardTerminals“ .....                 | 254 |
| 8402 | Tabelle 144: TAB_KON_565 Operation GetCards .....                             | 254 |
| 8403 | Tabelle 145: TAB_KON_566 Ablauf GetCards .....                                | 258 |
| 8404 | Tabelle 146: TAB_KON_567 Fehlercodes „GetCards“ .....                         | 259 |
| 8405 | Tabelle 147: TAB_KON_568 Operation GetResourceInformation .....               | 259 |
| 8406 | Tabelle 148: TAB_KON_569 Ablauf GetResourceInformation .....                  | 262 |
| 8407 | Tabelle 149: TAB_KON_570 Fehlercodes „GetResourceInformation“ .....           | 263 |
| 8408 | Tabelle 150: TAB_KON_571 Operation Subscribe .....                            | 263 |
| 8409 | Tabelle 151: TAB_KON_572 Ablauf Subscribe .....                               | 265 |
| 8410 | Tabelle 152: TAB_KON_573 Fehlercodes „Subscribe“ .....                        | 266 |
| 8411 | Tabelle 153: TAB_KON_574 Operation Unsubscribe .....                          | 266 |
| 8412 | Tabelle 154: TAB_KON_575 Ablauf Unsubscribe .....                             | 267 |
| 8413 | Tabelle 155: TAB_KON_576 Fehlercodes „Unsubscribe“ .....                      | 267 |
| 8414 | Tabelle 156: TAB_KON_792 Operation RenewSubscriptions .....                   | 268 |
| 8415 | Tabelle 157: TAB_KON_793 Ablauf RenewSubscriptions .....                      | 269 |
| 8416 | Tabelle 158: TAB_KON_794 Fehlercodes „RenewSubscriptions“ .....               | 269 |
| 8417 | Tabelle 159: TAB_KON_577 Operation GetSubscription .....                      | 270 |
| 8418 | Tabelle 160: TAB_KON_578 Ablauf GetSubscription .....                         | 271 |
| 8419 | Tabelle 161: TAB_KON_579 Fehlercodes „GetSubscription“ .....                  | 272 |
| 8420 | Tabelle 162: TAB_KON_580 Konfigurationswerte des Systeminformationsdienstes   |     |
| 8421 | (Administrator) .....   | 272 |
| 8422 | Tabelle 163: TAB_KON_581 Verschlüsselungsdienst-Operationen für               |     |
| 8423 | EVT_MONITOR_OPERATIONS .....  | 273 |
| 8424 | Tabelle 164: TAB_KON_747 KeyReference für Encrypt-/DecryptDocument .....      | 274 |
| 8425 | Tabelle 165: TAB_KON_859 Werteliste und Defaultwert des Parameters crypt bei  |     |
| 8426 | hybrider Verschlüsselung .....  | 275 |
| 8427 | Tabelle 166: TAB_KON_739 - TUC_KON_070 „Daten hybrid verschlüsseln“ .....     | 275 |
| 8428 | Tabelle 167: TAB_KON_073 Vorgaben zum Format verschlüsselter XML-Dokumente .. | 283 |
| 8429 | Tabelle 168: TAB_KON_740 Fehlercodes TUC_KON_070 „Daten hybrid verschlüsseln“ | 284 |
| 8430 | Tabelle 169: TAB_KON_140 – TUC_KON_071 „Daten hybrid entschlüsseln“ .....     | 285 |
| 8431 | Tabelle 170: TAB_KON_142 Fehlercodes TUC_KON_071 „Daten hybrid entschlüsseln“ | 289 |
| 8432 | Tabelle 171: TAB_KON_741 – TUC_KON_072 „Daten symmetrisch verschlüsseln“ .... | 289 |
| 8433 | Tabelle 172: TAB_KON_742 Fehlercodes TUC_KON_072 „Daten symmetrisch           |     |
| 8434 | verschlüsseln“ .....  | 290 |
| 8435 | Tabelle 173: TAB_KON_743 - TUC_KON_073 „Daten symmetrisch entschlüsseln“ .... | 290 |
| 8436 | Tabelle 174: TAB_KON_744 Fehlercodes TUC_KON_073 „Daten symmetrisch           |     |
| 8437 | entschlüsseln“ .....  | 291 |
| 8438 | Tabelle 175: TAB_KON_860 – TUC_KON_075 „Symmetrisch verschlüsseln“ .....      | 291 |

|      |   |     |
|------|---|-----|
| 8439 | Tabelle 176: TAB_KON_861 - TUC_KON_076 „Symmetrisch entschlüsseln“ .....              | 293 |
| 8440 | Tabelle 177: TAB_KON_745 Basisdienst Verschlüsselungsdienst.....                      | 294 |
| 8441 | Tabelle 178: TAB_KON_071 Operation EncryptDocument.....                               | 295 |
| 8442 | Tabelle 179: TAB_KON_746 Ablauf EncryptDocument.....                                  | 299 |
| 8443 | Tabelle 180: TAB_KON_141 Fehlercodes „EncryptDocument“ .....                          | 299 |
| 8444 | Tabelle 181: TAB_KON_075 Operation DecryptDocument.....                               | 300 |
| 8445 | Tabelle 182: TAB_KON_076 Ablauf DecryptDocument .....                                 | 302 |
| 8446 | Tabelle 183: TAB_KON_145 Fehlercodes „DecryptDocument“ .....                          | 302 |
| 8447 | Tabelle 184: TAB_KON_582 – Signaturverfahren Dokumentensignatur .....                 | 303 |
| 8448 | Tabelle 185: TAB_KON_778 – Einsatzbereich der Signaturvarianten für XAdES, CAdES      |     |
| 8449 | und PAdES .....   | 304 |
| 8450 | Tabelle 186: TAB_KON_583 – Default-Signaturverfahren .....                            | 307 |
| 8451 | Tabelle 187: TAB_KON_584 nonQES-Operationen für EVT_MONITOR_OPERATIONS...             | 307 |
| 8452 | Tabelle 188: TAB_KON_900 Zertifikate und private Schlüssel für Signaturerstellung und |     |
| 8453 | Signaturprüfung (QES und nonQES).....   | 308 |
| 8454 | Tabelle 189: TAB_KON_862 Werteliste und Defaultwert des Parameters crypt bei QES-     |     |
| 8455 | Erzeugung.....  | 309 |
| 8456 | Tabelle 190: TAB_KON_863 Werteliste und Defaultwert des Parameters crypt bei          |     |
| 8457 | nonQES-Erzeugung .....  | 309 |
| 8458 | Tabelle 191: TAB_KON_748 - TUC_KON_155 „Dokumente zur Signatur vorbereiten“ .         | 316 |
| 8459 | Tabelle 192: TAB_KON_586 Fehlercodes TUC_KON_155 „Dokumente zur Signatur              |     |
| 8460 | vorbereiten“ .....  | 319 |
| 8461 | Tabelle 193: TAB_KON_749 – TUC_KON_165 „Signaturvoraussetzungen für nonQES            |     |
| 8462 | prüfen“ .....   | 319 |
| 8463 | Tabelle 194: TAB_KON_587 Fehlercodes TUC_KON_165 „Signaturvoraussetzungen für         |     |
| 8464 | nonQES prüfen“ .....  | 320 |
| 8465 | Tabelle 195: TAB_KON_750 – TUC_KON_166 „nonQES Signaturen erstellen“ .....            | 320 |
| 8466 | Tabelle 196: TAB_KON_120 Fehlercodes TUC_KON_166 „nonQES Signaturen erstellen“        |     |
| 8467 | .....   | 321 |
| 8468 | Tabelle 197: TAB_KON_751 – TUC_KON_152 „Signaturvoraussetzungen für QES prüfen“       |     |
| 8469 | .....   | 322 |
| 8470 | Tabelle 198: TAB_KON_588 Fehlercodes TUC_KON_152 „Signaturvoraussetzungen für         |     |
| 8471 | QES prüfen“ .....   | 323 |
| 8472 | Tabelle 199: TAB_KON_752 – TUC_KON_154 „QES Signaturen erstellen“ .....               | 323 |
| 8473 | Tabelle 200: TAB_KON_126 Fehlercodes TUC_KON_154 „QES Signaturen erstellen“ ..        | 327 |
| 8474 | Tabelle 201: TAB_KON_293 - TUC_KON_168 „Einzelsignatur QES erstellen“ .....           | 327 |
| 8475 | Tabelle 202: TAB_KON_590 Fehlercodes TUC_KON_168 „Einzelsignatur QES erstellen“       |     |
| 8476 | .....   | 328 |
| 8477 | Tabelle 203: TAB_KON_870 – TUC_KON_158 „Komfortsignaturen erstellen“ .....            | 328 |

|      |   |     |
|------|---|-----|
| 8478 | Tabelle 204: TAB_KON_873 Fehlercodes TUC_KON_158 „Komfortsignaturen erstellen“      |     |
| 8479 | .....   | 331 |
| 8480 | Tabelle 205: TAB_KON_753 – TUC_KON_160 „Dokumente nonQES signieren“                 | 331 |
| 8481 | Tabelle 206: TAB_KON_127 Fehlercodes TUC_KON_160 „Dokumente nonQES signieren“       |     |
| 8482 | .....   | 334 |
| 8483 | Tabelle 207: TAB_KON_753 – TUC_KON_160 „Dokumente nonQES signieren“                 | 334 |
| 8484 | Tabelle 208: TAB_KON_127 Fehlercodes TUC_KON_160 „Dokumente nonQES signieren“       |     |
| 8485 | .....   | 337 |
| 8486 | Tabelle 209: TAB_KON_121 - TUC_KON_161 „nonQES Dokumentensignatur prüfen“           | 337 |
| 8487 | Tabelle 210: TAB_KON_124 Fehlercodes TUC_KON_161 „nonQES Dokumentensignatur         |     |
| 8488 | prüfen“   | 341 |
| 8489 | Tabelle 211: TAB_KON_754 Übersicht Status für Prüfung einer Dokumentensignatur      | 342 |
| 8490 | Tabelle 212: TAB_KON_430 – TUC_KON_162 „Kryptographische Prüfung der XML-           |     |
| 8491 | Dokumentensignatur“   | 344 |
| 8492 | Tabelle 213: TAB_KON_431 Fehlercodes TUC_KON_162 „Kryptographische Prüfung der      |     |
| 8493 | XML-Dokumentensignatur“   | 345 |
| 8494 | Tabelle 214: TAB_KON_755 – TUC_KON_150 „Dokumente QES signieren“                    | 345 |
| 8495 | Tabelle 215: TAB_KON_128 Fehlercodes TUC_KON_150 „Dokument QES signieren“           | 350 |
| 8496 | Tabelle 216: TAB_KON_192 Verhalten des Konnektors beim Abbruch einer Stapelsignatur |     |
| 8497 | .....   | 351 |
| 8498 | Tabelle 217: TAB_KON_591 - TUC_KON_151 „QES-Dokumentensignatur prüfen“              | 352 |
| 8499 | Tabelle 218: TAB_KON_592 Fehlercodes TUC_KON_151 „QES Dokumentensignatur            |     |
| 8500 | prüfen“   | 356 |
| 8501 | Tabelle 219: TAB_KON_593 Übersicht Status für Prüfung einer Dokumentensignatur      | 357 |
| 8502 | Tabelle 220: TAB_KON_871 – TUC_KON_170 „Dokumente mit Komfort signieren“            | 358 |
| 8503 | Tabelle 221: TAB_KON_872 Fehlercodes TUC_KON_170 „Dokumente mit Komfort             |     |
| 8504 | signieren“  | 361 |
| 8505 | Tabelle 222: TAB_KON_883 – TUC_KON_171 „Komfortsignatur einschalten“                | 361 |
| 8506 | Tabelle 223: TAB_KON_886 Fehlercodes TUC_KON_171 „Komfortsignatur einschalten“      |     |
| 8507 | .....   | 363 |
| 8508 | Tabelle 224: TAB_KON_884 – TUC_KON_172 „Komfortsignatur ausschalten“                | 363 |
| 8509 | Tabelle 225: TAB_KON_887 Fehlercodes TUC_KON_172 „Komfortsignatur ausschalten“      |     |
| 8510 | .....   | 364 |
| 8511 | Tabelle 226: TAB_KON_885 – TUC_KON_173 „Liefere Signaturmodus“                      | 365 |
| 8512 | Tabelle 227: TAB_KON_888 Fehlercodes TUC_KON_173 „Liefere Signaturmodus“            | 366 |
| 8513 | Tabelle 228: TAB_KON_197 Basisdienst Signaturdienst (nonQES und QES)                | 367 |
| 8514 | Tabelle 229: TAB_KON_065 Operation SignDocument (nonQES und QES)                    | 368 |
| 8515 | Tabelle 230: TAB_KON_756 Ablauf Operation SignDocument (nonQES und QES)             | 380 |
| 8516 | Tabelle 231: TAB_KON_757 Fehlercodes „SignDocument (nonQES und QES)“                | 381 |
| 8517 | Tabelle 232: TAB_KON_066 Operation VerifyDocument (nonQES und QES)                  | 381 |

|      |   |     |
|------|---|-----|
| 8518 | Tabelle 233: TAB_KON_760 Ablauf Operation VerifyDocument (nonQES und QES).....    | 386 |
| 8519 | Tabelle 234: TAB_KON_761 Fehlercodes „VerifyDocument (nonQES und QES)“ .....      | 386 |
| 8520 | Tabelle 235: TAB_KON_840 Operation StopSignature .....                            | 387 |
| 8521 | Tabelle 236: TAB_KON_841 Ablauf Operation StopSignature .....                     | 388 |
| 8522 | Tabelle 237: TAB_KON_842 Fehlercodes „StopSignature“ .....                        | 388 |
| 8523 | Tabelle 238: TAB_KON_843 Operation GetJobNumber .....                             | 388 |
| 8524 | Tabelle 239: TAB_KON_844 Ablauf Operation GetJobNumber .....                      | 389 |
| 8525 | Tabelle 240: TAB_KON_845 Fehlercodes „GetJobNumber“ .....                         | 389 |
| 8526 | Tabelle 241: TAB_KON_874 ActivateComfortSignature .....                           | 389 |
| 8527 | Tabelle 242: TAB_KON_877 Ablauf ActivateComfortSignature .....                    | 390 |
| 8528 | Tabelle 243: TAB_KON_879 Fehlercodes ActivateComfortSignature .....               | 391 |
| 8529 | Tabelle 244: TAB_KON_875 DeactivateComfortSignature .....                         | 391 |
| 8530 | Tabelle 245: TAB_KON_878 Ablauf DeactivateComfortSignature .....                  | 392 |
| 8531 | Tabelle 246: TAB_KON_880 Fehlercodes DeactivateComfortSignature .....             | 392 |
| 8532 | Tabelle 247: TAB_KON_876 GetSignatureMode .....                                   | 392 |
| 8533 | Tabelle 248: TAB_KON_882 Ablauf GetSignatureMode .....                            | 394 |
| 8534 | Tabelle 249: TAB_KON_881 Fehlercodes GetSignatureMode .....                       | 394 |
| 8535 | Tabelle 250: TAB_KON_596 Konfigurationswerte des Signaturdienstes (Administrator) |     |
| 8536 | .....   | 395 |
| 8537 | Tabelle 251: TAB_KON_853- intendedKeyUsage bei Zertifikatsprüfung .....           | 396 |
| 8538 | Tabelle 252: TAB_KON_858 Kartenobjekt in Abhängigkeit vom kryptographischen       |     |
| 8539 | Verfahren.....  | 398 |
| 8540 | Tabelle 253: TAB_KON_825 Fehlercodes „TLS-Verbindungsaufbau zum TSL-Dienst“ ..    | 400 |
| 8541 | Tabelle 254: TAB_KON_826 Fehlercodes „TLS-Verbindungsaufbau zum TSL-Dienst bei    |     |
| 8542 | Prüfung der technischen Rolle“ .....  | 400 |
| 8543 | Tabelle 255: TAB_KON_597 Operationen in EVT_MONITOR_OPERATIONS .....              | 402 |
| 8544 | Tabelle 256: TAB_KON_766 TUC_KON_032 „TSL aktualisieren“ .....                    | 402 |
| 8545 | Tabelle 257: TAB_KON_598 Fehlercodes TUC_KON_032 „TSL aktualisieren“ .....        | 405 |
| 8546 | Tabelle 258: TAB_KON_618 TUC_KON_031 „BNetzA-VL aktualisieren“ .....              | 407 |
| 8547 | Tabelle 259: TAB_KON_619 Fehlercodes TUC_KON_031 „BNetzA-VL aktualisieren“ ....   | 408 |
| 8548 | Tabelle 260: TAB_KON_767 TUC_KON_040 „CRL aktualisieren“ .....                    | 408 |
| 8549 | Tabelle 261: TAB_KON_599 Fehlercodes TUC_KON_040 „CRL aktualisieren“ .....        | 410 |
| 8550 | Tabelle 262: TAB_KON_768 TUC_KON_033 „Zertifikatsablauf prüfen“ .....             | 410 |
| 8551 | Tabelle 263: TAB_KON_600 Fehlercodes TUC_KON_033 „Zertifikatsablauf prüfen“ ....  | 413 |
| 8552 | Tabelle 264: TAB_KON_769 TUC_KON_037 „Zertifikat prüfen“ .....                    | 413 |
| 8553 | Tabelle 265: TAB_KON_601 Fehlercodes TUC_KON_037 „Zertifikat prüfen“ .....        | 418 |
| 8554 | Tabelle 266: TAB_KON_818 TUC_KON_042 „CV-Zertifikat prüfen“ .....                 | 418 |



|      |   |     |
|------|---|-----|
| 8555 | Tabelle 267: TAB_KON_819 Fehlercodes TUC_KON_042 „CV-Zertifikat prüfen“ .....       | 420 |
| 8556 | Tabelle 268: TAB_KON_770 TUC_KON_034 „Zertifikatsinformationen extrahieren“ ....    | 420 |
| 8557 | Tabelle 269: TAB_KON_602 Fehlercodes TUC_KON_034 „Zertifikatsinformationen          |     |
| 8558 | extrahieren“ .....  | 423 |
| 8559 | Tabelle 270: TAB_KON_771 Basisanwendung Zertifikatsdienst.....                      | 423 |
| 8560 | Tabelle 271: TAB_KON_676 Operation CheckCertificateExpiration .....                 | 424 |
| 8561 | Tabelle 272: TAB_KON_677 Ablauf CheckCertificateExpiration .....                    | 426 |
| 8562 | Tabelle 273: TAB_KON_603 Fehlercodes „CheckCertificateExpiration“ .....             | 427 |
| 8563 | Tabelle 274: TAB_KON_678 Operation ReadCardCertificate .....                        | 427 |
| 8564 | Tabelle 275: TAB_KON_679 Ablauf ReadCardCertificate .....                           | 430 |
| 8565 | Tabelle 276: TAB_KON_604 Fehlercodes „ReadCardCertificate“ .....                    | 431 |
| 8566 | Tabelle 277: TAB_KON_795 Operation VerifyCertificate .....                          | 431 |
| 8567 | Tabelle 278: TAB_KON_797 Ablauf VerifyCertificate .....                             | 432 |
| 8568 | Tabelle 279: TAB_KON_800 Fehlercodes „VerifyCertificate“ .....                      | 433 |
| 8569 | Tabelle 280: TAB_KON_772 TUC_KON_035 „Zertifikatsdienst initialisieren“ .....       | 434 |
| 8570 | Tabelle 281: TAB_KON_605 Fehlercodes TUC_KON_035 „Zertifikatsdienst initialisieren“ |     |
| 8571 | .....   | 435 |
| 8572 | Tabelle 282: TAB_KON_606 Konfiguration des Zertifikatsdienstes .....                | 435 |
| 8573 | Tabelle 283: TAB_KON_733 Einsehbare Konfigurationsparameter des Zertifikatsdienstes |     |
| 8574 | .....   | 437 |
| 8575 | Tabelle 284: TAB_KON_857 - Fehlercodes beim Import des Cross-Zertifikats für TI-    |     |
| 8576 | Vertrauensanker ECC.....  | 441 |
| 8577 | Tabelle 285: TAB_KON_607 – TUC_KON_271 „Schreibe Protokolleintrag“ .....            | 444 |
| 8578 | Tabelle 286: TAB_KON_608 Fehlercodes TUC_KON_271 „Schreibe Protokolleintrag“ ..     | 447 |
| 8579 | Tabelle 287: TAB_KON_609 Konfigurationswerte des Protokollierungsdienstes           |     |
| 8580 | (Administrator) .....   | 449 |
| 8581 | Tabelle 288: TAB_KON_610 – TUC_KON_272 „Initialisierung Protokollierungsdienst“ .   | 450 |
| 8582 | Tabelle 289: TAB_KON_611 Fehlercodes TUC_KON_272 „Initialisiere                     |     |
| 8583 | Protokollierungsdienst“ .....   | 451 |
| 8584 | Tabelle 290: TAB_KON_773 – TUC_KON_110 „Kartenbasierte TLS-Verbindung aufbauen“     |     |
| 8585 | .....   | 452 |
| 8586 | Tabelle 291: TAB_KON_612 Fehlercodes TUC_KON_110 „Kartenbasierte TLS-Verbindung     |     |
| 8587 | aufbauen“ .....   | 453 |
| 8588 | Tabelle 292: TAB_KON_774 - TUC_KON_111 „Kartenbasierte TLS-Verbindung abbauen“      |     |
| 8589 | .....   | 454 |
| 8590 | Tabelle 293: TAB_KON_613 Fehlercodes TUC_KON_111 „Kartenbasierte TLS-Verbindung     |     |
| 8591 | abbauen“ .....  | 455 |
| 8592 | Tabelle 294: TAB_KON_805 - TUC_KON_290 „LDAP-Verbindung aufbauen“ .....             | 456 |
| 8593 | Tabelle 295: TAB_KON_815 – TUC_KON_291 „Verzeichnis abfragen“ .....                 | 457 |
| 8594 | Tabelle 296: TAB_KON_816 – TUC_KON_292 „LDAP-Verbindung trennen“ .....              | 458 |

|      |   |     |
|------|---|-----|
| 8595 | Tabelle 297: TAB_KON_817 – TUC_KON_293 „Verzeichnisabfrage abbrechen“ .....         | 459 |
| 8596 | Tabelle 298: TAB_KON_780 – Signaturverfahren Externe Authentisierung .....          | 460 |
| 8597 | Tabelle 299: TAB_KON_839 Basisdienst Authentifizierungsdienst .....                 | 461 |
| 8598 | Tabelle 300: TAB_KON_781 Operation ExternalAuthenticate.....                        | 462 |
| 8599 | Tabelle 301: TAB_KON_782 Ablauf Operation ExternalAuthenticate .....                | 465 |
| 8600 | Tabelle 302: TAB_KON_783 Übersicht Fehler Operation ExternalAuthenticate .....      | 465 |
| 8601 | Tabelle 303: TAB_KON_784 Privater Schlüssel je Karte für ExternalAuthenticate ..... | 465 |
| 8602 | Tabelle 304: TAB_KON_680 Mapping der Netzwerksegmente.....                          | 468 |
| 8603 | Tabelle 305: TAB_KON_681 Definition der vom Konnektor verwendeten VPN-Tunnel .      | 469 |
| 8604 | Tabelle 306: TAB_KON_682 Definition der Konnektor IP-Adressen.....                  | 469 |
| 8605 | Tabelle 307: TAB_KON_614 - TUC_KON_305 „LAN-Adapter initialisieren“ .....           | 480 |
| 8606 | Tabelle 308: TAB_KON_615 Fehlercodes TUC_KON_305 „LAN-Adapter initialisieren“ ..    | 481 |
| 8607 | Tabelle 309: TAB_KON_616 - TUC_KON_306 „WAN-Adapter initialisieren“ .....           | 481 |
| 8608 | Tabelle 310: TAB_KON_617 Fehlercodes TUC_KON_306 „WAN-Adapter initialisieren“.      | 482 |
| 8609 | Tabelle 311: TAB_KON_622 - TUC_KON_304 „Netzwerk-Routen einrichten“ .....           | 483 |
| 8610 | Tabelle 312: TAB_KON_623 Fehlercodes TUC_KON_304 „Netzwerk-Routen einrichten“       |     |
| 8611 | .....   | 485 |
| 8612 | Tabelle 313: TAB_KON_683 LAN-Adapter IP-Konfiguration .....                         | 486 |
| 8613 | Tabelle 314: TAB_KON_684 LAN-Adapter Erweiterte Parameter .....                     | 487 |
| 8614 | Tabelle 315: TAB_KON_685 WAN-Adapter IP-Konfiguration.....                          | 487 |
| 8615 | Tabelle 316: TAB_KON_686 WAN-Adapter Erweiterte Parameter .....                     | 488 |
| 8616 | Tabelle 317: TAB_KON_624 – „Konfigurationsparameter der Anbindung LAN/WAN“ ...      | 489 |
| 8617 | Tabelle 318: TAB_KON_625 - Konfigurationsparameter Firewall-Schnittstelle .....     | 492 |
| 8618 | Tabelle 319: TAB_KON_626 „Liefere Netzwerkinformationen über DHCP“ .....            | 493 |
| 8619 | Tabelle 320: TAB_KON_627 „Aktivierung des DHCP-Servers“ .....                       | 495 |
| 8620 | Tabelle 321: TAB_KON_628 „Basiskonfiguration des DHCP-Servers“ .....                | 495 |
| 8621 | Tabelle 322: TAB_KON_629 „Client-Gruppenspezifische Konfigurationsoptionen des      |     |
| 8622 | Konnektor-DHCP-Servers“ .....   | 495 |
| 8623 | Tabelle 323: TAB_KON_630 - TUC_KON_343 „Initialisierung DHCP-Server“ .....          | 498 |
| 8624 | Tabelle 324: TAB_KON_631 Fehlercodes TUC_KON_343 „Initialisierung DHCP-Server“      | 498 |
| 8625 | Tabelle 325: TAB_KON_632 – TUC_KON_341 „DHCP Informationen beziehen“ .....          | 499 |
| 8626 | Tabelle 326: TAB_KON_633 Fehlercodes TUC_KON_341 „DHCP-Informationen beziehen“      |     |
| 8627 | .....   | 500 |
| 8628 | Tabelle 327: TAB_KON_634 „Konfiguration des DHCP-Clients“ .....                     | 501 |
| 8629 | Tabelle 328: TAB_KON_635 – TUC_KON_321 „Verbindung zu dem VPN-Konzentrator der      |     |
| 8630 | TI aufbauen“ .....  | 503 |
| 8631 | Tabelle 329: TAB_KON_636 Fehlercodes TUC_KON_321 „Verbindung zu dem VPN-            |     |
| 8632 | Konzentrator der TI aufbauen“ .....   | 505 |

|      |   |     |
|------|---|-----|
| 8633 | Tabelle 330: TAB_KON_637 – TUC_KON_322 „Verbindung zu dem VPN-Konzentrator der    |     |
| 8634 | SIS aufbauen“ .....   | 506 |
| 8635 | Tabelle 331: TAB_KON_638 Fehlercodes TUC_KON_322 „Verbindung zu dem VPN-          |     |
| 8636 | Konzentrator der SIS aufbauen“ .....  | 507 |
| 8637 | Tabelle 332: TAB_KON_639 – Konfigurationsparameter VPN-Client .....               | 508 |
| 8638 | Tabelle 333: TAB_KON_640 Zustandswerte für Konnektor NTP-Server .....             | 511 |
| 8639 | Tabelle 334: TAB_KON_776 TUC_KON_351 „Liefere Systemzeit“ .....                   | 511 |
| 8640 | Tabelle 335: TAB_KON_641 Fehlercodes TUC_KON_351 „Liefere Systemzeit“ .....       | 512 |
| 8641 | Tabelle 336: TAB_KON_642 Operation sync_Time.....                                 | 512 |
| 8642 | Tabelle 337: TAB_KON_643 Konfiguration des Konnektor NTP-Servers .....            | 513 |
| 8643 | Tabelle 338: TAB_KON_730 Einsehbare Konfigurationsparameter des Konnektor NTP-    |     |
| 8644 | Servers.....  | 513 |
| 8645 | Tabelle 339: TAB_KON_644 – TUC_KON_352 „Initialisierung Zeitdienst“ .....         | 514 |
| 8646 | Tabelle 340: TAB_KON_645 Fehlercodes TUC_KON_352 „Initialisierung Zeitdienst“ ... | 515 |
| 8647 | Tabelle 341: TAB_KON_687 DNS-Forwards des DNS-Servers.....                        | 516 |
| 8648 | Tabelle 342: TAB_KON_646 – TUC_KON_361 „DNS-Namen auflösen“ .....                 | 517 |
| 8649 | Tabelle 343: TAB_KON_647 Fehlercodes TUC_KON_361 „DNS Namen auflösen“ .....       | 518 |
| 8650 | Tabelle 344: TAB_KON_646 – TUC_KON_361 „DNS-Namen auflösen“ .....                 | 518 |
| 8651 | Tabelle 345: TAB_KON_647 Fehlercodes TUC_KON_361 „DNS Namen auflösen“ .....       | 519 |
| 8652 | Tabelle 346: TAB_KON_648 – TUC_KON_362 „Liste der Dienste abrufen“ .....          | 520 |
| 8653 | Tabelle 347: TAB_KON_649 Fehlercodes TUC_KON_362 „Liste der Dienste abrufen“ ..   | 520 |
| 8654 | Tabelle 348: TAB_KON_650 - TUC_KON_363 „Dienstdetails abrufen“ .....              | 521 |
| 8655 | Tabelle 349: TAB_KON_651 Fehlercodes TUC_KON_363 „Dienstdetails abrufen“ .....    | 522 |
| 8656 | Tabelle 350: TAB_KON_652 Basisanwendung Namensdienst.....                         | 522 |
| 8657 | Tabelle 351: TAB_KON_653 Operation GetIPAddress.....                              | 522 |
| 8658 | Tabelle 352: TAB_KON_654 - Konfigurationsparameter Namensdienst .....             | 523 |
| 8659 | Tabelle 353: TAB_KON_731 Einsehbare Konfigurationsparameter Namensdienst .....    | 524 |
| 8660 | Tabelle 354: TAB_KON_655 Konfigurationen der Benutzerverwaltung (Super-           |     |
| 8661 | Administrator) .....  | 529 |
| 8662 | Tabelle 355: TAB_KON_656 Konfigurationen der Benutzerverwaltung .....             | 530 |
| 8663 | Tabelle 356: TAB_KON_657 Konfigurationsparameter des Konnektornamens .....        | 530 |
| 8664 | Tabelle 357: TAB_KON_833 Bezeichner für persistente Konfigurationsdaten für       |     |
| 8665 | Fachmodule.....   | 533 |
| 8666 | Tabelle 358: TAB_KON_658 Aktivieren/Deaktivieren von Leistungsumfängen .....      | 534 |
| 8667 | Tabelle 359: TAB_KON_659 Konnektor Standalone einsetzen .....                     | 535 |
| 8668 | Tabelle 360: TAB_KON_661 Konfigurationsparameter der Konnektorfreischaltung ..... | 536 |
| 8669 | Tabelle 361: TAB_KON_732 Einsehbare Konfigurationsparameter der                   |     |
| 8670 | Konnektorfreischaltung .....  | 537 |



|      |  |     |
|------|--|-----|
| 8671 | Tabelle 362: TAB_KON_662 Zustandswerte der Konnektorfreisaltung .....                | 537 |
| 8672 | Tabelle 363: TAB_KON_851 Einschränkung der Rechte des Remote-Administrators          |     |
| 8673 | (Blacklist) .....  | 541 |
| 8674 | Tabelle 364: TAB_KON_663 Konfigurationen des Remote Managements.....                 | 542 |
| 8675 | Tabelle 365: TAB_KON_664 – TUC_KON_280 „Konnektoraktualisierung durchführen“         | 545 |
| 8676 | Tabelle 366: TAB_KON_665 Fehlercodes TUC_KON_280 „Konnektoraktualisierung            |     |
| 8677 | durchführen“ .....   | 547 |
| 8678 | Tabelle 367: TAB_KON_666 – TUC_KON_281 „Kartenterminalaktualisierung anstoßen“       |     |
| 8679 | .....  | 549 |
| 8680 | Tabelle 368: TAB_KON_667 Fehlercodes TUC_KON_281 „Kartenterminalaktualisierung       |     |
| 8681 | anstoßen“ .....  | 551 |
| 8682 | Tabelle 369: TAB_KON_668 – TUC_KON_282 „UpdateInformationen beziehen“ .....          | 551 |
| 8683 | Tabelle 370: TAB_KON_669 Fehlercodes TUC_KON_282 „UpdateInformationen beziehen“      |     |
| 8684 | .....  | 553 |
| 8685 | Tabelle 371: TAB_KON_799 – TUC_KON_283 „Infrastruktur Konfiguration aktualisieren“   |     |
| 8686 | .....  | 553 |
| 8687 | Tabelle 372: Tab_Kon_726 Fehlercodes TUC_KON_283 „Infrastruktur Konfiguration        |     |
| 8688 | aktualisieren“ .....   | 557 |
| 8689 | Tabelle 373: TAB_KON_833 – TUC_KON_285 „UpdateInformationen für Fachmodul            |     |
| 8690 | beziehen“.....   | 557 |
| 8691 | Tabelle 374: TAB_KON_834 Fehlercodes TUC_KON_285 „UpdateInformationen für            |     |
| 8692 | Fachmodul beziehen“ .....  | 559 |
| 8693 | Tabelle 375: TAB_KON_835 – TUC_KON_286 „Paket für Fachmodul laden“.....              | 560 |
| 8694 | Tabelle 376: TAB_KON_836 Fehlercodes TUC_KON_286 „Paket für Fachmodul laden“         | 561 |
| 8695 | Tabelle 377: TAB_KON_864 – TUC_KON_284 „KSR-Client initialisieren“ .....             | 561 |
| 8696 | Tabelle 378: TAB_KON_822 Fehlercodes TUC_KON_284 „KSR-Client initialisieren“.....    | 562 |
| 8697 | Tabelle 379: TAB_KON_670 Konfigurationsparameter der Software-Aktualisierung ....    | 562 |
| 8698 | Tabelle 380: TAB_KON_820 Einsehbare Konfigurationsparameter der Software-            |     |
| 8699 | Aktualisierung.....  | 563 |
| 8700 | Tabelle 381: TAB_KON_671 Anforderungen Klima.....                                    | 570 |
| 8701 | Tabelle 382: TAB_KON_672 Anforderungen Vibration .....                               | 570 |
| 8702 | Tabelle 383: TAB_KON_672 Anforderungen Vibration .....                               | 571 |
| 8703 | Tabelle 384: TAB_KON_779 „Profilierung der Signaturformate“ .....                    | 611 |
| 8704 | Tabelle 385: TAB_KON_775 „Profilierung der Dokumentformate und Nachrichten“ .....    | 618 |
| 8705 | Tabelle 386: TAB_KON_777 Events Interne Mechanismen .....                            | 630 |
| 8706 | Tabelle 387: TAB_KON_711 Architektur der TI-Plattform, Berechtigt Fachmodule .....   | 648 |
| 8707 | Tabelle 388: TAB_KON_712 Architektur der TI-Plattform, Berechtigt Clientsysteme ...  | 653 |
| 8708 | Tabelle 389: TAB_KON_713 Architektur der TI-Plattform, Berechtigt eHealth-KT .....   | 655 |
| 8709 | Tabelle 390: TAB_KON_714 Architektur der TI-Plattform, Berechtigt Administrator .... | 655 |
| 8710 | Tabelle 391: Aufzähltypen .....  | 678 |

8711

8712 **5.5 Referenzierte Dokumente**8713 **5.5.1 Dokumente der gematik**

8714 Die nachfolgende Tabelle enthält die Bezeichnung der in dem vorliegenden Dokument  
 8715 referenzierten Dokumente der gematik zur Telematikinfrastruktur. Der mit der  
 8716 vorliegenden Version korrelierende Entwicklungsstand dieser Konzepte und  
 8717 Spezifikationen wird pro Release in einer Dokumentenlandkarte definiert, Version und  
 8718 Stand der referenzierten Dokumente sind daher in der nachfolgenden Tabelle nicht  
 8719 aufgeführt. Deren zu diesem Dokument passende jeweils gültige Versionsnummer sind in  
 8720 der aktuellsten, von der gematik veröffentlichten Dokumentenlandkarte enthalten, in der  
 8721 die vorliegende Version aufgeführt wird.

| [Quelle]                  | Herausgeber: Titel  |
|---------------------------|---|
| [gemGlossar]              | gematik: Glossar  |
| [gemKPT_Arch_TIP]         | gematik: Konzept Architektur der TI-Plattform   |
| [gemKPT_Sich_Kon]         | gematik: Sicherheitskonzept Konnektor   |
| [gemKPT_Test]             | gematik: Testkonzept  |
| [gemSpec_COS]             | gematik: Spezifikation des Card Operating System (COS) – Elektrische Schnittstelle  |
| [gemSpec_Karten_Fach_TIP] | gematik: Befüllvorschriften für die Plattformanteile der Karten der TI  |
| [gemSpec_Kon_SigProxy]    | gematik: Spezifikation Konnektor Signaturproxy  |
| [gemSpec_Kon_TBAuth]      | gematik: Spezifikation Konnektor Basisdienst tokenbasierte Authentisierung  |
| [gemSpec_eGK_ObjSys]      | gematik: Spezifikation der elektronischen Gesundheitskarte eGK-Objektsystem - für Karten der Generation 2   |
| [gemSpec_eGK_ObjSys_G2.1] | gematik: Spezifikation der elektronischen Gesundheitskarte eGK-Objektsystem - für Karten der Generation 2.1   |
| [gemSpec_eGK_P1]          | gematik: Die Spezifikation der elektronische Gesundheitskarte; Teil 1 – Spezifikation der elektrischen Schnittstelle - für Karten der Generation 1+ |
| [gemSpec_eGK_P2]          | gematik: Die Spezifikation der elektronische Gesundheitskarte; Teil 2 – Grundlegende Applikationen - für Karten der Generation 1+                   |
| [gemSpec_gSMC-K_ObjSys]   | gematik: Spezifikation der gSMC-K Objektsystem  |
| [gemSpec_gSMC-KT_ObjSys]  | gematik: Spezifikation gSMC-KT Objektsystem   |
| [gemSpec_HBA_ObjSys]      | gematik: Spezifikation HBA Objektsystem   |
| [gemSpec_Krypt]           | gematik: Übergreifende Spezifikation Verwendung kryptographischer Algorithmen in der Telematikinfrastruktur   |

|                        |   |
|------------------------|---|
| [gemSpec_KSR]          | gematik: Spezifikation Konfigurationsdienst   |
| [gemSpec_KT]           | gematik: Spezifikation eHealth-Kartenterminal   |
| [gemSpec_Net]          | gematik: Spezifikation Netzwerk   |
| [gemSpec_OID]          | gematik: Spezifikation OID  |
| [gemSpec_OM]           | gematik: Übergreifende Spezifikation Operations und Maintenance   |
| [gemSpec_Perf]         | gematik: Performance und Mengengerüst TI-Plattform  |
| [gemSpec_PKI]          | gematik: Spezifikation PKI  |
| [gemSpec_SMC-B_ObjSys] | gematik: Spezifikation SMC-B Objektsystem   |
| [gemSpec_VPN_ZugD]     | gematik: Spezifikation VPN-Zugangsdienst  |
| [gemSpec_VZD]          | gematik: Spezifikation Verzeichnisdienst  |
| gemGitHub_tslSig       | <a href="https://github.com/gematik/examples-TelematikInterfaces/tree/master/tslService/detachedSignature">https://github.com/gematik/examples-TelematikInterfaces/tree/master/tslService/detachedSignature</a> |

8722

## 5.5.2 Weitere Dokumente

| [Quelle]           | Herausgeber (Erscheinungsdatum): Titel   |
|--------------------|--|
| [7816-4]           | ISO/IEC 7816-4: 2005 (2nd edition) Identification cards — Integrated circuit cards - Part 4: Organization, security and commands for interchange   |
| [ALGCAT]           | Bekanntmachung zur elektronischen Signatur nach dem Signaturgesetz und der Signaturverordnung (Übersicht über geeignete Algorithmen), Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahnen, vom 11.12.2015 (auch online verfügbar: <a href="https://www.bundesanzeiger.de">https://www.bundesanzeiger.de</a> mit dem Suchbegriff „BAnz AT 01.02.2016 B5“). |
| [Basic Profile1.2] | Basic Profile Version 1.2<br><a href="http://www.ws-i.org/Profiles/BasicProfile-1.2-2010-11-09.html">http://www.ws-i.org/Profiles/BasicProfile-1.2-2010-11-09.html</a>   |
| [Basic Profile2.0] | Basic Profile Version 2.0<br><a href="http://www.ws-i.org/Profiles/BasicProfile-2.0-2010-11-09.html">http://www.ws-i.org/Profiles/BasicProfile-2.0-2010-11-09.html</a>   |
| [BSI_GK]           | BSI:<br>IT-Grundschutz-Kataloge (15. Ergänzungslieferung 2016)<br><a href="https://download.gsb.bund.de/BSI/ITGSK/IT-Grundschutz-Kataloge_2016_EL15_DE.pdf">https://download.gsb.bund.de/BSI/ITGSK/IT-Grundschutz-Kataloge_2016_EL15_DE.pdf</a>  |
| [BSI-TR-03111]     | Technical Guideline BSI TR-03111<br>Elliptic Curve Cryptography, Version 2.10, Date: 2018-06-01<br><a href="https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/Tec">https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Publications/Tec</a>   |

|                |  |
|----------------|--|
|                | <a href="https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03111/BSI-TR-03111_V-2-1_pdf.pdf?__blob=publicationFile&amp;v=2">hGuidelines/TR03111/BSI-TR-03111_V-2-1_pdf.pdf?__blob=publicationFile&amp;v=2</a>  |
| [BSI-TR03114]  | BSI (22.10.2007): Technische Richtlinie – Stapelsignatur mit dem Heilberufsausweis; Version 2.0<br><a href="https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03114/BSI-TR-03114.pdf?__blob=publicationFile&amp;v=1">https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03114/BSI-TR-03114.pdf?__blob=publicationFile&amp;v=1</a>   |
| [BSI TR-03120] | BSI (23.10.2007): BSI - Technische Richtlinie – Sichere Kartenterminalidentität (Betriebskonzept); Version 1.0<br><a href="https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03120/BSI-TR-03120.pdf?__blob=publicationFile&amp;v=1">https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03120/BSI-TR-03120.pdf?__blob=publicationFile&amp;v=1</a>  |
| [CADES]        | ETSI: Electronic Signature Formats, Electronic Signatures and Infrastructures (ESI) – Technical Specification, ETSI TS 101 733 V2.2.1, via <a href="http://www.etsi.org">http://www.etsi.org</a>   |
| [Canon XML1.1] | Canonical XML Version 1.1<br><a href="http://www.w3.org/TR/2008/REC-xml-c14n11-20080502/">http://www.w3.org/TR/2008/REC-xml-c14n11-20080502/</a>   |
| [CDA]          | ISO/HL7 27932:2009 Data Exchange Standards -- HL7 Clinical Document Architecture, Release 2  |
| [CDA-Sig]      | Erstellung von XML-Signaturen für Dokumente nach Clinical Documents Architecture – R2, Elektronische Signatur von Arztbriefen, Ärztekammern in NRW im Auftrag der Bundesärztekammer, Version 1.6 vom 19.04.2010  |
| [COMMON_PKI]   | Common PKI Specifications for Interoperable Applications<br>Version 2.0, 20 January 2009<br><a href="http://www.t7ev.org/themen/entwickler/common-pki-v20-spezifikation.html">http://www.t7ev.org/themen/entwickler/common-pki-v20-spezifikation.html</a><br>ISIS-MTT Core Specification, 2004, Version 1.1<br><a href="https://www.teletrust.de/fileadmin/files/ISIS-MTT_Profile_SigGOptions_v1.1.pdf">https://www.teletrust.de/fileadmin/files/ISIS-MTT_Profile_SigGOptions_v1.1.pdf</a> |
| [CMS]          | Cryptographic Message Syntax (CMS), September 2009<br><a href="http://tools.ietf.org/html/rfc5652">http://tools.ietf.org/html/rfc5652</a>  |
| [DIN 66003]    | DIN 66003:1999<br>Informationsverarbeitung; 7-Bit-Code   |
| [HPC-P1]       | Spezifikation des elektronischen Heilberufsausweises<br>Version 2.3.2, 05.08.2009, Teil I: Kommandos, Algorithmen und Funktionen der COS Plattform   |
| [HPC-P2]       | Spezifikation des elektronischen Heilberufsausweises<br>Version 2.3.2, 05.08.2009, Teil II: HPC - Anwendungen und Funktionen   |

|              |   |
|--------------|---|
| [HPC-P3]     | Spezifikation des elektronischen Heilberufsausweises<br>Version 2.3.2, 05.08.2009, Teil III: SMC - Anwendungen und Funktionen   |
| [HüKo06]     | BSI (2006):<br>Hühnlein, Detlef/Korte, Ulrike: Grundlagen der elektronischen Signatur   |
| [IEEE 802.3] | Technical Committee Computer Communications of the IEEE Computer Society, USA (1985):<br>IEEE standards for local area networks: carrier sense multiple access with collision detection (CSMA/CD) access method and physical layer specifications<br>ISBN: 0-7381-4253-0  |
| [ISO 8601]   | International Organization for Standardization (2006-09): Data elements and interchange formats -- Information interchange -- Representation of dates and times   |
| [KVK]        | Spitzenverbände der Krankenkassen, Kassenärztliche Bundesvereinigung und Kassenzahnärztlichen Bundesvereinigung (gültig ab 25. November 2009):<br>Technische Spezifikation der Versichertenkarte Version: 2.08  |
| [MIME]       | <a href="#">RFC 2045</a> , <a href="#">RFC 2046</a> , <a href="#">RFC 2047</a> , <a href="#">RFC 2048</a> , <a href="#">RFC 2049</a>  |
| [NTPv4]      | Internet Engineering Task Force (IETF) (06/2010): Network Time Protocol Version 4: Protocol and Algorithms Specification<br><a href="http://www.ietf.org/rfc/rfc5905.txt">http://www.ietf.org/rfc/rfc5905.txt</a>   |
| [OASIS-AdES] | OASIS: Advanced Electronic Signature Profiles of the OASIS Digital Signature Service Version 1.0, OASIS Standard,<br><a href="http://docs.oasis-open.org/dss/v1.0/oasis-dss-profiles-AdES-spec-v1.0-os.pdf">http://docs.oasis-open.org/dss/v1.0/oasis-dss-profiles-AdES-spec-v1.0-os.pdf</a>  |
| [OASIS-DSS]  | OASIS: Digital Signature Service Core Protocols, Elements, and Bindings, Version 1.0, OASIS Standard, via<br><a href="http://docs.oasis-open.org/dss/v1.0/oasis-dss-core-spec-v1.0-os.pdf">http://docs.oasis-open.org/dss/v1.0/oasis-dss-core-spec-v1.0-os.pdf</a>  |
| [OASIS-SP]   | OASIS: Signature Policy Profile of the OASIS Digital Signature Services Version 1.0, Committee Draft 01, 18 May 2009, <a href="http://docs.oasis-open.org/dss-x/profiles/sigpolicy/oasis-dssx-1.0-profiles-sigpolicy-cd01.pdf">http://docs.oasis-open.org/dss-x/profiles/sigpolicy/oasis-dssx-1.0-profiles-sigpolicy-cd01.pdf</a>   |
| [OASIS-VR]   | OASIS: Profile for comprehensive multi-signature verification reports for OASIS Digital Signature Services Version 1.0, Committee Specification 01, 12 November 2010, <a href="http://docs.oasis-open.org/dss-x/profiles/verificationreport/oasis-dssx-1.0-profiles-vr-cs01.pdf">http://docs.oasis-open.org/dss-x/profiles/verificationreport/oasis-dssx-1.0-profiles-vr-cs01.pdf</a> |

|             |  |
|-------------|--|
| [PAdES-1]   | European Telecommunications Standards Institute (ETSI): Electronic Signatures and Infrastructures (ESI); PDF Advanced Electronic Signature Profiles; Part 1: PAdES Overview – a framework document for PAdES, ETSI TS 102 778-1 V1.1.1, Technical Specification, 2009    |
| [PAdES-3]   | European Telecommunications Standards Institute (ETSI): Electronic Signatures and Infrastructures (ESI); PDF Advanced Electronic Signature Profiles; Part 3: PAdES Enhanced – PAdES-BES and PAdES-EPES Profiles, ETSI TS 102 778-3 V1.1.2, Technical Specification, 2009 |
| [PAdES-4]   | European Telecommunications Standards Institute (ETSI): Electronic Signatures and Infrastructures (ESI); PDF Advanced Electronic Signature Profiles; Part 4: PAdES Long Term – PAdES-LTV Profile, ETSI TS 102 778-4 V1.1.2, Technical Specification, 2009                |
| [ISO 19005] | ISO 19005 – Document management – Electronic document file format for long-term preservation   |
| [PDF/A-2]   | ISO 19005-2:2011 – Document management – Electronic document file format for long-term preservation – Part 2: Use of ISO 32000-1 (PDF/A-2)   |
| [PP_NK]     | Common Criteria Schutzprofil (Protection Profile)<br>Schutzprofil 1: Anforderungen an den Netzkonnektor<br>BSI-CC-PP-0097  |
| [PP_KON]    | Common Criteria Schutzprofil (Protection Profile)<br>Schutzprofil 2: Anforderungen an den Konnektor:<br>BSI-CC-PP-0098   |
| [RFC792]    | IETF (September 1981) INTERNET CONTROL MESSAGE PROTOCOL<br><a href="http://tools.ietf.org/html/rfc792">http://tools.ietf.org/html/rfc792</a>   |
| [RFC1034]   | RFC 1034 (November 1987): Domain Names – Concepts and Facilities<br><a href="http://tools.ietf.org/html/rfc1034">http://tools.ietf.org/html/rfc1034</a>  |
| [RFC1122]   | RFC 1122 (Oktober 1989): Requirements for Internet Hosts --<br>Communication Layers<br><a href="http://tools.ietf.org/html/rfc1122">http://tools.ietf.org/html/rfc1122</a>   |
| [RFC1812]   | F. Baker (ed.): Requirements for IP Version 4 Routers, IETF RFC 1812,<br><a href="http://www.ietf.org/rfc/rfc1812.txt">http://www.ietf.org/rfc/rfc1812.txt</a>   |
| [RFC1918]   | RFC1918 (Februar 1996): Address Allocation for Private Internets<br><a href="http://tools.ietf.org/html/rfc1918">http://tools.ietf.org/html/rfc1918</a>  |
| [RFC2119]   | RFC 2119 (März 1997): Key words for use in RFCs to Indicate<br>Requirement Levels S. Bradner,<br><a href="http://tools.ietf.org/html/rfc2119">http://tools.ietf.org/html/rfc2119</a>   |

|           |   |
|-----------|---|
| [RFC2131] | Network Working Group (03/1997): Dynamic Host Configuration Protocol<br><a href="http://www.ietf.org/rfc/rfc2131.txt">http://www.ietf.org/rfc/rfc2131.txt</a>   |
| [RFC2132] | Network Working Group (03/1997): DHCP Options and BOOTP Vendor Extensions<br><a href="http://www.ietf.org/rfc/rfc2132.txt">http://www.ietf.org/rfc/rfc2132.txt</a>  |
| [RFC2617] | Network Working Group (06/1999): HTTP Authentication: Basic and Digest Access Authentication<br><a href="http://www.ietf.org/rfc/rfc2617.txt">http://www.ietf.org/rfc/rfc2617.txt</a>   |
| [RFC2818] | Network Working Group (05/2000): HTTP Over TLS<br><a href="http://www.ietf.org/rfc/rfc2818.txt">http://www.ietf.org/rfc/rfc2818.txt</a>   |
| [RFC3447] | B. Kaliski: PKCS #1: RSA Encryption, Version 2.1, RFC3447,  |
| [RFC2616] | Network Working Group (06/1999): Hypertext Transfer Protocol -- HTTP/1.1<br><a href="http://www.ietf.org/rfc/rfc2616.txt">http://www.ietf.org/rfc/rfc2616.txt</a>   |
| [RFC2644] | D. Senie: <i>Changing the Default for Directed Broadcasts in Routers</i> , IETF RFC 2644, <a href="http://www.ietf.org/rfc/rfc2644.txt">http://www.ietf.org/rfc/rfc2644.txt</a>   |
| [RFC2663] | P. Srisuresh, M. Holdrege: <i>IP Network Address Translator (NAT) Terminology and Considerations</i> , IETF RFC 2663, <a href="http://www.ietf.org/rfc/rfc2663.txt">http://www.ietf.org/rfc/rfc2663.txt</a>   |
| [RFC3022] | RFC 3022 (Januar 2001): Traditional IP Network Address Translator (Traditional NAT)<br><a href="http://tools.ietf.org/html/rfc3022">http://tools.ietf.org/html/rfc3022</a>  |
| [RFC3275] | D. Eastlake, J. Reagle, D. Solo: <i>(Extensible Markup Language) XML Signature Syntax and Processing</i> , IETF RFC 3275, via <a href="http://www.ietf.org/rfc/rfc3275.txt">http://www.ietf.org/rfc/rfc3275.txt</a>   |
| [RFC3279] | W. Polk, R. Hously, L. Bassham: <i>Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile</i> , IETF RFC 3279, <a href="http://www.ietf.org/rfc/rfc3279.txt">http://www.ietf.org/rfc/rfc3279.txt</a> |
| [RFC3629] | Network Working Group (11/2003): UTF-8, a transformation format of ISO 10646<br><a href="http://www.ietf.org/rfc/rfc3629.txt">http://www.ietf.org/rfc/rfc3629.txt</a>   |
| [RFC3927] | Network Working Group (05/2005): Dynamic Configuration of IPv4 Link-Local Addresses<br><a href="http://www.ietf.org/rfc/rfc3927.txt">http://www.ietf.org/rfc/rfc3927.txt</a>  |
| [RFC3986] | Network Working Group (01/2005): Uniform Resource Identifier (URI): Generic Syntax  |



|            |   |
|------------|---|
| [RFC4122]  | RFC 4122 (July 2005): A Universelly Unique Identifier UUID URN Namespace<br><a href="http://tools.ietf.org/html/rfc4122">http://tools.ietf.org/html/rfc4122</a>   |
| [RFC4632]  | Network Working Group (08/2006): Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan<br><a href="http://tools.ietf.org/html/rfc4632">http://tools.ietf.org/html/rfc4632</a>   |
| [RFC5246]  | RFC 5246 (August 2008): The Transport Layer Security (TLS) Protocol Version 1.2;<br><a href="http://tools.ietf.org/html/rfc5246">http://tools.ietf.org/html/rfc5246</a>   |
| [RFC5652]  | R. Housley:<br>Cryptographic Message Syntax (CMS), RFC 5652 (September 2009)<br><a href="http://tools.ietf.org/html/rfc5652">http://tools.ietf.org/html/rfc5652</a>   |
| [RFC 6598] | RFC 6598 (April 2012): IANA-Reserved IPv4 Prefix for Shared Address Space<br><a href="http://tools.ietf.org/html/rfc6598">http://tools.ietf.org/html/rfc6598</a>  |
| [RFC6931]  | RFC 6931 (April 2013): Additional XML Security Uniform Resource Identifiers (URIs)<br><a href="http://tools.ietf.org/html/rfc6931">http://tools.ietf.org/html/rfc6931</a>   |
| [RFC7159]  | RFC 7159 (March 2014): The JavaScript Object Notation (JSON) Data Interchange Format<br><a href="http://tools.ietf.org/html/rfc7159">http://tools.ietf.org/html/rfc7159</a>   |
| [S/MIME]   | RFC 5751 (Januar 2010): Secure/Multipurpose Internet Mail Extensions (S/MIME), Version 3.2, Message Specification<br><a href="http://www.ietf.org/rfc/rfc5751.txt">http://www.ietf.org/rfc/rfc5751.txt</a>  |
| [SOAP1.1]  | Simple Object Access Protocol (SOAP) 1.1<br>W3C Note (08 May 2000)<br><a href="https://www.w3.org/TR/2000/NOTE-SOAP-20000508/">https://www.w3.org/TR/2000/NOTE-SOAP-20000508/</a>   |
| [SOAP1.2]  | SOAP Version 1.2 Part 1: Messaging Framework (Second Edition)<br>W3C Recommendation (27 April 2007)<br><a href="http://www.w3.org/TR/2007/REC-soap12-part1-20070427/">http://www.w3.org/TR/2007/REC-soap12-part1-20070427/</a>  |
| [SICCT]    | TeleTrust (17.12.2010): SICCT Secure Interoperable ChipCard Terminal, Version 1.21<br><a href="https://www.teletrust.de/fileadmin/docs/projekte/sicct/SICCT-Spezifikation-1.21.pdf">https://www.teletrust.de/fileadmin/docs/projekte/sicct/SICCT-Spezifikation-1.21.pdf</a> |
| [TIFF6]    | TIFF Revision 6.0 (Final, June 3, 1992)<br><a href="https://www.adobe.io/open/standards/TIFF/_jcr_content/contentbody/download/file.res/TIFF6.pdf.html">https://www.adobe.io/open/standards/TIFF/_jcr_content/contentbody/download/file.res/TIFF6.pdf.html</a>              |



|                                |   |
|--------------------------------|---|
| [WSDL1.1]                      | W3C Note (15.03.2001):<br>Web Services Description Language (WSDL) 1.1<br><a href="http://www.w3.org/TR/wsdl">http://www.w3.org/TR/wsdl</a>   |
| [XAdES]                        | European Telecommunications Standards Institute (ETSI): Technical Specification XML Advanced Electronic Signatures (XAdES). ETSI Technical Specification TS 101 903, Version 1.4.2, 2010                        |
| [XMLDSig]                      | W3C Recommendation (06.2008):<br>XML-Signature Syntax and Processing<br><a href="http://www.w3.org/TR/2008/REC-xmlsig-core-20080610/">http://www.w3.org/TR/2008/REC-xmlsig-core-20080610/</a>                   |
| [XMLEnc]                       | XML Encryption Syntax and Processing<br>W3C Recommendation 11 April 2013<br><a href="http://www.w3.org/TR/xmlenc-core1/">http://www.w3.org/TR/xmlenc-core1/</a>   |
| [XPath]                        | W3C Recommendation (14 December 2010)<br>XML Path Language (XPath) 2.0 (Second Edition)<br><a href="http://www.w3.org/TR/2010/REC-xpath20-20101214/">http://www.w3.org/TR/2010/REC-xpath20-20101214/</a>        |
| [XSLT]                         | W3C Recommendation (23 January 2007)<br>XSL Transformations (XSLT) Version 2.0<br><a href="http://www.w3.org/TR/2007/REC-xslt20-20070123/">http://www.w3.org/TR/2007/REC-xslt20-20070123/</a>                   |
| [XAdES<br>Baseline<br>Profile] | European Telecommunications Standards Institute (ETSI): Electronic Signatures and Infrastructure (ESI); XAdES Baseline Profile; ETSI Technical Specification TS 103 171, Version 2.1.1, 2012-03                 |
| [CADES<br>Baseline<br>Profile] | European Telecommunications Standards Institute (ETSI): Electronic Signatures and Infrastructure (ESI); CADES Baseline Profile; ETSI Technical Specification TS 103 173, Version 2.1.1, 2013-04                 |
| [PADES<br>Baseline<br>Profile] | European Telecommunications Standards Institute (ETSI): Electronic Signatures and Infrastructure (ESI); PADES Baseline Profile; ETSI Technical Specification TS 103 172, Version 2.2.2, (2013-04)               |
| [XSL]                          | W3C Recommendation (05.12.2006):<br>Extensible Stylesheet language (XSL) Version 1.1<br><a href="http://www.w3.org/TR/2006/REC-xsl11-20061205/">http://www.w3.org/TR/2006/REC-xsl11-20061205/</a>               |
| [MTOM]                         | SOAP Message Transmission Optimization Mechanism<br>W3C Recommendation 25 January 2005<br><a href="http://www.w3.org/TR/2005/REC-soap12-mtom-20050125/">http://www.w3.org/TR/2005/REC-soap12-mtom-20050125/</a> |
| [MTOM-<br>SOAP1.1]             | W3C Member Submission 05 April 2006<br>SOAP 1.1 Binding for MTOM 1.0<br><a href="https://www.w3.org/Submission/soap11mtom10/">https://www.w3.org/Submission/soap11mtom10/</a>                                   |
| [WS-MTOM<br>Policy]            | W3C Member Submission 18 November 2007<br>MTOM Serialization Policy Assertion 1.1   |

|          |  |
|----------|--|
| [COS-G2] | Common Criteria Protection Profile, Card Operating System Generation 2, (PP COS G2), BSI-CC-PP-0082-V2 |
|----------|--|

8723  
8724

ENTWURF

8725  
8726

## 6 Anhang B – Profilierung der Signatur- und Verschlüsselungsformate (normativ)

### 8727 6.1 Profilierung der Verschlüsselungsformate

### 8728 6.2 Profilierung der Signaturformate

8729 Tabelle 384: TAB\_KON\_779 „Profilierung der Signaturformate“

| Aspekt<br>(QES/nonQES)                         | Festlegung<br>(XML-Signatur/CMS-Signatur/PDF-Signatur)   |
|--|--|
| <b>Zertifikatsreferenz</b><br>(QES und nonQES) | <p><u>XML-Signatur</u><br/>Bei der Signaturerstellung ist das XML-Element <code>SigningCertificate</code> gemäß den Vorgaben aus XAdES Kapitel 7.2.2 „The SigningCertificate element“ anzulegen. Bei der Signaturprüfung ist es gemäß XAdES Kapitel G.2.2.5 „Verification technical rules“ [XAdES] zu prüfen. Grundsätzlich sind auch Signaturen zu prüfen, die keine Zertifikatsreferenz enthalten. Das Prüfergebnis muss dann widerspiegeln, dass diese Sicherheitsfunktion nicht enthalten war.</p> <p><u>CMS-Signatur</u><br/>Bei der Signaturerstellung ist das Attribut <code>signing certificate reference</code> gemäß CAAdES Kapitel 5.7.3 „Signing Certificate Reference Attributes“ [CAAdES] anzulegen. Bei der Signaturprüfung ist es gemäß CAAdES Kapitel 5.6.3 „Message signature verification process“ [CAAdES] zu prüfen. Grundsätzlich sind auch Signaturen zu prüfen, die keine Zertifikatsreferenz enthalten. Das Prüfergebnis muss dann widerspiegeln, dass diese Sicherheitsfunktion nicht enthalten war.</p> <p><u>PDF-Signatur</u><br/>Bei der Signaturerstellung ist das Attribut <code>signing certificate reference</code> gemäß den Vorgaben aus [PAdES-3] Kapitel 4.4.3 „Signing Certificate Reference Attribute“ anzulegen. Bei der Signaturprüfung ist es gemäß [PAdES-3] Kapitel 4.6.1 „Signing Certificate Reference Validation“ zu prüfen. Grundsätzlich sind auch Signaturen zu prüfen, die keine Zertifikatsreferenz enthalten. Das Prüfergebnis muss dann widerspiegeln, dass diese Sicherheitsfunktion nicht enthalten war.</p> |

|  |   |
|--|---|
| <b>Signaturablage</b>  | <u>PDF-Signatur</u><br>Sie Signatur wird als Incremental Update gemäß [PDF/A-2] Kapitel 7.5.6 an das Dokument angefügt.   |
| <b>Parallelsignatur</b><br>(QES und nonQES)                    | <u>XML-Signatur</u><br>Parallele Signaturen werden durch je ein <code>ds:signature</code> -Element pro Signatur abgebildet. Für die Signaturvariante „enveloping“ werden parallele Signaturen nicht angeboten.<br><u>CMS-Signatur:</u><br>Parallele Signaturen werden durch je einen <code>SignerInfo</code> -Container pro Signatur realisiert.<br><u>PDF-Signatur:</u><br>Parallele Signaturen werden nicht angeboten.  |
| <b>Dokumentexkludierende Gegensignatur</b><br>(QES und nonQES) | <u>XML-Signatur</u><br>Die Implementierung erfolgt mittels Countersignature gemäß [XAdES], Kapitel 7.2.4. Jede vorhandene Parallel-Signatur wird gegensigniert.<br><u>CMS-Signatur:</u><br>Die Implementierung erfolgt mittels der Countersignature gemäß CMS-Spezifikation [RFC5652]. Jede vorhandene Parallel-Signatur wird gegensigniert.<br><u>PDF-Signatur:</u><br>Dokumentexkludierende Gegensignaturen werden nicht angeboten.   |
| <b>Referenzierung</b><br>(QES und nonQES)                      | <u>XML-Signatur</u><br>Bei der Signaturerzeugung verwendet der Konnektor in der Signatur nur ID-basierte Referenzen. Bei der Signaturprüfung reagiert der Konnektor bei Abweichungen hiervon mit Fehler 4208.   |
|  | <u>XML-Signatur / CMS-Signatur</u><br>Bei XML-Dokumenten und XML-Signaturen kann die Referenzierung von Objekten auf zwei Arten erfolgen: <ul style="list-style-type: none"> <li>– Ist kein XML-Schema vorhanden, so werden die Werte des ID-Attributs des referenzierten Elements verwendet.</li> <li>– Wird ein XML-Schema übergeben, so muss dieses die ID-Attribute zur Referenzierung festlegen.</li> </ul> Bei Abweichungen reagiert der Konnektor mit Fehlercode 4115. |
| <b>Anzahl unterstützter Signaturen</b><br>(QES und nonQES)     | <u>XML-Signatur / CMS-Signatur / PDF-Signatur</u><br>Es müssen mindestens 20 Signaturen pro Dokument unterstützt werden. Sind mehr als die unterstützte Anzahl von Signaturen in einem Dokument enthalten, wird die Operation mit Fehler 4001 abgebrochen.  |

## 6.3 Profilierung VerificationReport

### Anforderung eines ausführlichen Prüfberichts

Folgende Aufrufparameter müssen unterstützt werden:

```
<ReturnVerificationReport
  xmlns="urn:oasis:names:tc:dss-x:1.0:profiles:verificationreport:schema#"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="urn:oasis:names:tc:dss-x:1.0:profiles:verificationreport:schema#
    oasis-dssx-1.0-profiles-vr-cd1.xsd">
  <IncludeVerifier>false</IncludeVerifier>
  <IncludeCertificateValues>true</IncludeCertificateValues>
  <IncludeRevocationValues>true</IncludeRevocationValues>
  <ExpandBinaryValues>false</ExpandBinaryValues>
  <ReportDetailLevel>
    urn:oasis:names:tc:dss-
    x:1.0:profiles:verificationreport:reportdetail:allDetails
  </ReportDetailLevel>
</ReturnVerificationReport>
```

### Verwendung des erzeugten VerificationReport

Für die folgenden Inhalte müssen die angegebenen Strukturen benutzt werden. Im Standard angegebene Pflichtfelder von erzeugten Strukturen müssen ggf. zusätzlich gefüllt werden:

1. Prüfzeitpunkt (Systemzeit des Konnektors zum Zeitpunkt der Prüfung)

```
/VerificationReport/
  dss:VerificationTimeInfo/
    dss:VerificationTime
```

2. Signaturzeitpunkt(Ermittelter\_Signaturzeitpunkt\_Eingebettet)

```
/VerificationReport/
  IndividualReport/
    SignedObjectIdentifier/
      SignedProperties/
        SignedSignatureProperties/
          XAdES:SigningTime
```

Die Signierzeit SigningTime ist nicht nur für XAdES-Signaturen, sondern allgemein für Signaturen gemäß AdES-Baseline-Profilierung, also auch für CADES und PAdES zu füllen.

- 8767 3. Angenommener Signaturzeitpunkt gemäß TIP1-A\_5540 (QES) und TIP1-A\_5545  
8768 (nonQES)
- 8769 /VerificationReport/  
8770 IndividualReport/  
8771 Details/  
8772 dss:VerificationTimeInfo/  
8773 dss:VerificationTime  
8774
- 8775 4. der binäre Wert der Signatur
- 8776 /VerificationReport/  
8777 IndividualReport/  
8778 SignedObjectIdentifier/  
8779 SignatureValue
- 8780 5. Kurztext
- 8781 Der signierte Kurztext wird in folgendem XML-Element zurückgegeben:
- 8782 /VerificationReport/  
8783 IndividualReport/  
8784 SignedObjectIdentifier/  
8785 SignedProperties/  
8786 Other/  
8787 SIG:ShortText  
8788
- 8789 6. Das folgende Element mit den Werten true/false gibt an, ob eine  
8790 Zertifikatsreferenz gemäß Anhang B2 vorhanden ist (true) oder nicht (false):
- 8791 /VerificationReport/  
8792 IndividualReport/  
8793 SignedObjectIdentifier/  
8794 SignedProperties/  
8795 Other/  
8796 SIG:ReferenceToSignerCertificate
- 8797 7. Sämtliche signierte Attribute, deren Rückgabe nicht explizit über andere Elemente  
8798 geregelt ist, werden als direkt anzeigbare Key/Value-Paare zurückgeben. Dabei  
8799 sind sowohl Key und Value bereits für die Anzeige formatiert. Der Key wird in  
8800 einer Zeile dargestellt. Der Value wird in mehreren Zeilen dargestellt, wobei ein  
8801 Zeilenumbruch durch 'CARRIAGE RETURN (CR)' 'LINE FEED (LF)' erzeugt wird und  
8802 keine weiteren Steuerzeichen erlaubt sind.
- 8803 /VerificationReport/  
8804 IndividualReport/  
8805 SignedObjectIdentifier/  
8806 SignedProperties/

8807 Other/  
 8808 SIG:DisplayableAttributes

8809 8. das Ergebnis der Signaturprüfung  
 8810 /VerificationReport/  
 8811 IndividualReport/  
 8812 Result

8813 9. handelt es sich bei der Signatur um eine Gegensignatur wird diese als solche  
 8814 markiert  
 8815  
 8816 /DetailedSignatureReport/  
 8817 Properties/  
 8818 UnsignedProperties/  
 8819 Other/  
 8820 SIG:CounterSignatureMarker

8821 und mit  
 8822  
 8823 /DetailedSignatureReport/  
 8824 Properties/  
 8825 UnsignedProperties/  
 8826 Other/  
 8827 SIG:CounterSignatureMarker/  
 8828 SignatureValueReference/  
 8829 @IdRef  
 8830  
 8831

8832 auf jede (eine oder mehrere) gegensignierte Signaturen verwiesen. Dabei  
 8833 zeigt IdRef auf den jeweiligen gegensignierten Signaturwert  
 8834  
 8835 /VerificationReport/  
 8836 IndividualReport/  
 8837 SignedObjectIdentifier/  
 8838 ds:SignatureValue/  
 8839 @Id

8840 10. das Ergebnis der Zertifikatsprüfung,  
 8841 /VerificationReport/  
 8842 IndividualReport/  
 8843 Details/  
 8844 DetailedSignatureReport/  
 8845 CertificatePathValidity/  
 8846 PathValiditySummary/

8847 ResultMajor

8848 11. Inhalt des Zertifikates, auf dem beruhend signiert wurde

8849 /VerificationReport/

8850 IndividualReport/

8851 Details/

8852 DetailedSignatureReport/

8853 CertificatePathValidity/

8854 PathValidityDetail/

8855 CertificateValidity/

8856 CertificateValue

8857 12. den Signaturalgorithmus der Dokumentensignatur (URI, angelehnt an den

8858 Wertebereich des Feldes ds:SignatureMethod),

8859 /VerificationReport/

8860 IndividualReport/

8861 Details/

8862 DetailedSignatureReport/

8863 SignatureOK/

8864 SignatureAlgorithm

8865

8866 13. aussagekräftiger Hinweis zum verminderten Beweiswert hinsichtlich Authentizität

8867 und Integrität der Signatur, wenn einer der bei der Signaturprüfung identifizierten

8868 und unterstützten Algorithmen zum Zeitpunkt der Signaturprüfung nicht mehr laut

8869 Algorithmenkatalog [ALGCAT] als geeignet eingestuft wird. Auszuwerten sind die

8870 Festlegungen des ALGCAT sowohl bezogen auf die Vergangenheit als auch auf die

8871 Zukunft.

8872 Für alle geprüften Zertifikate:

8873 ../

8874 vr:CertificateValidity/

8875 vr:SignatureOK/

8876 vr:SignatureAlgorithm/

8877 vr:Suitability/

8878 ./ResultMajor= urn:oasis:names:tc:dss:1.0:detail:invalid

8879 ./ResultMessage="Algorithmen seit <Jahr> als unsicher eingestuft"

8880 14. PathValidity bis zur TrustAnchor-TSL

8881 //CertificateValidity/ChainingOK/ResultMajor (ab dem zweiten Zertifikat in der

8882 Kette)

8883 //CertificateValidity/CertificateStatus/CertStatusOK/ResultMajor

8884 //CertificateValidity/CertificateValue

8885 Für das Feld TrustAnchor ist



8886 "urn:oasis:names:tc:dss-  
8887 x:1.0:profiles:verificationreport:trustanchor:certDataBase"  
8888 zu verwenden.

8889 15. Prüfergebnis des Gültigkeitszeitraums  
8890 /VerificationReport/  
8891 IndividualReport/  
8892 Details/  
8893 DetailedSignatureReport/  
8894 CertificatePathValidity/  
8895 PathValidityDetail/  
8896 CertificateValidity/  
8897 ValidityPeriodOK/  
8898 ResultMajor

8899 16. Prüfung der Extensions  
8900 /VerificationReport/  
8901 IndividualReport/  
8902 Details/  
8903 DetailedSignatureReport/  
8904 CertificatePathValidity/  
8905 PathValidityDetail/  
8906 CertificateValidity/  
8907 ExtensionsOK/  
8908 ResultMajor

8909 17. Zeitstempel und Herkunft der OCSP-Antwort für das Signaturzertifikat  
8910 /VerificationReport/  
8911 IndividualReport/  
8912 Details/  
8913 DetailedSignatureReport/  
8914 CertificatePathValidity/  
8915 PathValidityDetail/  
8916 CertificateValidity/  
8917 CertificateStatus/  
8918 RevocationEvidence/  
8919 OCSPValidity/  
8920 OCSPIdentifier/  
8921 ./XAdES:ResponderID/XAdES:ByName  
8922 ./XAdES:ProducedAt

8923 18. OCSP Antwort für das Signaturzertifikats

8924 /VerificationReport/

8925 IndividualReport/

8926 Details/

8927 /vr:DetailedSignatureReport/

8928 vr:CertificatePathValidity/

8929 vr:PathValidityDetail/

8930 vr:CertificateValidity/

8931 vr:CertificateStatus/

8932 vr:RevocationEvidence/

8933 vr:OCSPValidity/

8934 vr:OCSPValue

### 8935 **Sonderfälle:**

#### 8936 **Dokument mit parallelen Signaturen**

8937 Für jede Signatur wird ein IndividualReport erzeugt.

#### 8938 **Dokument mit Signatur und Gegensignatur**

8939 Für jede Signatur wird ein IndividualReport erzeugt.

8940

## 8941 **6.4 Profilierung der Dokumentenformate und Nachrichten**

8942 **Tabelle 385: TAB\_KON\_775 „Profilierung der Dokumentformate und Nachrichten“**

|              |  |
|--------------|--|
| XML-Dokument | Es gelten folgende Mindestanforderungen, die der Konnektor bezüglich Dokumentenstruktur und Dokumenteninhalt unterstützen muss:<br>- Hierarchietiefe des Dokumentenbaums: 30 Ebenen<br>- Anzahl von XML-Elementen im Dokument: 30.000<br>- Anzahl von XML-Attributen je XML-Element: 20<br>- Anzahl von direkten Kindern eines XML-Elements: 50<br>- Länge von XML-Bezeichnern (z. B. Elementnamen, Attributnamen, Namespace-Prefixes, usw.): 200<br>- Anzahl von Transformationen: 64<br>- Element-Größe pro Einzelknoten im Base64-codierten Dokument: 30 MB |
|              | Es dürfen keine ENTITY-Deklarationen im XML-Dokument vorkommen.  |
|              | Zu verifizierende XML-Dokumente dürfen im <Transforms>-Teil ihrer Referenzen weder XPath-Ausdrücke noch XSL-Transformationen enthalten.  |

|  |  |
|--|--|
|  | Bei Referenzen (ReferenceType) darf das Optionale URI-Attribut nicht vorhanden sein, oder es muss leer sein. |
|  | XInclude darf nicht unterstützt werden.  |
|  | Die Attribute schemaLocation und noNamespaceSchemaLocation dürfen nicht unterstützt werden.                  |

8943

8944

ENTWURF

## 7-Anhang D—Übersicht über die verwendeten Versionen

Für den Fall, dass Schnittstellenversionen unterstützt werden müssen, die den gleichen TargetNamespace nutzen, kann der Konnektor zu diesen Schnittstellenversionen einheitlich einen SOAP-Endpunkt anbieten, der die höchste der Schnittstellenversionen implementiert.

**Tabelle 385: TAB\_KON\_688 Version der Schemas aus dem Namensraum des Konnektors**

| Schemas aus dem Namensraum des Konnektors „ <a href="http://ws.gematik.de/conn">http://ws.gematik.de/conn</a> “ |   |  |
|---|---|--|
|   |   |  |
| XSD-Name  | CardEvents.xsd  |  |
| XSD-Schemaversion   | 6.0.0   |  |
| TargetNamespace   | <a href="http://ws.gematik.de/conn/CardEvents/v6.0">http://ws.gematik.de/conn/CardEvents/v6.0</a>   |  |
|   |   |  |
| XSD-Name  | CardService_v8_1_3.xsd  |  |
| XSD-Schemaversion   | 8.1.3   |  |
| TargetNamespace   | <a href="http://ws.gematik.de/conn/CardService/v8.1">http://ws.gematik.de/conn/CardService/v8.1</a> |  |
|   |   |  |
| XSD-Name  | CardService_v8_1_1.xsd  |  |
| XSD-Schemaversion   | 8.1.1   |  |
| TargetNamespace   | <a href="http://ws.gematik.de/conn/CardService/v8.1">http://ws.gematik.de/conn/CardService/v8.1</a> |  |
|   |   |  |
| XSD-Name  | CardService.xsd   |  |
| XSD-Schemaversion   | 8.1.0   |  |
| TargetNamespace   | <a href="http://ws.gematik.de/conn/CardService/v8.1">http://ws.gematik.de/conn/CardService/v8.1</a> |  |
|   |   |  |

|  |                   |   |
|--|-------------------|---|
|  | XSD-Name          | CardServiceCommon.xsd   |
|  | XSD-Schemaversion | 2.0.0   |
|  | TargetNamespace   | <a href="http://ws.gematik.de/conn/CardServiceCommon/v2.0">http://ws.gematik.de/conn/CardServiceCommon/v2.0</a>     |
|  |                   |   |
|  | XSD-Name          | CardTerminalInfo.xsd  |
|  | XSD-Schemaversion | 8.1.0   |
|  | TargetNamespace   | <a href="http://ws.gematik.de/conn/CardTerminalInfo/v8.1">http://ws.gematik.de/conn/CardTerminalInfo/v8.1</a>       |
|  |                   |   |
|  | XSD-Name          | CardTerminalService.xsd   |
|  | XSD-Schemaversion | 1.1.2   |
|  | TargetNamespace   | <a href="http://ws.gematik.de/conn/CardTerminalService/v1.1">http://ws.gematik.de/conn/CardTerminalService/v1.1</a> |
|  |                   |   |
|  | XSD-Name          | CertificateService_v6_0_2.xsd   |
|  | XSD-Schemaversion | 6.0.2   |
|  | TargetNamespace   | <a href="http://ws.gematik.de/conn/CertificateService/v6.0">http://ws.gematik.de/conn/CertificateService/v6.0</a>   |
|  |                   |   |
|  | XSD-Name          | CertificateService.xsd  |
|  | XSD-Schemaversion | 6.0.1   |
|  | TargetNamespace   | <a href="http://ws.gematik.de/conn/CertificateService/v6.0">http://ws.gematik.de/conn/CertificateService/v6.0</a>   |
|  |                   |   |
|  | XSD-Name          | CertificateServiceCommon.xsd  |
|  | XSD-Schemaversion | 2.0.1   |

|  |                   |   |
|--|-------------------|---|
|  | TargetNamespace   | <a href="http://ws.gematik.de/conn/CertificateServiceCommon/2.0">http://ws.gematik.de/conn/CertificateServiceCommon/2.0</a> |
|  |                   |   |
|  | XSD-Name          | ConnectorCommon.xsd   |
|  | XSD-Schemaversion | 5.0.0   |
|  | TargetNamespace   | <a href="http://ws.gematik.de/conn/ConnectorCommon/v5.0">http://ws.gematik.de/conn/ConnectorCommon/v5.0</a>                 |
|  |                   |   |
|  | XSD-Name          | ConnectorContext.xsd  |
|  | XSD-Schemaversion | 2.0.0   |
|  | TargetNamespace   | <a href="http://ws.gematik.de/conn/ConnectorContext/v2.0">http://ws.gematik.de/conn/ConnectorContext/v2.0</a>               |
|  |                   |   |
|  | XSD-Name          | EncryptionService_v6_1_2.xsd  |
|  | XSD-Schemaversion | 6.1.2   |
|  | TargetNamespace   | <a href="http://ws.gematik.de/conn/EncryptionService/v6.1">http://ws.gematik.de/conn/EncryptionService/v6.1</a>             |
|  |                   |   |
|  | XSD-Name          | EncryptionService.xsd   |
|  | XSD-Schemaversion | 6.1.1   |
|  | TargetNamespace   | <a href="http://ws.gematik.de/conn/EncryptionService/v6.1">http://ws.gematik.de/conn/EncryptionService/v6.1</a>             |
|  |                   |   |
|  | XSD-Name          | EventService.xsd  |
|  | XSD-Schemaversion | 7.2.1   |
|  | TargetNamespace   | <a href="http://ws.gematik.de/conn/EventService/v7.2">http://ws.gematik.de/conn/EventService/v7.2</a>                       |
|  |                   |   |

|   |   |
|---|---|
| XSD-Name  | ServiceDirectory.xsd                            |
| XSD-Schemaversion   | 3.1.0   |
| TargetNamespace   | http://ws.gematik.de/conn/ServiceDirectory/v3.1 |
|   |   |
| XSD-Name  | SignatureService_V7_5_2.xsd                     |
| XSD-Schemaversion   | siehe XSD-Name                                  |
| TargetNamespace   | http://ws.gematik.de/conn/SignatureService/v7.5 |
|   |   |
| XSD-Name  | SignatureService_V7_4_4.xsd                     |
| XSD-Schemaversion   | siehe XSD-Name                                  |
| TargetNamespace   | http://ws.gematik.de/conn/SignatureService/v7.4 |
|   |   |
| XSD-Name  | SignatureService.xsd                            |
| XSD-Schemaversion   | 7.4.2   |
| TargetNamespace   | http://ws.gematik.de/conn/SignatureService/v7.4 |
| Der Signatordienst des Konnektors MUSS sich an der Außenschnittstelle konform zu den gehärteten Schemas XSD_HARDENED verhalten. Die konkreten Schemas zu dem Bezeichner XSD_HARDENED legt die Dokumentenlandkarte fest. |   |

Tabelle 386: TAB\_KON\_798 Schnittstellenversionen

**Pro-Dienst mit Operationen an der Außenschnittstelle:  
WSDLs des Konnektors und verwendete XSDs aus dem Namensraum der  
gematik http://ws.gematik.de**

**Kartendienst (CardService)**

|                                   |  |
|-----------------------------------|--|
| WSDL Name                         | CardService.wsdl   |
| WSDL Version                      | 8.1.2  |
| TargetNamespace                   | http://ws.gematik.de/conn/CardService/WSDL/v8.1  |
| verwendete XSDs                   | CardService.xsd,<br>CardServiceCommon.xsd,<br>ConnectorCommon.xsd,<br>ConnectorContext.xsd,<br>ProductInformation.xsd,<br>TelematikError.xsd |
| <b>Kartendienst (CardService)</b> |  |
| WSDL Name                         | CardService.wsdl   |
| WSDL Version                      | 8.1.1  |
| TargetNamespace                   | http://ws.gematik.de/conn/CardService/WSDL/v8.1  |
| verwendete XSDs                   | CardService.xsd,<br>CardServiceCommon.xsd,<br>ConnectorCommon.xsd,<br>ConnectorContext.xsd,<br>ProductInformation.xsd,<br>TelematikError.xsd |
| <b>Kartendienst (CardService)</b> |  |
| WSDL Name                         | CardService.wsdl   |
| WSDL Version                      | 8.1.0  |
| TargetNamespace                   | http://ws.gematik.de/conn/CardService/WSDL/v8.1  |



|   |                 |  |
|---|-----------------|--|
|   | verwendete XSDs | CardService.xsd,<br>CardServiceCommon.xsd,<br>ConnectorCommon.xsd,<br>ConnectorContext.xsd,<br>ProductInformation.xsd,<br>TelematikError.xsd   |
|   |                 |  |
| <b>Kartenterminaldienst (CardTerminalService)</b> |                 |  |
|   | WSDL Name       | CardTerminalService.wsdl   |
|   | WSDL Version    | 1.1.0  |
|   | TargetNamespace | http://ws.gematik.de/conn/CardTerminalService/ WSDL/v1.1   |
|   | verwendete XSDs | CardTerminalService.xsd,<br>CardService.xsd,<br>CardTerminalService.xsd,<br>CardServiceCommon.xsd,<br>ConnectorCommon.xsd,<br>ConnectorContext.xsd,<br>TelematikError.xsd                  |
|   |                 |  |
| <b>Systeminformationsdienst (EventService)</b>    |                 |  |
|   | WSDL Name       | EventService.wsdl  |
|   | WSDL Version    | 7.2.0  |
|   | TargetNamespace | http://ws.gematik.de/conn/EventService/ WSDL/v7.2  |
|   | verwendete XSDs | CardService.xsd,<br>CardServiceCommon.xsd,<br>CardTerminalInfo.xsd,<br>ConnectorCommon.xsd,<br>ConnectorContext.xsd,<br>EventService.xsd,<br>ProductInformation.xsd,<br>TelematikError.xsd |

|   |                 |  |
|---|-----------------|--|
|   |                 |  |
| <b>Zertifikatsdienst (CertificateService)</b>     |                 |  |
|   | WSDL Name       | CertificateService.wsdl  |
|   | WSDL Version    | 6.0.1  |
|   | TargetNamespace | http://ws.gematik.de/conn/CertificateService/ WSDL/v6.0  |
|   | verwendete XSDs | CertificateService.xsd,<br>CertificateServiceCommon.xsd,<br>ConnectorCommon.xsd,<br>ConnectorContext.xsd |
|   |                 |  |
| <b>Zertifikatsdienst (CertificateService)</b>     |                 |  |
|   | WSDL Name       | CertificateService.wsdl  |
|   | WSDL Version    | 6.0.0  |
|   | TargetNamespace | http://ws.gematik.de/conn/CertificateService/ WSDL/v6.0  |
|   | verwendete XSDs | CertificateService.xsd,<br>CertificateServiceCommon.xsd,<br>ConnectorCommon.xsd,<br>ConnectorContext.xsd |
|   |                 |  |
| <b>Verschlüsselungsdienst (EncryptionService)</b> |                 |  |
|   | WSDL Name       | EncryptionService.wsdl   |
|   | WSDL Version    | 6.1.1  |
|   | TargetNamespace | http://ws.gematik.de/conn/EncryptionService/ WSDL/v6.1   |

|   |                 |  |
|---|-----------------|--|
|   | verwendete XSDs | ConnectorCommon.xsd,<br>ConnectorContext.xsd,<br>EncryptionService.xsd                   |
| <b>Verschlüsselungsdienst (EncryptionService)</b> |                 |  |
|   | WSDL Name       | EncryptionService.wsdl   |
|   | WSDL Version    | 6.1.0  |
|   | TargetNamespace | http://ws.gematik.de/conn/EncryptionService/ WSDL/v6.1                                   |
|   | verwendete XSDs | ConnectorCommon.xsd,<br>ConnectorContext.xsd,<br>EncryptionService.xsd                   |
| <b>Signaturdienst (SignatureService)</b>          |                 |  |
|   | WSDL Name       | SignatureService_V7_5_2.wsdl   |
|   | WSDL Version    | siehe WSDL Name  |
|   | TargetNamespace | http://ws.gematik.de/conn/SignatureService/WSDL/v7.5                                     |
|   | verwendete XSDs | ../tel/error/TelematikError.xsd,<br>ConnectorContext.xsd,<br>SignatureService_V7_5_2.xsd |
| <b>Signaturdienst (SignatureService)</b>          |                 |  |
|   | WSDL Name       | SignatureService_V7_4_2.wsdl   |
|   | WSDL Version    | siehe WSDL Name  |
|   | TargetNamespace | http://ws.gematik.de/conn/SignatureService/WSDL/v7.4                                     |

|  |                 |   |
|--|-----------------|---|
|  | verwendete XSDs | ../tel/error/TelematikError.xsd,<br>ConnectorContext.xsd,<br>SignatureService_V7_4_4.xsd  |
|  |                 |   |
| <b>Signaturdienst (SignatureService)</b>               |                 |   |
|  | WSDL Name       | SignatureService.wsdl   |
|  | WSDL-Version    | 7.4.0   |
|  | TargetNamespace | <a href="http://ws.gematik.de/conn/SignatureService/WSDL/v7.4">http://ws.gematik.de/conn/SignatureService/WSDL/v7.4</a>         |
|  | verwendete XSDs | ../tel/error/TelematikError.xsd,<br>ConnectorContext.xsd,<br>SignatureService.xsd   |
|  |                 |   |
| <b>Authentifizierungsdienst (AuthSignatureService)</b> |                 |   |
|  | WSDL Name       | AuthSignatureService_v7_4_1.wsdl  |
|  | WSDL-Version    | 7.4.1   |
|  | TargetNamespace | <a href="http://ws.gematik.de/conn/AuthSignatureService/WSDL/v7.4">http://ws.gematik.de/conn/AuthSignatureService/WSDL/v7.4</a> |
|  | verwendete XSDs | ../tel/error/TelematikError.xsd,<br>ConnectorContext.xsd,<br>SignatureService.xsd   |
|  |                 |   |
| <b>Authentifizierungsdienst (AuthSignatureService)</b> |                 |   |
|  | WSDL Name       | AuthSignatureService.wsdl   |
|  | WSDL-Version    | 7.4.0   |
|  | TargetNamespace | <a href="http://ws.gematik.de/conn/AuthSignatureService/WSDL/v7.4">http://ws.gematik.de/conn/AuthSignatureService/WSDL/v7.4</a> |

|  |                 |   |
|--|-----------------|---|
|  | verwendete XSDs | ../tel/error/TelematikError.xsd,<br>ConnectorContext.xsd,<br>SignatureService.xsd |
|--|-----------------|---|

8954  
8955

ENTWURF

8956

## 87 Anhang F – Übersicht Events

8957

Tabelle 386: TAB\_KON\_777 Events Interne Mechanismen

| Topic Ebene1<br>/Topic Ebene2<br>/Topic Ebene3                                   | Typ | Schwere | P<br>r<br>o<br>t | A<br>n<br>C<br>l<br>i<br>e<br>n<br>t<br>s | Parameter   | Bedeutung   | Auslöser<br>(TUC/Op) |
|--|-----|---------|------------------|---|---|---|----------------------|
| <b>Interne Mechanismen</b>   |     |         |                  |   |   |   |                      |
| BOOTUP<br>/BOOTUP_COMPLETE   | Op  | Info    | x                | x   |   | Änderung des Betriebszustand des  |                      |
| OPERATIONAL_STATE<br>/EC_CardTerminal_Software_Out_Of_Date<br>(\$ctId)           | Op  | Info    | x                | x   | Value=true/false;<br>CtID=\$ctId;<br>Bedeutung=\$EC.description | Änderung des Betriebszustand des durch Änderung im Fehlerzustand (Änderung im Value). |                      |
| OPERATIONAL_STATE<br>EC_CardTerminal_SMC-KT_Certificate_Expires_Soon<br>(\$ctId) | Op  | Info    | x                | x   | Value=true/false;<br>CtID=\$ctId;<br>Bedeutung=\$EC.description | Änderung des Betriebszustand des durch Änderung im Fehlerzustand (Änderung im Value). | TUC_KON_050          |
| OPERATIONAL_STATE<br>/EC_Connector_Software_Out_Of_Date                          | Op  | Info    | x                | x   | Value=true/false;<br>Bedeutung=\$EC.description                 | "   |                      |
| OPERATIONAL_STATE<br>/EC_FW_Not_Valid_Status_Blocked                             | Sec | Fatal   | x                | x   | Value=true/false;<br>Bedeutung=\$EC.description                 | "   |                      |

|  |     |             |   |   |   |   |                 |
|--|-----|-------------|---|---|---|---|-----------------|
| OPERATIONAL_STATE<br>/EC_Time_Sync_Not_Successful  | Op  | Info        | x | x | Value=true/<br>false;<br>LastSyncAttempt<br>=<br>\$lastSyncAttempt<br>Timestamp;<br>LastSyncSuccess<br>=<br>\$lastSyncSuccess<br>Timestamp;<br>Bedeutung=<br>\$EC.description | " |                 |
| OPERATIONAL_STATE<br>/EC_TSL_Update_Not_Successful | Op  | Info        | x | x | Value=true/<br>false;<br>Bedeutung=<br>\$EC.description<br>;<br>LastUpdateTSL=<br>\$lastUpdate<br>TSLTimestamp  | " |                 |
| OPERATIONAL_STATE<br>/EC_TSL_Expiring              | Sec | Info        | x | x | Value=true/<br>false;<br>NextUpdateTSL=<br>\$NextUpdate-<br>Element der<br>TSL;<br>Bedeutung=<br>\$EC.description   | " |                 |
| OPERATIONAL_STATE<br>/EC_TSL_Trust_Anchor_Expiring | Sec | Info        | x | x | Value=true/<br>false;<br>ExpiringDate<br>TrustAnchor=<br>Ablaufdatum<br>der Vertrauens<br>ankergültigkeit<br>;<br>Bedeutung=<br>\$EC.description                              | " |                 |
| OPERATIONAL_STATE<br>/EC_LOG_OVERFLOW              | Op  | Warni<br>ng | x | x | Value=true/<br>false;<br>Protokoll=\$Prot<br>okoll;<br>Bedeutung=<br>\$EC.description   | " | TUC_KON_2<br>71 |

|  |     |             |   |   |   |   |  |
|--|-----|-------------|---|---|---|---|--|
| OPERATIONAL_STATE<br>/EC_CRL_Expiring                            | Sec | Warni<br>ng | x | x | Value=true/<br>false;<br>NextUpdateTSL=<br>\$NextUpdate-<br>Element der<br>TSL;<br>Bedeutung=<br>\$EC.description   | " |  |
| OPERATIONAL_STATE<br>/EC_Time_Sync_Pending_Warnin<br>g           | Sec | Warni<br>ng | x | x | Value=true/<br>false;<br>LastSyncSuccess<br>=<br>\$lastSyncSucces<br>s<br>Timestamp;<br>Bedeutung=<br>\$EC.description  | " |  |
| OPERATIONAL_STATE<br>/EC_TSL_Out_Of_Date_Within_G<br>race_Period | Sec | Warni<br>ng | x | x | Value=true/<br>false;<br>NextUpdateTSL=<br>\$NextUpdate-<br>Element der<br>TSL;<br>GracePeriodTSL=<br>CERT_TSL_<br>DEFAULT_GRACE_<br>PERIOD_DAYS;<br>Bedeutung=<br>\$EC.description | " |  |
| OPERATIONAL_STATE<br>/EC_CardTerminal_Not_Availab<br>le(\$ctId)  | Op  | Error       | x | x | Value=true/<br>false;<br>CtID=\$ctId;<br>Bedeutung<br>=\$EC.descriptio<br>n   | " |  |
| OPERATIONAL_STATE<br>/EC_No_VPN_TI_Connection                    | Op  | Error       | x | x | Value=true/<br>false;<br>Bedeutung=<br>\$EC.description   | " |  |
| OPERATIONAL_STATE<br>/EC_No_VPN_SIS_Connection                   | Op  | Error       | x | x | Value=true/<br>false;<br>Bedeutung=<br>\$EC.description   | " |  |



|  |     |       |   |   |   |   |  |
|--|-----|-------|---|---|---|---|--|
| OPERATIONAL_STATE<br>/EC_No_Online_Connection                | Op  | Error | x | x | Value=true/<br>false;<br>Bedeutung=<br>\$EC.description   | " |  |
| OPERATIONAL_STATE<br>/EC_IP_Addresses_Not_Availabl<br>e      | Sec | Error | x | x | Value=true/<br>false;<br>Bedeutung=<br>\$EC.description   | " |  |
| OPERATIONAL_STATE<br>/EC_CRL_Out_Of_Date                     | Sec | Fatal | x | x | Value=true/<br>false;<br>NextUpdateCRL=<br>\$NextUpdate der<br>CRL;<br>Bedeutung=<br>\$EC.description | " |  |
| OPERATIONAL_STATE<br>/EC_Firewall_Not_Reliable               | Sec | Fatal | x | x | Value=true/<br>false;<br>Bedeutung=<br>\$EC.description   | " |  |
| OPERATIONAL_STATE<br>/EC_Random_Generator_Not_Rel<br>iable   | Sec | Fatal | x | x | Value=true/<br>false;<br>Bedeutung=<br>\$EC.description   | " |  |
| OPERATIONAL_STATE<br>/EC_SecureKeyStore_Not_Avail<br>able    | Sec | Fatal | x | x | Value=true/<br>false;<br>Bedeutung=<br>\$EC.description   | " |  |
| OPERATIONAL_STATE<br>/EC_Security_Log_Not_Writabl<br>e       | Op  | Fatal | x | x | Value=true/<br>false;<br>Bedeutung=<br>\$EC.description   | " |  |
| OPERATIONAL_STATE<br>/EC_Software_Integrity_Check<br>_Failed | Sec | Fatal | x | x | Value=true/<br>false;<br>Bedeutung=<br>\$EC.description   | " |  |

|  |     |       |   |   |  |   |  |
|--|-----|-------|---|---|--|---|--|
| OPERATIONAL_STATE<br>/EC_Time_Difference_Intolerable         | Sec | Fatal | x | x | Value=true/<br>false;<br>Bedeutung=<br>\$EC.description<br>;<br>NtpTimedifference=<br>Zeitabweichung;<br>NtpMaxAllowedTime<br>difference=<br>NTP_MAX_TIMEDIFFERENCE;<br>Bedeutung=<br>\$EC.description | " |  |
| OPERATIONAL_STATE<br>/EC_Time_Sync_Pending_Critical          | Sec | Fatal | x | x | Value=true/<br>false;<br>LastSyncSuccess=<br>\$lastSyncSuccess<br>Timestamp;<br>NtpGracePeriod=<br>NTP_GRACE_PERIOD;<br>Bedeutung=<br>\$EC.description   | " |  |
| OPERATIONAL_STATE<br>/EC_TSL_Trust_Anchor_Out_Of_Date        | Sec | Fatal | x | x | Value=true/<br>false;<br>ExpiringDate<br>TrustAnchor=<br>Ablaufdatum der<br>Vertrauensanker<br>gültigkeit;<br>Bedeutung=<br>\$EC.description   | " |  |
| OPERATIONAL_STATE<br>/EC_TSL_Out_Of_Date_Beyond_Grace_Period | Sec | Fatal | x | x | Value=true/<br>false;<br>Next<br>UpdateTSL=<br>\$NextUpdate-<br>Element der<br>TSL;<br>GracePeriodTSL=<br>CERT_TSL_<br>DEFAULT_<br>GRACE_PERIOD_DAYS;<br>Bedeutung=<br>\$EC.description                | " |  |

|  |        |                |   |   |  |   |  |
|--|--------|----------------|---|---|--|---|--|
| OPERATIONAL_STATE<br>/EC_CRYPTOPERATION_ALARM                | Sec    | Warni<br>ng    | x | x | Value=true/<br>false;<br>Operation=<br>\$Operationsname<br>;<br>Count=\$Summenwe<br>rt;<br>Arbeitsplatz<br>=\$<Liste<br>operations-<br>aufrufenden<br>workplace IDs>;<br>Meldung='<br>Auffällige<br>Häufung von<br>Operationsaufru<br>fen in den<br>letzten<br>10 Minuten' | " |  |
| OPERATIONAL_STATE<br>/EC_OTHER_ERROR_STATE (\$no)            | \$Type | \$Seve<br>rity | x | x | Value=true/<br>false;<br>Bedeutung=<br>\$EC.description  | " |  |
| OPERATIONAL_STATE<br>/EC_BNetzA_VL_Update_Not_Suc<br>cessful | Op     | Info           | x | x | Value=true/<br>false;<br>LastUpdate<br>BNetzAVL=<br>\$lastUpdateBNet<br>zAVL<br>Timestamp;<br>Bedeutung=<br>\$EC.description<br>;  | " |  |
| OPERATIONAL_STATE<br>/EC_BNetzA_VL_not_valid                 | Sec    | Warni<br>ng    | x | x | Value=true/<br>false;<br>NextUpdate<br>BNetzAVL=<br>\$NextUpdate-<br>Element der<br>BNetzA-VL;<br>Bedeutung=<br>\$EC.description<br>;  | " |  |
| <b>Zugriffsberechtigungsdienst</b>                           |        |                |   |   |  |   |  |
|  |        |                |   |   |  |   |  |
| <b>Dokumentvalidierungsdienst</b>                            |        |                |   |   |  |   |  |
|  |        |                |   |   |  |   |  |

| Dienstverzeichnisdienst          |                 |            |   |   |   |  |                            |
|----------------------------------|-----------------|------------|---|---|---|--|----------------------------|
|                                  |                 |            |   |   |   |  |                            |
| Kartenterminaldienst             |                 |            |   |   |   |  |                            |
| CT<br>/ERROR                     | \$Error<br>Type | \$Severity | x | x | CtID=\$CT.ID;<br>Name=\$CT.HOSTNAME;<br>Error=\$Fehlercode;<br>Bedeutung=\$Fehlertext   | Bei der Kommunikation mit dem KT ist ein Fehler aufgetreten  | TUC_KON_051<br>TUC_KON_053 |
| CT<br>/CONNECTED                 | Op              | Info       | x | x | CtID=\$CT.CTID;<br>Hostname=\$CT.HOSTNAME   | Die Verbindung zu einem Kartenterminal wurde hergestellt   |                            |
| CT<br>/DISCONNECTED              | Op              | Info       | x | x | CtID=\$CT.CTID;<br>Hostname=\$CT.HOSTNAME   | Die Verbindung zu einem Kartenterminal wurde unterbrochen  |                            |
| CT<br>/TLS_ESTABLISHMENT_FAILURE | \$Error<br>Type | \$Severity | x | x | CtID = \$CT.ID;<br>Name=\$CT.HOSTNAME;<br>Error=\$Fehlercode;<br>Bedeutung=\$Fehlertext | Im Rahmen des Verbindungsaufbaus sind Fehler aufgetreten   | TUC_KON_050                |
| CT<br>/CT_ADDING_ERROR           | \$Error<br>Type | \$Severity | x | x | IP=\$IP-Adresse;<br>Name=\$Hostname;<br>Error=\$Fehlercode;<br>Bedeutung=\$Fehlertext   | Bei dem Versuch ein KT der Verwaltung zuzufügen ist ein Fehler aufgetreten   | TUC_KON_054                |
| CT<br>/SLOT_FREE                 | Op              | Info       | - | - | CtID=\$CT.CTID;<br>SlotNo=\$CT.SLOTS_USED[X]  | Internes Event von Kartenterminaldienst<br>--> Kartendienst. Informiert, dass ein Slot frei wurde. Wird im Kartendienst ausgewertet und verursacht dort CARD/REMOVED |                            |

|                     |    |      |   |   |   |   |  |
|---------------------|----|------|---|---|---|---|--|
| CT<br>/SLOT_IN_USE  | Op | Info | - | - | CtID=\$CT.CTID;<br>SlotNo=<FU-<br>Nummer<br>aus<br>Ereignisnachric<br>ht>   | Internes Event<br>von<br>Kartenterminal<br>dienst<br>--><br>Kartendienst.<br>Informiert,<br>dass<br>ein Slot belegt<br>wurde. Wird im<br>Kartendienst<br>ausgewertet<br>und verursacht<br>dort<br>CARD/INSERTED |  |
| <b>Kartendienst</b> |    |      |   |   |   |   |  |
| CARD<br>/INSERTED   | Op | Info | x | x | CardHandle=<br>\$CARD.CARDHANDL<br>E;<br>CardType=\$CARD.<br>TYP;<br>CardVersion=<br>\$CARD.VER;<br>ICCSN=\$CARD.ICC<br>SN;<br>CtID=\$CARD.CTID<br>;<br>SlotID=<br>\$CARD.SLOTID;<br>InsertTime=<br>\$CARD.INSERTTIM<br>E;<br>CardHolderName=<br>\$CARD.CARD<br>HOLDERNAME;<br>KVNR=\$CARD.KVNR     | Eine Karte<br>wurde<br>gesteckt   | TUC_KON_0<br>01<br>(als<br>Reaktion<br>auf<br>CTM<br>/SLOT_IN_<br>USE) |
| CARD<br>/REMOVED    | Op | Info | x | x | CardHandle=<br>\$CARD.CARDHANDL<br>E;<br>CardType=\$CARD.<br>TYP;<br>CardVersion=<br>\$CARD.VER;<br>ICCSN=\$CARD.ICC<br>SN;<br>CtID=\$CARD.CTID<br>;<br>SlotID=<br>\$CARD.SLOTID;<br>InsertTime=<br>\$CARD.INSERTTIM<br>E;<br>CardHolderName=<br>\$CARD.CARDHOLDE<br>R<br>NAME;<br>KVNR=\$CARD.KVNR | Eine Karte<br>wurde<br>gezogen  | Reaktion<br>auf<br>CTM/SLOT_<br>FREE                                   |

|                                  |    |      |   |   |  |   |                 |
|----------------------------------|----|------|---|---|--|---|-----------------|
| CARD<br>/PIN<br>/VERIFY_STARTED  | Op | Info | - | x |  |   |                 |
| CARD<br>/PIN<br>/VERIFY_FINISHED | Op | Info | - | x |  |   |                 |
| CARD<br>/PIN<br>/CHANGE_STARTED  | Op | Info | - | x |  |   |                 |
| CARD<br>/PIN<br>/CHANGE_FINISHED | Op | Info | - | x |  |   |                 |
| CARD<br>/PIN<br>/ENABLE_STARTED  | Op | Info | - | x | CardHandle=\$;<br>CardType=\$;<br>ICCSN=\$;<br>CtID=\$;<br>SlotID=\$;<br>PinRef=\$PinRef;<br>PinInputCtID<br>=\$PinInputKT | PIN-Schutz<br>anschalten<br>beginnt       | TUC_KON_0<br>27 |
| CARD<br>/PIN<br>/ENABLE_FINISHED | Op | Info | - | x | CardHandle=\$;<br>CardType=\$;<br>ICCSN=\$;<br>CtID=\$;<br>SlotID=\$;<br>PinRef=\$PinRef;<br>PinInputCtID<br>=\$PinInputKT | PIN-Schutz<br>anschalten<br>wurde beendet | TUC_KON_0<br>27 |
| CARD<br>/PIN<br>/DISABLE_STARTED | Op | Info | - | x | CardHandle=\$;<br>CardType=\$;<br>ICCSN=\$;<br>CtID=\$;<br>SlotID=\$;<br>PinRef=\$PinRef;<br>PinInputCtID<br>=\$PinInputKT | PIN-Schutz<br>ausschalten<br>beginnt      | TUC_KON_0<br>27 |

|                                     |    |       |   |   |  |   |   |
|-------------------------------------|----|-------|---|---|--|---|---|
| CARD<br>/PIN<br>/DISABLE_FINISHED   | Op | Info  | - | x | CardHandle=\$;<br>CardType=\$;<br>ICCSN=\$;<br>CtID=\$;<br>SlotID=\$;<br>PinRef=\$PinRef;<br>PinInputCtID<br>=\$PinInputKT | PIN-Schutz<br>ausschalten<br>wurde beendet  | TUC_KON_0<br>27   |
| <b>Systeminformationsdienst</b>     |    |       |   |   |  |   |   |
|                                     |    |       |   |   |  |   |   |
| <b>Verschlüsselungsdienst</b>       |    |       |   |   |  |   |   |
|                                     |    |       |   |   |  |   |   |
| <b>Signaturdienst</b>               |    |       |   |   |  |   |   |
| SIG<br>/SIGNDOC<br>/NEXT_SUCCESSFUL | Op | Info  | - | X | \$Jobnummer  | Die nächste<br>Signatur aus<br>einem<br>Signaturstapel<br>wurde<br>erfolgreich<br>erstellt. | TUC_KON_1<br>66<br>„nonQES<br>Signaturen<br>erstellen“<br>TUC_KON_1<br>54<br>„QES<br>Signaturen<br>erstellen“ |
| <b>Zertifikatsdienst</b>            |    |       |   |   |  |   |   |
| CERT<br>/TSL<br>/IMPORT             | Op | Error | x | - | \$Fehlerbeschrei<br>bung   | Manueller<br>Import der<br>TSL<br>fehlgeschlagen  | TUC_KON_0<br>32<br>"TSL<br>aktualisiere<br>n"   |
| CERT<br>/TSL<br>/UPDATED            | Op | Info  | x | - |  | Eine neue TSL<br>wurde<br>erfolgreich in<br>den TrustStore<br>eingespielt                   | TUC_KON_0<br>32<br>"TSL<br>aktualisiere<br>n"   |
| CERT<br>/CRL<br>/INVALID            | Op | Error | x | - |  | Prüfung der<br>Signatur<br>der CRL<br>fehlgeschlagen  | TUC_KON_0<br>40<br>"CRL<br>aktualisiere<br>n"   |

|                              |    |             |   |   |  |  |  |
|------------------------------|----|-------------|---|---|--|--|--|
| CERT<br>/CRL<br>/IMPORT      | Op | Error       | x | - | \$Fehlerbeschrei<br>bung   | Manueller<br>Import der CRL<br>fehlgeschlagen          | TUC_KON_0<br>40<br>"CRL<br>aktualisiere<br>n"        |
| CERT<br>/CRL<br>/UPDATED     | Op | Info        | x | - |  | Die CRL wurde<br>erfolgreich<br>aktualisiert           | TUC_KON_0<br>40<br>"CRL<br>aktualisiere<br>n"        |
| CERT<br>/CARD<br>/EXPIRATION | Op | Warni<br>ng | x | x | CARD_TYPE=gSMC-<br>K;<br>ICCSN=\$ICCSN;<br>Konnektor=<br>\$MGM_KONN_HOSTN<br>AME;<br>ZertName=<Name<br>des<br>Zertifikatsobje<br>kts>;<br>Expiration<br>Date=\$validity  | gSMC-K<br>abgelaufen                                   | TUC_KON_0<br>33<br>"Zertifikatsa<br>blauf<br>prüfen" |
| CERT<br>/CARD<br>/EXPIRATION | Op | Warni<br>ng | - | x | CARD_TYPE=\$Type<br>;<br>ICCSN=\$ICCSN;<br>CARD_HANDLE=<br>\$CardHandle;<br>CardHolderName=<br>\$CardHolderName<br>;<br>ZertName=<Name<br>des<br>Zertifikatsobje<br>kts>;<br>ExpirationDate=<br>\$validity       | Sonstige Karte<br>abgelaufen                           | TUC_KON_0<br>33<br>"Zertifikatsa<br>blauf<br>prüfen" |
| CERT<br>/CARD<br>/EXPIRATION | Op | Info        | - | x | CARD_TYPE=gSMC-<br>K;<br>ICCSN=\$ICCSN;<br>Konnektor=<br>\$MGM_KONN_HOSTN<br>AME;<br>ZertName=<Name<br>des<br>Zertifikatsobje<br>kts>;<br>ExpirationDate=<br>\$validity;<br>DAYS_LEFT=<br>\$validity-<br>\$Today | gSMC-K läuft<br>innerhalb von<br>DAYS_LEFT<br>Tagen ab | TUC_KON_0<br>33<br>"Zertifikatsa<br>blauf<br>prüfen" |



|                                |                 |            |   |   |   |   |   |
|--------------------------------|-----------------|------------|---|---|---|---|---|
| CERT<br>/CARD<br>/EXPIRATION   | Op              | Info       | - | x | CARD_TYPE=\$Type;<br>ICCSN=\$ICCSN;<br>CARD_HANDLE=\$CardHandle;<br>CardHolderName=\$CardHolderName;<br>ZertName=<Name des Zertifikatsobjekts>;<br>ExpirationDate=\$validity;<br>DAYS_LEFT=\$validity-\$Today | Sonstige Karte läuft innerhalb von DAYS_LEFT Tagen ab               | TUC_KON_033<br>"Zertifikatsablauf prüfen" |
| CERT<br>/BNETZA_VL<br>/UPDATED | Op              | Info       | x | - |   | Eine neue BNetzA-VL wurde erfolgreich in den TrustStore eingespielt | TUC_KON_031<br>" BNetzA-VL aktualisieren" |
| CERT<br>/BNETZA_VL<br>/IMPORT  | Op              | Error      | x | - | \$Fehlerbeschreibung  | Manueller Import der BNetzA-VL fehlgeschlagen                       | TUC_KON_031<br>" BNetzA-VL aktualisieren" |
| <b>Protokollierungsdienst</b>  |                 |            |   |   |   |   |   |
| LOG<br>/ERROR                  | \$Error<br>Type | \$Severity | - | - | Error=\$Fehlercode  | Im Protokollierungsdienst auftretende Fehler werden verteilt        | TUC_KON_271                               |
| LOG<br>/CRYPTO_OP              | Sec             | Info       | x | - | Operation=\$Operationsname;<br><für alle betroffenen Schlüssel:><br>Karte=\$ICCSN;<br>Keyref=<Referenz auf den Schlüssel>;<br>CARD_HANDLE=\$CardHandle;<br>CardHolderName=\$CardHolderName                    |   |   |
| <b>TLS-Dienst</b>              |                 |            |   |   |   |   |   |

|                                      |    |             |   |   |                               |  |                                   |
|--------------------------------------|----|-------------|---|---|-------------------------------|--|-----------------------------------|
|                                      |    |             |   |   |                               |  |                                   |
| <b>Anbindung LAN/WAN</b>             |    |             |   |   |                               |  |                                   |
| ANLW<br>/LAN<br>/IP_CHANGED          | Op | Warni<br>ng | x | - | IP=\$dieNeueIP                | Wenn der LAN-Adapter eine neue IP oder Netzwerk bekommen hat | DHCP, Management<br>schnittstelle |
| ANLW<br>/WAN<br>/IP_CHANGED          | Op | Info        | x | - | IP=\$dieNeueIP                | Wenn der WAN-Adapter eine neue IP oder Netzwerk bekommen hat | DHCP, Management<br>schnittstelle |
| <b>DHCP-Server</b>                   |    |             |   |   |                               |  |                                   |
| DHCP<br>/SERVER<br>/STATECHANGED     | Op | Info        | x | x | STATE=\$DHCP_SERVER_STATE     |  | Administrator                     |
| <b>DHCP Client</b>                   |    |             |   |   |                               |  |                                   |
| DHCP<br>/LAN_CLIENT<br>/RENEW        | Op | Info        | x | x | IP_ADDRESS=<Belegung>         |  | TUC_KON_341                       |
| DHCP<br>/WAN_CLIENT<br>/RENEW        | Op | Info        | x | x | IP_ADDRESS=<Belegung>         |  | TUC_KON_341                       |
| DHCP<br>/LAN_CLIENT<br>/STATECHANGED | Op | Info        | x | x | STATE=\$DHCP_CLIENT_LAN_STATE |  |                                   |
| DHCP<br>/WAN_CLIENT<br>/STATECHANGED | Op | Info        | x | x | STATE=\$DHCP_CLIENT_WAN_STATE |  |                                   |
| <b>VPN-Client</b>                    |    |             |   |   |                               |  |                                   |
| NETWORK<br>/VPN_TI<br>/UP            | Op | Info        | x | x |                               | Wenn der VPN-Tunnel zur TI erfolgreich aufgebaut worden ist. |                                   |

|  |    |       |   |   |  |   |                             |
|--|----|-------|---|---|--|---|-----------------------------|
| NETWORK<br>/VPN_TI<br>/DOWN                      | Op | Info  | x | x |  | Wenn der VPN-Tunnel zur TI nicht mehr zur Verfügung steht.                                  | AFO                         |
| NETWORK<br>/VPN<br>/CONFIG_CHANGED               | Op | Info  | x | - |  | Wenn die Konfiguration des VPN-Clients angepasst wurde.                                     | Management<br>schnittstelle |
| NETWORK<br>/VPN_SIS<br>/UP                       | Op | Info  | x | x |  | Wenn der VPN-Tunnel zum SIS erfolgreich aufgebaut worden ist.                               |                             |
| NETWORK<br>/VPN_SIS<br>/DOWN                     | Op | Info  | x | x |  | Wenn der VPN-Tunnel zum SIS nicht mehr zur Verfügung steht.                                 | AFO                         |
| <b>Zeitdienst</b>                                |    |       |   |   |  |   |                             |
| NTP<br>/ENTERCRITICALSTATE                       | Op | FATAL | x | - | MESSAGE=<br>„CRITICALTIME<br>DEVIATION“  | Zeitabweichung von mehr als einer Stunde entdeckt   |                             |
| <b>Namensdienst und Dienstlokalisierung</b>      |    |       |   |   |  |   |                             |
|  |    |       |   |   |  |   |                             |
| <b>Leistungsumfänge und Standalone-Szenarios</b> |    |       |   |   |  |   |                             |
| MGM<br>/ADMINCHANGES                             | Op | Info  | x | - | User=<br>\$AdminUsername;<br>RefID=\$Referenz<br>ID;<br>NewVal=<br>\$NeuEingestellter<br>Wert“ | Änderungen die der Admin vornimmt werden protokolliert                                      |                             |
| MGM<br>/CONFIG_EXIMPORT                          | Op | Info  | x | - | User=<br>\$AdminUsername;<br>Mode=<br>[Export/Import]  | Dokumentiert (via Mode), dass die Konnektor konfiguration exportiert oder importiert wurde. |                             |

|   |                     |                |   |   |   |  |                   |
|---|---------------------|----------------|---|---|---|--|-------------------|
| MGM<br>/FACTORYSETTINGS                             | Op                  | Info           | x | - | User=<br>\$AdminUsername  | Ein ausgelöster<br>Werksreset<br>wird<br>protokolliert   |                   |
| MGM<br>/REMOTE_SESSION                              | Op                  | Info           | x | - | InitUser=<br>\$AdminUsername;<br>RemoteID=<Kennu<br>ng der<br>Gegenstelle>;<br>Mode=<br>[InitSuccess/<br>InitFail/Exit] | Protokollierung<br>des Versuchs,<br>des Beginns<br>und des Endes<br>einer Remote-<br>Management<br>Session |                   |
| MGM<br>/LU_CHANGED<br>/LU_ONLINE                    | Op                  | Info           | x | x | Active=<br>\$MGM_LU_ONLINE  | Leistungsumfan<br>g Online wurde<br>aktiviert/<br>deaktiviert  | Administrat<br>or |
| MGM<br>/LU_CHANGED<br>/LU_SAK                       | Op                  | Info           | x | x | Active=<br>\$MGM_LU_SAK   | Leistungsumfan<br>g Signatur<br>anwendungs<br>komponente<br>wurde<br>aktiviert/deakti<br>viert             | Administrat<br>or |
| MGM<br>/STANDALONE_CHANGED                          | Op                  | Info           | x | x | Active=<br>\$MGM_STANDALONE<br>_KON   | Festlegung des<br>Konnektors als<br>"Alleinstehend"<br>wurde geändert                                      | Administrat<br>or |
| <b>In- und Außerbetriebnahme</b>                    |                     |                |   |   |   |  |                   |
| MGM<br>/TI_ACCESS_GRANTED                           | Op                  | Info           | x | - | Active=<br>\$MGM_TI_ACCESS_<br>GRANTED  | Der Konnektor<br>wurde<br>erfolgreich<br>freigeschaltet  | Administrat<br>or |
| <b>Software- Aktualisierungsdienst (KSR-Client)</b> |                     |                |   |   |   |  |                   |
| KSR<br>/ERROR                                       | \$Err<br>or<br>Type | \$Seve<br>rity | x | x | Target=Konnekto<br>r;<br>Name=<br><MGM_KONN_HOSTN<br>AME>;<br>Error=\$Fehlerco<br>de;<br>Bedeutung=<br>\$Fehlertext     | Während der<br>Konnektor<br>aktualisierung<br>ist ein Fehler<br>aufgetreten                                | TUC_KON_2<br>80   |

|                            |                 |            |   |   |  |  |                                    |
|----------------------------|-----------------|------------|---|---|--|--|------------------------------------|
| KSR<br>/ERROR              | \$Error<br>Type | \$Severity | x | x | Target=KT;<br>Name=<br><KT-Friendly<br>Name>;<br>CtID=\$CtID;<br>Error=\$Fehlercode;<br>Bedeutung=<br>\$Fehlertext   | Während einer<br>Kartenterminal<br>aktualisierung<br>ist ein Fehler<br>aufgetreten                     | TUC_KON_2<br>81                    |
| KSR<br>/ERROR              | \$Error<br>Type | \$Severity | x | x | Error=\$Fehlercode;<br>Bedeutung=<br>\$Fehlertext  | Im KSR-Client<br>ist ein Fehler<br>aufgetreten   | TUC_KON_2<br>82                    |
| KSR<br>/UPDATE<br>/START   | Sec             | Info       | x | x | <u>für TUC KON 280</u><br>Target=Konnektor;<br>Name=<br><MGM_KONN_HOSTNAME><br><br><u>für TUC KON 281</u><br>Target=KT;<br>CtID=\$CtID   | Ein<br>Updateprozess<br>im Konnektor<br>wird gestartet,<br>Ziel Konnektor<br>oder<br>Kartenterminal    | TUC_KON_2<br>80<br>TUC_KON_2<br>81 |
| KSR<br>/UPDATE<br>/SUCCESS | Sec             | Info       | x | x | <u>für TUC KON 280</u><br>Target=Konnektor;<br>Name=<br><MGM_KONN_HOSTNAME>;<br>NewFirmwareversion=<br><UpdateInformation.<br>FirmwareVersion<br>>;<br>ConfigurationChanged<br>=<Ja/Nein>;<br>ManualInputNeeded=<br><Ja/Nein><br><br><u>für TUC KON 281</u><br>Target=KT;<br>Name=<br><KT-FriendlyName>;<br>CtID=\$CtID;<br>NewFirmwareversion=<br><UpdateInformation.<br>FirmwareVersion<br>> | Die Firmware<br>des<br>Konnektors/<br>eines<br>Kartenterminals<br>wurde<br>erfolgreich<br>aktualisiert | TUC_KON_2<br>80<br>TUC_KON_2<br>81 |

|   |     |      |   |   |  |  |                                    |
|---|-----|------|---|---|--|--|------------------------------------|
| KSR<br>/UPDATE<br>/END                    | Sec | Info | x | x | für TUC KON 280<br>Target=Konnekto<br>r;<br>Name=<br><MGM_KONN_HOSTN<br>AME><br><br>für TUC KON 281<br>Target=KT;<br>CtID=\$CtID   | Ein<br>Updateprozess<br>im Konnektor<br>wurde beendet  | TUC_KON_2<br>80<br>TUC_KON_2<br>81 |
| KSR<br>/UPDATE<br>/KONNEKTOR_DOWNLOAD_END | Op  | Info | x | x | <b>Je<br/>heruntergeladene<br/>m FW-Paket:</b><br>ProductVendorID<br>=<br>\$UpdateInformat<br>ion/<br>ProductVendorID<br>;<br><br>ProductCode=<br>\$UpdateInformat<br>ion/<br>ProductCode;<br><br>ProductName=<br>\$UpdateInformat<br>ion/<br>ProductName;<br><br>FirmwareVersion<br>=<br>\$UpdateInformat<br>ion/<br>Firmware/FWVers<br>ion;<br><br>Deadline=<br>\$UpdateInformat<br>ion/<br>DeploymentInfor<br>mation/<br>Deadline;<br><br>FWPriority=<br>\$UpdateInformat<br>ion/<br>Firmware/FWPrio<br>rity;<br><br>FirmwareRelease<br>Notes=<br>\$UpdateInformat<br>ion/<br>Firmware/<br>FirmwareRelease<br>Notes | Download der<br>Konnektor<br>Firmware<br>abgeschlossen | TIP1-<br>A_6025                    |

|                           |    |      |   |   |   |  |             |
|---------------------------|----|------|---|---|---|--|-------------|
| KSR<br>/UPDATES_AVAILABLE | Op | Info | - | x | <b>Je gefundenem FW-Paket:</b><br>ProductVendorID=<br>\$UpdateInformation/<br>ProductVendorID;<br><br>ProductCode=<br>\$UpdateInformation/<br>ProductCode;<br><br>ProductName=<br>\$UpdateInformation/<br>ProductName;<br><br>FirmwareVersion=<br>\$UpdateInformation/<br>FirmwareVersion;<br><br>Deadline=<br>\$UpdateInformation/<br>DeploymentInformation/<br>Deadline;<br><br>FWPriority=<br>\$UpdateInformation/<br>Firmware/FWPriority;<br>FirmwareReleaseNotes=<br>\$UpdateInformation/<br>Firmware/<br>FirmwareReleaseNotes | Ein oder mehrere Updates auf neuere Versionen sind verfügbar | TIP1-A_4836 |
| KSR<br>/UPDATE_KONFIG     | Op | Info | x | - | AlteVersion,<br>NeueVersion   | Aktualisierung Bestandsnetze                                 | TUC_KON_283 |

8958

8959 Die Abbildungsvorschrift von Fehler- auf Event-Type lautet:

8960

- Security → Security,

8961

- Technical → Operation,

8962

- Infrastructure → Infrastructure,

8963

- Business → Business,

8964

- Other → Other

## 98 Anhang H – Mapping von „Architektur der TI-Plattform“ auf Konnektorspezifikation

**Tabelle 387: TAB\_KON\_711 Architektur der TI-Plattform, Berechtigt Fachmodule**

| Interface              | Operation                    | → Funktionsmerkmal                     | Interface  |
|------------------------|------------------------------|--|--|
| I_Cert_Verification    | verify_Certificate           | → Zertifikatsdienst                    | TUC_KON_037<br>"Zertifikat prüfen"   |
| I_Crypt_Operations     | decrypt_Document             | → Verschlüsselungsdienst               | TUC_KON_071<br>"Daten hybrid entschlüsseln"  |
|                        | encrypt_Document             | →                                      | TUC_KON_070<br>"Daten hybrid verschlüsseln"  |
| I_DNS_Name_Information | get_FQDN                     | → Namensdienst und Dienstlokalisierung | TUC_KON_364<br>„DNS Reverse Lookup durchführen“                                      |
|                        | get_IP_Address               | →                                      | TUC_KON_361<br>„DNS Namen auflösen“  |
|                        | get_Service_Information      | → Namensdienst und Dienstlokalisierung | TUC_KON_362<br>„Liste der Dienste abrufen“<br>TUC_KON_363<br>„Dienstdetails abrufen“ |
| I_IP_Transport         | send_Data_TI                 | →                                      |  |
| I_KT_Operations        | interact_with_User           | → Kartenterminaldienst                 | TUC_KON_051<br>"Mit Anwender über Kartenterminal interagieren"                       |
| I_KV_Card_Handling     | discard_Card_Usage_Reference | → ---                                  | --- keine Umsetzung notwendig. Erfolgt implizit                                      |



|                      |                          |   |                   |   |
|----------------------|--------------------------|---|-------------------|---|
|                      | get_Card_Usage_Reference | → | ---               | --- keine Umsetzung notwendig. Erfolgt implizit                         |
| I_KV_Card_Operations | decrypt_Data             | → | Kartendienst      | TUC_KON_219<br>"Entschlüssele"  |
|                      | do_Reset                 | → |                   | TUC_KON_024<br>"Karte zurücksetzen"                                     |
|                      | erase_Card_Data          | → |                   | TUC_KON_211<br>„LöschRecordInhalt“<br>TUC_KON_204<br>„LöschDateiInhalt“ |
|                      | extract_card_data        | → | Zertifikatsdienst | TUC_KON_034<br>"Zertifikatsinformationen extrahieren"                   |
|                      | read_Card_Data           | → | Kartendienst      | TUC_KON_202<br>"LeseDatei"  |
|                      |                          | → |                   | TUC_KON_209<br>"LeseRecord"   |
|                      |                          | → |                   | TUC_KON_215<br>"SucheRecord"  |
|                      | read_KVK                 | → |                   | TUC_KON_202<br>"Lese Datei"   |
|                      | send_APDU                | → |                   | TUC_KON_200<br>"SendeAPDU"  |
|                      | sign_Data                | → |                   | TUC_KON_218<br>"Signiere"   |
|                      | verify_eGK               | → |                   | TUC_KON_018<br>"eGK-Sperrung prüfen"                                    |
|                      | write_Card_Data          | → |                   | TUC_KON_203<br>"SchreibeDatei"  |

|                            |                           |   |                              |   |
|----------------------------|---------------------------|---|------------------------------|---|
|                            |                           | → |                              | TUC_KON_210<br>"SchreibeRecord"                   |
|                            |                           | → |                              | TUC_KON_214<br>"FügeHinzuRecord"                  |
|                            | write_eGK_Protocol        | → |                              | TUC_KON_006<br>"Datenzugriffsaudit eGK schreiben" |
| I_KV_Card_Reservati<br>on  | handle_Session            | → | Kartendienst                 | TUC_KON_023<br>"Karte reservieren"                |
| I_KV_Card_Unlockin<br>g    | authorize_Card            | → | Kartendienst                 | TUC_KON_005<br>"Card-to-Card authentisieren"      |
|                            | change_PIN                | → |                              | TUC_KON_019<br>"PIN ändern"                       |
|                            | enable_PIN<br>disable_PIN | → |                              | TUC_KON_027<br>„PIN-Schutz ein-/ ausschalten“     |
|                            | do_C2C                    | → |                              | TUC_KON_005<br>"Card-to-Card authentisieren"      |
|                            | get_PIN_Status            | → |                              | TUC_KON_022<br>"Liefere PIN-Status"               |
|                            | initialize_PIN            | → |                              | TUC_KON_019<br>"PIN ändern"                       |
|                            | unblock_PIN               | → |                              | TUC_KON_021<br>"PIN entsperren"                   |
|                            | verify_PIN                | → |                              | TUC_KON_012<br>"PIN verifizieren"                 |
| I_Notification_From_<br>FM | notify                    | → | Systeminformatio<br>nsdienst | TUC_KON_256<br>"Systemereignis absetzen"          |

|                           |                                       |   |                         |   |
|---------------------------|---------------------------------------|---|-------------------------|---|
| I_Local_Storage           | write_Data<br>read_Data<br>erase_Data | → | Konnektormanagement     | TIP1-A_5484                                     |
| I_Poll_System_Information | get_Ressource_Information             | → | Systeminformationdienst | TUC_KON_254<br>"Liefere Ressourcendetails"      |
|                           | get_Ressource_List                    | → |                         | TUC_KON_252<br>"Liefere KT_Liste"               |
|                           | get_Ressource_List                    | → |                         | TUC_KON_253<br>"Liefere Karten_Liste"           |
| I_Reg_Notification        | register_for_Notifications            | → | ---                     | --- keine Umsetzung notwendig. Erfolgt implizit |
| I_Role_Information        | get_Role                              | → | Kartendienst            | TUC_KON_036<br>„LiefereFachliche Rolle“         |
| I_SAK_Operations          | sign_Document_QES                     | → | Signaturdienst          | TUC_KON_150<br>„Dokumente QES signieren“        |
|                           | verify_Document_QES                   | → |                         | TUC_KON_151<br>"QES Dokumentensignatur prüfen"  |
| I_Sign_Operations         | sign_Document                         | → | Signaturdienst          | TUC_KON_160<br>„Dokumente nonQES signieren“     |
|                           | external_Authenticate                 | → |                         | TUC_KON_160<br>„Dokumente nonQES signieren“     |
|                           | verify_Document                       | → |                         | TUC_KON_161<br>„nonQES Dokumentsignatur prüfen“ |

|                            |                            |   |                        |   |
|----------------------------|----------------------------|---|------------------------|---|
|                            | get_Certificate            | → | Kartendienst           | TUC_KON_216<br>„LeseZertifikat“   |
| I_Symm_Crypt_Operations    | decrypt_Document_Symmetric | → | Verschlüsselungsdienst | TUC_KON_073<br>"Daten symmetrisch entschlüsseln"                            |
|                            | encrypt_Document_Symmetric | → |                        | TUC_KON_072<br>"Daten symmetrisch verschlüsseln"                            |
| I_Synchronised_System_Time | get_Time                   | → | Zeitdienst             | TUC_KON_351<br>"Liefere Systemzeit"   |
| I_TLS_Client               | send_Secure                | → | TLS-Dienst             | TUC_KON_110<br>"Kartenbasierte TLS-Verbindung aufbauen"                     |
|                            |                            | → |                        | TUC_KON_111<br>"Kartenbasierte TLS-Verbindung abbauen"                      |
|                            |                            | → | Anbindung LAN/WAN      | AFOs: Routing der IP-Pakete von Fachmodul (=Konnektor intern)<br>--> VPN_TI |
| I_Directory_Query          | search_Directory           | → | LDAP-Proxy             | TUC_KON_290<br>„LDAP-Verbindung aufbauen“                                   |
|                            |                            | → |                        | TUC_KON_291<br>„Verzeichnis abfragen“                                       |
|                            |                            | → |                        | TUC_KON_292<br>„LDAP-Verbindung trennen“                                    |

|                   |                         |   |   |   |
|-------------------|-------------------------|---|---|---|
|                   |                         | → |   | TUC_KON_293<br>„Verzeichnisabfrage<br>abbrechen“                  |
| I_KSRC_FM_Support | list_available_Packages | → | Software-Aktualisierung<br>(KSR-Client) | TUC_KON_285<br>„UpdateInformationen für<br>Fachmodul<br>beziehen“ |
|                   | load_Package            | → |   | TUC_KON_286<br>„Paket für<br>Fachmodul<br>laden“                  |

8968

8969

8970

**Tabelle 388: TAB\_KON\_712 Architektur der TI-Plattform, Berechtig Clientssysteme**

| Interface             | Operation                    | → Funktionsmerkmal                     | Interface:Operation                                      |
|-----------------------|------------------------------|--|--|
| I_Crypt_Operations    | decrypt_Document             | → Verschlüsselungsdienst               | EncryptionService:DecryptDocument                        |
|                       | encrypt_Document             | →                                      | EncryptionService:EncryptDocument                        |
| I_DNS_Name_Resolution | get_FQDN                     | → Namensdienst und Dienstlokalisierung | GetFQDN  |
|                       | get_IP_Address               | →                                      | GetIPAddress   |
| I_IP_Transport        | send_Data_External           | → Anbindung LAN/WAN                    | AFOs:<br>Routing der IP-Pakete von Client<br>--> VPN_SIS |
| I_KV_Card_Handling    | discard_Card_Usage_Reference | → ---                                  | --- keine Umsetzung notwendig.<br>Erfolgt implizit       |
|                       | get_Card_Usage_Reference     | → ---                                  | --- keine Umsetzung notwendig.<br>Erfolgt implizit       |

|                           |                            |   |                           |                                      |
|---------------------------|----------------------------|---|---------------------------|--------------------------------------|
| I_KV_Card_Unlocking       | change_PIN                 | → | Kartendienst              | CardService :ChangePin               |
|                           | disable_PIN                | → |                           | CardService :EnablePin               |
|                           | enable_PIN                 | → |                           | CardService :DisablePin              |
|                           | get_PIN_Status             | → |                           | CardService :GetPinStatus            |
|                           | initialize_PIN             | → |                           | CardService :ChangePin               |
|                           | unlock_PIN                 | → |                           | CardService :UnlockPin               |
|                           | verify_PIN                 | → |                           | CardService :VerifyPin               |
| I_Poll_System_Information | get_Ressource_Information  | → | Systeminformationssdienst | EventService :GetResourceInformation |
|                           | get_Ressource_List         | → |                           | EventService :GetCardTerminals       |
|                           | get_Ressource_List         | → |                           | EventService :GetCards               |
| I_Reg_Notification        | register_for_Notifications | → | Systeminformationssdienst | EventService :Subscribe              |
|                           |                            | → |                           | EventService :Unsubscribe            |
|                           |                            | → |                           | EventService :GetSubscription        |
| I_SAK_Operations          | sign_Document_QES          | → | Signaturdienst            | SignatureService :SignDocument       |
|                           | verify_Document_QES        | → |                           | SignatureService :VerifyDocument     |
| I_Sign_Operations         | sign_Document              | → | Signaturdienst            | SignatureService :SignDocument       |

|                            |                       |   |                              |   |
|----------------------------|-----------------------|---|------------------------------|---|
|                            | verify_Document       | → |                              | SignatureService<br>:VerifyDocument                   |
|                            | external_Authenticate | → | Authentifizierungs<br>dienst | AuthSignatureSer<br>vice<br>:ExternalAuthenti<br>cate |
|                            | get_Certificate       | → | Zertifikatsdienst            | CertificateService<br>:ReadCardCertific<br>ate        |
| I_NTP_Time_Infor<br>mation | sync_Time             | → | Zeitdienst                   | I_NTP_Time_Infor<br>mation<br>:sync_Time              |
| I_Directory_Query          | search_Directory      | → | LDAP-Proxy                   | LDAP-Operation<br>(TIP1-A_5521)                       |

8971  
8972  
8973

Tabelle 389: TAB\_KON\_713 Architektur der TI-Plattform, Berechtig eHealth-KT

| Interface      | Operation | → | Funktionsmerkmal | Interface:Operation                              |
|----------------|-----------|---|------------------|--|
| I_Notification | notify    | → | SICCT            | Ereignisdienst<br>:SICCT-<br>Ereignisnachrichten |
|                |           | → | SICCT            | Ereignisdienst<br>:ServiceAnnouncement           |

8974  
8975  
8976

Tabelle 390: TAB\_KON\_714 Architektur der TI-Plattform, Berechtig Administrator

| Interface                         | Operation               | -<br>> | Funktionsmerkmal                            | Interface:Operation   |
|-----------------------------------|-------------------------|--------|---|---|
| I_Change_System_Ti<br>me          | set_System_Ti<br>me     | -<br>> | Zeitdienst                                  | TIP1-A_4793<br>Konfigurierbarkeit<br>des<br>Konnektor NTP-<br>Servers |
| I_Facade_Access_Conf<br>iguration | add_Clientsyste<br>m    | -<br>> | Fachliche<br>Anbindung der<br>Clientsysteme | TIP1-A_4518<br>Konfiguration der<br>Anbindung<br>Clientsysteme        |
|                                   | remove_Clients<br>ystem |        |   |   |

|                         |                        |  |  |   |
|-------------------------|------------------------|--|--|---|
|                         | set_CS_Access_Mode     |  |  |   |
| I_KSRC_Local_Management | do_local_Update        | - Software-<br>> Aktualisierung (KSR-Client) |  | TUC_KON_280<br>"Konnektoraktualisierung durchführen"  |
| I_KSRC_Management       | do_Update              | - Software-<br>> Aktualisierung (KSR-Client) |  | TUC_KON_280<br>"Konnektoraktualisierung durchführen"  |
|                         |                        |  |  | TUC_KON_281<br>"Kartenterminalaktualisierung anstoßen"  |
|                         | list_available_Updates |  |  | TUC_KON_282<br>"Update Informationen beziehen"  |
| I_KTV_Management        | configure_KTs          | - Kartenterminalverwaltung<br>>              |  | Managementschnittstelle<br>:TIP1-A_4555<br>Manuelles Hinzufügen eines Kartenterminals         |
|                         |                        |  |  | Managementschnittstelle<br>:TIP1-A_4540<br>Reaktion auf KT Service Announcement               |
|                         |                        |  |  | Managementschnittstelle<br>:TIP1-A_4556<br>Pairing mit Kartenterminal durchführen             |
|                         |                        |  |  | Managementschnittstelle<br>:TIP1-A_4557<br>Ändern der Korrelationswerte eines Kartenterminals |



---

## 8977 **109 Anhang I – Umsetzungshinweise (informativ)**

---

8978 In diesem Anhang finden sich Darstellungen und Informationen, die ein  
8979 Konnektorhersteller zur Umsetzung der normativen Anforderungen in ein konkretes  
8980 Produkt berücksichtigen kann. Sie wurden im Rahmen der Erhebung der normativen  
8981 Anforderungen erarbeitet, um die Umsetzbarkeit der Anforderungen zu bestätigen.  
8982 Dieser Anhang soll als Unterstützung für eine Umsetzung verstanden werden und erhebt  
8983 keinen Anspruch auf Korrektheit und Vollständigkeit.

### 8984 **10.19.1 Systemüberblick**

#### 8985 **10.1.19.1.1 – Hinweise zur Sicherheitsevaluierung nach Common** 8986 **Criteria**

8987 Gemäß dem Sicherheitskonzept des Konnektors [gemKPT\_Sich\_Kon] muss die Software  
8988 des Konnektors nach Common Criteria (CC) evaluiert und geprüft werden.

8989 Diese Software erbringt Sicherheitsleistungen in zwei wesentlichen Funktionsblöcken.  
8990 Durch diese Aufteilung ist es möglich, dass die einzelnen Funktionsblöcke zeitlich  
8991 voneinander unabhängig bzw. sogar von unterschiedlichen Herstellern implementiert,  
8992 evaluiert und geprüft werden können. Es werden zwei Schutzprofile (Protection Profile)  
8993 für die Funktionsblöcke des Konnektors erstellt. Es handelt sich dabei um die  
8994 Schutzprofile des Netzkonnektors (KONN.NK) sowie des Anwendungskonnektors  
8995 (KONN.AK) inklusive der Signaturanwendungskomponente. Das Schutzprofil des  
8996 Sicherheitsmoduls für den Konnektor (SM-K) wird in diesem Kapitel nicht betrachtet.

8997 Diese Schutzprofile definieren eine implementierungsunabhängige Menge von  
8998 Sicherheitsanforderungen für die einzelnen Konnektorfunktionsblöcke bzw.  
8999 Konnektorbestandteile. Anhand dieser Schutzprofile werden von den Herstellern der  
9000 Konnektoren die Sicherheitsvorgaben (Security Targets) für die konkreten Umgebungen  
9001 erstellt, welche als Eingangsdokumente für den Zertifizierungsprozess der jeweiligen  
9002 konkreten Komponenten eingesetzt werden. Diese zu evaluierenden Komponenten  
9003 werden als Evaluierungsgegenstand (EVG) bezeichnet.

#### 9004 **10.1.1.19.1.1.1 Separationsmechanismen des Konnektors**

9005 Damit es nach einer erfolgreichen Evaluierung eines Konnektors auch weiterhin möglich  
9006 bleibt, Software oder Daten, die keinen direkten Einfluss auf Sicherheitsfunktionen der  
9007 EVGs aufweisen, ohne eine Re-Evaluierung definiert auszutauschen, hinzuzufügen oder  
9008 zu erweitern, ist eine Separation der Komponenten des EVG dringend anzuraten.

9009 Implementiert der Hersteller keine bzw. nicht ausreichende Separationsmechanismen, so  
9010 ist bei bestimmten Update-Arten von einer aufwändigen Re-Evaluierung des  
9011 entsprechenden EVGs auszugehen. Die Separation dient also der Trennung zwischen  
9012 ausführbarem Code des EVG, welcher Sicherheitsfunktionen umsetzt, und zusätzlichem  
9013 ausführbarem Code auf dem Konnektor, welcher keine Sicherheitsfunktionen umsetzt.

9014 Die Wahl der Separationsmechanismen steht dem Hersteller frei und muss in den  
9015 Sicherheitsvorgaben für den EVG beschrieben und als solcher evaluiert werden. Aus  
9016 diesen Sicherheitsvorgaben ergibt sich auch, welche Update-Arten bei welchen  
9017 Separationsmechanismen eine Re-Evaluierung des EVG erfordern und wie aufwendig  
9018 diese Re-Evaluierung ausfällt.

9019 Unter diese Update-Arten können beispielsweise – je nach Konnektorarchitektur, CC-  
 9020 Dokumentation oder Konnektorimplementierung – Bestandteile des unter dem Konnektor  
 9021 arbeitenden Betriebssystems, die Installation dezentraler Komponenten von Fachlogik  
 9022 oder Konfigurationsdaten des Konnektors fallen.

9023 Als Beispiel für Separationsmechanismen sei auf die folgende informative Aufzählung  
 9024 verwiesen, welche jedoch keinen Anspruch auf Vollständigkeit besitzen kann und nur  
 9025 mögliche Alternativen aufzeigt:

- 9026 • Java-Sandbox-Konzept,
- 9027 • Interpreter mit restriktiver Laufzeitprüfung,
- 9028 • vom Betriebssystem bereitgestellte Prozess- und Speichertrennung,
- 9029 • virtuelle Maschinen,
- 9030 • physische Trennung durch separierte Hardware.

9031 Je nach gewähltem Architekturansatz des Herstellers sind nicht alle hier genannten  
 9032 Alternativen für die Separation des EVG auf dem Konnektor anwendbar.

9033 Insbesondere sollte der Hersteller den eigentlichen Update-Prozess und die dafür  
 9034 verantwortliche Komponente mit besonderer Sorgfalt beschreiben, spezifizieren und  
 9035 implementieren. Bei einer fehlerhaften Implementierung dieser Komponente besteht die  
 9036 Gefahr einer Schwächung oder des Ausschaltens von Sicherheitsfunktionen des EVG. Die  
 9037 Update-Komponente muss eine sichere Zuweisung der Updates zu den separierten  
 9038 Bestandteilen des EVGs gewährleisten. Auch ist zu betonen, dass der EVG immer die  
 9039 Integrität der Daten des Updates und die Authentizität des Absenders prüfen muss, bevor  
 9040 ein Update akzeptiert wird. Der Update-Komponente muss somit besondere Beachtung  
 9041 geschenkt werden.

## 9042 **10.1.1.29.1.1.2 Granularität der TSF**

9043 Die TSF (TOE Security Functionality) eines EVG besteht aus Subsystemen und Modulen,  
 9044 wobei ein Modul die genaueste Beschreibung einer Funktionalität darstellt und unterhalb  
 9045 der Subsysteme angesiedelt ist. Subsysteme beschreiben das Design des EVG und  
 9046 können wiederum – je nach Komplexität eines EVGs – aus weiteren Subsystemen  
 9047 bestehen. Ein Entwickler sollte außer der Modulbeschreibung keine weiteren  
 9048 Informationen zur Implementierung der dort beschriebenen Funktionalität benötigen.

9049 Die Subsysteme und Module der TSF gliedern sich in drei Klassen:

- 9050 (a) SFR-Enforcing Subsysteme und Module. Hierunter fallen die Subsysteme und  
 9051 Module, welche eine funktionale Sicherheitsanforderung direkt durchsetzen.
- 9052 (b) SFR-Supporting Subsysteme und Module. Hierunter fallen die Subsysteme und  
 9053 Module, welche bei der Durchsetzung einer funktionalen Sicherheitsanforderung  
 9054 unterstützend wirken.
- 9055 (c) SFR-Non-Interfering Subsysteme und Module. Hierunter fallen die Subsysteme  
 9056 und Module, welche keine Leistung bei der Erfüllung einer funktionalen  
 9057 Sicherheitsanforderung erbringen.

9058 Sollte nach einer erfolgreichen CC-Evaluierung eines Konnektors die Notwendigkeit zur  
 9059 Änderung der Software des Konnektors gegeben sein, so ist unter Umständen eine Re-  
 9060 Evaluierung des EVG erforderlich. Diese Notwendigkeit kann sich aus der Behebung von  
 9061 nachträglich erkannten Fehlern, aufgetretenen Sicherheitslücken, Schwächen eines  
 9062 Standardverfahrens oder einer erforderlichen Erweiterung der Funktionalität ergeben.

9063 Im Rahmen der Aufzählung der Anforderungen an die Beschreibung des EVG-Design  
 9064 (ADV\_TDS) wird bereits die Aufteilung der TSF auf Subsysteme und Module beschrieben.  
 9065 Trotzdem soll hiermit ausdrücklich geraten werden, die Aufteilung der TSF auf die  
 9066 Subsysteme und Module selbst und die Aufteilung der Subsysteme und Module auf die  
 9067 drei o. g. Klassen möglichst feingranular durchzuführen.

9068 Denn so

- 9069 1. können einfacher umfassende Tests durchgeführt und die Testabdeckung  
 9070 sichergestellt werden,
- 9071 2. kann bei der Veränderung von Programmcode der Evaluator die Auswirkungen auf  
 9072 SFR-Enforcing, Supporting oder Non-Interfering SFRs einfacher herausfinden und  
 9073 damit die Kosten und den zeitlichen Aufwand einer Re-Evaluierung senken.
- 9074 3. kann bei der Veränderung von Programmcode, welcher als SFR-Non-Interfering  
 9075 eingestuft wird, das Maintenance-Verfahren anstelle einer Re-Evaluierung  
 9076 angewandt werden, welches einen erheblichen zeitlichen und damit auch  
 9077 monetären Vorteil gegenüber dem Re-Evaluierungsverfahren darstellt.

## 9078 **10-29.2 Übergreifende Festlegungen**

### 9079 **10-2-19.2.1 Interne Mechanismen**

#### 9080 **10-2-1-19.2.1.1 Zufallszahlen und Schlüssel**

9081 Der Konnektor kann zur Erzeugung von Zufallszahlen und Einmalschlüsseln einen  
 9082 Hardware- oder Software-Generator verwenden. Als Quelle für Zufallszahlen kann der  
 9083 Konnektor die gSMC-K verwenden.

## 9084 **10-39.3 Funktionsmerkmale**

### 9085 **10-3-19.3.1 Anwendungskonnektor**

#### 9086 **10-3-1-19.3.1.1 Administration des Informationsmodells**

9087 Wie die Administration der persistenten Entitäten und Beziehungen des  
 9088 Informationsmodells im Detail über die bereitzustellende Administrationsoberfläche  
 9089 erfolgt, entscheidet der Hersteller.

9090 Es wird folgende Reihenfolge für die Pflege des Informationsmodells empfohlen.

- 9091 1. Mandantenübergreifende Administration:
  - 9092 • Es werden die Entitäten Arbeitsplätze, Clientsysteme mit  
 9093 Authentifizierungsmerkmalen CS-AuthMerkmal und SMC-B\_Verwaltet  
 9094 erfasst.  
 9095 Die Eingabe der Kartenterminals erfolgt über die Kartenterminalverwaltung.
  - 9096 • Es wird die Beziehungen zwischen Arbeitsplatz und Kartenterminals „lokal“ und  
 9097 „entfernt (zentral)“ gepflegt.
- 9098 2. Mandantenbezogene Administration:
  - 9099 • Die Definition bzw. Auswahl eines Mandanten bildet den Einstiegspunkt.

- Pro Mandanten werden aus den bereits eingepflegten Entitäten „Kartenterminal“, „Arbeitsplatz“, „Clientsystem“, „SMC-B\_Verwaltet“ die für den Mandanten im Zugriff erlaubten zugeordnet.
- Pro Mandant erfolgt eine Zuordnung der Arbeitsplätze zu Clientsystemen.
- Pro Mandant erfolgt eine Zuordnung der lokalen Kartenterminals, über die jeweils pro Arbeitsplatz die Eingabe der Remote-PIN erfolgen darf.

### **10.3.1.29.3.1.2 Vorgehensvariante für das Handling von CardSessions**

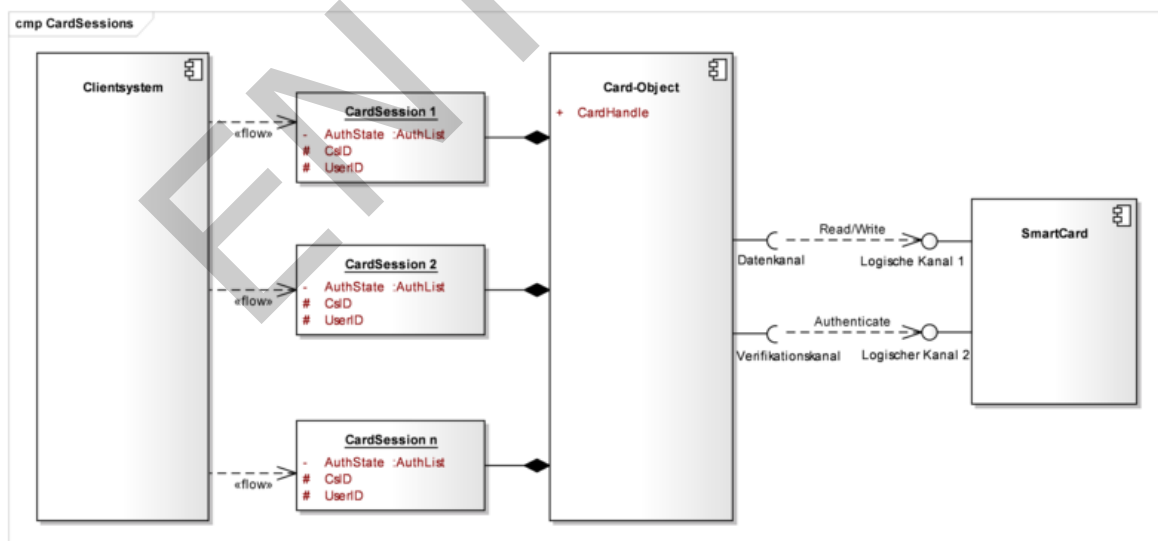
Das in der [TIP1-A\_4560] „Rahmenbedingungen für Kartensitzungen“ geforderte Verhalten, ließe sich über folgenden Mechanismus umsetzen:

Verschiedene Clientsystem (oder verschiedene Nutzer an einem Clientsystem) möchten auf Daten der über CardHandle adressierten Smartcard zugreifen.

Für die Zugriffe müssen, je nach Definition der Zugriffsbedingung in der Zielkarte, bestimmte Sicherheitszustände erreicht werden (durch Verifikation einer PIN oder durch C2C). Diese erreichten Sicherheitszustände werden innerhalb einer Karte jeweils an einen logischen Kanäle (bzw. den Basiskanal) gebunden, d. h., das Erhöhen oder Absenken eines Sicherheitszustands wirkt nicht außerhalb des logischen Kanals, in dem die Veränderung verursacht wurde.

Finden nun Clientsystemzugriffe in unterschiedlichen Kontexten (Mandant, Clientsystem, Arbeitsplatz und Nutzer verschieden) auf die gleiche Karte statt, so muss sichergestellt sein, dass PIN-Eingaben und durchgeführte C2C nur für den Kontext wirksam sind, in welchem sie durchgeführt wurden. Dies ließe sich erreichen, wenn jeder Kontext auf einen eigenen logischen Kanal der Karte abgebildet würde. Leider unterstützen der HBA und die SMC-B nur vier, die eGK nur einen logischen Kanal. Mehrere gleichzeitige unterschiedliche Kontexte wären somit nicht möglich.

Eine mögliche Lösung für beliebig viele gleichzeitige Kontexte:



**Abbildung 23: PIC\_KON\_120 Abbildung von CardSessions auf logische Kanäle**

Der Kartendienst fungiert als Multiplexer. Er spiegelt die Zugriffsrechte der Karte und wendet deren Regeln selbständig gegen die unterschiedlichen Zugriffe durch Clientsysteme an.

- 9132 Für jedes Card-Objekt wird „in Richtung Clientsystem“ pro Kontext genau eine  
 9133 CardSession erzeugt und verwaltet. Zugriffe des Clientsystems erfolgen somit „im  
 9134 Kontext“ einer CardSession.
- 9135 In Richtung Karte verwendet das Card-Object genau zwei Kanäle (zwei logische oder  
 9136 einen logischen und einen Basiskanal):
- 9137 • Einen Datenkanal, über den die Datenbewegungen und kryptographischen  
 9138 Operationen laufen und
  - 9139 • Einen Verifikationskanal, der ausschließlich für Authentisierungszwecke verwendet  
 9140 wird
- 9141 In jeder CardSession werden die in ihrem Kontext erreichten Sicherheitszustände  
 9142 vermerkt. Das Vorgehen für die Durchführung der Verifikationen und des Vermerkens der  
 9143 erreichten Sicherheitszustände, sowie der Datenzugriffe folgt folgenden Regeln (hier für  
 9144 PIN-Verifikation, sinngleich auch für C2C):
- 9145 • Soll über eine CardSession eine PIN-Verifikation für PinRef\_A gegen eine Karte  
 9146 durchgeführt werden und der erhöhte Sicherheitszustand für PinRef\_A ist noch  
 9147 nicht erreicht (bsp. direkt nach einem Karten-Reset), dann leite die Verifikation  
 9148 über den Datenkanal (initiale Freischaltung des Datenkanals für folgende  
 9149 Datenzugriffe).
  - 9150 • Soll über eine CardSession eine PIN-Verifikation für PinRef\_A gegen eine Karte  
 9151 durchgeführt werden und der erhöhte Sicherheitszustand für PinRef\_A ist auf dem  
 9152 Datenkanal bereits erreicht, dann leite die Verifikation über den  
 9153 Verifikationskanal.
  - 9154 • Wurde durch eine CardSession eine Verifikation für PinRef\_A erfolgreich  
 9155 durchgeführt, wird dieser erreichte Sicherheitszustand für PinRef\_A in der  
 9156 zugreifenden CardSession vermerkt
  - 9157 • Datenzugriffe auf oder Kryptooperationen mit Karten werden durch den  
 9158 Kartendienst nur zugelassen, wenn die zugreifende CardSession über einen für  
 9159 diese Zugriffe benötigten erhöhten Sicherheitszustand verfügt. Ist der benötigte  
 9160 Vermerk für die zugreifende CardSession nicht vorhanden, beantwortet der  
 9161 Kartendienst die Anfrage mit der passenden Kartenfehlermeldung. Es erfolgt keine  
 9162 Interaktion mit der Karte.
- 9163 Diese Regeln führen dazu, dass eine durch CardSession Y fehlgeschlagene Verifikation für  
 9164 PinRef\_A die zuvor erfolgreich durch CardSession X durchgeführte Verifikation nicht  
 9165 beeinflusst. Kartenzugriffe auf dem Datenkanal sind für CardSession X weiterhin möglich,  
 9166 da der Verlust des erhöhten Sicherheitszustands durch fehlerhafte Verifikation immer nur  
 9167 im Verifikationskanal erfolgt.
- 9168 Dieser Mechanismus funktioniert mit zwei Kanälen zu einer Karte für beliebig viele  
 9169 CardSessions.

### 9170 **10.3.1.39.3.1.3 Darstellung von Terminal-Anzeigen auf einem** 9171 **Kartenterminal**

- 9172 Die folgenden Ausführungen dienen der Klarstellung für die korrekte Verwendung der zur  
 9173 Verfügung stehenden Datenobjekte (DO) zur Darstellung von Terminal-Anzeigen an  
 9174 einem Kartenterminal nach SICCT- und eHealth-Kartenterminal-Spezifikation.
- 9175 Die SICCT-Spezifikation enthält eine Liste von Datenobjekten (DO), die von den  
 9176 Kartenterminals unterstützt werden müssen oder können. Dabei gibt es zwei  
 9177 Datenobjekte zur Anzeige von Terminal-Anzeigen: APPL DO und SMTBD DO.

9178 Kartenterminals müssen APPL DO (steht für Application Label Data Object) unterstützen.  
9179 APPL DOs müssen immer eine 7 Bit ISO646DE/DIN66003-Codierung enthalten  
9180 [DIN66003].

9181 Kartenterminals können SMTBD DO (steht für SICCT Message-To-Be-Displayed Data  
9182 Object) unterstützen, müssen dieses aber nicht. Über SMTBD DOs können weitere  
9183 Zeichensätze am Display angezeigt werden.

9184 Der Konnektor soll APPL DOs verwenden. Er kann SMTBD DOs verwenden, wenn er  
9185 sicherstellt, dass das angesteuerte Kartenterminal diese unterstützt und die dargestellte  
9186 Meldung der Klartextmeldung entspricht, die mittels APPL DO erreicht worden wäre.

9187 Um dem Kartenterminal den Umbruch längerer Texte über das Zeilenende hinaus zu  
9188 erleichtern, enthalten die Terminal-Anzeigen das Zeichen 0x0B als „Soll-  
9189 Zeilenumbrüche“. Die „Soll-Zeilenumbrüche“ werden nicht als Textzeichen gezählt. Sie  
9190 zeigen einen potentiellen Zeilenumbruch an. Diese müssen vom Kartenterminal  
9191 herausgefiltert werden und werden nicht durch andere Zeichen ersetzt.

9192 Die Maximallänge für Terminal-Anzeigen beträgt ohne PIN-Eingabe (OUTPUT [O]) 48  
9193 Zeichen.

9194 Besonderheit bei Terminal-Anzeigen, die zu einer PIN-Eingabe (INPUT [I]) auffordern:

9195 Für die PIN-Eingabe wird eine strukturierte Terminal-Anzeige übergeben, aufgeteilt auf  
9196 maximal 40 Zeichen für die Terminal-Anzeige plus maximal 10 Zeichen für den sog. PIN-  
9197 Prompt (bei Platz für zusätzliche 6 Zeichen für die PIN-Eingabe). Ein gültiger String hat  
9198 die Form: <Terminal-Anzeige>0x0F<PIN-Prompt>. Auch die Terminal-Anzeige für  
9199 Eingaben soll mit „Soll-Zeilenumbrüchen“ versehen werden.

9200 Bei der Übertragung der Terminal-Anzeige ist auf die korrekte Codierung der  
9201 Zeichenkette zu achten. Der einzige Zeichensatz, der von allen Kartenterminals  
9202 unterstützt werden MUSS, ist (7 Bit) ISO646DE/DIN66003 [DIN66003]. Dadurch darf  
9203 eine Terminal-Anzeige auch deutsche Sonderzeichen enthalten.

9204  
9205

| Hex<br>Cod<br>e | ...0   | ...<br>1 | ...<br>2 | ...<br>3 | ...<br>4 | ...<br>5 | ...<br>6 | ...<br>7 | ...<br>8 | ...<br>9 | ...<br>A | ...<br>B | ...<br>C | ...<br>D | ...<br>E | ...<br>F        |
|-----------------|--|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|----------|-----------------|
| 0...            | <i>diverse Steuerzeichen - nicht verwendet -</i> |          |          |          |          |          |          |          |          |          |          |          |          |          |          |                 |
| 1...            | <i>diverse Steuerzeichen - nicht verwendet -</i> |          |          |          |          |          |          |          |          |          |          |          |          |          |          |                 |
| 2...            | <i>space</i>                                     | !        | "        | #        | \$       | %        | &        | '        | (        | )        | *        | +        | ,        | -        | .        | /               |
| 3...            | 0  | 1        | 2        | 3        | 4        | 5        | 6        | 7        | 8        | 9        | :        | ;        | <        | =        | >        | ?               |
| 4...            | §  | A        | B        | C        | D        | E        | F        | G        | H        | I        | J        | K        | L        | M        | N        | O               |
| 5...            | P  | Q        | R        | S        | T        | U        | V        | W        | X        | Y        | Z        | Ä        | Ö        | Ü        | ^        | —               |
| 6...            | `  | a        | b        | c        | d        | e        | f        | g        | h        | i        | j        | k        | l        | m        | n        | o               |
| 7...            | p  | q        | r        | s        | t        | u        | v        | w        | x        | y        | z        | ä        | ö        | ü        | ß        | <i>de<br/>/</i> |

**Abbildung 24: PIC\_KON\_007 Übersicht Zeichensatz ISO646DE/DIN66003**

9206

9207

9208

ENTWURF



## 11.10 Anhang K – Szenarien im dezentralen Umfeld

Die folgenden Szenarien für den Einsatz der Produkte der Telematikinfrastruktur beschreiben informativ Varianten und Optionen, die durch die Spezifikationen abgedeckt werden.

Die vorliegenden Abbildungen in diesem Anhang fokussieren auf das dezentrale Umfeld und verzichten daher auf die Darstellung der zentralen Anteile, wie das zentrale Netzwerk der Telematikinfrastruktur, welches über den „VPN-Konzentrator TI“ erreichbar ist.

Der Konnektor, sowie die Netzwerkkomponenten Switch und IAG (Internet Access Gateway) sind in den folgenden Szenarien zum Schutz vor unerlaubtem Zugriff gemäß den Annahmen des Sicherheitskonzeptes vor unbefugten physischen Zugriffen geschützt installiert.

Die folgenden Abschnitte stellen jeweils ein Szenario in der Übersicht als Diagramm, eine Beschreibung sowie eine kurze Auflistung der Voraussetzungen und Auswirkungen dar.

### 11.10.1 Szenario 1: Einfache Installation ohne spezielle Anforderungen und ohne bestehende Infrastruktur

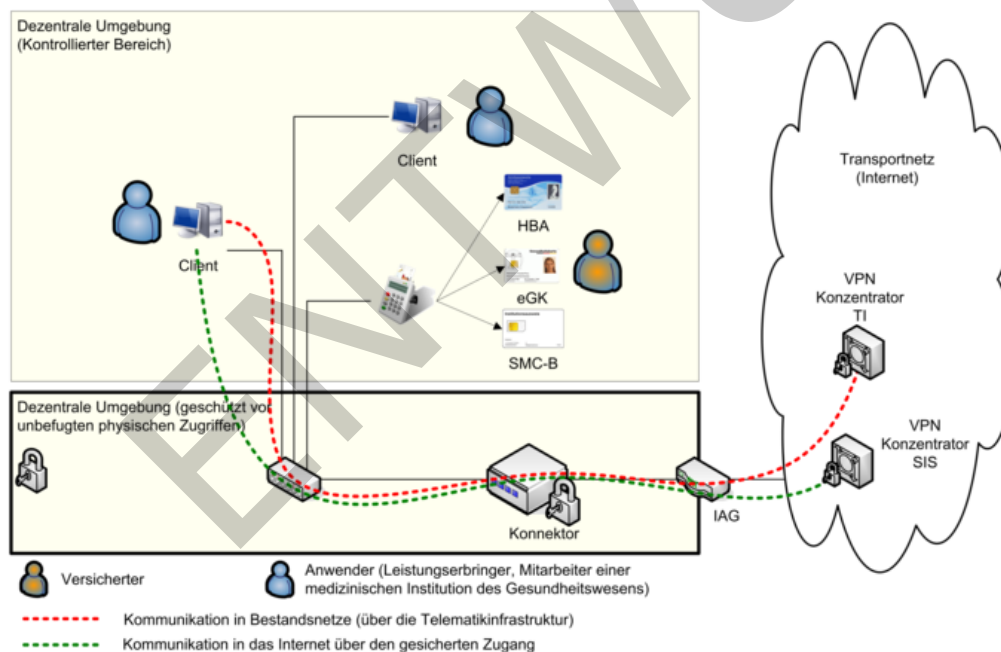


Abbildung 25: Szenario einer einfachen Installation

#### 11.1.10.1.1 Beschreibung des Szenarios

Abbildung 25 zeigt ein einfaches Szenario für das dezentrale Umfeld. Es wird der Konnektor als Default-Gateway für jegliche IP-Kommunikation aus dem LAN in das WAN eingesetzt. Dabei übernimmt der Konnektor das Routing der Kommunikation in das Internet über den SIS (Secure Internet Service) und in die an die TI angeschlossenen Bestandsnetze. Die Bezeichnung IAG (Internet Access Gateway) steht für die Geräte, die



9233 den Internetzugang ermöglichen und typischerweise vom Internet Service Provider (ISP)  
9234 zur Verfügung gestellt werden (z.B. DSL Router und DSL Modem).

9235 Ein oder mehrere Clientsysteme können über den Konnektor Anwendungsfälle der  
9236 Telematikinfrastruktur initiieren und über den Konnektor und die zentrale TI-Plattform in  
9237 Bestandsnetze kommunizieren (rote gestrichelte Linie). Dabei ist die Nutzung der  
9238 Anwendungsfälle der Telematikinfrastruktur je nach Konfiguration des Konnektors  
9239 entweder nur authentifizierten Clients möglich oder beliebigen Clients.

9240 In diesem einfachen Szenario werden über ein einziges Kartenterminal die SMC-B, der  
9241 HBA und auch die eGK des Versicherten gelesen, es können dazu alternativ auch  
9242 mehrere Kartenterminals genutzt werden.

9243 Darüber hinaus können die Clientsysteme über den SIS (Secure Internet Service) auf  
9244 Dienste des Internets zugreifen.

### 9245 ~~11.1.2~~ 10.1.2 Voraussetzungen

- 9246 • Anbindung der bestehenden Clientsysteme an ein zum Konnektor kompatibles  
9247 LAN muss möglich sein.
- 9248 • Konfiguration des Konnektors als Default-Gateway in den Clientsystemen und  
9249 Konfiguration der notwendigen VPN-Tunnel im Konnektor, um in die  
9250 verschiedenen Netze zu routen.
- 9251 • Verfügbarkeit einer SMC-B

### 9252 ~~11.1.3~~ 10.1.3 Auswirkungen

- 9253 • Die Clientsysteme können über den Konnektor Anwendungsfälle der TI initiieren
- 9254 • Die Clientsysteme können über den Konnektor auf das Internet und  
9255 Bestandsnetze zugreifen

## 11.2.10.2 Szenario 2: Installation mit mehreren Behandlungsräumen

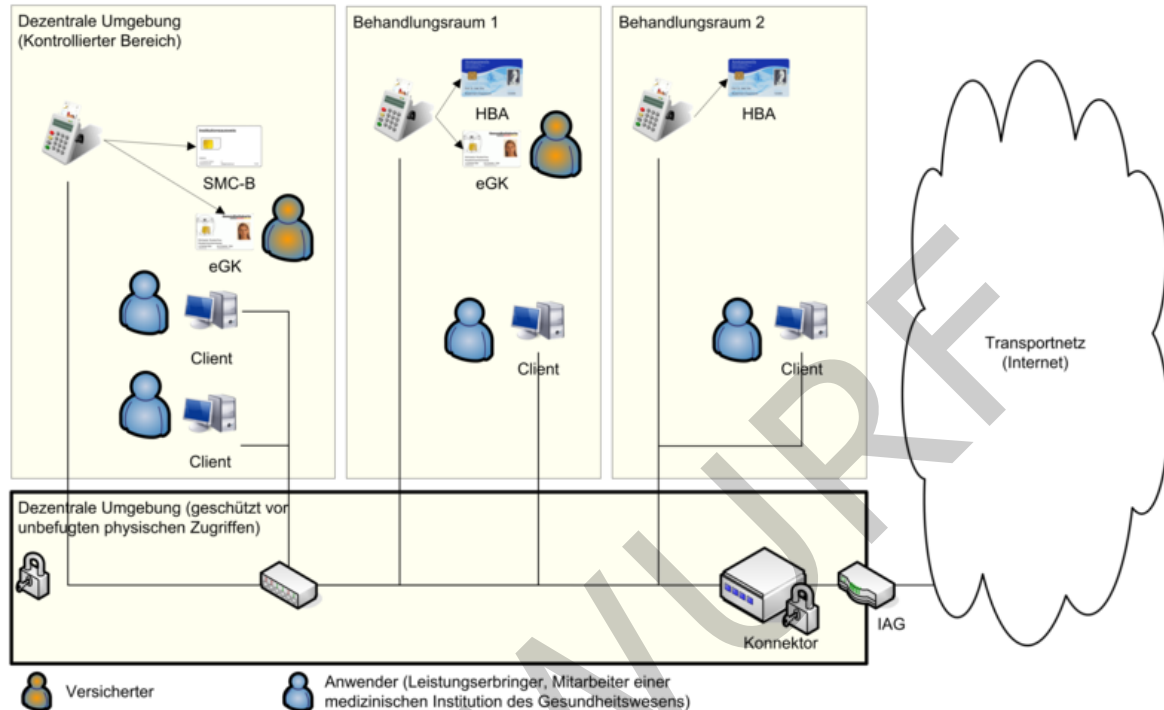


Abbildung 26: Szenario einer Installation mit mehreren Behandlungsräumen

### 11.2.10.2.1 Beschreibung des Szenarios

Mit der in Szenario 1 skizzierten Topologie kann auch ein Szenario bedient werden, bei dem mehrere Behandlungsräume unterstützt werden (siehe Abbildung 26). Dabei ist in jedem Behandlungsraum mindestens ein Kartenterminal vorzusehen, so dass die eGK gelesen werden kann.

Auf die Darstellung der Kommunikationswege in zentrale Netze wurde in Abbildung 26 verzichtet, da sich hier keine Änderung gegenüber Szenario 1 ergibt.

Durch die Ressourcenverwaltung des Konnektors wird sichergestellt, dass bei Anwendungsfällen diejenigen Kartenterminals angesprochen werden, welche dem Arbeitsplatz zugeordnet sind, von dem aus der Anwendungsfall initiiert wurde.

### 11.2.10.2.2 Voraussetzungen

- Anbindung der bestehenden Clientsysteme an ein zum Konnektor kompatibles LAN muss möglich sein.
- Konfiguration des Konnektors als Default-Gateway in den Clientsystemen und Einrichtung der notwendigen VPN-Tunnel im Konnektor, um in die verschiedenen Netze zu routen.
- Verfügbarkeit einer SMC-B und mehrerer Kartenterminals und Clientsysteme

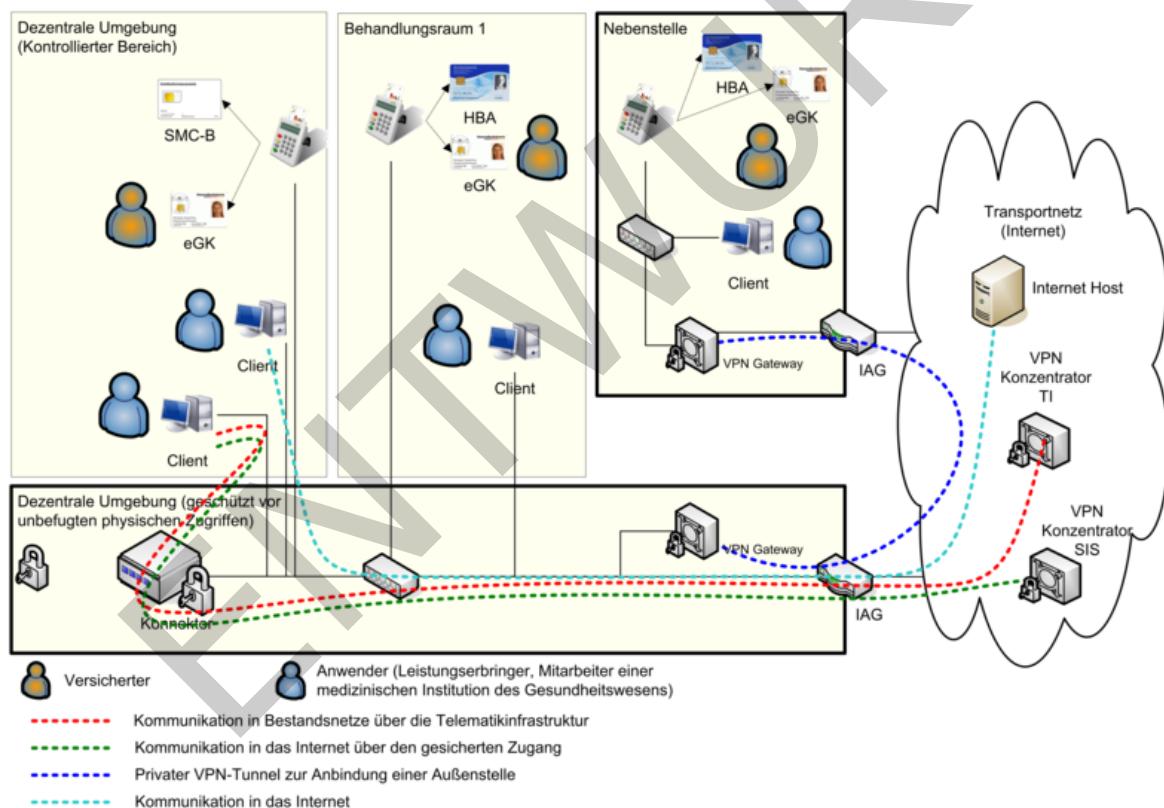
- 9278 • Die Clientsysteme und Kartenterminals und deren Relationen sind dem Konnektor  
9279 über Konfiguration bekannt gemacht worden.

### 9280 ~~11.2.3~~ 10.2.3 Auswirkungen

- 9281 • Die Clientsysteme können über den Konnektor Anwendungsfälle der TI initiieren  
9282 • Die Clientsysteme können über den Konnektor auf das Internet (über den SIS)  
9283 und Bestandsnetze zugreifen  
9284 • Der HBA-Inhaber muss seinen HBA mit sich führen und kann diesen in den  
9285 einzelnen Kartenterminals der Behandlungsräume nutzen.

### 9286 ~~11.3~~ 10.3 Szenario 3: Integration in bestehende Infrastruktur ohne 9287 Netzsegmentierung

9288



9289  
9290

9291 **Abbildung 27: Szenario einer Integration der TI Produkte in eine bestehende**  
9292 **Infrastruktur**

### 9293 ~~11.3.1~~ 10.3.1 Beschreibung des Szenarios

9294 Im Falle einer bereits vorhandenen Infrastruktur im dezentralen Bereich können die  
9295 Produkte der TI, insbesondere der Konnektor, so in die Infrastruktur integriert werden,  
9296 dass Bestandsanwendungen bereits erprobte Kommunikationswege weiter nutzen  
9297 können.

9298 Wie in Abbildung 27 beispielhaft dargestellt, existiert bereits eine Infrastruktur, die  
 9299 sowohl einen Internetzugang für die Arbeitsplätze ermöglicht (gestrichelte Linie in  
 9300 türkis), als auch eine Nebenstelle über VPN anbindet (gestrichelte Linie in blau). In  
 9301 diesem Fall wird der Konnektor als zusätzliches Gerät an das bestehende Netzwerk  
 9302 angeschlossen und nutzt den bereits vorhandenen Internetanschluss zur Kommunikation  
 9303 in die TI.

9304 Für die Clientsysteme muss in diesem Szenario je nach individuellem Anforderungsprofil  
 9305 entschieden werden, ob das jeweilige Clientsystem über die Telematikinfrastruktur  
 9306 kommunizieren können soll und den gesicherten Internetzugang (SIS) nutzen soll oder  
 9307 nicht.

9308 Soll ein Clientsysteme nicht über die Telematikinfrastruktur kommunizieren, bleibt der  
 9309 IAG als Default-Gateway dieses Clientsystems konfiguriert. In diesem Fall routet der IAG  
 9310 die eingehenden IP-Pakete mit öffentlichen Zieladressen weiter in das Internet. Die  
 9311 gestrichelte Linie in türkis zeigt beispielhaft einen Zugriff in das Internet.

9312 Soll ein Clientsystem über die Telematikinfrastruktur kommunizieren oder den  
 9313 gesicherten Internetzugang (SIS) nutzen, muss der Konnektor als Default-Gateway  
 9314 konfiguriert werden. In diesem Fall routet der Konnektor die eingehenden IP-Pakete, die  
 9315 nicht für ihn bestimmt sind, entweder durch den VPN-Tunnel der TI über die  
 9316 Telematikinfrastruktur in ein angeschlossenes Bestandsnetz, (gestrichelte Linie in rot)  
 9317 oder durch den VPN-Tunnel zum SIS (Secure Internet Service) in das Internet  
 9318 (gestrichelte Linie in grün). Sollte kein sicherer Internetzugang konfiguriert sein, so  
 9319 würde der Konnektor den Traffic verwerfen und ggf. per ICMP dem Client eine anderes  
 9320 Gateway (IAG) vorschlagen. Alternativ können die von den Clients benötigten Routing-  
 9321 Informationen manuell oder per DHCP konfiguriert werden.

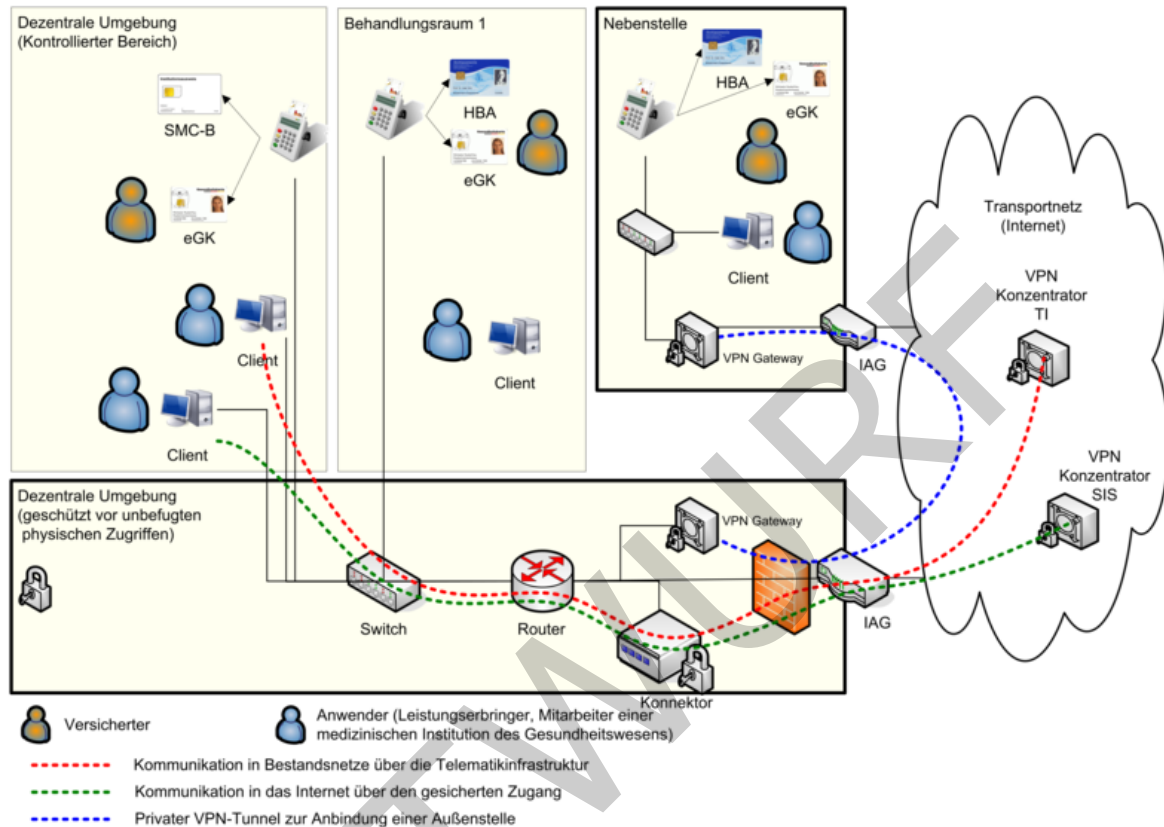
## 9322 ~~11.3.2~~ 10.3.2 Voraussetzungen

- 9323 • Konnektor ist kompatibel zur bestehenden Infrastruktur (Vernetzung)
- 9324 • Die bestehende Infrastruktur verfügt über einen Internetzugang
- 9325 • Verfügbarkeit einer SMC-B und mehrerer Kartenterminals
- 9326 • Die Clientsysteme und Kartenterminals und deren Relationen sind dem Konnektor  
 9327 über Konfiguration bekannt gemacht worden.

## 9328 ~~11.3.3~~ 10.3.3 Auswirkungen

- 9329 • Produkte der Telematik können „minimal-invasiv“ in die bestehende Infrastruktur  
 9330 integriert werden. Bestehende Kommunikationswege können weiter genutzt  
 9331 werden.
- 9332 • Für Clients kann je nach individuellen Anforderungsprofil der sichere  
 9333 Internetzugang über den Konnektor genutzt werden oder der direkte  
 9334 Internetzugang über den bestehenden IAG

## 11.4.10.4 Szenario 4: Integration in bestehende Infrastruktur mit Netzsegmentierung



**Abbildung 28: Szenario einer Integration der TI Produkte in eine bestehende Infrastruktur mit existierendem Router**

### 11.4.110.4.1 Beschreibung des Szenarios

Das vorliegende Szenario skizziert eine etwas komplexere dezentrale Umgebung, in der das Netzwerk segmentiert ist und dedizierte Router als Default-Gateway für die Clientsysteme genutzt werden. In diesem Fall kann die Konfiguration der Clients unverändert bleiben und der Konnektor wird als zusätzliches Gerät in das Netzwerk integriert und dem Router bekanntgemacht als Gateway für den sicheren Internetzugang und den Zugang zu den an die Telematikinfrastruktur angeschlossenen Bestandsnetze.

### 11.4.210.4.2 Voraussetzungen

- Konnektor ist kompatibel zur bestehenden Infrastruktur (Vernetzung)
- Verfügbarkeit einer SMC-B und mehrerer Kartenterminals
- Der Konnektor ist dem bestehenden Router als Gateway bekannt gemacht.

### 11.4.3 10.4.3 Auswirkungen

- Produkte der Telematik können „minimal-invasiv“ in die bestehende Infrastruktur integriert werden. Bestehende Kommunikationswege können weiter genutzt werden.
- Die Default-Gateway-Konfiguration der Clients muss nicht geändert werden.

### 11.5 10.5 Szenario 5: Zentral gesteckter HBA

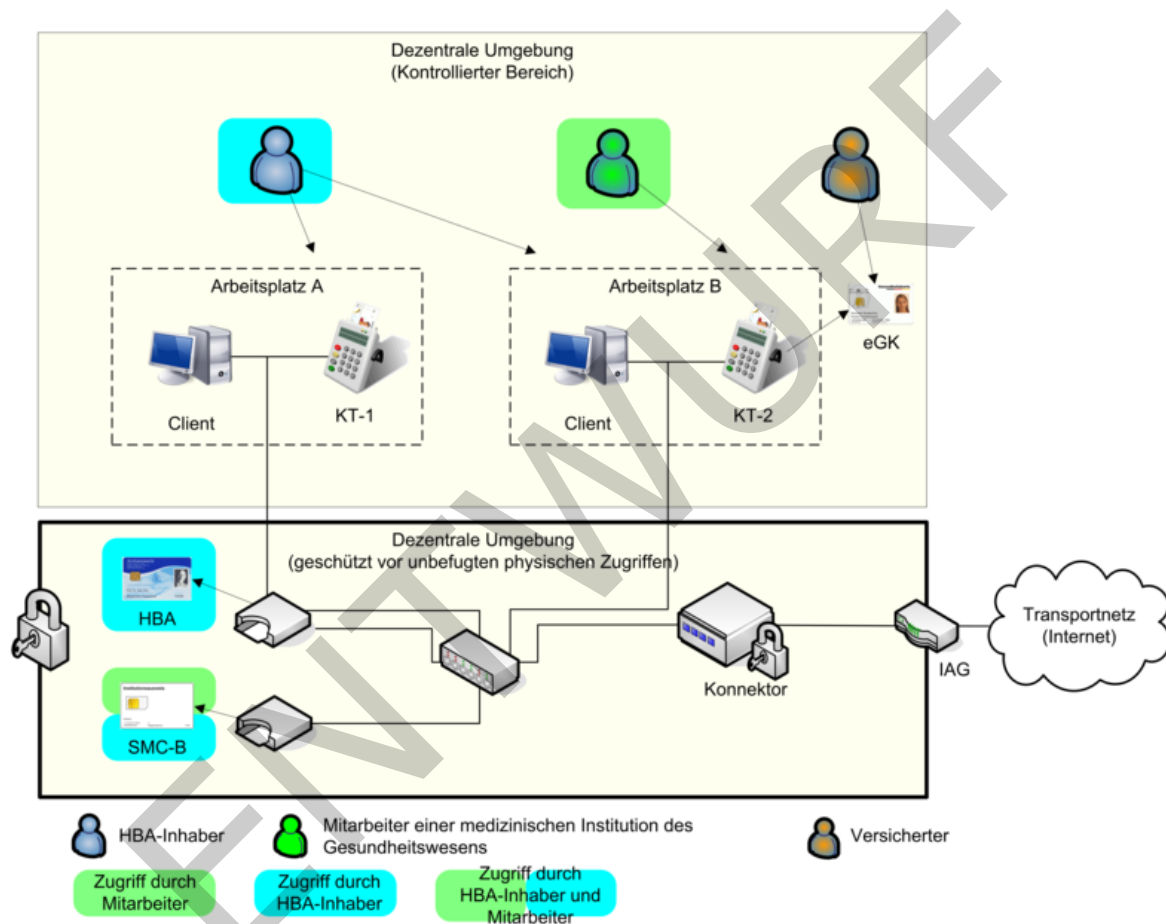


Abbildung 29: Szenario mit zentral gesteckten HBA und SMC-B

### 11.5.1 10.5.1 Beschreibung des Szenarios

Dieses Szenario zeichnet sich dadurch aus dass ein HBA nicht durch seinen Inhaber mitgeführt und am Arbeitsplatz gesteckt wird, sondern zentral und geschützt vor unbefugten physischen Zugriffen gesteckt bleibt.

Der HBA-Inhaber greift über jeden konfigurierten Arbeitsplatz auf seinen HBA zu. Die Remote-PIN-Eingabe erfolgt unter Verwendung des lokal am Arbeitsplatz vorhandenen eHealth-Kartenterminals.

Die Mechanismen zum Zugriff auf eine zentral gesteckte SMC-B funktionieren analog zum HBA.

9370 **~~11.5.2~~10.5.2 Voraussetzungen**

9371 Folgende zusätzliche Punkte müssen erfüllt sein, um dieses Szenario umzusetzen:

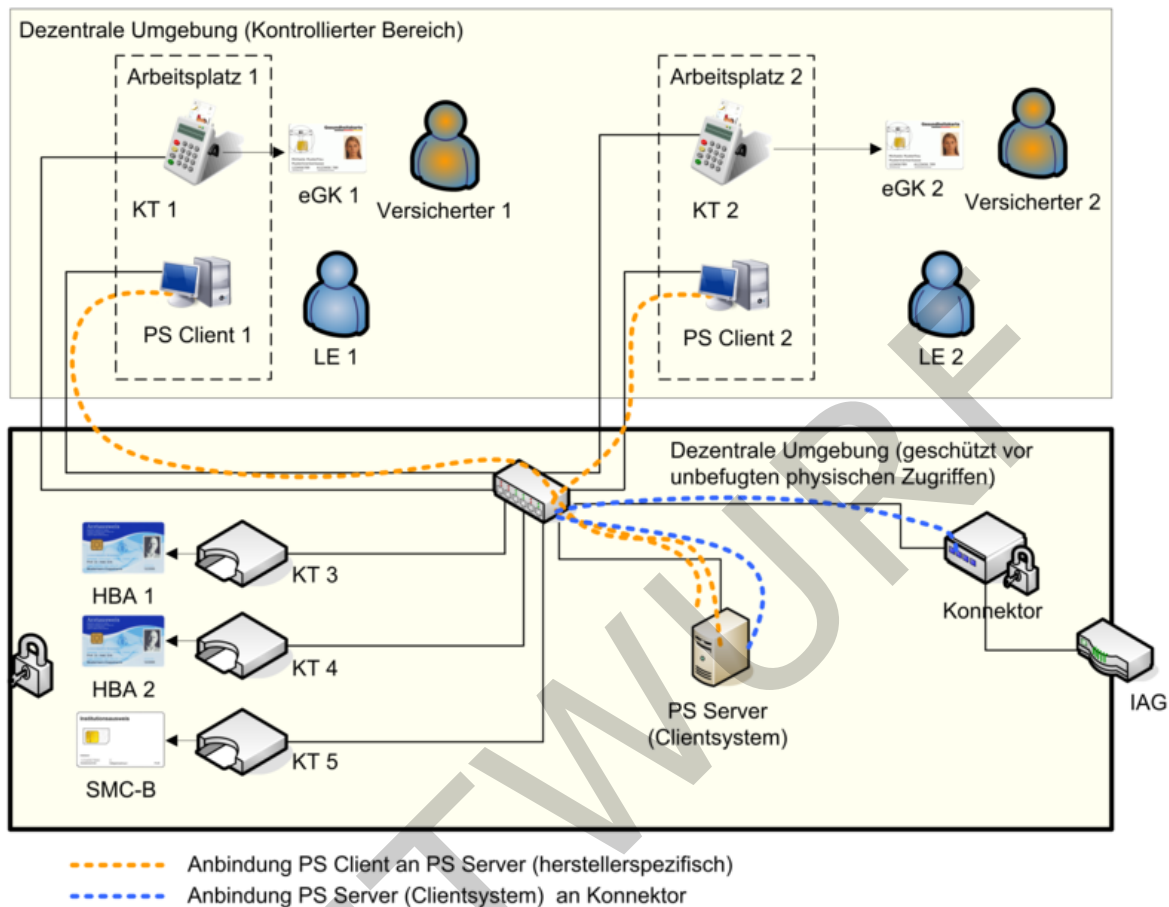
- 9372 • Stecken der zentral gesteckten Karten HBA und SMC-B (ohne direkte Aufsicht)
- 9373 und Sicherstellung des Schutzes vor unbefugtem physischen Zugriff
- 9374 • Konfiguration im Konnektor: Lokales eHealth-Kartenterminals als lokales eHealth-
- 9375 Kartenterminal für eine Remote-PIN-Eingabe eines bestimmten Arbeitsplatzes.
- 9376 *Im abgebildeten Beispiel KT-1 für Arbeitsplatz A und KT-2 für Arbeitsplatz B.*
- 9377 • Konfiguration im Konnektor: Assoziation der gewünschten Arbeitsplätze zum
- 9378 jeweiligen Kartenterminal mit zentral gesteckter Karte.
- 9379 *Im abgebildeten Beispiel Arbeitsplatz A assoziiert mit dem eHealth-Kartenterminal*
- 9380 *des HBAs und Arbeitsplatz B mit eHealth-Kartenterminal des HBAs und dem*
- 9381 *eHealth-Kartenterminal der SMC-B.*

9382 **~~11.5.3~~10.5.3 Auswirkung**

- 9383 • HBA muss nicht mehr durch seinen Inhaber mitgeführt werden
- 9384 • SMC-B muss nicht mehr unter ständiger Aufsicht eines Mitarbeiters einer
- 9385 Organisation des Gesundheitswesens sein.



## 11.6.10.6 Szenario 6: Installation mit zentralem PS



**Abbildung 30: Szenario mit zentralem Primärsystem als Clientsystem**

### 11.6.110.6.1 Beschreibung des Szenarios

Das Szenario skizziert eine dezentrale Konfiguration, bei der das Primärsystem aus einem Serveranteil „PS Server“ und mehreren Clientanteilen „PS Client“ besteht. Die Anbindung zwischen dem „PS Server“ und den „PS Clients“ ist herstellerspezifisch. Der „PS Server“ fungiert als ein einziges Clientsystem gegenüber der TI bzw. dem Konnektor (z.B. als Terminalserver). Die Clientsystemschnittstelle des Konnektors wird ausschließlich vom „PS Server“ genutzt. Der „PS Server“ muss bei der Kommunikation mit dem Konnektor eine Übersetzung der zugreifenden „PS Clients“ auf die entsprechende Entität „Arbeitsplatz“ des Konnektors durchführen.

Beispielhaft zeigt das Szenario zwei Arbeitsplätze mit jeweils einem Kartenterminal für die eGK sowie zentral gesteckte SMC-B und HBAs. Alternativ sind auch lokal am Arbeitsplatz gesteckte HBAs möglich.



9403 **~~11.6.2~~10.6.2 Voraussetzungen**

- 9404 • Netzanbindung aller Komponenten (u. a. KT, PS Client, PS Server, Konnektor) in  
9405 der dezentralen Umgebung bis einschließlich zur Netzwerkschicht (IP-Ebene)
- 9406 • Konfiguration des Primärsystems mit seinen Anteilen „PS Server“ und ggf.  
9407 mehreren „PS Clients“ passend zum Informationsmodell des Konnektors  
9408 (herstellerspezifisch).
- 9409 • Konfiguration des Konnektors. U. a.:
- 9410 • Informationsmodell:  
9411 Beim Beispielszenario u.a Entitäten „Clientsystem“ für „PS Server“,  
9412 „Arbeitsplatz“ für „Arbeitsplatz 1“ und Arbeitsplatz 2“, „Kartenterminal“ und  
9413 „KT-Slot“ für „KT 1“ – „KT 5“, „Mandat“ für die vorgesehene Anzahl von  
9414 Mandaten, „SM-B\_Verwaltet“ sowie entsprechende Entitätenbeziehungen.
- 9415 • Anbindung PS Server (ggf. über TLS)
- 9416 • Pairing der Kartenterminals
- 9417 • Gesteckte Karten (SMC-B, HBA, eGK)
- 9418 • Anmeldung Nutzer am „PS Client“

9419 **~~11.6.3~~10.6.3 Auswirkungen**

- 9420 • An den verschiedenen Arbeitsplätzen können für die definierten Mandaten und  
9421 Nutzer Anwendungsfälle der TI initiiert werden.
- 9422 • HBA-Inhaber müssen entsprechen der gewählten HBA-Deployment-Varianten
- 9423 • ihren HBA zentral stecken und über das Remote-PIN-Verfahren zugreifen
- 9424 • ihren HBA mit sich führen und lokal in Kartenterminal der Arbeitsplätze stecken

## 11.7.10.7 Szenario 7: Mehrere Mandanten

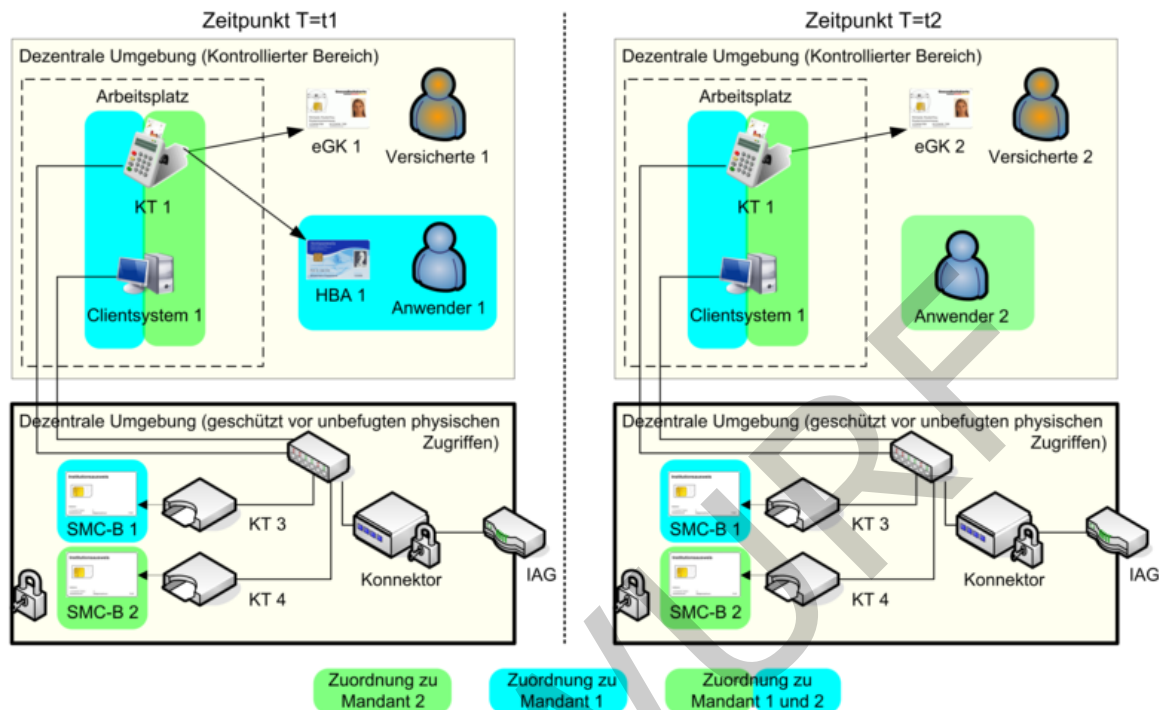


Abbildung 31: Szenario für den Zugriff

### 11.7.10.7.1 Beschreibung des Szenarios

Das Szenario skizziert eine dezentrale Konfiguration, bei der mehrere Mandanten vorhanden sind, wobei jedem Mandant eine eigene SMC-B zugeordnet ist. Die SMC-Bs sind zentral zusammen mit dem Konnektor geschützt vor unbefugten physischen Zugriffen installiert. Die Komponenten Arbeitsplätze, Clientsysteme und Kartenterminals müssen eine Zuordnung zum Mandanten haben, wobei Zuordnungen zu mehreren Mandanten möglich sind. Das Beispiel zeigt einen Arbeitsplatz mit „Clientsystem 1“ und „KT 1“, der zu unterschiedlichen Zeiten durch verschiedene Mandanten verwendet wird. Zum Zeitpunkt T=t1 greift ein Anwender 1 mit HBA 1 über einen Anwendungsfall im Kontext Mandat 1 auf die TI zu, wobei der Versicherte 1 mit eGK 1 am Anwendungsfall beteiligt ist. Zum Zeitpunkt T=t2 wird ein anderer Anwendungsfall im Kontext von Mandat 2 durch einen Anwender 2 ohne HBA initiiert, wobei der Versicherte 2 mit eGK 2 am Anwendungsfall beteiligt ist. Das Clientsystem stellt hierbei den Mandantenbezug sowie die Nutzer Authentisierung sicher. Als Variante können auch mehrere Mandanten eine Zuordnung zu einer einzelnen SMC-B haben. Weiterhin können auch in diesem Szenario HBAs zentral gesteckt werden.

### 11.7.210.7.2 Voraussetzungen

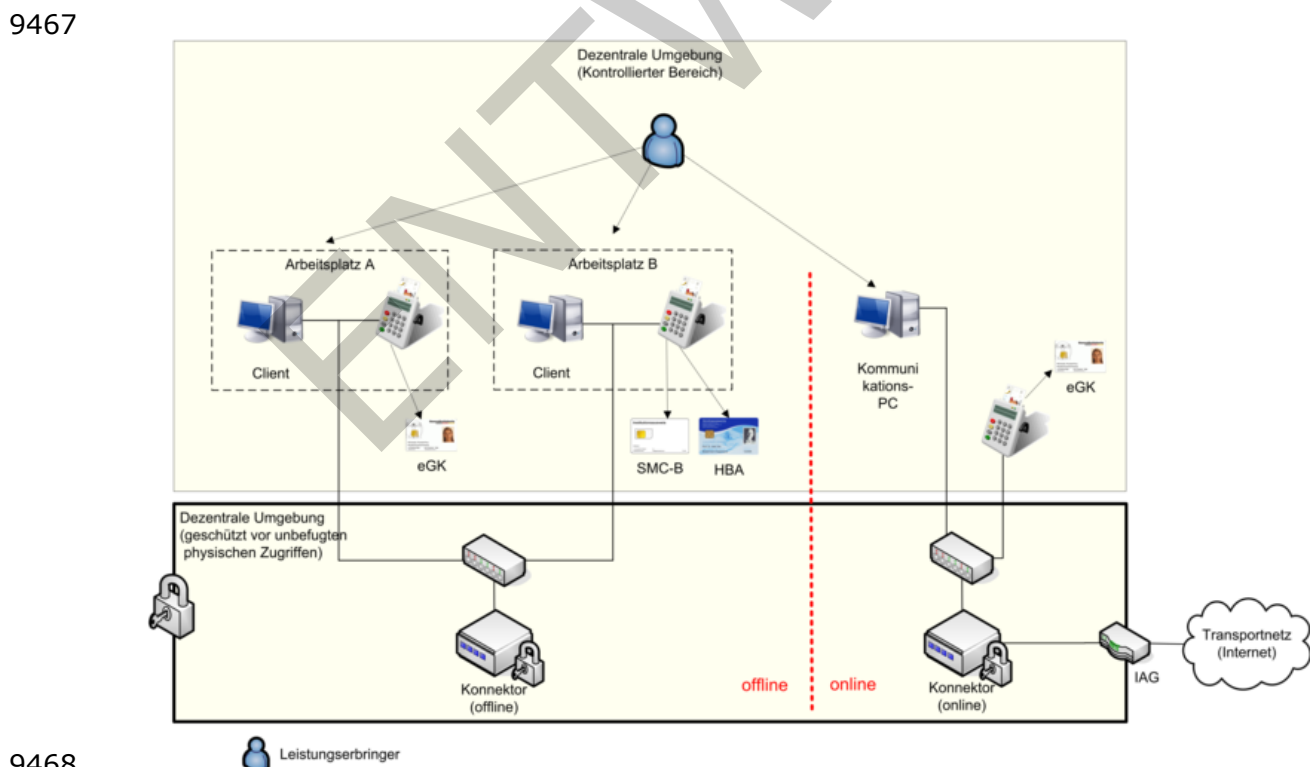
- Netzerkanbindung aller Komponenten (u. a. KT, Clientsystem, Konnektor) in der dezentralen Umgebung bis einschließlich zur Netzwerkschicht (IP-Ebene)
- Konfiguration der Clientsysteme („Clientsystem 1“), passend zum Informationsmodell des Konnektors (herstellerspezifisch).

- 9450 • Konfiguration des Konnektors. U. a.:
- 9451 • Konfiguration Konnektor:
- 9452 Beim Beispielszenario u.a Entitäten „Clientsystem“ für „Clientsystem 1“,
- 9453 „Arbeitsplatz“ für „Arbeitsplatz 1“, „Kartenterminal“ und „KT-Slot“ für „KT 1“ –
- 9454 „KT 4“, „Mandat“ für „Mandant 1“ und „Mandant 2“, „SM-B\_Verwaltet“ für
- 9455 „SMC-B 1“ und SMC-B 2“ sowie entsprechende Entitätenbeziehungen
- 9456 • Anbindung „Clientsystem 1“ (ggf. über TLS)
- 9457 • Pairing der Kartenterminals
- 9458 • Gesteckte Karten (SMC-B 1, SMC-B 2, HBA 1, eGK 1, eGK 2)
- 9459 • Anmeldung eines Anwenders mit Mandantenbezug am Clientsystem

### 11-7-310.7.3 Auswirkungen

- 9461 • An den verschiedenen Arbeitsplätzen können für die definierten Mandaten und
- 9462 Anwender Anwendungsfälle der TI initiiert werden.
- 9463 • HBA-Inhaber müssen entsprechen der gewählten HBA-Deployment-Varianten
- 9464 • ihren HBA zentral stecken und über das Remote-PIN-Verfahren zugreifen
- 9465 • ihren HBA mit sich führen und lokal in Kartenterminal der Arbeitsplätze stecken

### 11-810.8 Szenario 9: Standalone Konnektor - Physische Trennung



**Abbildung 32: Standalone-Szenario mit physischer Trennung im Konnektor**

## 9470 ~~11.8.1~~10.8.1 Beschreibung des Szenarios

9471 Dieses Szenario stellt eine Variante des Standalone-Szenarios dar, bei dem eine  
9472 physische Trennung der Konnektoren eingesetzt wurde.

9473 Im Standalone-Szenario besteht eine Trennung zwischen den Praxissystemen der  
9474 dezentralen Umgebung, welche offline (also, ohne Anbindung an die zentrale TI-  
9475 Plattform) betrieben werden und den für das Update der eGK durch die Fachanwendung  
9476 VSDM notwendigen Komponenten, welche online (also, mit Verbindung in die zentrale TI-  
9477 Plattform) betrieben werden.

9478 Die physische Trennung im Standalone-Szenario zeichnet sich dadurch aus, dass  
9479 getrennte Komponenten zum Einsatz kommen. Der Online-Konnektor ist mit der  
9480 zentralen TI-Plattform verbunden und ermöglicht das VSDM Update der eGKs. Ein am  
9481 Online-Konnektor angebundener Kommunikations-PC kann darüber hinaus über den  
9482 sicheren Internetzugang der TI auf das Internet und über den VPN-Konzentrator TI auf  
9483 Bestandsnetze zugreifen.

9484 Sollten die Online-/Offline-Systeme nicht netztechnisch voneinander getrennt sein, so  
9485 obliegt es dem Administrator der Praxissysteme sicherzustellen, dass die netztechnische  
9486 Verbindung keine Gefährdung für die Praxissysteme zur Folge hat.

9487 Im Offline-Konnektor sind einzelne Funktionen nicht verfügbar, andere haben einen  
9488 eingeschränkten Funktionsumfang. So kann z.B. eine QES erzeugt oder geprüft aber  
9489 dabei keine aktuelle Statusauskunft (OCSP-Response) für die eingesetzten Zertifikate  
9490 eingeholt werden. Dies hat zur Folge, dass bei Erzeugung einer QES keine Statusauskunft  
9491 für das Signaturzertifikat in die Signatur eingebettet werden kann und bei einer Prüfung  
9492 der QES nur eine eventuell in die Signatur eingebettet Statusauskunft des Zertifikats  
9493 berücksichtigt werden kann.

9494 Der Nutzer muss in diesem Fall selber entscheiden ob der gebotene Funktionsumfang für  
9495 seinen Anwendungsfall ausreichend ist.

## 9496 ~~11.8.2~~10.8.2 Voraussetzungen

9497 Folgende zusätzliche Punkte müssen erfüllt sein, um dieses Szenario umzusetzen:

- 9498 • Konfiguration im Konnektor: Es muss konfiguriert werden, welche Komponenten  
9499 von welchem Konnektor (online/offline) verwendet werden dürfen.
- 9500 • Ein eHealth-Kartenterminal oder ein Arbeitsplatz darf immer nur mit einem der  
9501 Konnektoren verbunden sein.
- 9502 • Konfiguration im Konnektor: Im Offline-Konnektor wird kein VPN-Kanal  
9503 konfiguriert.
- 9504 • Clients bzw. Kommunikations-PC müssen sicherstellen, dass sie nur den jeweils  
9505 richtigen Konnektor ansprechen.
- 9506 • Es sollte eine netztechnische Trennung des Online- und Offline-Segmentes  
9507 erfolgen. Wird dies nicht umgesetzt, dann obliegt es dem Administrator der  
9508 Praxissysteme sicherzustellen, dass die netztechnische Verbindung keine  
9509 Gefährdung für die Praxissysteme zur Folge hat.  
9510 Sollte keine netztechnische Trennung erfolgen, so kann nur einer der Konnektoren  
9511 als DHCP-Server agieren. Es wird empfohlen hier den Offline-Konnektor zu  
9512 verwenden, da dort tendenziell mehr Systeme angeschlossen sind. Die am Online-  
9513 Konnektor angeschlossenen Systeme müssen dann direkt konfiguriert werden.

9514 **~~11.8.3~~10.8.3 Auswirkung**

- 9515 • Erhöhter Aufwand durch separate Konnektoren und separate eHealth-  
9516 Kartenterminals.
- 9517 • Trennung der Praxissysteme von der zentralen TI-Plattform ist für den  
9518 Leistungserbringer nachweislich sichergestellt.
- 9519 • Eingeschränkte Funktionalität der TI für Praxissysteme (nur Offline-Funktionalität)
- 9520 • Notwendige Prüfung des Leistungserbringers, ob eingeschränkte Funktionalität  
9521 (insbesondere bei Sicherheitsfunktionen) akzeptabel ist.
- 9522 • Sicherer Internetzugang der TI nur über den Kommunikations-PC nutzbar.
- 9523 • Zugang zu Bestandsnetzen über den VPN-Konzentrator TI nur über den  
9524 Kommunikations-PC nutzbar

## **1211 Anhang L – Datentypen von Eingangs- und Ausgangsdaten**

**Tabelle 391: Aufzähltypen**

| Typname              | Werteliste   |
|----------------------|--|
| [Boolean]            | {true   false}   |
| [EncryptionType]     | {CMS   XMLEnc   S/MIME}  |
| [EventType]          | {Op   Sec   Perf}  |
| [EventSeverity]      | {Debug   Info   Warn   Err   Fatal}                                      |
| [KtOutputMode]       | {Input   OutputWait   OutputConfirm   OutputKeep   OutputErase}          |
| [PinStatus]          | {VERIFIED   VERIFYABLE   BLOCKED   TRANSPORT_PIN   EMPTY_PIN   DISABLED} |
| [PinResult]          | {OK   REJECTED   BLOCKED   ERROR}  |
| [PukResult]          | {OK   REJECTED   BLOCKED   ERROR}  |
| [VerificationResult] | {VALID   INVALID   INCONCLUSIVE}   |