Einführung der Gesundheitskarte

Spezifikation

Konnektor

Version:	4.7.0
Revision:	\main\rel_online\rel_ors1\208
Stand:	17.07.2015
Status:	freigegeben
Klassifizierung:	öffentlich
Referenzierung:	[gemSpec_Kon]

Dokumentinformationen

Änderungen zur Vorversion

Einarbeitung der KOM-LE-bedingten Änderungen.

Einarbeitung von Erratas.

Alle Kapitel und Anforderungen, die nicht in Phase 1 umgesetzt werden, sind türkis markiert.

Dokumentenhistorie

Version	Stand	Kap./ Seite	Grund der Änderung, besondere Hinweise	Bearbeitung
3.1.0	21.08.12		zur Abstimmung freigegeben	gematik
4.0.0	15.10.12		Einarbeitung Gesellschafterkommentare	gematik
4.1.0	12.11.12		Einarbeitung Kommentare aus der übergreifenden Konsistenzprüfung	gematik
4.2.0	06.06.13		Einarbeitung Gesellschafterkommentare	gematik
4.3.0	22.08.13		Einarbeitung lt. Änderungsliste vom 08.08.13	gematik
	13.02.14		Arbeitsstand zur fachlichen Prüfung durch den LA in Bezug auf die offenen Punkte der Konnektor-Spezifikation	gematik
4.4.0	21.02.14		Losübergreifende Synchronisation	gematik
4.5.0	17.06.14		Einarbeitung von Änderungsmeldungen der P11-Änderungsliste (einzelne Korrekturen und Präzisierungen)	gematik
4.6.0	26.08.14		Einarbeitung von Änderungsmeldungen der P12-Änderungsliste (einzelne Korrekturen und Präzisierungen)	gematik
			Einarbeitung von Änderungen für KOM-LE Einarbeitung der Errata 1.4.1 bis 1.4.6	gematik
4.7.0	17.07.15		freigegeben	gematik

Inhaltsverzeichnis

Dokumentinformationen **2**

Inhaltsverzeichnis **3**

1 Einordnung des Dokumentes **13**

1.1 Zielsetzung **13**

1.2 Zielgruppe **13**

1.3 Geltungsbereich **13**

1.4 Abgrenzung des Dokuments **14**

1.5 Methodik **14**

1.5.1 Anforderungen 14
1.5.2 Offene Punkte 14
1.5.3 Erläuterungen zur Spezifikation des Außenverhaltens 14
1.5.4 Erläuterungen zur Dokumentenstruktur und „Dokumentenmechanismen“ . 15
1.5.4.1 Modulare Spezifikation über Funktionsmerkmale *15*
1.5.4.2 Technische Use Cases - TUCs *16*
1.5.4.3 Event-Mechanismus *16*
1.5.4.4 Konfigurationsparameter und Zustandswerte *17*
1.5.5 Phasen „VSDM“ und „VSDM und Basisdienst QES“ 17

2 Systemüberblick **19**

2.1 Logische Struktur **21**

2.2 Sicherer Datenspeicher **22**

2.3 Überblick Konnektoridentität **23**

2.4 Mandantenfähigkeit **23**

2.5 Versionierung **24**

2.6 Fachanwendungen **24**

2.7 Netzseitige Einsatzszenarien **25**

Parameter ANLW_ANBINDUNGS_MODUS 25
Parameter ANLW_INTERNET_MODUS 25

2.8 Lokale und entfernte Kartenterminals **26**

2.9 Standalone-Szenario **26**

3 Übergreifende Festlegungen **27**

3.1 Konnektoridentität und gSMC-K **30**

3.1.1 Organisatorische Anforderungen und Sperrprozesse 30

3.2 Bootup-Phase 32
3.3 Betriebszustand 33
3.3.1 Betriebsaspekte 42
3.4 Fachliche Anbindung der Clientsysteme 43
3.4.1 Betriebsaspekte 45
3.5 Clientsystemschnittstelle 47
3.5.1 SOAP-Schnittstelle 47
3.5.2 Statusrückmeldung und Fehlerbehandlung 47
3.6 Verwendung manuell importierter CA-Zertifikate 48
3.7 Testunterstützung 49

4 Funktionsmerkmale 52
4.1 Anwendungskonnektor 52
4.1.1 Zugriffsberechtigungsdienst 52
4.1.1.1 Funktionsmerkmalweite Aspekte 52
4.1.1.2 Durch Ereignisse ausgelöste Reaktionen 62
4.1.1.3 Interne TUCs, nicht durch Fachmodule nutzbar 62
4.1.1.4 Interne TUCs, auch durch Fachmodule nutzbar 62
4.1.1.4.1 TUC_KON_000 „Prüfe Zugriffsberechtigung“ 62
4.1.1.5 Operationen an der Außenschnittstelle 70
4.1.1.6 Betriebsaspekte 70
4.1.2 Dokumentvalidierungsdienst 71
4.1.2.1 Funktionsmerkmalweite Aspekte 71
4.1.2.2 Durch Ereignisse ausgelöste Reaktionen 71
4.1.2.3 Interne TUCs, nicht durch Fachmodule nutzbar 71
4.1.2.4 Interne TUCs, auch durch Fachmodule nutzbar 71
4.1.2.4.1 TUC_KON_080 “Dokument validieren” 71
4.1.2.5 Operationen an der Außenschnittstelle 75
4.1.2.6 Betriebsaspekte 75
4.1.3 Dienstverzeichnisdienst 75
4.1.3.1 Funktionsmerkmalweite Aspekte 75
4.1.3.2 Durch Ereignisse ausgelöste Reaktionen 79
4.1.3.3 Interne TUCs, nicht durch Fachmodule nutzbar 79
4.1.3.4 Interne TUCs, auch durch Fachmodule nutzbar 79
4.1.3.4.1 TUC_KON_041 „Einbringen der Endpunktinformationen während der Bootup-Phase“ 79
4.1.3.5 Operationen an der Außenschnittstelle 80
4.1.3.6 Betriebsaspekte 81
4.1.4 Kartenterminaldienst 81
4.1.4.1 Funktionsmerkmalweite Aspekte 85
4.1.4.2 Durch Ereignisse ausgelöste Reaktionen 87
4.1.4.3 Interne TUCs, nicht durch Fachmodule nutzbar 88
4.1.4.3.1 TUC_KON_050 „Beginne Kartenterminalsitzung“ 88
4.1.4.3.2 TUC_KON_054 „Kartenterminal hinzufügen“ 93
4.1.4.3.3 TUC_KON_053 „Paire Kartenterminal“ 95

4.1.4.3.4 TUC_KON_055 „Befülle CT-Object“ 99
4.1.4.4 Interne TUCs, auch durch Fachmodule nutzbar....100
4.1.4.4.1 TUC_KON_051 „Mit Anwender über Kartenterminal interagieren“.100
4.1.4.4.2 TUC_KON_056 „Karte anfordern“ 102
4.1.4.4.3 TUC_KON_057 „Karte auswerfen“ 104
4.1.4.5 Operationen an der Außenschnittstelle....105
4.1.4.5.1 RequestCard 106
4.1.4.5.2 EjectCard 108
4.1.4.6 Betriebsaspekte....110
4.1.4.6.1 Allgemeine Betriebsaspekte 110
4.1.4.6.2 Kartenterminals pflegen....111
4.1.4.6.3 Import der Kartenterminal-Informationen 115
4.1.5 Kartendienst 116
4.1.5.1 Funktionsmerkmalweite Aspekte 118
4.1.5.2 Durch Ereignisse ausgelöste Reaktionen 122
4.1.5.3 Interne TUCs, nicht durch Fachmodule nutzbar....123
4.1.5.3.1 TUC_KON_001 „Karte öffnen“....123
4.1.5.4 Interne TUCs, auch durch Fachmodule nutzbar....125
4.1.5.4.1 TUC_KON_026 „Liefere CardSession“ 125
4.1.5.4.2 TUC_KON_012 „PIN verifizieren“ 126
4.1.5.4.3 TUC_KON_019 „PIN ändern“ 130
4.1.5.4.4 TUC_KON_021 „PIN entsperren“ 133
4.1.5.4.5 TUC_KON_022 „Liefere PIN-Status“ 135
4.1.5.4.6 TUC_KON_023 „Karte reservieren“ 137
4.1.5.4.7 TUC_KON_005 „Card-to-Card authentisieren“ 138
4.1.5.4.8 TUC_KON_202 „LeseDatei“ 142
4.1.5.4.9 TUC_KON_203 „SchreibeDatei“....143
4.1.5.4.10 TUC_KON_209 „LeseRecord“....145
4.1.5.4.11 TUC_KON_210 „SchreibeRecord“....146
4.1.5.4.12 TUC_KON_214 „FügeHinzuRecord“ 148
4.1.5.4.13 TUC_KON_215 „SucheRecord“....149
4.1.5.4.14 TUC_KON_018 „eGK-Sperrung prüfen“ 150
4.1.5.4.15 TUC_KON_006 „Datenzugriffsaudit eGK schreiben“ 153
4.1.5.4.16 TUC_KON_218 „Signiere“....154
4.1.5.4.17 TUC_KON_219 „Entschlüssele“ 155
4.1.5.4.18 TUC_KON_200 „SendeAPDU“ 156
4.1.5.4.19 TUC_KON_024 „Karte zurücksetzen“....158

4.1.5.4.20 TUC_KON_216 „LeseZertifikat“ 159
4.1.5.5 Operationen an der Außenschnittstelle 160
4.1.5.5.1 VerifyPin 160
4.1.5.5.2 ChangePin 163
4.1.5.5.3 GetPinStatus 165
4.1.5.5.4 UnblockPin 168
4.1.5.5.5 AuthorizeSMC 170
4.1.5.6 Betriebsaspekte 171
4.1.5.6.1 TUC_KON_025 "Initialisierung Kartendienst" 172
4.1.5.6.2 Kartenübersicht und PIN-Management 172
4.1.6 Systeminformationsdienst 173
4.1.6.1 Funktionsmerkmalweite Aspekte 174
4.1.6.2 Durch Ereignisse ausgelöste Reaktionen 176
4.1.6.3 Interne TUCs, nicht durch Fachmodule nutzbar 176
4.1.6.4 Interne TUCs, auch durch Fachmodule nutzbar 176
4.1.6.4.1 TUC_KON_256 „Systemereignis absetzen“ 176
4.1.6.4.2 TUC_KON_252 „Liefere KT_Liste“ 181
4.1.6.4.3 TUC_KON_253 „Liefere Karten_Liste“ 182
4.1.6.4.4 TUC_KON_254 „Liefere Ressourcendetails“ 183
4.1.6.5 Operationen an der Außenschnittstelle 185
4.1.6.5.1 GetCardTerminals 186
4.1.6.5.2 GetCards 188
4.1.6.5.3 GetResourceInformation 192
4.1.6.5.4 Subscribe 195
4.1.6.5.5 Unsubscribe 197
4.1.6.5.6 RenewSubscriptions 199
4.1.6.5.7 GetSubscription 200
4.1.6.6 Betriebsaspekte 202
4.1.7 Verschlüsselungsdienst 203
4.1.7.1 Funktionsmerkmalweite Aspekte 203
4.1.7.2 Durch Ereignisse ausgelöste Reaktionen 204
4.1.7.3 Interne TUCs, nicht durch Fachmodule nutzbar 204
4.1.7.4 Interne TUCs, auch durch Fachmodule nutzbar 204
4.1.7.4.1 TUC_KON_070 “Daten hybrid verschlüsseln” 204
4.1.7.4.2 TUC_KON_071 “Daten hybrid entschlüsseln” 210
4.1.7.4.3 TUC_KON_072 “Daten symmetrisch verschlüsseln” 214
4.1.7.4.4 TUC_KON_073 “Daten symmetrisch entschlüsseln” 215
4.1.7.5 Operationen an der Außenschnittstelle 216
4.1.7.5.1 EncryptDocument 217

4.1.7.5.2 DecryptDocument 221
4.1.7.6 Betriebsaspekte 224
4.1.8 Signaturdienst und SAK 224
4.1.8.1 Funktionsmerkmalweite Aspekte 224
4.1.8.1.1 Dokumentensignatur 224
4.1.8.1.2 Signaturrichtlinien 229
4.1.8.1.3 Signaturzeitpunkt 230
4.1.8.1.4 Extended Trusted Viewer Service 230
4.1.8.1.5 Schnittstelle zwischen SAK (im Konnektor) und xTV 232
4.1.8.1.6 Job-Nummer 233
4.1.8.1.7 Externe Authentisierung 234
4.1.8.2 Durch Ereignisse ausgelöste Reaktionen 234
4.1.8.3 Interne TUCs, nicht durch Fachmodule nutzbar 235
4.1.8.3.1 TUC_KON_155 “Dokumente zur Signatur vorbereiten” 236
4.1.8.3.2 TUC_KON_165 „Signaturvoraussetzungen für nonQES prüfen“ ... 239
4.1.8.3.3 TUC_KON_166 „nonQES Signaturen erstellen“ 240
4.1.8.3.4 TUC_KON_152 "Signaturvoraussetzungen für QES prüfen" 241
4.1.8.3.5 TUC_KON_153 “Dokumentenliste im xTV anzeigen” 242
4.1.8.3.6 TUC_KON_154 "QES Signaturen erstellen" 244
4.1.8.3.7 TUC_KON_168 „Einzelsignatur QES erstellen“ 248
4.1.8.4 Interne TUCs, auch durch Fachmodule nutzbar 249
4.1.8.4.1 TUC_KON_160 “Dokumente nonQES signieren” 249
4.1.8.4.2 TUC_KON_161 “nonQES Dokumentsignatur prüfen ” 251
4.1.8.4.3 TUC_KON_150 “Dokumente QES signieren” 257
4.1.8.4.4 Verhalten des xTV bei der Signaturerstellung 261
4.1.8.4.5 Anforderungen an die Stapelsignatur 264
4.1.8.4.6 TUC_KON_151 "QES Dokumentensignatur prüfen" 266
4.1.8.4.7 Verhalten des xTV bei der Signaturprüfung 272
4.1.8.5 Operationen an der Außenschnittstelle 273
4.1.8.5.1 SignDocument (nonQES und QES) 274
4.1.8.5.2 VerifyDocument (nonQES und QES) 284
4.1.8.5.3 ExternalAuthenticate 289
4.1.8.5.4 SetXTVView 292
4.1.8.5.5 GetXTVView 294
4.1.8.6 Betriebsaspekte 295
4.1.9 Zertifikatsdienst 296
4.1.9.1 Funktionsmerkmalweite Aspekte 296
4.1.9.2 Durch Ereignisse ausgelöste Reaktionen 298

4.1.9.3 Interne TUCs, nicht durch Fachmodule nutzbar ... 298
4.1.9.3.1 TUC_KON_032 „TSL aktualisieren“ ... 298
4.1.9.3.2 TUC_KON_040 „CRL aktualisieren“ ... 300
4.1.9.3.3 TUC_KON_033 „Zertifikatsablauf prüfen“ ... 302
4.1.9.4 Interne TUCs, auch durch Fachmodule nutzbar ... 303
4.1.9.4.1 TUC_KON_037 „Zertifikat prüfen" ... 303
4.1.9.4.2 TUC_KON_034 „Zertifikatsinformationen extrahieren“ ... 307
4.1.9.5 Operationen an der Außenschnittstelle ... 309
4.1.9.5.1 CheckCertificateExpiration ... 309
4.1.9.5.2 ReadCardCertificate ... 311
4.1.9.5.3 VerifyCertificate ... 314
4.1.9.6 Betriebsaspekte ... 317
4.1.9.6.1 TUC_KON_035 „Zertifikatsdienst initialisieren“ ... 317
4.1.10 Protokollierungsdienst ... 320
4.1.10.1 Funktionsmerkmalweite Aspekte ... 320
4.1.10.2 Durch Ereignisse ausgelöste Reaktionen ... 322
4.1.10.3 Interne TUCs, nicht durch Fachmodule nutzbar ... 322
4.1.10.4 Interne TUCs, auch durch Fachmodule nutzbar ... 322
4.1.10.4.1 TUC_KON_271 „Schreibe Protokolleintrag“ ... 322
4.1.10.5 Operationen an der Außenschnittstelle ... 324
4.1.10.6 Betriebsaspekte ... 324
4.1.10.6.1 TUC_KON_272 “Initialisierung Protokollierungsdienst ... 326
4.1.11 TLS-Dienst ... 327
4.1.11.1 Funktionsmerkmalweite Aspekte ... 328
4.1.11.2 Durch Ereignisse ausgelöste Reaktionen ... 328
4.1.11.3 Interne TUCs, nicht durch Fachmodule nutzbar ... 328
4.1.11.4 Interne TUCs, auch durch Fachmodule nutzbar ... 328
4.1.11.4.1 TUC_KON_110 "Kartenbasierte TLS-Verbindung aufbauen" ... 328
4.1.11.4.2 TUC_KON_111 "Kartenbasierte TLS-Verbindung abbauen" ... 329
4.1.11.5 Operationen an der Außenschnittstelle ... 330
4.1.11.6 Betriebsaspekte ... 330
4.1.12 LDAP-Proxy ... 331
4.1.12.1 Funktionsmerkmalweite Aspekte ... 331
4.1.12.2 Durch Ereignisse ausgelöste Reaktionen ... 331
4.1.12.3 Interne TUCs, nicht durch Fachmodule nutzbar ... 331
4.1.12.4 Interne TUCs, auch durch Fachmodule nutzbar ... 331
4.1.12.4.1 TUC_KON_290 „LDAP-Verbindung aufbauen“ ... 331
4.1.12.4.2 TUC_KON_291 „Verzeichnis abfragen“ ... 332
4.1.12.4.3 TUC_KON_292 „LDAP-Verbindung trennen" ... 333
4.1.12.4.4 TUC_KON_293 „Verzeichnisabfrage abbrechen" ... 334
4.1.12.5 Operationen an der Außenschnittstelle ... 335
4.1.12.5.1 Unterstützte LDAPv3 Operationen ... 335

4.1.12.6 Betriebsaspekte ... 335

4.2 Netzkonnektor ... 336
4.2.1 Anbindung LAN/WAN ... 336
4.2.1.1 Funktionsmerkmalweite Aspekte ... 336
4.2.1.1.1 Netzwerksegmentierung ... 337
4.2.1.1.2 Routing und Firewall ... 339
4.2.1.2 Durch Ereignisse ausgelöste Reaktionen ... 349
4.2.1.3 Interne TUCs, nicht durch Fachmodule nutzbar ... 350
4.2.1.3.1 TUC_KON_305 „LAN-Adapter initialisieren“ ... 350
4.2.1.3.2 TUC_KON_306 „WAN-Adapter initialisieren“ ... 351
4.2.1.3.3 TUC_KON_304 „Netzwerk-Routen einrichten“ ... 352
4.2.1.4 Interne TUCs, auch durch Fachmodule nutzbar ... 355
4.2.1.5 Operationen an der Außenschnittstelle ... 355
4.2.1.6 Betriebsaspekte ... 355
4.2.2 DHCP-Server ... 362
4.2.2.1 Funktionsmerkmalweite Aspekte ... 363
4.2.2.2 Durch Ereignisse ausgelöste Reaktionen ... 363
4.2.2.3 Interne TUCs, nicht durch Fachmodule nutzbar ... 363
4.2.2.4 Interne TUCs, auch durch Fachmodule nutzbar ... 363
4.2.2.5 Operationen an der Außenschnittstelle ... 363
4.2.2.5.1 Liefere Netzwerkinformationen über DHCP ... 363
4.2.2.6 Betriebsaspekte ... 364
4.2.2.6.1 TUC_KON_343 „Initialisierung DHCP-Server“ ... 367
4.2.3 DHCP-Client ... 368
4.2.3.1 Funktionsmerkmalweite Aspekte ... 368
4.2.3.2 Durch Ereignisse ausgelöste Reaktionen ... 369
4.2.3.3 Interne TUCs, nicht durch Fachmodule nutzbar ... 369
4.2.3.3.1 TUC_KON_341 „DHCP-Informationen beziehen“ ... 369
4.2.3.4 Interne TUCs, auch durch Fachmodule nutzbar ... 370
4.2.3.5 Operationen an der Außenschnittstelle ... 370
4.2.3.6 Betriebsaspekte ... 370
4.2.4 VPN-Client ... 371
4.2.4.1 Funktionsmerkmalweite Aspekte ... 371
4.2.4.2 Durch Ereignisse ausgelöste Reaktionen ... 372
4.2.4.3 Interne TUCs, nicht durch Fachmodule nutzbar ... 373
4.2.4.3.1 TUC_KON_321 „Verbindung zu dem VPN-Konzentrator der TI aufbauen“ 373
4.2.4.3.2 TUC_KON_322 „Verbindung zu dem VPN-Konzentrator des SIS aufbauen“ 375
4.2.4.4 Interne TUCs, auch durch Fachmodule nutzbar ... 377
4.2.4.5 Operationen an der Außenschnittstelle ... 377
4.2.4.6 Betriebsaspekte ... 377
4.2.5 Zeitdienst ... 379
4.2.5.1 Funktionsmerkmalweite Aspekte ... 379
4.2.5.2 Durch Ereignisse ausgelöste Reaktionen ... 380

4.2.5.3 Interne TUCs, nicht durch Fachmodule nutzbar ... 380
4.2.5.4 Interne TUCs, auch durch Fachmodule nutzbar ... 381
4.2.5.4.1 TUC_KON_351 Liefere Systemzeit ... 381
4.2.5.5 Operationen an der Außenschnittstelle ... 382
4.2.5.5.1 Sync_Time ... 382
4.2.5.6 Betriebsaspekte ... 382
4.2.5.6.1 TUC_KON_352 Initialisierung Zeitdienst ... 383
4.2.6 Namensdienst und Dienstlokalisierung ... 384
4.2.6.1 Funktionsmerkmalseite Aspekte ... 384
4.2.6.2 Durch Ereignisse ausgelöste Reaktionen ... 386
4.2.6.3 Interne TUCs, nicht durch Fachmodule nutzbar ... 386
4.2.6.4 Interne TUCs, auch durch Fachmodule nutzbar ... 386
4.2.6.4.1 TUC_KON_361 „DNS-Namen auflösen“ ... 386
4.2.6.4.2 TUC_KON_362 „Liste der Dienste abrufen“ ... 387
4.2.6.4.3 TUC_KON_363 „Dienstdetails abrufen“ ... 388
4.2.6.5 Operationen an der Außenschnittstelle ... 390
4.2.6.5.1 GetIPAddress ... 390
4.2.6.6 Betriebsaspekte ... 391
4.3 Konnektormanagement ... 392
4.3.1 Zugang und Benutzerverwaltung des Konnektormanagements ... 394
4.3.2 Konnektorname und Versionsinformationen ... 397
4.3.3 Konfigurationsdaten: Persistieren sowie Export- Import ... 397
4.3.4 Administration von Fachmodulen ... 399
4.3.5 Neustart und Werksreset ... 400
4.3.6 Leistungsumfänge und Standalone-Szenarios ... 400
4.3.7 Online-Anbindung verwalten ... 402
4.3.8 Remote Management ... 406
4.3.9 Software- und Konfigurationsaktualisierung (KSR-Client) ... 407
4.3.9.1 Funktionsmerkmalweite Aspekte ... 407
4.3.9.2 Durch Ereignisse ausgelöste Reaktionen ... 408
4.3.9.3 Interne TUCs, nicht durch Fachmodule nutzbar ... 408
4.3.9.3.1 TUC_KON_280 „Konnektoraktualisierung durchführen“ ... 408
4.3.9.3.2 TUC_KON_281 „Kartenterminalaktualisierung anstoßen“ ... 412
4.3.9.3.3 TUC_KON_282 „UpdateInformationen beziehen“ ... 413
4.3.9.3.4 TUC_KON_283 „Infrastruktur Konfiguration aktualisieren“ ... 415
4.3.9.4 Interne TUCs, auch durch Fachmodule nutzbar ... 417
4.3.9.5 Operationen an der Außenschnittstelle ... 417
4.3.9.6 Betriebsaspekte ... 417
4.3.9.6.1 TUC_KON_284 KSR-Client initialisieren ... 417
4.3.10 Konnektorstatus ... 421
4.4 Hardware-Merkmale des Konnektors ... 421

Anhang A - Verzeichnisse ... 424

A1 – Abkürzungen **424**
A2 – Glossar **424**
A3 – Abbildungsverzeichnis **424**
A4 – Tabellenverzeichnis **425**
A5 - Referenzierte Dokumente **437**
A5.1 – Dokumente der gematik 437
A5.2 – Weitere Dokumente 438

Anhang B - Profilierung der Signatur- und Verschlüsselungsformate (normativ) **444**
B1 – Profilierung der Verschlüsselungsformate **444**
B2 – Profilierung der Signaturformate **444**
B3 – Profilierung der Transformation von XML-Dokumenten für die Anzeige **445**
B4 – Bildungsvorschrift „ShortTextSecure“ für PDF/A, TIFF und Text **450**

Anhang C - QES-Dokumentenformate und -Signaturrichtlinien (normativ) **451**
C1 – Dokumentenformat DF_SV_PDFA **451**
C2 – Dokumentenformat DF_SV_TIFF **451**
C3 – Signaturrichtlinie SR_DF_SV_VHITG_ARZTBRIEF **451**
C4 – Signaturrichtlinie SR_DF_SV_EGES **455**

Anhang D - Übersicht über die verwendeten Versionen **459**

Anhang E - Übersicht Konfigurationsparameter und Zustandswerte **462**

Anhang F - Übersicht Events **488**

Anhang G - Fehlercodes **501**

Anhang H - Mapping von „Architektur der TI-Plattform“ auf Konnektorspezifikation **511**

Anhang I - Umsetzungshinweise (informativ) **519**
I1 - Systemüberblick **519**
I1.1 - Hinweise zur Sicherheitsevaluierung nach Common Criteria. 519
I1.1.1 - Separationsmechanismen des Konnektors *519*
I1.1.2 - Granularität der TSF *520*
I2 - Übergreifende Festlegungen **521**
I2.1 - Interne Mechanismen 521
I2.1.1 - Zufallszahlen und Schlüssel *521*
I3 - Funktionsmerkmale **521**
I3.1 - Anwendungskonnektor 521
I3.1.1 - Administration des Informationsmodells *521*

I3.1.2 - Vorgehensvariante für das Handling von CardSessions *522*
I3.1.3 - Darstellung von Terminal-Anzeigen auf einem Kartenterminal *524*
I3.2 – Signaturanwenungskomponente 525
Adressatengerechte und benutzerfreundliche Präsentation *525*
Schnittstelle zwischen SAK_{KON} xTV *526*

Anhang K - Szenarien im dezentralen Umfeld 527

Szenario 1: Einfache Installation ohne spezielle Anforderungen und ohne bestehende Infrastruktur 527
Beschreibung des Szenarios 528
Voraussetzungen 528
Auswirkungen 528

Szenario 2: Installation mit mehreren Behandlungsräumen 529
Beschreibung des Szenarios 529
Voraussetzungen 529
Auswirkungen 530

Szenario 3: Integration in bestehende Infrastruktur ohne Netzsegmentierung .. 530
Beschreibung des Szenarios 530
Voraussetzungen 531
Auswirkungen 531

Szenario 4: Integration in bestehende Infrastruktur mit Netzsegmentierung 532
Beschreibung des Szenarios 532
Voraussetzungen 532
Auswirkungen 533

Szenario 5: Zentral gesteckter HBA 533
Beschreibung des Szenarios 533
Voraussetzungen 534
Auswirkung 534

Szenario 6: Installation mit zentralem PS 535
Beschreibung des Szenarios 535
Voraussetzungen 536
Auswirkungen 536

Szenario 7: Mehrere Mandanten 537
Beschreibung des Szenarios 537
Voraussetzungen 537
Auswirkungen 538

Szenario 8: Standalone Konnektor - Logische Trennung 539
Beschreibung des Szenarios 539
Voraussetzungen 540
Auswirkung 540

Szenario 9: Standalone Konnektor - Physische Trennung 541
Beschreibung des Szenarios 541
Voraussetzungen 542
Auswirkung 542

1 Einordnung des Dokumentes

1.1 Zielsetzung

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

Dieses Dokument beschreibt die dezentrale Komponente zur sicheren Anbindung von Clientsystemen der Leistungserbringer/der Kostenträgergeschäftsstellen an die Telematikinfrastruktur – den Konnektor. Der Konnektor ist einerseits verantwortlich für den Zugriff auf die beim Leistungserbringer/der Kostenträgergeschäftstelle befindlichen Kartenterminals sowie Karten und andererseits für die Kommunikation mit den zentralen Diensten der TI-Plattform und fachanwendungsspezifischen Diensten.

Aus den Kommunikationsbeziehungen mit Clientsystem, Kartenterminals, Karten und zentralen Diensten der TI-Plattform und fachanwendungsspezifischen Diensten resultieren vom Konnektor anzubietende Schnittstellen, die gemeinsam in diesem Dokument sowie den fachanwendungsspezifischen Fachmodulspezifikationen normativ geregelt werden. Vom Konnektor genutzte Schnittstellen liegen zumeist in anderen Verantwortungsbereichen (zentrale TI-Plattform aber auch Schnittstellen der Kartenterminals und Karten). Diese werden in den übergreifenden Spezifikationen der TI, sowie den Produkttypspezifikationen definiert.

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

1.2 Zielgruppe

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

1.3 Geltungsbereich

Dieses Dokument enthält normative Festlegungen zur Telematikinfrastruktur des Deutschen Gesundheitswesens. Der Gültigkeitszeitraum der vorliegenden Version und deren Anwendung in Zulassungsverfahren wird durch die gematik GmbH in gesonderten Do-kumenten (z.B. Dokumentenlandkarte, Produkttypsteckbrief, Leistungsbeschreibung) festgelegt und bekannt gegeben.

Wichtiger Schutzrechts-/Patentrechtshinweis

Die nachfolgende Spezifikation ist von der gematik allein unter technischen Gesichtspunkten erstellt worden. Im Einzelfall kann nicht ausgeschlossen werden, dass die Implementierung der Spezifikation in technische Schutzrechte Dritter eingreift. Es ist allein Sache des Anbieters oder

Herstellers, durch geeignete Maßnahmen dafür Sorge zu tragen, dass von ihm aufgrund der Spezifikation angebotene Produkte und/oder Leistungen nicht gegen Schutzrechte Dritter verstoßen und sich ggf. die erforderlichen Erlaubnisse/Lizenzen von den betroffenen Schutzrechtsinhabern einzuholen. Die gematik GmbH übernimmt insofern keinerlei Gewährleistungen.

1.4 Abgrenzung des Dokuments

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

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

1.5 Methodik

1.5.1 Anforderungen

Anforderungen als Ausdruck normativer Festlegungen werden durch eine eindeutige ID sowie die dem RFC 2119 [RFC2119] entsprechenden, in Großbuchstaben geschriebenen deutschen Schlüsselworte MUSS, DARF NICHT, SOLL, SOLL NICHT, KANN gekennzeichnet.
Sie werden im Dokument wie folgt dargestellt:

⌦ **TIP1-A_0000 <Titel der Afo>**

Text / Beschreibung⌫

Dabei umfasst die Anforderung sämtliche innerhalb der Textmarken angeführten Inhalte.

1.5.2 Offene Punkte

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

Die XYZ müssen noch definiert werden.

1.5.3 Erläuterungen zur Spezifikation des Außenverhaltens

Der Konnektor stellt einen vergleichsweise komplexen Produkttyp dar, dessen Beschreibung eine Herausforderung darstellt und somit in vielen verschiedenen Varianten möglich wäre. An dieser Stelle folgen daher wesentliche Informationen, die das korrekte Verstehen der Spezifikation fördern:

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

Normative Festlegungen, die eine Festlegung des inneren Verhalten vermuten lassen (beispielsweise die Definitionen der Technischen Use Cases - TUCs) sind nur in so weit normativ, wie ihre Festlegungen auf die Außenschnittstelle wirken. Sie legen explizit nicht die intern zu verwendende Implementierung fest. Die Notwendigkeit für diese Art der „scheinbaren internen Beschreibung“ ergibt sich aus der Komplexität der Gesamtkomponente, sowie dem Bedarf, wiederholt ähnlich Verhaltsweisen in Außenschnittstellen darstellen zu müssen. In diesem Fall werden die sich wiederholenden Verhaltensanteile in internen TUCs zur editoriellen Wiederverwendung gekapselt. Die konkrete konnektorinterne Modularisierung bleibt dem Hersteller freigestellt. Insbesondere bleibt es dem Hersteller freigestellt, intern bereits Mechanismen für kommende Releases zu realisieren, sofern diese an der Außenschnittstelle keine Auswirkung zeigen (bspw. Vorab-Implementierung von ELC für Gen 2-Karten)

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

1.5.4 Erläuterungen zur Dokumentenstruktur und „Dokumentenmechanismen “

1.5.4.1 Modulare Spezifikation über Funktionsmerkmale

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

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

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

1. Funktionsmerkmalweite Aspekte
2. Durch Ereignisse ausgelöste Reaktionen
3. Interne TUCs, **nicht** durch Fachmodule nutzbar
4. Interne TUCs, **auch** durch Fachmodule nutzbar
5. Operationen an der Außenschnittstelle

6. Betriebsaspekte

Die Unterkapitel 1-5 dienen der funktionalen Beschreibung des Funktionsmerkmals.

Punkte, die zum Funktionieren des Funktionsmerkmals relevant sind: Initialisierungsaspekte, durch den Administrator festzulegenden Konfigurationsparameter etc., werden im Unterkapitel Betriebsaspekte erfasst.

In jedem Funktionsmerkmal sind immer alle Unterkapitel enthalten, auch wenn es im konkreten Einzelfall dort keine Inhalte gibt. Diese feste Struktur innerhalb der Funktionsmerkmale erleichtert die Orientierung und erhöht somit die Lesbarkeit.

1.5.4.2 Technische Use Cases - TUCs

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

Die Beschreibung der TUCs erfolgt nach folgendem Muster:

- TUC-Tabelle
- Aktivitäts- oder Sequenzdiagramm (optional)
- Fehlercodetabelle

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

Die Spezifikation, in der ein TUC definiert wird, ist an den mittleren drei Buchstaben der TUC-Referenz zu erkennen: TUC_KON_001 entsprechend in dieser Konnektorspezifikation, TUC_PKI_019 in der PKI-Spezifikation [gemSpec_PKI] und TUC_VPN_001 in der Spezifikation des VPN-Zugangsdienstes [gemSpec_VPN_ZugD].

1.5.4.3 Event-Mechanismus

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

Um den Ursprung eines Events erkennen zu können, verwenden alle Events ein Haupt-Topic passend zum Funktionsmerkmal: „DHCP/LAN_CLIENT/RENEW" wird im Funktionsmerkmal DHCP ausgelöst, "CARD/INSERTED" wird im Funktionsmerkmal Kartendienst ausgelöst usw.

1.5.4.4 Konfigurationsparameter und Zustandswerte

Werte die der Administrator des Konnektors einsehen oder verändern können muss, werden zusätzlich zu den Festlegungen in Kapitel 4.3 Konnektormanagement auch pro Funktionsmerkmal in den jeweiligen Unterkapiteln „Betriebsaspekte" erhoben. Diese **Konfigurationsparameter** werden über eine ReferenzID definiert. Definierte Konfigurationsparameter können in allen Kapiteln der Spezifikation referenziert werden. Den Ort, an welchem ein solcher Konfigurationsparameter definiert/erhoben und somit dessen Bedeutung beschrieben wird, lässt sich über den Präfix der ReferenzID erkennen: `CERT_CRL_DOWNLOAD_ADDRESS` (also „Cert") wird im Zertifikatsdienst definiert, `MGM_LU_ONLINE` (also „MGM") wird im Konnektormanagement definiert usw.

Die Referenz-IDs der Konfigurationsparameter besitzen in ihrer Schreibweise nur innerhalb des Dokuments Gültigkeit. In der Umsetzung können für die Konfigurationswerte herstellerspezifische Beschreibungen und Labels verwendet werden.

Vergleichbar zu diesen Konfigurationsparametern, sind **Zustandswerte.** Auch diese werden über Referenz-IDs definiert, nur können sie nicht durch den Administrator verändert oder eingesehen werden. Sie finden nur konnektorintern Verwendung und sind für die Beschreibung der Verhaltensweise notwendig, Beispiele sind `CTM_CT_LIST` für die Liste der durch den Konnektor verwalteten Kartenterminals oder `CM_CARD_LIST` für die Liste der aktuell erreichbaren Karten. Zustandswerte verwenden die gleichen Präfixe wie Konfigurationsparameter.

1.5.5 Phasen „VSDM" und „VSDM und Basisdienst QES"

Der Online-Rollout (Stufe 1) erfolgt gemäß Leistungsbeschreibung in zwei Phasen.

In **Phase „VSDM"** liegt der Fokus auf der Bereitstellung der von der Anwendung VSDM benötigten Dienste sowie der Bereitstellung des sicheren Internetzugangs (SIS) und der Anbindung von Bestandsnetzen am Beispiel des Zugangs zum sicheren Netz der KVen (SNK).

In der zweiten **Phase „VSDM und Basisdienst QES"** kommen der Basisdienst QES und die HSM-B Ausprägung der SM-B hinzu.

Definiert wird der Leistungsumfang des Konnektors in Phase „VSDM" durch sämtliche Anforderungen an den Konnektor reduziert um die Anforderungen, die folgende vier Leistungen betreffen:

- Signaturdienst und SAK (QES und nonQES),
- Verschlüsselungsdienst,
- HBA-Unterstützung und
- die HSM-B Ausprägung der SM-B.

Entsprechend der Reduktion um diese vier Leistungen sind folgende Kapitel insgesamt nicht normativ für Phase „VSDM“:

- 4.1.2 Dokumentenvalidierungsdienst
- 4.1.5.4.11 TUC_KON_210 „SchreibeRecord“
- 4.1.5.4.13 TUC_KON_215 „SucheRecord“
- 4.1.5.4.16 TUC_KON_218 „Signiere“
- 4.1.5.4.17 TUC_KON_219 „Entschlüssele“
- 4.1.7 Verschlüsselungsdienst
- 4.1.8 Signaturdienst und SAK

Außerdem entfallen einzelne Anforderungen für Phase „VSDM“ aus Kapitel 3 Übergreifende Festlegungen.

Alle Kapitel und Anforderungen im vorliegenden Dokument, die nicht in Phase „VSDM“ umgesetzt werden, sind türkis markiert. Anforderungen, die in Phase „VSDM“ umgesetzt werden, die aber spezifische Anteile der Phase „VSDM und Basisdienst QES“ enthalten, sind nicht markiert. Die Anteile, die auf in Phase „VSDM“ nicht vorhandene Schnittstellen wirken, gelten automatisch als erfüllt.

2 Systemüberblick

Der Konnektor ist ein Produkttyp der TI gemäß [gemKPT_Arch_TIP#5.3.9].

Er bietet seine Basisdienste sowohl intern den in ihm laufenden Fachmodulen an, als auch externen Clientsystemen über die Konnektoraußenschnittstellen.

Im lokalen Netz des Leistungserbringers/Kostenträgers (LEKTR LAN) kommuniziert das Clientsystem mit dem Konnektor über dessen LAN-seitiges Ethernet-Interface. Alleinig der Konnektor kommuniziert mit den in lokalen Netzen angeschlossenen Kartenterminals und Karten. Auch die Kommunikation mit den zentralen Diensten der TI-Plattform und fachanwendungsspezifischen Diensten erfolgt ausschließlich über den Konnektor über dessen WAN-seitiges Ethernet-Interface. Abbildung 1 stellt die Schnittstellen im Umfeld des Konnektors dar.

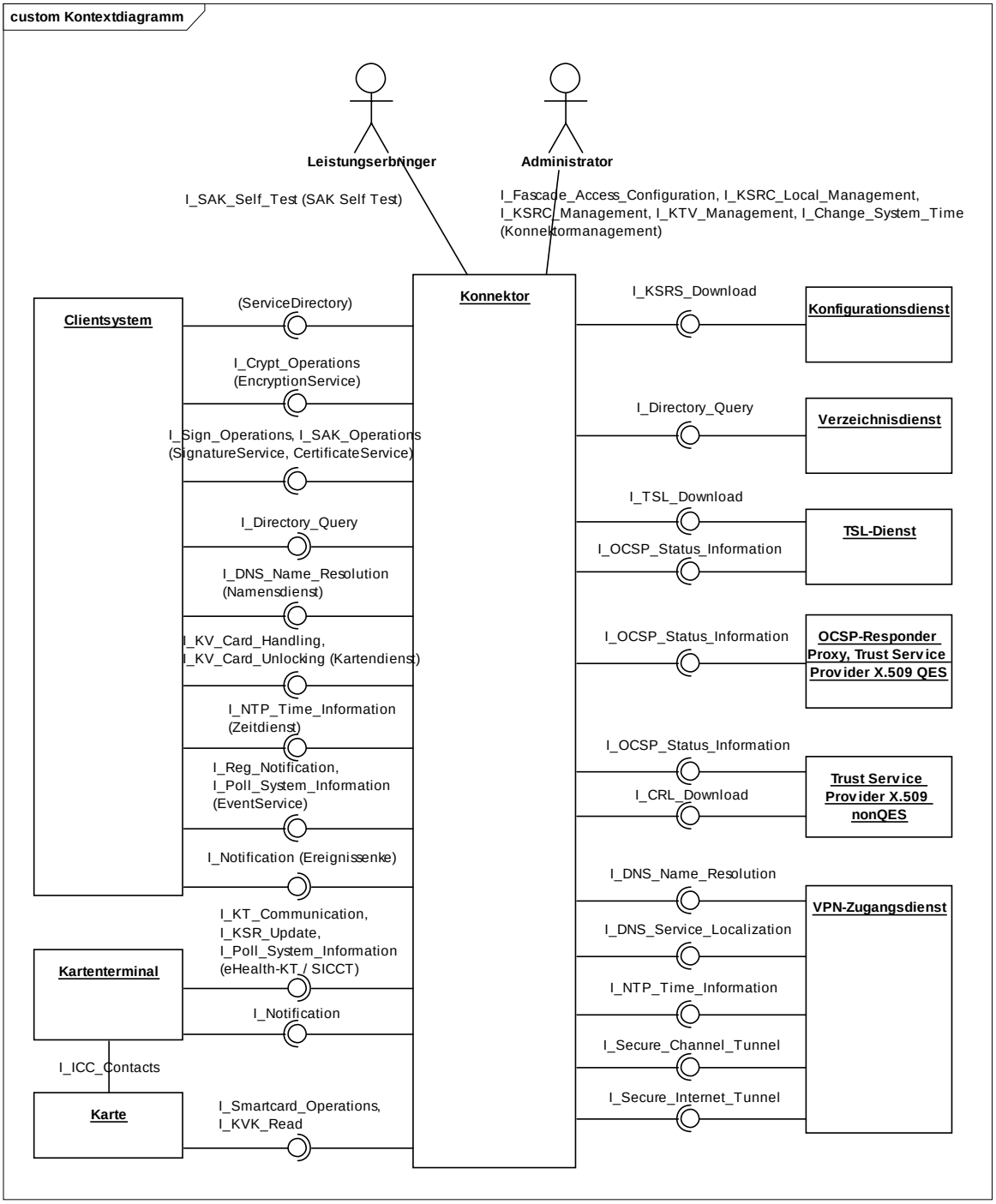


Abbildung 1 PIC_KON_116 Schnittstellen des Konnektors von und zu anderen Produkttypen

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 Übersicht der Zuordnung „logische Schnittstellen → technische Schnittstellen“ findet sich in Anhang H.

Zum Nachweis der Sicherheit müssen Konnektoren im Rahmen der Zulassung nach Common Criteria gegen die Schutzprofile [PP NK] und [PP KON_ORS1] evaluiert und

zertifiziert werden. Da es sich bei dem Konnektor auch um eine SAK nach [SigG01]/[SigV01] handelt, muss der Konnektor auch nach [SigG01]/[SigV01] bestätigt werden.

Die zu verwendenden kryptographischen Verfahren und zugehörige Parameter (z. B. Schlüssellängen) für alle kryptographischen Operationen innerhalb der Telematikinfrastruktur, werden durch das Dokument „Verwendung kryptographischer Algorithmen in der Telematikinfrastruktur“ [gemSpec_Krypt] normativ geregelt.

2.1 Logische Struktur

Der Produkttyp Konnektor besitzt eine Vielzahl verschiedenster Operationen und Verhaltsweisen an seiner Außenschnittstelle. Um sein komplexes Gesamtverhalten sinnvoll beschreiben zu können, wird der Konnektor innerhalb dieser Spezifikation logisch unterteilt und strukturiert. Es wird primär zwischen Anwendungs- und Netzkonnektor unterschieden, begleitet von Mechanismen, die blockübergreifend beschrieben werden.

Der logische Aufbau des Konnektors ist in Abbildung 2 dargestellt.

- Der Anwendungskonnektor bietet anwendungsnahe Basisdienste (inklusive SAK) und Fachmodule zur Nutzung durch ein Clientsystem an.
- Der Netzkonnektor bietet transportnahe Basisdienste und verbindet das lokale Netz des Leistungserbringers / Kostenträgers mit der zentralen TI-Plattform.
- Die gSMC-K ist zwar ein eigenständiger Produkttyp innerhalb der TI, wird im Konnektor jedoch als Verbaukomponente betrachtet. Sie enthält die kryptographischen Identitäten des Konnektors, sowie Steuerdaten (Umgebungsinformationen TU/RU/PU, zugehörige Adressbereiche, herstellerspezifische Konfigurationsdaten), die aus Sicherheitsgründen unveränderlich in den Konnektor eingebracht werden müssen.
- Das Konnektormanagement dient der administrativen Verwaltung und Steuerung des gesamten Konnektors.
- Der Sichere Datenspeicher dient der integeren, vertraulichen und authentischen Persistierung von veränderlichen Daten (siehe auch Kapitel 2.2).
- Der Extended Trusted Viewer (xTV) gehört logisch zum Konnektor, wird jedoch abgesetzt von diesem am Arbeitsplatz des Nutzers betrieben. Es bildet die Interaktionskomponente das SAK zum Signierenden bzw. Signatur-Prüfenden.

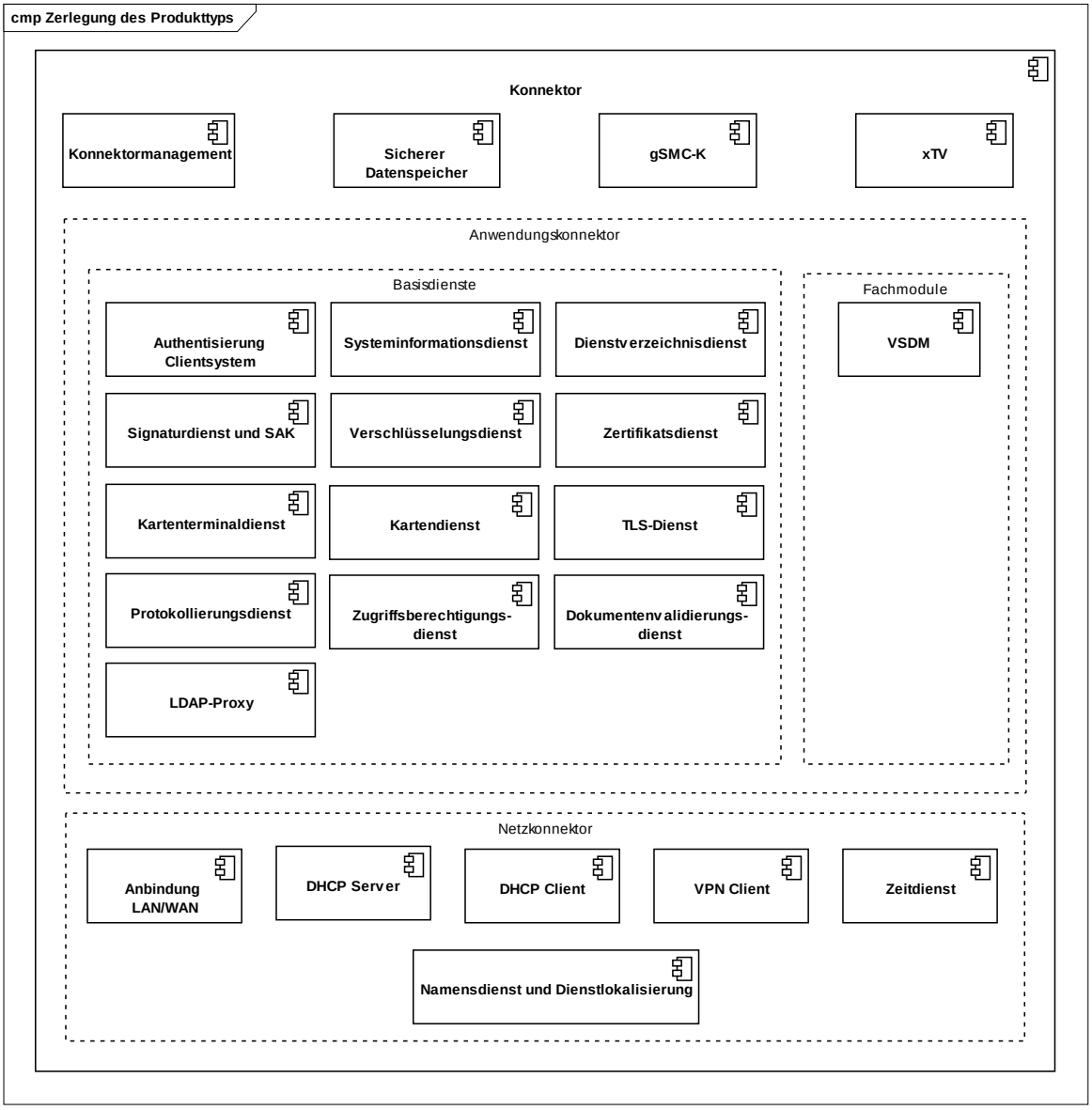


Abbildung 2 PIC_KON_117 Logische Zerlegung des Konnektors in Anwendungs- und Netzkonnektor

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.

2.2 Sicherer Datenspeicher

Wie im vorherigen Kapitel dargestellt, wird für den Konnektor ein Datenspeicher angenommen, in welchem der Konnektor alle sicherheitskritischen, veränderlichen Daten dauerhaft speichert, die für seinen Betrieb relevant sind. Dieser Datenspeicher sichert die Integrität, Authentizität und Vertraulichkeit der in ihm hinterlegten Daten bzw. der aus ihm entnommenen Daten. Alleinig der Konnektor hat auf diesen Datenspeicher Zugriff. Für

folgende, im weiteren Verlauf der Spezifikation anfallende Daten wird angenommen, dass diese im Sicheren Datenspeicher persistiert werden:

- Der Trust Store des Zertifikatsdienstes
- Die Konfigurationsdaten des Konnektormanagements
- Die Konfigurationsdaten aller Funktionsmerkmale

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

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

2.3 Überblick Konnektoridentität

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

- ID.NK.VPN für den Netzkonnektor
  Die Identität des Netzkonnektors dient der Authentisierung gegenüber den zentralen Netzwerkdiensten und wird für die Anmeldung an den VPN-Konzentrator genutzt.
- ID.AK.AUT für den Anwendungskonnektor
  Die Identität des Anwendungskonnektors dient der Authentisierung gegenüber den Clientsystemen im Rahmen von TLS-Verbindungen.
- ID.SAK.AUT für die im Anwendungskonnektor enthaltene Signaturanwendungskomponente
  Die Identität der SAK dient zur Authentisierung gegenüber den Kartenterminals und dem Extended Trusted Viewer (xTV). Darüber hinaus muss sich die SAK des Konnektors gegenüber dem Heilberufsausweis mittels eines kartenverifizierbaren Zertifikats (C.SAK.AUTD_CVC) mit entsprechendem Profil ausweisen, um Stapelsignaturen durchführen zu können.

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

Die Geräteidentitäten werden durch asymmetrische Schlüssel und X.509-Zertifikate umgesetzt. Es werden RSA-Schlüssel verwendet.

2.4 Mandantenfähigkeit

Den Anforderungen aus [gemKPT_Arch_TIP#TIP1-A_2200] folgend, wird die Mandantenfähigkeit innerhalb des Konnektors nicht durch eine einzelne Funktion, sondern durch Berücksichtigung in einer Reihe von Funktionsmerkmalen umgesetzt.

Die Mandantenfähigkeit wirkt dabei auf:

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

2.5 Versionierung

Gemäß [gemSpec_OM] müssen Konnektor und Kartenterminals über eine Versionierung verfügen. Die relevanten Versionsinformationen sind durch das O&M-Schema Product-Information.xsd definiert. Ferner definiert [gemSpec_OM], dass Konnektor und Kartenterminal das Konzept der Firmware-Gruppe verwenden müssen. Daher verfügen die beiden Produkttypen auch über eine aktuelle Firmware-Gruppenversion.

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

- Dienstverzeichnisdienst (Kapitel 4.1.3): Ausgabe der Konnektorversion über SOAP
- Kartenterminaldienst (Kapitel 4.1.4): Anzeige der Versionsinformationen der verwalteten Kartenterminals
- Konnektormanagement (Kapitel 4.3):
  - Anzeige der Versionsinformationen des Konnektors (Kapitel 4.3.2)
  - Software-Aktualisierung (KSR-Client) für Konnektor und Kartenterminals (Kapitel 4.3.9)

2.6 Fachanwendungen

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

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

In dieser Ausbaustufe unterstützt der Konnektor die Fachanwendung VSDM über ein Fachmodul.

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

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 Leistungserbringer 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 des LE verwendet den DHCP-Server des Konnektors (siehe Kap. 4.2.2).
- Ein bestehender DHCP-Server im Netz des LE 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 des LE 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.

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).

2.8 Lokale und entfernte Kartenterminals

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

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

2.9 Standalone-Szenario

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

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

Ein solcher alleinstehender Konnektor mit Zugang zu TI muss zur Durchführung der Fachanwendungen durch einen weiteren Konnektor unterstützt werden, der in direkter Verbindung zum Clientsystem steht, selbst aber keine Online-Anbindung besitzt. Die im Gesetz geforderte informationstechnische Trennung (Online-Bereich vs. Offline-Bereich im lokalen Netz des Leistungserbringers) kann auch durch eine logische Separation innerhalb eines Konnektors erreicht werden, die der Administrator zuschalten kann (siehe Kapitel 4.3.6). Ist dieser Modus aktiviert, so garantiert der Konnektor, dass keine Daten aus Richtung der TI zum Clientsystem gelangen (und umgekehrt). In dieser Betriebsart ist demnach statt zwei Konnektoren nur ein Konnektor notwendig.

3 Übergreifende Festlegungen

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

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

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

Dokumentformate

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

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

Folgende Bezeichner werden verwendet:

`Alle_DocFormate`: XML, PDF/A, Text, TIFF, Binär

`nonQES_DocFormate`: XML, PDF/A, Text, TIFF, Binär

`QES_DocFormate`: XML, PDF/A, Text, TIFF

Für nonQES_DocFormate wird, trotz Gleichheit zu Alle_DocFormate, ein eigener Referenzbezeichner verwendet, da sich diese Liste noch ändern könnte. TIFF wird durch [gemKPT_Arch_TIP] nicht für die nonQES verlangt. Die Unterstützung dieses Formats für nonQES bedeutet jedoch keinen Mehraufwand, da die Routinen durch QES bereits implementiert sind und nachgenutzt werden können.

Die QES_DocFormate müssen hinsichtlich ihrer Anzeigequalität im xTV weiter verfeinert werden:

- `QES_DF_SecView`: Dokumentformate, die im xTV sicher angezeigt werden können (Sicherheit der Anzeige bestätigt nach SigG/SigV)
  - `DF_SV_PDFA`: PDF/A-2b [PDF/A-2] unter Beachtung der in Anhang C festgelegten Einschränkungen

- `DF_SV_Text`: Text-Dokument (Zeichensatz ISO-8859-15 oder UTF-8)
- `DF_SV_TIFF`: TIFF 6.0: Part 1 Baseline TIFF [TIFF6] unter Beachtung der in Anhang C festgelegten Einschränkungen
- `DF_SV_XML`: Liste aller sicher anzeigbaren XML-Dokumentformate. Dies sind:
  - `DF_SV_VHITG_ARZTBRIEF`: VHiTG-Arztbrief, siehe [CDA], und / eArztbrief, siehe [eAB]
  - `DF_SV_EGES`: eGesamtaufstellung, siehe [eGES]

- `QES_DF_NonSecView`: Dokumentformate, die im xTV angezeigt werden können, bei welcher der xTV die Korrektheit und Vollständigkeit der Anzeige jedoch nicht garantieren kann (bestmögliche Darstellung mit entsprechendem Warnhinweis, siehe 4.1.8.3.5). Dies umfasst alle Dokumente, die zwar zur Gruppe der QES_DocFormate gehören, aber im Einzelfall in konkreten Teilen von den unter QES_DF_SecView benannten Formaten abweichen (beispielsweise unerlaubte Subelemente beinhalten).

Die in DF_SV_XML definierten XML-Formate werden innerhalb der Signaturrichtlinien in Anhang C spezifiziert.

⌦ TIP1-A_4500 Dokumentgrößen von 25 MB

Der Konnektor MUSS für alle Außenschnittstellen, in denen ein Dokument verarbeitet wird, Dokumente mit einer Größe <= 25 MB unterstützen. Der Konnektor KANN Dokumente mit einer Größe > 25 MB unterstützen. ⌫

⌦ TIP1-A_4502 Zeichensatzcodierungen UTF-8 und ISO-8859-15

Der Konnektor MUSS bei der Verarbeitung von Dokumenten der Formate XML und Text die Zeichensatzkodierungen UTF-8 und ISO-8859-15 unterstützen. Das verarbeitete Dokument MUSS der Konnektor mit demselben Zeichensatz kodieren, in dem das Eingangsdokument kodiert war.⌫

⌦ TIP1-A_5541 Referenzen in Dokumenten nicht dynamisch auflösen

Der Konnektor DARF die in Dokumenten eventuell vorhandenen Referenzen auf externe Ressourcen NICHT durch Aufrufe über Netzwerkverbindungen hinweg auflösen, es sei denn, dies wird im Einzelfall normativ gefordert.⌫

Kartentypen

Der Konnektor unterstützt eine Reihe von Kartentypen. Die folgende Tabelle enthält die Liste der Referenzbezeichner für die verschiedenen Kartentypen, wie sie im weiteren Verlauf verwendet werden. Die Unterstützung von Karten der Generation 2 beschränkt

sich bei diesen auf die Datenstrukturen und Schlüssel, die aus Gründen der Abwärtskompatibilität zu den Karten der Generation 1 vorhanden sind. Eine Ausnahme hiervon bilden die Geräte-CVCs, die bereits für dieses Release basierend auf ELC verwendet werden.

Tabelle 1: TAB_KON_500 Wertetabelle Kartentypen

Referenz-ID Kartentyp	Karten-generation	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]
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 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 Referenz-IDs
HBA-VK		Adressiert die HBA-Vorläuferkarten HBA-qSig und ZOD_2.0. 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) 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. Wird dieser Referenzbezeichner verwendet, gelten die zugehörigen Aussagen und Festlegungen für beide Typen.

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 gSCM-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**

Der Konnektor MUSS folgende Container der gSMC-K als Quelle seiner Identitäten verwenden:

- Für ID.NK.VPN: MF/DF.NK/EF.C.NK.VPN.R2048
- Für ID.AK.AUT: MF/DF.AK/EF.C.AK.AUT.R2048
- Für ID.SAK.AUT: MF/DF.SAK/EF.C.SAK.AUT.R2048
- Für C.SAK.AUTD_CVC: MF/DF.SAK/EF.C.SAK.AUTD_CVC.E256⌫

3.1.1 Organisatorische Anforderungen und Sperrprozesse

⌦ **TIP1-A_5392 gSMC-K-Verantwortung durch den Hersteller des Konnektors**

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

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

⌦ **TIP1-A_5393 Dokumentation der Konnektorzertifikatszuordnungen**

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

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

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

3.2 Bootup-Phase

⌦ **TIP1-A_4507 Isolation während der Bootup-Phase**

Da während der Bootup-Phase des Konnektors noch nicht alle Sicherheitsmechanismen ihre Leistung erbringen können, DÜRFEN die Dienste des Konnektors während dem Bootup über physikalische Schnittstellen von außen NICHT erreichbar sein. ⌫

⌦ **TIP1-A_4508 Konnektorzustand nach Bootup**

Der Konnektor MUSS nach Beendigung der Bootup-Phase die Initialisierung der Funktionsmerkmale durchlaufen haben. Die Startreihenfolge der Funktionsmerkmale kann unter Berücksichtigung von TIP1-A_4507 herstellerspezifisch gestaltet werden.

Im Rahmen der Bootup-Phase MÜSSEN folgende TUCs ausgeführt werden: TUC_KON_025, TUC_KON_035, TUC_KON_272, TUC_KON_341, TUC_KON_343, TUC_KON_352 (die Reihenfolge der TUC-Ausführung ist herstellerspezifisch).

Treten während der Bootup-Phase Fehler auf, so MUSS die Bootup-Phase, sofern möglich, abgeschlossen werden.

Sobald die Bootup-Phase abgeschlossen ist, MUSS TUC_KON_256 „Systemereignis absetzen“ mit folgenden Parameter aufgerufen werden:

TUC_KON_256{"BOOTUP/BOOTUP_COMPLETE"; Op; Info; ""}⌫

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

3.3 Betriebszustand

⌦ **TIP1-A_4509 Betriebszustand erfassen**

Der Konnektor MUSS seinen Betriebszustand gemäß Tabelle TAB_KON_503 Betriebszustand_Fehlerzustandsliste über Fehlerzustände $EC erfassen.

Tritt die in Spalte „Beschreibung“ charakterisierte Fehlersituation eines Fehlerzustandes $EC ein, wird sein Wert $EC.value = true. Sobald die Fehlersituation beendet ist, springt der Wert auf $EC.value = false. Die Fehlerzustände müssen dabei innerhalb der „max. Feststellungszeit“ (Tabellenspalte) erfasst werden. Eine maximale Feststellungszeit von einen Tag (1 day) verlangt, dass einmal am Tag der Zustand geprüft werden muss, unabhängig davon, welche TUCs aufgerufen werden. Eine maximale Feststellungszeit unter einer Minute (1 sec, 10 sec) verlangt, dass nach der Feststellung einer Fehlfunktion innerhalb eines TUCs die Zustandsänderung innerhalb der angegebenen Zeit stattfinden muss.

Nach Abschluss des Boot-Vorgangs müssen sämtliche Fehlerzustände mit einer „max. Feststellungszeit“ von „1 day“ erfasst worden sein. ⌫

⌦ **TIP1-A_4597 Unterstützung von Missbrauchserkennungen**

Der Konnektors MUSS zur Unterstützung von Missbrauchserkennungen für alle Operationen, die in EVT_MONITOR_OPERATIONS gelistet sind und deren Alarmwert > 0 ist, kontinuierlich folgende Aktivitäten durchlaufen:

1. Minütlich gleitende 10-Minuten-Summe je in EVT_MONITOR_OPERATIONS gelistete Operation berechnen. Dazu gehen
   - erfolgreiche Abschlüsse der Operation mit dem OK_Val der Operation ein
   - eine fehlerhaft beendete Operation mit dem NOK_Val der Operation ein
2. Überschreitet der gleitende 10-Minuten-Summenwert einer in EVT_MONITOR_OPERATIONS gelisteten Operation den zugehörigen Alarmwert, so setze EC_CRYPTOPERATION_ALARM auf True.⌫

Erklärung „Minütlich gleitende 10-Minuten-Summe“: Für die jeweilige Operation wird die Summe aller OK_Val und NOK_Val der letzten 10 Minuten gebildet. Diese Summe wird jede Minute neu berechnet.

⌦ **TIP1-A_4510 Sicherheitskritische Fehlerzustände**

Der Konnektor MUSS bei eingetretenem Fehlerzustand aus Tabelle Tab_ Kon_503 Betriebszustand_Fehlerzustandsliste mit Severity=Fatal dafür sorgen, dass von den

Operationen der Basisdienste und Technische Use Cases (TUCs) der Basisdienste, die relevant für Fachanwendungen sind, nur erlaubte Operationen und TUCs gestartet und ausgeführt werden.

Welche Operationen und TUCs je eingetretenem Fehlerzustand ausgeführt werden dürfen, legt Tabelle „TAB_KON_504 Ausführungserlaubnis für Dienste in kritischen Fehlerzuständen“ fest: Jede Erlaubnis ist dort durch ein „x“ definiert.

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

Tabelle 2 TAB_KON_502 Fehlercodes „Betriebszustand“

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

⌫

⌦ **TIP1-A_4512 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{ "OPERATIONAL_STATE/$EC.errorcondition ";
$EC.type;
$EC.severity;
{value=$EC.value; $EC.parameterlist}
} ⌫

Tabelle 3 TAB_KON_503 Betriebszustand_Fehlerzustandsliste

ErrorCondition[1]	Beschreibung	Type	Severity	max. Feststellungs-zeit	Parameterlist
EC_CardTerminal_Software_Out_Of_Date ($ctId)	Software auf Kartenterminal($ctId) ist nicht aktuell	Op	Info	1 d	CtId=$ctId; Bedeutung=$EC.description
EC_Connector_Software_Out_Of_Date	I_KSRS_Download::list_Updates liefert mindestens eine UpdateInformation mit einer UpdateInformation/Firmware/FWVersion > aktuelle Version der Konnektorsoftware, deren UpdateInformation/Firmware/FWPriority = „Kritisch“	Op	Info	1 d	Bedeutung=$EC.description
EC_Time_Sync_Not_Successful	der letzte Synchronisationsversuch der Systemzeit war nicht erfolgreich.	Op	Info	1 sec	LastSyncAttempt= $lastSyncAttemptTimestamp; LastSyncSuccess =$lastSyncSuccessTimestamp; Bedeutung=$EC.description
EC_TSL_Update_Not_Successful	das letzte Update der TSL war nicht erfolgreich.	Op	Info	1 sec	Bedeutung=$EC.description; LastUpdateTSL=$lastUpdateTSLTimestamp
EC_TSL_Expiring	Systemzeit t mit t > NextUpdate-Element der TSL – 7 Tage und t <= NextUpdate-Element der TSL	Sec	Info	1 d	NextUpdateTSL =$NextUpdate-Element der TSL Bedeutung=$EC.description
EC_TSL_Trust_Anchor_Expiring	Gültigkeit des Vertrauensankers ist noch nicht abgelaufen, läuft aber innerhalb von 30 Tagen ab.	Sec	Info	1 d	ExpiringDateTrustAnchor=Ablaufdatum der Vertrauensankergültigkeit Bedeutung=$EC.description
EC_LOG_OVERFLOW	Wenn im Rahmen der Regeln für die rollierende Speicherung von Logging-Einträgen Einträge gelöscht werden, die nicht älter als LOG_DAYS bzw. FM_<fmName>_LOG_DAYS sind, tritt der Fehlerzustand ein.	Op	Warning	1 sec	Bedeutung=$EC.description

[1] Jeder Fehlerzustand wird durch einen eindeutigen ErrorCondition identifiziert. Dieser kann einen Parameter enthalten. Sind etwa die Kartenterminals mit ctId=47 und das mit ctId=93 nicht erreichbar, so lauten die ErrorCondition „EC_CardTerminal_Not_Available(47)“ und „EC_CardTerminal_Not_Available(93)“.

ErrorCondition[1]	Beschreibung	Type	Severity	max. Feststellungs-zeit	Parameterlist
	Der Fehlerzustand kann nur durch einen Administrator wieder zurückgesetzt werden.				
EC_CRL_Expiring	Systemzeit t > NextUpdate der CRL – 3 Tage	Sec	Warning	1 d	ExpiringDateCRL=NextUpdate der CRL Bedeutung=$EC.description
EC_Time_Sync_Pending_Warning	MGM_LU_ONLINE=Enabled und keine erfolgreiche Synchronisation der Systemzeit seit d Tagen und d > NTP_WARN_PERIOD und d <= NTP_GRACE_PERIOD. Nach einer Korrektur oder Bestätigung der Systemzeit durch einen Administrator muss der Konnektor wie nach einer erfolgreichen Zeitsynchronisation verfahren, d.h., der Tagezähler wird auf 0 zurückgesetzt.	Sec	Warning	1 d	LastSyncSuccess =$lastSyncSuccessTimestamp; Bedeutung=$EC.description
EC_TSL_Out_Of_Date _Within_Grace_Period	Systemzeit t mit t > NextUpdate-Element der TSL und t <= NextUpdate-Element der TSL + CERT_TSL_DEFAULT_GRACE_PERIOD_DAYS und eine neue TSL ist nicht verfügbar	Sec	Warning	1 day	NextUpdateTSL =$NextUpdate-Element der TSL; GracePeriodTSL =CERT_TSL_DEFAULT_GRACE_PERIOD_DAYS; Bedeutung=$EC.description
EC_CardTerminal_Not_Available ($ctId)	Bekanntes Kartenterminal($ctId) ist nicht verfügbar.	Op	Error	1 sec	CtId=$ctId; Bedeutung=$EC.description
EC_No_VPN_TI_Connection	kein sicherer Kanal (VPN) in die Telematikinfrastruktur aufgebaut. Der Wert 300 sec ist abgeleitet aus der maximalen Verbindungsaufbauzeit bei einem Standortausfall des VPN-Zugangsdienstes.	Op	Error	300 sec	Bedeutung=$EC.description
EC_No_VPN_SIS_Connection	kein sicherer Kanal (VPN) zu den Sicheren Internet Services aufgebaut. Der Wert 300 sec ist abgeleitet aus der	Op	Error	300 sec	Bedeutung=$EC.description

ErrorCondition[1]	Beschreibung	Type	Severity	max. Feststellungs-zeit	Parameterlist
	maximalen Verbindungsaufbauzeit bei einem Standortausfall des VPN-Zugangsdienstes.				
EC_No_Online_Connection	Konnektor kann Dienste im Transportnetz nicht erreichen.	Op	Error	10 sec	Bedeutung=$EC.description
EC_FeatureOrTUC_Not_Available ($Dienst/$Operation)	Dienst $Dienst oder Operation $Operation nicht verfügbar	Op	Error	1 sec	Dienst=$Dienst; Operation=$Operation; Bedeutung=$EC.description
EC_IP_Adresses_Not_Available	Die IP-Adressen des Netzkonnektors sind nicht oder falsch gesetzt.	Sec	Error	1 sec	Bedeutung=$EC.description
EC_CRL_Out_Of_Date	Systemzeit t > Next Update der CRL	Sec	Fatal	1 day	NextUpdateCRL=$NextUpdate der CRL; Bedeutung=$EC.description
EC_Firewall_Not_Reliable	Firewall-Regeln konnten nicht fehlerfrei generiert werden oder beim Laden der Firewall-Regeln ist ein Fehler aufgetreten.	Sec	Fatal	1 sec	Bedeutung=$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. Nach einer Korrektur oder Bestätigung der Systemzeit durch einen Administrator muss der	Sec	Fatal	1 sec	NtpTimedifference=Zeitabweichung; NtpMaxAllowedTimedifference =NTP_MAX_TIMEDIFFERENCE; Bedeutung=$EC.description

ErrorCondition[1]	Beschreibung	Type	Severity	max. Feststellungs-zeit	Parameterlist
	Konnektor den Fehlerzustand zurücksetzen.				
EC_Time_Sync_Pending_Critical	MGM_LU_ONLINE=Enabled und keine erfolgreiche Synchronisation der Systemzeit seit d Tagen und d > 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 =$lastSyncSuccessTimestamp; NtpGracePeriod=NTP_GRACE_PERIOD Bedeutung=$EC.description
EC_TSL_Trust_Anchor_Out_Of_Date	Gültigkeit des Vertrauensankers ist abgelaufen	Sec	Fatal	1 day	ExpiringDateTrustAnchor=Ablaufdatum der Vertrauensankergültigkeit; Bedeutung=$EC.description
EC_TSL_Out_Of_Date _Beyond_Grace_Period	Systemzeit t mit t > NextUpdate-Element der TSL + CERT_TSL_DEFAULT_GRACE_PERIOD_DAYS und eine neue TSL ist nicht verfügbar	Sec	Fatal	1 day	NextUpdateTSL =$NextUpdate-Element der TSL; GracePeriodTSL =CERT_TSL_DEFAULT_GRACE_PERIOD_DAYS; Bedeutung=$EC.description
EC_CRYPTOPERATION_ALARM	Gemäß TIP1-A_4597 wurde ein potentieller Missbrauch einer Kryptooperation erkannt. Nur der Administrator kann die Alarmmeldung zurücksetzen.	Sec	Warning	1 minute	Operation=$Operationsname; Count=$Summenwert; Arbeitsplatz=$<Liste operationsaufrufenden workplaceIDs>; Meldung='Auffällige Häufung von Operationsaufrufen in den letzten 10 Minuten'

ErrorCondition[1]	Beschreibung	Type	Severity	max. Feststellungs-zeit	Parameterlist
EC_OTHER_ERROR_STATE($no)	Herstellerspezifische Fehlerzustände, die per $no (von 1 aufsteigend nummeriert) identifiziert werden. $Type, $Severity und $ParameterList legt der Hersteller nach Bedarf fest.	$Type	$Severity	<= 1 day	Bedeutung=$EC.description

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

Tabelle 4 TAB_KON_504 Ausführungserlaubnis für Dienste in kritischen Fehlerzuständen

		EC_Software_Integrity_Check_Failed	EC_Random_Generator_Not_Reliable	EC_Security_Log_Not_Writable	EC_Time_Sync_Pending_Critical	EC_Time_Difference_Intolerable	EC_CRL_Out_Of_Date	EC_TSL_Out_Of_Date_Beyond_Grace	EC_TSL_Trust_Anchor_Out_Of_Date	EC_Firewall_Not_Reliable	EC_Secure_KeyStore_Not_Available
Technische Use Cases (TUCs) der Basisdienste relevant für Fachanwendung und die Kommunikation mit SNK und SIS											
Zugriffsberechtigungsdienst											
TUC_KON_000	PrüfeAufrufkontext	-	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	-	-
TUC_KON_256	Systemereignis absetzen	-	x	x	x	x	x	x	x	x	x
Kartendienst											
TUC_KON_005	Card-to-Card authentisieren	-	-	-	-	-	x	x	x	-	-
TUC_KON_006	Datenzugriffsaudit eGK schreiben	-	-	-	-	-	x	x	x	-	-
TUC_KON_018	eGK-Sperrung prüfen	-	-	-	-	-	x	x	x	-	-
TUC_KON_024	Karte zurücksetzen	-	-	-	-	-	x	x	x	-	-
TUC_KON_200	SendeAPDU	-	-	-	-	-	x	x	x	-	-
TUC_KON_202	LeseDatei	-	-	-	-	-	x	x	x	-	-
TUC_KON_203	SchreibeDatei	-	-	-	-	-	x	x	x	-	-
TUC_KON_209	LeseRecord	-	-	-	-	-	x	x	x	-	-
Verschlüsselungsdienst											
TUC_KON_072	Daten symmetrisch verschlüsseln	-	-	-	x	x	x	x	x	-	-
Zertifikatsdienst											
TUC_KON_034	Zertifikatsinformationen extrahieren	-	-	-	x	x	x	x	x	-	-
Protokollierungsdienst											
TUC_KON_271	Schreibe Protokolleintrag	-	-	-	-	-	x	x	x	-	-
TLS-Dienst											
TUC_KON_110	Kartenbasierte TLS-Verbindung aufbauen	-	-	-	-	-	-	-	-	-	-
Verbindung zum VPN-Konzentrator											

	EC_Software_Integrity_Check_Failed	EC_Random_Generator_Not_Reliable	EC_Security_Log_Not_Writable	EC_Time_Sync_Pending_Critical	EC_Time_Difference_Intolerable	EC_CRL_Out_Of_Date	EC_TSL_Out_Of_Date_Beyond_Grace	EC_TSL_Trust_Anchor_Out_Of_Date	EC_Firewall_Not_Reliable	EC_Secure_KeyStore_Not_Available
TUC_KON_321 Verbindung zu dem VPN-Konzentrator der TI aufbauen	-	-	-	-	-	-	-	-	-	-
TUC_KON_322 Verbindung zum dem VPN-Konzentrator des SIS aufbauen	-	-	-	-	-	-	-	-	-	-
Operationen der Basisdienste										
Kartendienst										
AuthorizeSMC	-	-	-	-	-	X	X	X	-	-
VerifyPin	-	-	-	-	-	X	X	X	-	-
UnblockPin	-	-	-	-	-	X	X	X	-	-
ChangePin	-	-	-	-	-	X	X	X	-	-
GetPinStatus	-	-	-	-	-	X	X	X	-	-
Systeminformationsdienst										
Schnittstelle der 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
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	-	-
DecryptDocument	-	-	-	-	-	X	X	X	-	-
Signaturdienst und SAK										
SignDocument	-	-	-	-	-	X	X	X	-	-
VerifyDocument	-	-	-	-	-	X	X	X	-	-
SetXTVView	-	-	-	-	-	X	X	X	-	-
GetXTVView	-	-	-	-	-	X	X	X	-	-
Zertifikatsdienst										
ReadCardCertificate	-	-	-	-	-	X	X	X	X	X
CheckCertificateExpiration	-	-	-	-	-	X	X	X	X	X
VerifyCertificate	-	-	-	-	-	X	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

In den kritischen Fehlerzuständen, in denen keine TLS-Verbindung ins LAN aufgebaut werden (EC_Random_Generator_Not_Reliable, EC_Software_Integrity_Check_Failed, EC_Security_Log_Not_Writable, EC_Time_Sync_Pending_Critical, EC_Time_Difference_Intolerable), kann keine Verbindung zu den Kartenterminals aufgebaut werden. Infolge sind hier keine Kartenoperationen zugelassen.

Wenn keine Verbindung zum VPN-Konzentrator des SIS aufgebaut werden kann, ist infolge das Internet nicht über den Konnektor erreichbar. Wenn keine Verbindung zum VPN-Konzentrator der TI aufgebaut werden kann, ist das sichere Netz der KVen (SNK) nicht erreichbar.

Die Architektur der TI ist so angelegt, dass die Fehlerzustände mit Severity=Fatal in den Tabellen TAB_KON_504 und TAB_KON_503 mit vernachlässigbarer Wahrscheinlichkeit von externen Einflüssen abhängen. Die SLAs für Dienste der zentralen TI-Plattform sind so gefasst, dass diese schwerwiegend verletzt werden müssten, um dadurch einen Konnektor in einen solchen kritischen Zustand zu bringen (externer Fehler aus Sicht des Konnektors). Dass beispielsweise die TSL über den Zeitraum der Grace-Period-TSL (typisch: 7 Tage) nicht erreichbar ist (ErrorCondition „EC_Time_Sync_Pending_Critical“), kann nur bei massiver Verletzung der für zentrale Dienste festgelegten SLAs eintreten.

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

⌦ **TIP1-A_5148 Performance - Konnektor - Mittlerer Abstand zwischen Ausfällen**

Der Konnektorhersteller MUSS den mittleren Zeitabstand zwischen Ausfällen (MTBF) als Produkteigenschaft ausweisen. Der Konnektor soll einen mittleren Zeitabstand zwischen Ausfällen (MTBF) von mindestens 50 Jahren haben.

Ein „Ausfall“ gilt dann als eingetreten, wenn

- der Konnektor nicht mehr gebootet werden kann, d. h. kein „BOOTUP/BOOTUP_COMPLETE“ Event ausgelöst wird, und dies nicht auf einen externen Fehler zurückzuführen ist,
- oder sich der Konnektor in einem Fehlerzustand mit Severity=Fatal befindet, der nicht auf einen externen Fehler zurückzuführen ist,
- oder Funktionen des Konnektors ausgefallen sind, ohne dass dies auf externe Fehler zurückzuführen ist. ⌫

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

3.3.1 Betriebsaspekte

Der Konnektor soll per Signaleinrichtung am Konnektor die Fehlerzustände mit Severity „Error“ und „Fatal“ anzeigen (siehe [TIP1-A_4843]).

⌦ **TIP1-A_4513 Betriebszustände anzeigen und Fehlerzustände zurücksetzen**

Der Konnektor MUSS es dem Administrator ermöglichen, den aktuellen Betriebszustand einzusehen und Fehlerzustände zurückzusetzen, soweit diese Möglichkeit

in Tabelle „TAB_KON_503 Betriebszustand_Fehlerzustandsliste“ für den jeweiligen Fehlerzustand festgelegt ist.

Ferner MUSS es die Managementschnittstelle dem Administrator ermöglichen Konfigurationsänderungen gemäß Tabelle TAB_KON_505 vorzunehmen:

Tabelle 5 TAB_KON_505 Konfigurationswerte Missbrauchserkennung

Referenz-ID	Belegung	Bedeutung und Administrator-Interaktion
EVT_MONITOR_OPERATIONS	Liste von: - Operationsname - OK_Val (Nummer) - NOK_Val (Nummer) - Alarmwert (Nummer)	Der Administrator MUSS in der Liste der zur Missbrauchserkennung überwachbaren Operationen alle Listeneinträge einsehen können. Er MUSS den jeweiligen Alarmwert editieren können (0-9999, 0=deaktiviert). OK_VAL und NOK_VAL DÜRFEN durch den Administrator NICHT veränderbar sein.

⌫

3.4 Fachliche Anbindung der Clientsysteme

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

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

Dabei werden die folgenden zwei Nutzungsszenarien nicht unterschieden:

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

Sowohl die Anbindung zur Administration des Konnektors, als auch die Anbindung zur Nutzung von Bestandsnetzen oder dem gesicherten Internetzugang sind nicht Gegenstand dieser Schnittstellenfestlegungen. Für die Anbindung zu Administration wird diese im Kapitel Konnektormanagement beschrieben, für die Anbindung von Bestandsnetzen bzw. dem gesicherten Internetzugang ist diese Art der Regelung nicht erforderlich, da es sich dort um Routing-Funktionen handelt.

Die seitens des Administrators einstellbaren Werte und Listen sind, der allgemeinen Struktur dieses Dokuments folgend, im Unterkapitel 3.4.1 Betriebsaspekte beschrieben.

⌦ **TIP1-A_4514 Verfügbarkeit einer TLS Schnittstelle**

Der Konnektor MUSS TLS in Richtung der Clientsysteme für alle Außenschnittstellen der Basisdienste:

- Dienstverzeichnisdienst
- Kartenterminaldienst
- Systeminformationsdienst
- Verschlüsselungsdienst
- Signaturdienst und SAK
- Zertifikatsdienst
- Kartendienst
- LDAP-Proxy

unterstützen.

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

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

⌦ **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 Clientsystem präsentierte Credential in ANCL_CUP_LIST enthalten sein

Für eine erfolgreiche Prüfung mit zertifikatsbasierter Client-Authentifizierung MUSS:

- das seitens des Clientsystem 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.⌫

⌦ **TIP1-A_5009 Mögliche Authentifizierungsvarianten des Clientsystems**

Der Konnektor MUSS als Authentifizierungsmethode des Clientsystems ausschließlich folgende Varianten erlauben:

- TLS-Server-Authentifizierung ohne Client-Authentifizierung
- TLS-Server-Authentifizierung und TLS-Client-Authentifizierung
- TLS-Server-Authentifizierung und Basic-Client-Authentifizierung
- Keine Authentifizierung des Clients und des Servers

Alle anderen Verbindungsversuche von Clientsystemen MÜSSEN vom Konnektor abgelehnt werden.⌫

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 Client-Authentisierung erzeugen und exportieren sowie X.509-Zertifikate importieren**

Der Konnektor MUSS die Erstellung und den Export von X.509-Zertifikaten für Clientsysteme und den zugehörigen privaten Schlüsseln durch den Administrator über das Managementinterface ermöglichen. 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.

Wird ANCL_TLS_MANDATORY auf ENABLED gewechselt, MÜSSEN alle nicht per TLS gesicherten http-Verbindungen geschlossen werden, sobald die in den Verbindungen jeweils aktuell laufenden Außenschnittstelle-Operationen abgeschlossen wurden, mit Ausnahme von http-Verbindungen zum Dienstverzeichnisdienst.

Der Konnektor MUSS den Administrator geeignet und verständlich auf seine Verantwortung für die Sicherung der Kommunikation hinweisen.

Tabelle 6 TAB_KON_506 Konfigurationsparameter der Clientsystem-Authentisierung

Referenz ID	Belegung	Bedeutung und Administrator-Interaktion
ANCL_TLS_MANDATORY	Enabled / Disabled	Der Administrator MUSS die verpflichtende Verwendung eines TLS gesicherten Kanals an- oder abschalten können. 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 Leistungserbringer die Verantwortung für die Sicherstellung der vertraulichen Übertragung übernimmt. Default-Wert: Enabled
ANCL_CAUT_MANDATORY	Enabled / Disabled	Der Administrator MUSS die verpflichtende Authentifizierung der Clientsysteme an- oder abschalten können. Default-Wert: Enabled
ANCL_CAUT_MODE	CERTIFICATE / PASSWORD	Der Administrator MUSS konfigurieren können, welche Client Authentifizierungsmodus genutzt werden kann bzw. genutzt werden muss. Default-Wert: CERTIFICATE
ANCL_CCERT_LIST	Liste von X.509-Zertifikat zugeordnet zu ClientID	Whitelist an importierten oder vom Konnektor erzeugten X.509-Zertifikaten und dazugehörenden Clientsystem IDs. Der Administrator MUSS die Liste der Zertifikate und den zugehörenden Clientsystemen verwalten können, der Inhalt der Zertifikate MUSS menschlich lesbar dargestellt werden.
ANCL_CUP_LIST	Liste von Benutzer/Passwort Kombinationen, zugeordnet zu ClientID	Whitelist an UserCredentials und dazugehörenden Clientsystem IDs. Der Administrator MUSS eine Liste von Credentials und zugehörendem 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 immer (ENABLED) auch über einen ungesicherten http-Kanal erfolgen kann, unabhängig von ANCL_TLS_MANDATORY = ENABLED oder ANCL_CAUT_MANDATORY = ENABLED. Default-Wert: Enabled

⌫

3.5 Clientsystemschnittstelle

⌦ TIP1-A_5401 Parallele Nutzbarkeit Clientsystemschnittstelle

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

3.5.1 SOAP-Schnittstelle

Für die Beschreibung der SOAP-Schnittstelle zum Clientsystem wird in dieser Spezifikation WSDL Version 1.1 [WSDL1.1] eingesetzt. Die Interoperabilität zwischen verschiedenen SOAP-Implementierungen wird durch die Vorgaben des WS-I Basic Profile V1.2 [BasicProfile1.2] erreicht.

⌦ TIP1-A_4519 Web-Services konform zu [BasicProfile1.2]

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

Abweichend von R1012 in [BasicProfile1.2] 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.

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 über eine SOAP-Fault-Nachricht gemeldet (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. ⌫

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 möglich, einen Error als Warnung zu interpretieren und an den eigenen internen oder externen Aufrufer zurückzumelden. Diese dabei erzeugte Fehlerkette wird in Form einer Fehler-Trace-Struktur abgebildet, um eine Nachverfolgung von Fehlern zu ermöglichen.

Operationen an der Außenschnittstelle senden die Fehlerkette zu Informationszwecken in der SOAP-Antwort an das Clientsystem. Dazu enthält jede SOAP-Antwort das Element Status, dass gemäß dem XML-Schema [ConnectorCommon.xsd] aufgebaut ist (siehe auch Abbildung 3 PIC_KON_107 XML-Struktur des Status-Elements einer SOAP-Antwort).

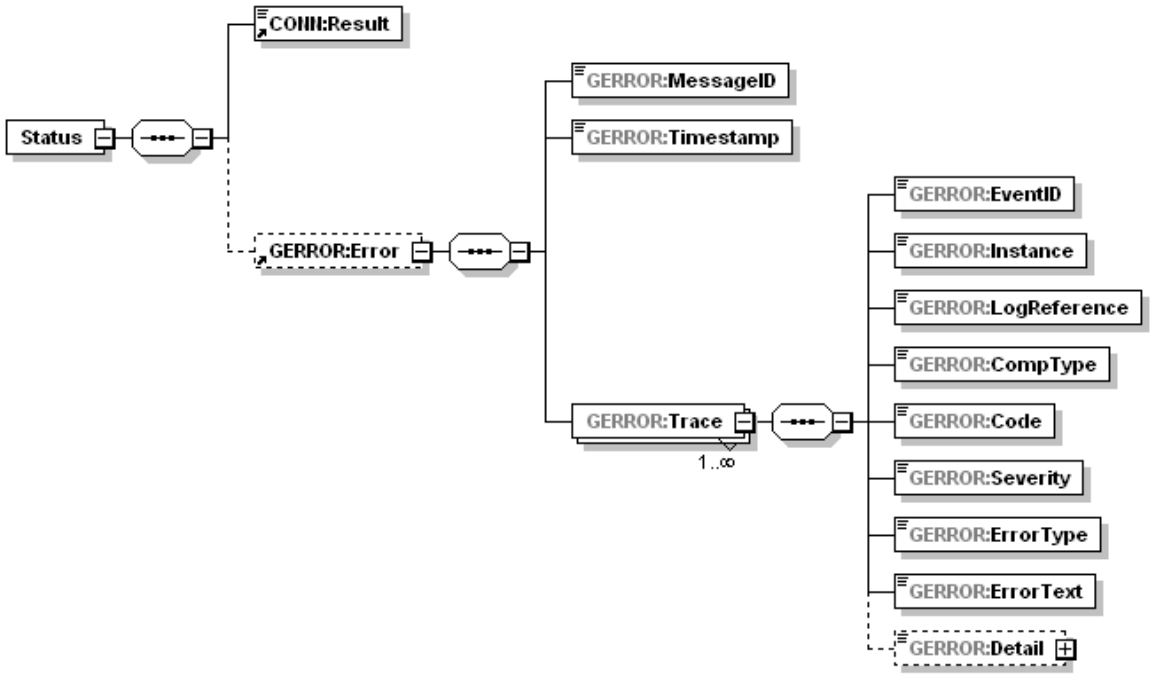


Abbildung 3 PIC_KON_107 XML-Struktur des Status-Elements einer SOAP-Antwort

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

⌦ **TIP1-A_4520 Bildung von Fehler-Trace-Strukturen**

Der Konnektor MUSS sicherstellen, dass Fehlermeldungen den Trace in der Software von der Fehlerursache zurück bis zum auslösenden Operationsaufruf vollständig nachvollziehbar machen. ⌫

⌦ **TIP1-A_4521 Protokollierung von Fehlern inkl. Trace-Struktur**

Der Konnektor MUSS Fehler protokollieren, die in TUCs, Operationen oder herstellerspezifisch definiert sind und den Schweregrad (Severity) Warning, Error oder Fatal haben. Er MUSS zum Fehlerprotokolleintrag auch die Fehler-Traces ablegen.⌫

3.6 Verwendung manuell importierter CA-Zertifikate

TI-fremde X.509-Zertifikate werden im Rahmen des Verschlüsselungsdienstes verwendet. Um den Vertrauensraum für diese Zertifikate abzubilden, erlaubt der Konnektor, X.509-CA-Zertifikate zu diesen TI-fremden X.509-Zertifikaten in eine interne Liste (CERT_IMPORTED_CA_LIST) zu importieren.

Der Konnektor kann dann im Rahmen der Hybridverschlüsselung den symmetrischen Schlüssel empfängerspezifisch mit dessem TI-fremden X.509-Zertifikat verschlüsseln. Die

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

⌦ **TIP1-A_5433 Manuell importierte X.509-CA-Zertifikate nur für hybride Verschlüsselung**

Der Konnektor DARF Zertifikate, die unter Verwendung von manuell importierten X.509-CA-Zertifikaten (CERT_IMPORTED_CA_LIST) herausgegeben wurden, NICHT für andere Zwecke als zur hybriden Verschlüsselung von Dokumenten verwenden. ⌫

⌦ **TIP1-A_5660 Hinweise im Handbuch für manuell importierte X.509-CA-Zertifikate**

Das Handbuch des Konnektors MUSS sinngemäß folgende Hinweise enthalten:

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

3.7 Testunterstützung

Gemäß Testkonzept Online-Rollout (Stufe 1) [gemKPT_Test_ORS1#TIP1-A_2839] muss ein Hersteller eines Konnektors seine Modelle in drei Ausführungen vorsehen: Eine für die Testumgebung, eine für die Referenzumgebung und eine für die Produktivumgebung.

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

⌦ **TIP1-A_4981 Steuerung der Betriebsumgebung via gSMC-K**

Der Produkttyp Konnektor MUSS sowohl in der Testumgebung (TU), der Referenzumgebung (RU) wie auch der Produktivumgebung (PU) betreibbar sein.

Die Information, ob eine Konnektorinstanz in der TU/RU oder PU betrieben wird, MUSS der Konnektor dem File MF/EF.EnvironmentSettings der gSMC-K entnehmen.

Abhängig von der ermittelten Betriebsumgebung MÜSSEN die Konfigurationswerte gemäß Tabelle TAB_KON_812 verwendet werden.

Tabelle 7: 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#Tab_Adrkonzept_Test]	Siehe TAB_KON_680. Dieser Wert MUSS für den Administrator über die Managementschnittstelle mit dem Konfigurationswert voreingestellt und änderbar sein.
	NET_TI_GESICHERTE_FD	siehe [gemSpec_Net#Tab_Adrkonzept_Test]	Siehe TAB_KON_680. Dieser Wert MUSS für den Administrator über die Managementschnittstelle mit dem Konfigurationswert voreingestellt und änderbar sein.
	NET_TI_OFFENE_FD	siehe [gemSpec_Net#Tab_Adrkonzept_Test]	Siehe TAB_KON_680. 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. Dieser Wert MUSS für den Administrator über die Managementschnittstelle einsehbar, aber nicht änderbar sein.

⌫

⌦ TIP1-A_4707 Betrieb in Test- und Referenzumgebung

Der Produkttyp Konnektor MUSS auch in der Test- und Referenzumgebung betrieben werden können. Dafür MUSS der Vertrauensanker des Konnektors für diese Umgebung ausgetauscht werden können. Dies KANN durch Lieferung eines neuen

Konnektors oder durch Austausch der gSMC-K durch den Hersteller ermöglicht werden. Der Hersteller MUSS sicherstellen, dass Konnektoren ausschließlich mit den zu ihrer Einsatzumgebung gehörenden Vertrauensankern ausgestattet werden.⌫

⌦ **TIP1-A_4982 Anzeige von TU/RU in der Managementschnittstelle**

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

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

⌦ **TIP1-A_5013 Anzeige von TU/RU im xTV**

Der Extended Trusted Viewer MUSS, wenn der Konnektor in der Testumgebung (TU) oder Referenzumgebung (RU) betrieben wird, die Umgebungsbezeichnung zu jeder Zeit erkennbar anzeigen.

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

4 Funktionsmerkmale

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

4.1 Anwendungskonnektor

4.1.1 Zugriffsberechtigungsdienst

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

Der Zugriffsberechtigungsdienst definiert über ein Informationsmodell die erlaubten Zugriffsmöglichkeiten. Um dies zu erreichen, modelliert es Mandanten und ordnet ihnen Clientsysteme sowie die vom Konnektor verwalteten externen Ressourcen (Kartenterminal mit Slots, Arbeitsplatz mit ExtendedTrustedViewer und SMC-Bs) zu. Diese durch einen Administrator persistent zu modellierenden Entitäten und Beziehungen beinhalten die erlaubten Zugriffswege vom Clientsystem über Arbeitsplatz zum Kartenterminal und dessen Slots. Sie werden im Konnektor administrativ konfiguriert.

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

Abbildung 4 stellt das Informationsmodell dar. Die persistenten Entitäten haben einen grünen Hintergrund, die transienten einen weißen.

Tabelle 8 beschreibt die Entitäten und legt ihren Identitätsschlüssel fest. Tabelle 9 beschreibt die Attribute. Tabelle 10 beschreibt die Entitätsbeziehungen und referenziert dabei die in Abbildung 4 durch Zahlen in eckigen Klammern markierten Beziehungen. Tabelle 11 definiert Constraints, die zusätzlich zu den in Abbildung 4 definierten Kardinalitäten gelten. Die Constraints werden mittels Object Constraint Language (OCL) definiert.

4.1.1.1 Funktionsmerkmalweite Aspekte

⌦ TIP1-A_4522 Zugriffsberechtigungs-Informationsmodell des Konnektors

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

Das Informationsmodell ist definiert durch das UML-Diagramm „Abb_Kon_100 Informationsmodell des Konnektors“ und die Tabelle „TAB_KON_510 Informationsmodell Constraints“. Der Konnektor darf nur Daten in sein Informationsmodell übernehmen,

die alle Eigenschaften des Informationsmodells, insbesondere die Constraints, erfüllen.⌫

Die Entitäten werden in Tabelle „TAB_KON_507 Informationsmodell Entitäten“ beschrieben, die Attribute in Tabelle „TAB_KON_508 Informationsmodell Attribute“ und die Beziehungen in Tabelle „TAB_KON_509 Informationsmodell Entitätenbeziehungen“.

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

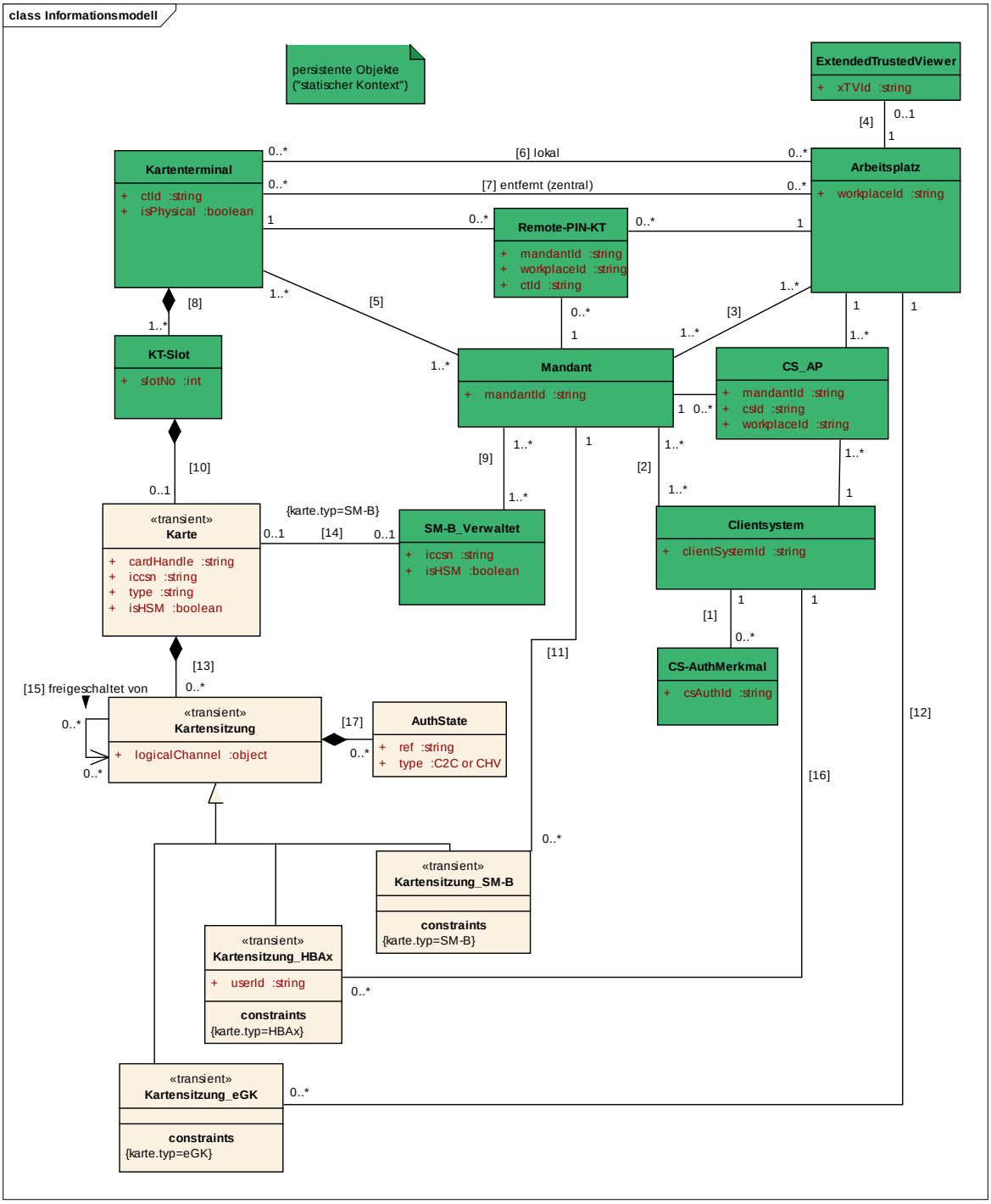


Abbildung 4: PIC_Kon_100 Informationsmodell des Konnektors

Tabelle 8 TAB_KON_507 Informationsmodell Entitäten

Entität	persistent/ transient	Identitäts -schlüssel	Beschreibung
Mandant	persistent	mandantId	Zu Mandanten und Mandantenfähigkeit siehe Kapitel Mandantenfähigkeit.

Entität	persistent/ transient	Identitäts -schlüssel	Beschreibung
Clientsystem	persistent	clientSystemId	Unter einem Clientsystem wird hier ein einzelnes oder eine Gruppe von Systemen verstanden, welche im LAN des Leistungserbringers 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
ExtendedTrustedViewer	persistent	xTVId	Aufgrund der eindeutigen Zuordnung von Extended Trusted Viewer zu Arbeitsplätzen ergibt sich der Zugriff auf bestimmte Instanzen automatisch aus dem Aufrufkontext. Eine weitergehende Berechtigungsprüfung bei der Nutzung von Extended Trusted Viewer im Rahmen der vom Clientsystem angestoßenen Abläufe ist daher nicht erforderlich. Extended Trusted Viewer werden daher im Folgenden nicht weiter betrachtet.
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 vom Clientsystem über das cardHandle adressiert. Die iccsn erlaubt eine dauerhafte Adressierung einer Karte. Für den Kartentyp „SM-B“ kann hier auch eine in einem HSM-B enthaltene

Entität	persistent/ transient	Identitäts -schlüssel	Beschreibung
			virtuelle SMC-B abgebildet werden.
Kartensitzung (CardSession)	transient	siehe konkrete Kartensitzungen	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. Zu einer Karte kann es mehrere Kartensitzungen geben, die voneinander logisch unabhängige Sicherheitsstatus einer Karte verwalten. Der Konnektor führt alle Zugriffe auf eine Karte im Kontext einer Kartensitzung zu dieser Karte aus. Das Attribut logischerKanal bezeichnet den logischen Kanal zur Karte, der im Rahmen der Kartensitzung verwendet wird[2].
Kartensitzung_eGK (CardSession_eGK)	transient	cardHandle	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.
Kartensitzung_SM-B (CardSession_SM-B)	transient	cardHandle, mandantId	Kartensitzung für eine SM-B
Kartensitzung_HBAx (CardSession_HBAx)	transient	cardHandle, clientSystemId, userId	Kartensitzung für einen HBAx. Unter dem Typ „HBAx“ sind auch die Vorläuferkarten wie „HBA-qSig” und ”ZOD_2.0” inkludiert.
SM-B_Verwaltet (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, clientSystemId, workplaceId	CS_AP legt die von einem Clientsystem pro Mandanten nutzbaren Arbeitsplätze fest. Ein Clientsystem kann dabei mehrere Arbeitsplätze bedienen.

[2] gemäß Standard [7816–4].

Entität	persistent/ transient	Identitäts -schlüssel	Beschreibung
			Ebenso können Arbeitsplätze von mehreren Clientsystemen, auch gleichzeitig, genutzt werden, z. B. bei zwei unterschiedlichen, voneinander unabhängigen Praxisprogrammen.
Remote-PIN-KT	persistent	mandantId, 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, (clientSystemId), (userId), ref	Zu einer Kartensitzung gibt es höhere AuthorizationStates, die durch (type =C2C) Freischaltung oder durch PIN-Eingabe (type=CHV) erreicht werden können.

Tabelle 9 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.

Attribut	Beschreibung
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“. 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.
xTVId	Das Identifikationsmerkmal eines ExtendedTrustedViewer.

Tabelle 10 TAB_KON_509 Informationsmodell Entitätenbeziehungen

Entitätenbeziehung	persistent/ 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. Arbeitsplätze können von mehreren Mandanten genutzt werden. Z. B. kann ein von mehreren Mandanten genutzter gemeinsamer Empfang als ein Arbeitsplatz modelliert werden.

Entitätenbeziehung	persistent/ transient	Beschreibung
Extended Trusted Viewer eines Arbeitsplatzes [4]	persistent	Den Arbeitsplätzen werden eindeutig Extended Trusted Viewer zugeordnet, wobei nicht jedem Arbeitsplatz ein Extended Trusted Viewer zugeordnet sein muss. Somit steht im Anwendungsfall durch Vorgabe des Arbeitsplatzes implizit fest, welcher Extended Trusted Viewer zu verwenden ist. An Arbeitsplätzen ohne Extended Trusted Viewer können demzufolge Anwendungsfälle, die einen solchen voraussetzen, nicht durchgeführt 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.

Entitätenbeziehung	persistent/ transient	Beschreibung
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.

Tabelle 11 TAB_KON_510 Informationsmodell Constraints

#	Beschreibung	Definition mittels OCL[3]

C1	Eine eGK muss eine oder keine Kartensitzung haben.	`context Karte`
		`inv: self.type = "eGK" implies`
		`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.	`context Kartensitzung-HBAx`
		`inv: forAll(k1, k2 : Kartensitzung-HBAx |`
		`k1.karte = k2.karte`
		`and k1.clientsystem = k2.clientsystem`
		`and k1.userId = k2.userId`
		`implies`
		`k1 = k2)`

C3	Wenn zwei SM-B-Kartensitzungen einer Karte dem gleichen Mandanten zugeordnet sind, dann müssen die beiden Kartensitzungen identisch sein.	`context Kartensitzung-SM-B`
		`inv: forAll(k1, k2 : Kartensitzung-SM-B |`
		`k1.karte = k2.karte`
		`and k1.mandant = k2.mandant implies`
		`k1 = k2)`

C4	Die Seriennummer iccsn einer Karte muss eindeutig sein.	`context Karte`
		`inv: Karte.allInstances ->`
		`isUnique(iccsn)`

C5	Die Seriennummer iccsn einer Karte muss für die vom Konnektor verwalteten SM-Bs eindeutig sein.	`context SM-B_Verwaltet`
		`inv: SM-B_Verwaltet.allInstances ->`
		`isUnique(iccsn)`

C6	Das CardHandle einer Karte muss eindeutig sein.	`context Karte`
		`inv: Karte.allInstances ->`
		`isUnique(cardHandle)`

[3] Die Constraints werden im UML ergänzenden Standard OCL definiert.

#	Beschreibung	Definition mittels OCL[3]

C7	Die Identifikationsnummer des Clientsystems muss eindeutig sein.	`context Clientsystem`
		`inv: Clientsystem.allInstances ->`
		`isUnique(clientSystemId)`

C8	Die Identifikationsnummer des Mandanten muss eindeutig sein.	`context Mandant`
		`inv: Mandant.allInstances ->`
		`isUnique(mandantId)`

C9	Die Identifikationsnummer des Arbeitsplatzes muss eindeutig sein.	`context Arbeitsplatz`
		`inv: Arbeitsplatz.allInstances ->`
		`isUnique(workplaceId)`

C10	Die Identifikationsnummer des Kartenterminals muss eindeutig sein.	`context Kartenterminal`
		`inv: Kartenterminal.allInstances ->`
		`isUnique(ctId)`

C11	Die Identifikationsnummer (slotNo) des Kartenterminal-Slots für ein gegebenes Kartenterminal muss eindeutig sein.	`context Kartenterminal`
		`inv: 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.	`context CS-AP`
		`inv: self.arbeitsplatz.mandant.includes(`
		`self.mandant)`
		`inv: 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.	`context Remote-PIN-KT`
		`inv: self.arbeitsplatz.mandant.includes(`
		`self.mandant)`
		`inv: self.kartenterminal.mandant.includes(`
		`self.mandant)`

C14	Zur Remote-PIN-Eingabe muss ein lokales Kartenterminal ausgewählt sein.	`context Remote-PIN-KT`
		`inv: self.arbeitsplatz`
		`.localKartenterminal`
		`.includes(self.kartenterminal)`
		`inv: not self.arbeitsplatz`
		`.entferntKartenterminal`
		`.includes(self.kartenterminal)`

C15	Zur Remote-PIN-Eingabe darf pro Mandanten und Arbeitsplatz nicht mehr als ein Kartenterminal ausgewählt werden.	`context Remote-PIN-KT`
		`inv: forAll(r1, r2 : Remote-PIN-KT |`
		`r1.arbeitsplatz = r2.arbeitsplatz`
		`and r1.mandant = r2.mandant implies`
		`r1 = r2)`

C16	Eine Kartensitzung-HBAx muss immer eine zugehörige userId haben.	`context Kartensitzung-HBAx` `inv: self.userId <> null`

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

⌦ **TIP1-A_4523 Sicherung der Aktualität des Informationsmodells Zugriffsberechtigungsdienst**

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

4.1.1.2 Durch Ereignisse ausgelöste Reaktionen

Keine.

4.1.1.3 Interne TUCs, nicht durch Fachmodule nutzbar

Keine.

4.1.1.4 Interne TUCs, auch durch Fachmodule nutzbar

4.1.1.4.1 TUC_KON_000 „Prüfe Zugriffsberechtigung“

Vor Ausführung jeder Operation an der Außenschnittstelle muss der Konnektor prüfen, ob die Operation ausgeführt werden darf (Autorisierung). Diese Prüfung auf Zugriffsberechtigung wird in TUC_KON_000 „Prüfe Zugriffsberechtigung“ gekapselt.

TUC_KON_000 „Prüfe Zugriffsberechtigung“ hat als Aufrufparameter den Aufrufkontext der Operation (siehe Abbildung 5), 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.

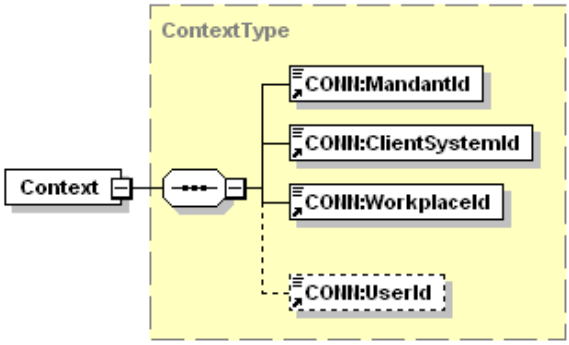


Abbildung 5: PIC_KON_101 Aufrufkontext der Operation

⌦ TIP1-A_4524 TUC_KON_000 "Prüfe Zugriffsberechtigung"

Der Konnektor MUSS den technischen Use Case TUC_KON_000 „Prüfe Zugriffsberechtigung" umsetzen.

Tabelle 12: 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.
Eingangs-anforderungen	keine
Auslöser und Vorbedingungen	Aufruf einer Operation des Konnektors durch das Clientsystem.
Eingangsdaten	• mandantId • clientSystemId • workplaceId • userId (optional) • ctId (optional) • cardHandle (optional) • needCardSession (needCardSession=true; doNotNeedCardSession=false; default: true; optional; wenn der Parameter leer ist, gilt der Default-Wert) Verwendet der aufrufende TUC eine Kartensitzung ist der Wert true, verwendet er keine Kartensitzung ist der Wert false. Die Berechtigungsprüfung geht im Default-Fall, davon aus, dass eine Kartensitzung benötigt wird, und prüft für diesen Fall die Berechtigung mit. • allWorkplaces (allWorkplaces=true; allWorkplace=false; default: false; optional; wenn der Parameter leer ist, gilt der Default-Wert) 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.
Komponenten	Konnektor
Ausgangsdaten	• keine (Autorisierung erteilt) • Fehler (Autorisierung nicht erteilt, siehe technische Fehlermeldung)

Element	Beschreibung
Standardablauf	1. Prüfe, ob die Pflichtparameter (mandantId, clientSystemId, workplaceId) vollständig gesetzt sind. 2. Prüfe, ob die gemäß [TIP1-A_4516] durchgeführte Authentifizierung über ein dem Clientsystem zugeordnetes CS-AuthMerkmal erfolgte. 3. Ermittle Zugriffsregel R zu den Aufrufparametern: 3.1. Falls der Parameter cardHandle nicht null ist, muss das Kartenobjekt des Informationsmodells Karte(cardHandle) ermittelt werden. 3.2. Zu den Parametern (ctId, cardHandle, needCardSession, allWorkplaces) muss mittels Tabelle „TAB_KON_513 Zugriffsregeln Regelzuordnung“ die Zugriffsregel R ermittelt werden. 4. Prüfe die Bedingungen der in Schritt 3 ermittelten Regel R: 4.1. Zur Regel R muss die relevante Spalte in Tabelle „TAB_KON_514 Zugriffsregeln Definition“ ermittelt werden. 4.2. Jede Zeile, die in der Spalte R ein „x“ hat, muss geprüft werden: 4.2.1. Prüfe, ob die in Spalte „Bedingung“ mittels OCL formulierte Bedingung für die Eingangsdaten erfüllt ist.
Varianten/ Alternativen	Bei einem Aufruf mit einem cardHandle zu den Kartentypen SMC-KT und UNKNOWN wird Schritt 3 in folgender Variante durchlaufen: 3. Ermittle Zugriffsregel R zu den Aufrufparametern: 3.1. ctId wird zum cardHandle bestimmt 3.2. Zu den Parametern (ctId, cardHandle: null, needCardSession: false, allWorkplaces: false) muss mittels Tabelle „TAB_KON_513 Zugriffsregeln Regelzuordnung“ die Zugriffsregel R ermittelt werden.
Fehlerfälle	Fehler im Ablauf (Standardablauf oder Varianten) führen in den folgend ausgewiesenen Schritten zu einem Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes: (→1) Es sind nicht alle Pflichtparameter gesetzt, Fehlercode: 4021 (→2.) Clientsystem aus dem Aufrufkontext nicht authentifiziert, Fehlercode: 4204 (→3.1) Karte nicht als gesteckt identifiziert, Fehlercode: 4008 (→3.2) Zu den Parametern konnte keine Regel ermittelt werden, Fehlercode: 4019 (→4.2.1) Bedingung nicht erfüllt Fehlercode: wie in Spalte „ErrorCode“ der geprüften Zeile aus Tabelle „TAB_KON_514 Zugriffsregeln Definition“
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	PIC_KON_118 Aktivitätsdiagramm zu „TUC_KON_000 Prüfe Zugriffsberechtigung“

⌫

Eine Beschreibung aller Zugriffsregeln gibt Tabelle 13.

Tabelle 13: 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.

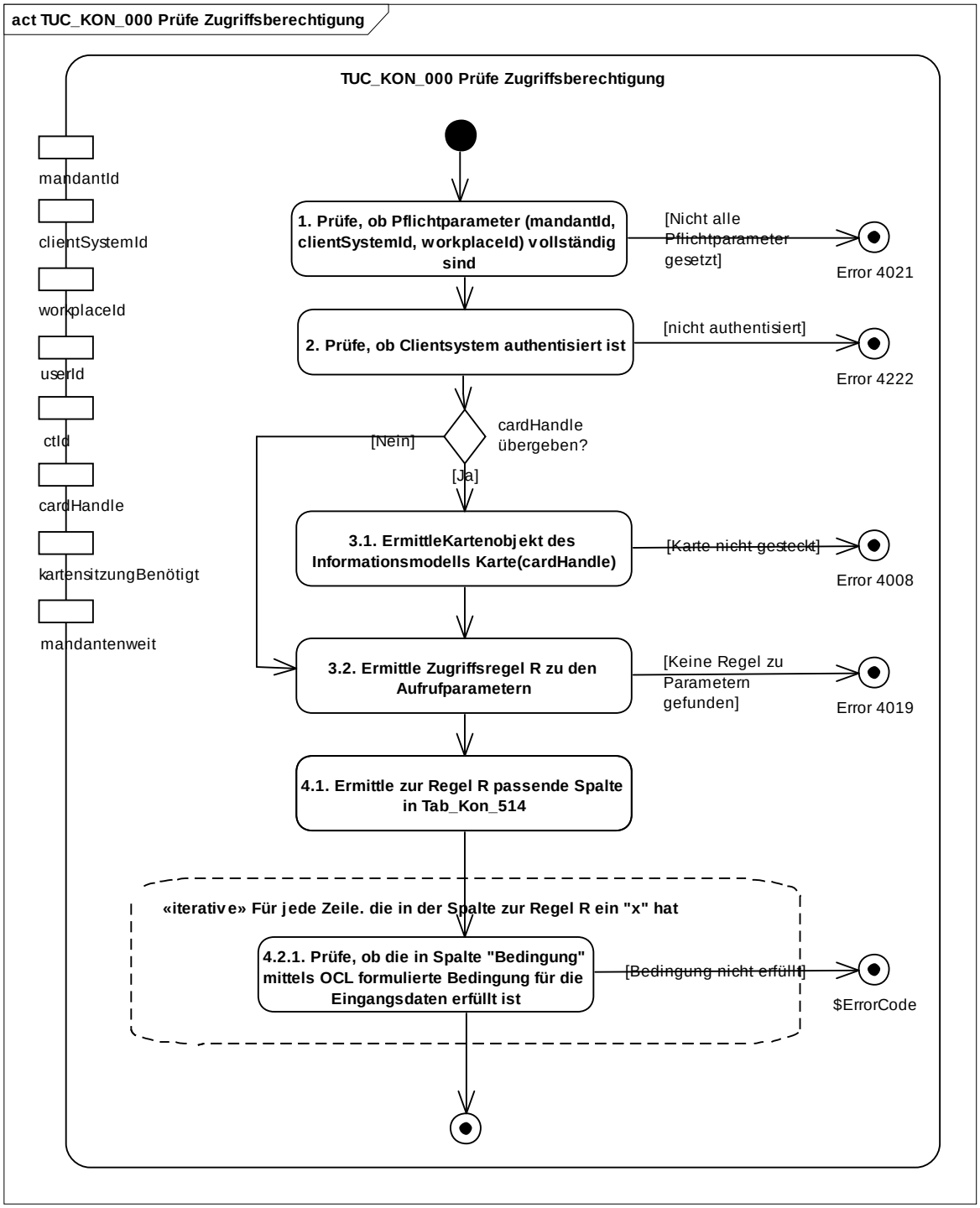


Abbildung 6: PIC_KON_118 Aktivitätsdiagramm zu „TUC_KON_000 Prüfe Zugriffsberechtigung“

Welche Zugriffsregel für einen gegebenen Satz an Aufrufparametern anzuwenden ist, wird in Tabelle 14 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 14: 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 15 definiert einzelne Bedingungen, ordnet sie den Regeln zu und definiert ErrorCodes für den Fall, dass eine Bedingung nicht erfüllt ist.

Die Bedingungen in Tabelle 15 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 benötigen Zugriffbeziehungen 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 listet Tabelle 16.

Tabelle 15: TAB_KON_514 Zugriffsregeln Definition

	Bedingung[4]	R1	R2	R3	R4	R5	R6	R7	R8	R9	Error Code
Entität[5]	`inv : userId <> null`								x	x	4003
	`let m : Mandant = Mandant(mandantId)` `inv : m <> null`	x	x	x	x	x	x	x	x	x	4004

	`let cs : Clientsystem`
	`= Clientsystem(clientSystemId)`
	`inv : cs <> null`
		x	x	x	x	x	x	x	x	x	4005

	`let ap : Arbeitsplatz`
	`= Arbeitsplatz(workplaceId)`
	`inv : ap <> null`
				x	x	x	x	x	x	x	4006

	`let kt : Kartenterminal`
	`= Kartenterminal(ctId)`
	`inv : kt <> null`
			x		x						4007

	`let k : Karte = Karte(cardHandle)` `inv : k <> null`					x	x	x	x	x	4008

Mandantbezug	`let m : Mandant = Mandant(mandantId)`
	`let cs : Clientsystem`
	`= Clientsystem(clientSystemId)`
	`inv : cs.mandant.includes(m)`
		x	x	x	x	x	x	x	x	x	4010

	`let m : Mandant = Mandant(mandantId)`
	`let ap : Arbeitsplatz`
	`= Arbeitsplatz(workplaceId)`
	`inv : ap.mandant.includes(m)`
				x	x	x	x	x	x	x	4011

	`let m : Mandant = Mandant(mandantId)`
	`let kt : Kartenterminal`
	`= Kartenterminal(ctId)`
	`inv : kt.mandant.includes(m)`
			x		x						4012

	`let m : Mandant = Mandant(mandantId)`
	`let k : Karte = Karte(cardHandle)`
	`inv : k.kT-Slot.kartenterminal.mandant`
	`.includes(m)`
						x	x	x	x	x	4012

Relation	`let k : Karte = Karte(cardHandle)` `inv : k.SM-B_Verwaltet <> null`							x			4009

	`let m : Mandant = Mandant(mandantId)`
	`let k : Karte = Karte(cardHandle)`
	`inv : k.SM-B_Verwaltet.mandant`
	`-> includes(m)`
								x			4013

	`let m : Mandant = Mandant(mandantId)`
	`let cs : Clientsystem`
	`= Clientsystem(clientSystemId)`
	`let ap : Arbeitsplatz`
	`= Arbeitsplatz(workplaceId)`
	`inv : CS_AP.allInstances`
	`-> exists(c : CS_AP |`
	`c.mandant = m`
	`and c.arbeitsplatz = ap`
	`and c.clientsystem = cs)`
				x	x	x	x	x	x	x	4014

	`let ap : Arbeitsplatz`
	`= Arbeitsplatz(workplaceId)`
	`let kt : Kartenterminal`
	`= Kartenterminal(ctId)`
	`inv :`
	`ap.lokalKartenterminal.includes(kt) or`
					x						4015

[4] Jede Bedingung ist als Constraint mittels OCL definiert, ist einzeln prüfbar und hat als Eingangsparameter mandantId, clientSystemId, workplaceId, ctId, cardHandle und userId.

[5] Zur Bezeichnung einer Objektinstanz, die im Informationsmodell vorhanden ist, wird die Notation <<Entitätsbezeichner>>(<<Komma separierte Liste der Identitätsschlüssel>> verwendet.

	Bedingung[4]	R1	R2	R3	R4	R5	R6	R7	R8	R9	Error Code
	`ap.entferntKartenterminal.includes(kt)`										

	`let ap : Arbeitsplatz = Arbeitsplatz(workplaceId)`
	`let kt : Kartenterminal = Karte(cardHandle).kT-Slot.kartenterminal`
	`inv : ap.lokalKartenterminal.includes(kt) or ap.entferntKartenterminal.includes(kt)`
								x	x	x	4015

	`let m : Mandant = Mandant(mandantId)` `let kt : Kartenterminal = Kartenterminal(ctId)` `let cs : Clientsystem = Clientsystem(clientSystemId)` `inv : CS_AP.allInstances -> 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)`		x								4020

	`let ap : Arbeitsplatz = Arbeitsplatz(workplaceId)`
	`let kt : Kartenterminal = Karte(cardHandle).kT-Slot.kartenterminal`
	`inv : ap.lokalKartenterminal.includes(kt)`
						x	x				4016

Kartensitzungen	`let ap : Arbeitsplatz = Arbeitsplatz(workplaceId)`
	`let k : Karte = Karte(cardHandle)`
	`inv : k.kartensitzung -> not exists(ks : Kartensitzung | ks.arbeitsplatz <> ap)`
							x				4017

Kartensitzungen	`let k : Karte = Karte(cardHandle)` `let cs : Clientsystem = Clientsystem(clientSystemId)` `inv : k.kartensitzung -> not exists(ks : Kartensitzung	ks.clientsystem = cs and ks.userId <> userId and ks.authState.size() > 0)`									x	4018

Tabelle 16 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

Fehlercode	ErrorType	Severity	Fehlertext
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.[6]
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.

4.1.1.5 Operationen an der Außenschnittstelle

Keine

4.1.1.6 Betriebsaspekte

⌦ TIP1-A_4525 Initialisierung Zugriffsberechtigungsdienst

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

[6] Sicherheitszustand wird bei PIN-Eingabe erhöht.

⌦ **TIP1-A_4526 Bearbeitung Informationsmodell Zugriffsberechtigungsdienst**

Für die Administration MUSS der Konnektor eine Administrationsoberfläche zur Pflege des Informationsmodells zur Verfügung stellen. Die Oberfläche muss es ermöglichen, sämtliche persistente Entitäten und Beziehungen des durch Abbildung „Abb_Kon_100 Informationsmodell des Konnektors“ und Tabelle „TAB_KON_510 Informationsmodell Constraints“ definierten Informationsmodells initial anzulegen, zu ändern und zu löschen.⌫

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

4.1.2 Dokumentvalidierungsdienst

Der Dokumentvalidierungsdienst ist ein Dienst, der nur intern genutzt wird, d. h., dass dessen definierte Verhaltensweisen nur in anderen TUCs des Konnektors nachgenutzt werden. Er bietet Schnittstellen zum Validieren von Dokumenten an. Dabei werden diejenigen spezifischen Dokumentformate unterstützt, die an den Außenschnittstellen anderer Dienste wie Signatur- und Verschlüsselungsdienst auftreten können (`Alle_DocFormate` gemäß Kapitel 3).

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

4.1.2.1 Funktionsmerkmalweite Aspekte

Keine.

4.1.2.2 Durch Ereignisse ausgelöste Reaktionen

Keine.

4.1.2.3 Interne TUCs, nicht durch Fachmodule nutzbar

Keine.

4.1.2.4 Interne TUCs, auch durch Fachmodule nutzbar

4.1.2.4.1 TUC_KON_080 “Dokument validieren”

⌦ **TIP1-A_4527 TUC_KON_080 “Dokument validieren”**

Der Konnektor MUSS den technischen Use Case TUC_KON_080 “Dokument validieren” umsetzen.

Tabelle 17: 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 `Alle_DocFormate(`außer „Binär“).

Element	Beschreibung
	Darüber hinaus prüft der TUC optional für die QES_DocFormate, ob der Dokumenteninhalt sicher im Extended Trusted Viewer angezeigt werden kann (d.h. ob es sich um ein QES_DF_SecView-Dokument handelt).
Auslöser	• Aufruf durch Fachmodul • Aufruf durch Basisdienst
Vorbedingungen	Keine
Eingangsdaten	• Zu validierendes Dokument. • Formatangabe für das Dokument (Dokumentformat) • Flag ob sichere Anzeigbarkeit geprüft werden soll (CheckDisplayability, yes/no) Optional für XML-Dokumente: • XML-Schema und ggf. weitere vom Hauptschema benutzte Schemata • XSL-Stylesheet und ggf. weitere vom Haupt-Stylesheet benutzte Stylesheets
Komponenten	Konnektor
Ausgangsdaten	• Angabe, ob das Dokument sicher anzeigbar ist (SecurelyDisplayable yes/no) optional, nur wenn Eingangsparameter CheckDisplayability = yes • Prüfprotokoll (DocumentValidation) Das Prüfprotokoll informiert im Fall SecurelyDisplayable = no über den Grund, wieso nicht sicher angezeigt werden kann. Die Ausprägung dieses Konnektor internen Parameters erfolgt herstellerspezifisch.
Nachbedingungen	Keine
Standardablauf	**Validierung der Dokumente auf Typkonformität** Der Konnektor führt je nach Format des Dokuments eine der folgenden Prüfungen durch: <u>A) XML-Dokumentvalidierung</u> Im Fall eines XML-Dokuments prüft der Konnektor: • Die XML-Wohlgeformtheit des Dokumentes • Falls ein XML-Schema übergeben wurde, validiert der Konnektor das XML-Schema selbst und prüft die Validität des XML-Dokuments in Bezug auf das XML-Schema. <u>B) PDF/A-Dokumentvalidierung</u> PDF/A-Dokumente werden auf Konformität zum PDF/A Standard geprüft [ISO 19005]. <u>C) TIFF-Dokumentvalidierung</u> Der Konnektor prüft, ob das Dokument konform zum TIFF-Standard [TIFF6] ist. <u>D) MIME-Dokumentvalidierung</u> Die Struktur von MIME-Dokumenten wird entsprechend [MIME] validiert. <u>E) Text-Dokumentvalidierung</u>

Element	Beschreibung
	Der Konnektor prüft die Konformität zum im Dokumentenformat vorgegebenen Character-Encoding. Der Konnektor prüft, dass keine Werte im Dokument enthalten sind, die gemäß übergebenem Dokumentformat nicht definiert sind. Für Binärdokumente findet keine Validierung statt. 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.
Varianten/Alternativen	**Wenn CheckDisplayability = yes:** **Prüfung der sicheren Anzeigbarkeit des Dokumenteninhalts** Falls auch die sichere Anzeigbarkeit im Extended Trusted Viewer geprüft werden soll, werden zusätzlich zu den im Standardablauf spezifizierten Prüfungen folgende weitere Prüfungen durchgeführt. Diese Prüfungen weisen nach, ob das konkrete Dokument zur Gruppe QES_DF_SecView gehört. Dabei werden neben den hier ausgewiesenen Prüfungen weitere Vorgaben bezüglich der einzelnen Dokumentformate in Anhang C getroffen. <u>A) XML-Dokumentprüfung</u> Wenn ein XSL-Stylesheet übergeben wurde, führt der Konnektor die XSL-Transformation durch. Für XML-Dokumente ist in den folgenden Fällen eine sichere Anzeige **nicht** möglich: • Das XML-Format gehört nicht zur Gruppe DF_SV_XML. • Einer der Prüfschritte aus dem Standardablauf ist fehlgeschlagen. • Die XSL-Transformation ist fehlgeschlagen. <u>B) PDF/A-Dokumentprüfung</u> Die Konformität des PDF/A-Dokument zu DF_SV_PDFA wird geprüft. Ist diese Prüfung erfolgreich und im Standardablauf sind keine Fehler aufgetreten kann das PDF/A-Dokument sicher angezeigt werden. <u>C) TIFF-Dokumentprüfung</u> Die Konformität des TIFF-Dokument zu DF_SV_TIFF wird geprüft. Ist diese Prüfung erfolgreich und im Standardablauf sind keine Fehler aufgetreten kann das TIFF-Dokument sicher angezeigt werden. <u>E) Text-Dokumentvalidierung</u> Die Konformität des Text-Dokument zu DF_SV_TEXT wird geprüft. Ist diese Prüfung erfolgreich und im Standardablauf sind keine Fehler aufgetreten kann das Text-Dokument sicher angezeigt werden. Ergibt die Dokumentprüfung, dass das Dokument sicher anzeigbar ist, so wird SecurelyDisplayable=yes zurückgegeben. Fehler bei der Dokumentenprüfung führen zu SecurelyDisplayable=no und zu einem Eintrag in das Prüfprotokoll, warum die Dokumentprüfung fehlgeschlagen ist und das Dokument nicht sicher angezeigt werden kann. Das optionale Ergebnis der Prüfung auf sichere Anzeigbarkeit

Element	Beschreibung
	(SecurelyDisplayable) und das Prüfprotokoll (DocumentValidation) werden an den Aufrufer zurückgegeben.
Fehlerfälle	**Standardablauf:** Bei der Dokumentenvalidierung protokolliert der TUC alle aufgetretenen Fehler im Rückgabewert DocumentValidation. (→A) Fehlerfälle bei XML-Dokumentvalidierung Wenn keine Schema übergeben wurde: Fehlercode 4193 Wenn eines der übergebenen Schemata selbst nicht wohlgeformt oder invalide ist, wird Fehlercode 4026 gemeldet. Wenn das XML-Dokument nicht wohlgeformt ist, wird Fehlercode 4022 gemeldet. Das XML-Dokument ist nicht valide in Bezug auf das zur Validierung benutzte Schema: Fehlercode 4023. (→B) Fehlerfälle bei PDF/A-Dokumentvalidierung Bei fehlgeschlagener Validierung: Fehlercode 4024, Dokumentformat = PDF/A (→C) Fehlerfälle bei TIFF-Dokumentvalidierung Bei fehlgeschlagener Validierung: Fehlercode 4024, Dokumentformat = TIFF (→D) Fehlerfälle bei MIME-Dokumentvalidierung Bei fehlgeschlagener Validierung: Fehlercode 4024, Dokumentformat = MIME (→E) Fehlerfälle bei Text-Dokumentvalidierung Bei fehlgeschlagener Validierung: Fehlercode 4024, Dokumentformat = Text **Variante - Prüfung der sicheren Anzeigbarkeit des Dokumenteninhalts:** Zusätzlich zu den Fehlerfällen des Standardablaufs können folgende Fehlerfälle auftreten: (→A) Fehlerfälle bei XML-Dokumentprüfung Wenn die XSL-Transformation nicht durchgeführt werden konnte: Fehlercode 4195
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 18: TAB_KON_144 Übersicht Fehlercodes für „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>)

Fehlercode	ErrorType	Severity	Fehlertext
			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
4195	`Security`	Error	Fehler bei XSL-Transformation

⌫

4.1.2.5 Operationen an der Außenschnittstelle

Keine

4.1.2.6 Betriebsaspekte

Keine

4.1.3 Dienstverzeichnisdienst

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

4.1.3.1 Funktionsmerkmalweite Aspekte

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

Der prinzipielle Ablauf sieht dabei folgendermaßen aus:

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

Der Konnektor stellt die Liste der Dienste, der Versionen und die Endpunkte der Dienste in einem XML-Dokument zusammen. Jeder über SOAP erreichbare Basisdienst des Konnektors wird in dieser Liste geführt. Ferner können Fachmodule ihre eigenen Endpunkte über TUC_KON_041 „Einbringen der Endpunktinformationen während der Bootup-Phase“ einbringen. Die so erstellte Liste der Dienste wird als Antwort an das Clientsystem übergeben.

Das Clientsystem prüft, ob die gewünschten Dienste und Versionen unterstützt werden und merkt sich die Endpunkte der Dienste für die späteren Aufrufe. Danach kann das 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.s

ds 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 Formatierung der Ausgabedatei**

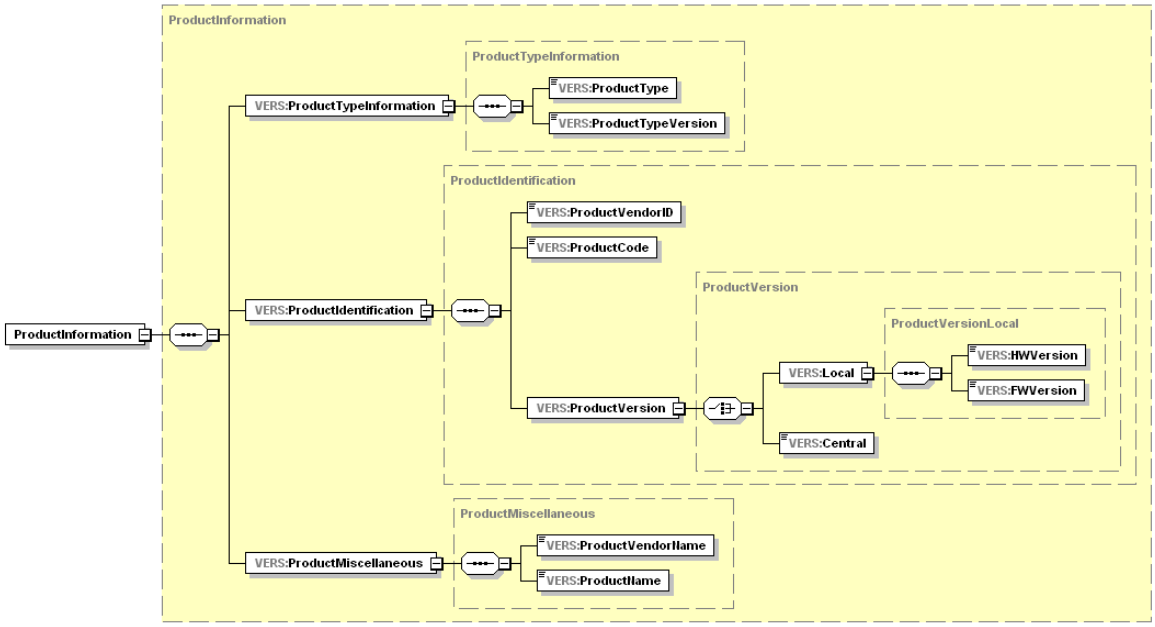
Das XML-Dokument, welches als „connector.sds“ dem Aufrufer zurückgeliefert wird, MUSS gemäß dem Schema „`conn/ServiceDirectory.xsd`“ formatiert sein. `conn/ServiceDirectory.xsd` referenziert die Schemata „`tel/version/ProductInformation.xsd`“ (siehe [gemSpec_OM]) und „`conn/ServiceInformation.xsd`“.

TAB_KON_516, TAB_KON_517 und TAB_KON_518 beschreiben die Elemente der zu verwendenden Schemastruktur.

Tabelle 19: TAB_KON_516 Basisanwendung Dienstverzeichnisdienst

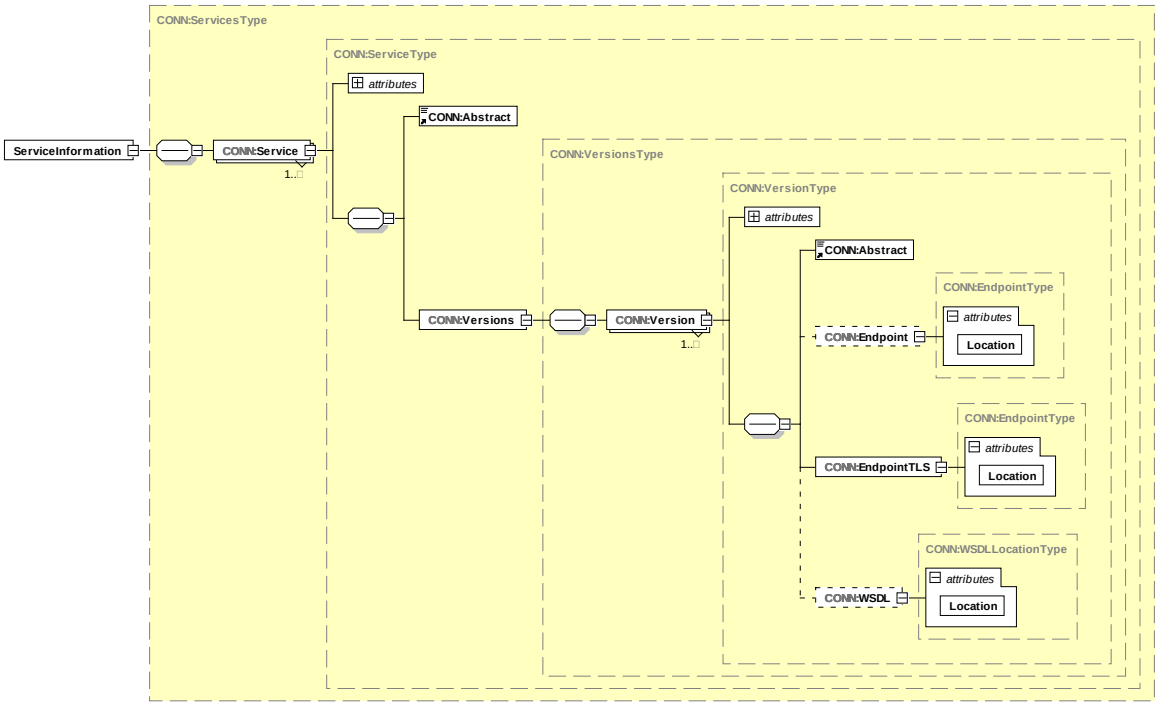
Name	ConnectorServiceDirectory
Version	Siehe Anhang D
Namensraum	Siehe Anhang D
Namensraum-Kürzel	CONN
Operationen	Lesen der vom Konnektor unterstützten Dienste
WSDL	Keine
Schema	ServiceDirectory.xsd

Tabelle 20 TAB_KON_517 Schemabeschreibung Produktinformation (ProductInformation.xsd)



Element	Bedeutung
ProductInformation/ProductTypeInformation/ ProductType	Produkttyp (Konnektor)
ProductInformation/ProductTypeInformation/ 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

Tabelle 21 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 340).
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 340.
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 Endpunktinformationen während der Bootup-Phase“

⌦ **TIP1-A_4531 TUC_KON_041 „Einbringen der Endpunktinformationen 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 22 TAB_KON_519 - TUC_KON_041 "Einbringen der Endpunktinformationen während der Bootup-Phase"

Element	Beschreibung
Name	TUC_KON_041 Einbringen der Endpunktinformationen während der Bootup-Phase
Beschreibung	Fachmodule MÜSSEN ihre Endpunktinformationen während der Bootup-Phase in den Dienstverzeichnisdienst einbringen können.
Auslöser und Vorbedingungen	Keine
Eingangsdaten	Die Eingangsdaten sind gemäß dem Ausgabeschema „Serviceinformation.xsd“ zu formatieren. Eine Beschreibung des Schemas befindet sich in TAB_KON_518.
Komponenten	Konnektor, Fachmodule
Ausgangsdaten	Keine
Standardablauf	Die übergebenen Serviceinformationen des Fachmoduls werden in die Gesamtstruktur „connector.sds“ aufgenommen.

Element	Beschreibung
	Falls beim Speichern der eingebrachten Endpunktinformationen ein Fehler auftritt wird Fehler 4027 ausgelöst.
Varianten/Alternativen	Keine
Fehlerfälle	4027: Die Endpunktinformationen konnten nicht übernommen werden.
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 23: TAB_KON_520 Übersicht Fehler TUC_KON_041 "Einbringen der Endpunktinformationen während der Bootup-Phase"

Fehlercode	ErrorType	Severity	Fehlertext
4027	Technical	Error	Die Endpunktinformationen konnten nicht übernommen werden.

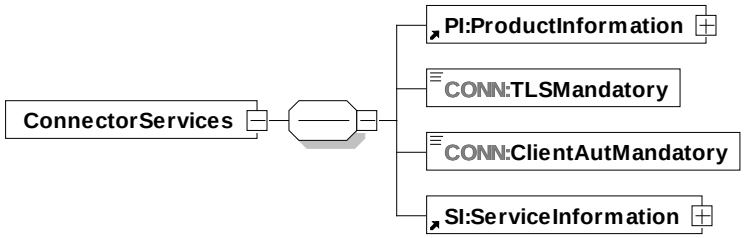
⌫

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 24: TAB_KON_521 Schnittstelle der Basisanwendung Dienstverzeichnisdienst

Dienstname	ConnectorServiceDirectory
Beschreibung	Der Aufruf liefert Angaben über den Hersteller, über das Konnektormodell und die Liste der Dienste, Konnektordienstversionen (KDV) und Endpunkte der Dienste.
Aufruf	GET /connector.sds HTTP/1.1 Host: ANLW_LAN_IP_ADDRESS oder MGM_KONN_HOSTNAME
Rückgabe	Das Antwortdokument ist in der Schemadatei `ServiceDirectory.xsd` beschrieben.
	ConnectorServices
	

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

⌫

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 Anwendungskonnektor den Clientsystemen anbietet, inklusive der Services der Fachmodule.⌫

4.1.4 Kartenterminaldienst

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

Für die TLS-Verbindungen zu den Kartenterminals muss der Konnektor die Vorgaben aus [gemSpec_Krypt#3.3.2] befolgen.

Innerhalb des Kartenterminaldienstes werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): „`CT`“

- Konfigurationsparameter: „CTM_“

Der Kartenterminaldienst verwaltet hinsichtlich der Kartenterminals mindestens die in der informativen Tabelle 25 TAB_KON_522 Parameterübersicht des Kartenterminaldienstes ausgewiesenen Parameter weitere herstellerspezifische Parameter sind möglich. Die normative Festlegung wann welche Parameter mit welchen Werten belegt werden, erfolgt in den folgenden Abschnitten und Unterkapiteln.

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

Tabelle 25 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	Identifier	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.SLOTCOUNT	Nummer	Anzahl der Slots des Kartenterminals
CT.SLOTS_USED	Liste	Liste der aktuell mit Karten belegten Slots
CT.PRODUCTINFORMATION	Inhalt ProductInformation.xsd	Die Herstellerinformationen zum Kartenterminal gemäß [gemSpec_OM]
CT.VALID_VERSION	Boolean	True, wenn die Version des Kartenterminals durch den Konnektor unterstützt wird, d.h. zu den in CTM_SUPPORTED_KT_VERSIONS passt

ReferenzID	Belegung	Zustandswerte
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 zugewiesen gepairt aktiv aktualisierend	Der Korrelationsstatus zum Konnektor: • bekannt (über Service Announcement/Service Discovery gelernte Kartenterminals), • zugewiesen (durch den Administrator aus dem Bereich der bekannten Kartenterminals oder manuell konfigurierte Kartenterminals), • gepairt (Pairing erfolgreich aber noch nicht zum Verbindungsaufbau freigegeben) • aktiv (durch Administrator zum Verbindungsaufbau freigegeben), • 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"),
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

Zum besseren Verständnis sind die Zustände, die ein Kartenterminal einnehmen kann, im nachfolgenden Zustandsdiagramm PIC_KON_071 dargestellt.

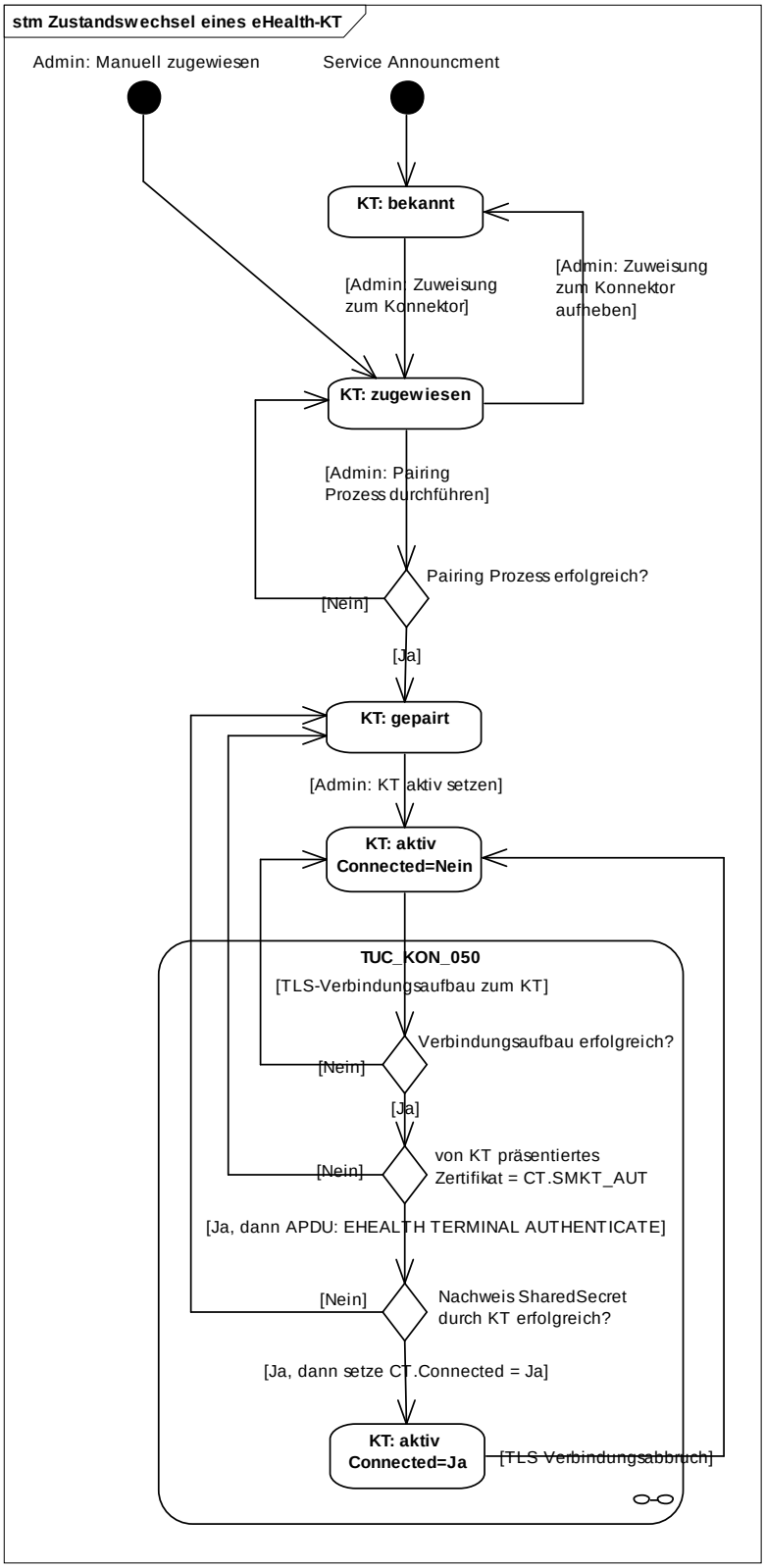


Abbildung 7 PIC_KON_071 Korrelationszustände eines eHealth-KT

4.1.4.1 Funktionsmerkmalweite Aspekte

TIP1-A_4534 Kartenterminals nach eHealth-KT-Spezifikation

Der Kartenterminaldienst MUSS Kartenterminals nach der eHealth-Kartenterminal Spezifikation [gemSpec_KT] unterstützen.

Zur Unterstützung von HSM-Bs benötigt der Konnektor virtuelle Kartenterminals (`CT.IS_PHYSICAL=Nein`), in denen virtuelle SMC-Bs „stecken“ können (siehe Kapitel 4.1.5). Diese Kartenterminals werden innerhalb des Zugriffsberechtigungsdienstes sowie des Systeminformationsdienstes wie normale Kartenterminals berücksichtigt. Weitere Details zu den logischen Kartenterminals finden sich im Kapitel Betriebsaspekte.

TIP1-A_4535 Unterstützung logischer Kartenterminals für HSMs

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

- Jedes logische KT MUSS als CT-Object mit eindeutiger CTID in CTM_CT_LIST enthalten sein
- Die CT-Attribute MÜSSEN gemäß TAB_KON_522 Parameterübersicht des Kartenterminaldienstes gesetzt werden.

TIP1-A_4536 TLS-Verbindung zu Kartenterminals halten

Der Kartenterminaldienst MUSS jede mit einem Kartenterminal etablierte Verbindung durch Nutzung des in [SICCT#6.1.4.5] definierten Keep-Alive Mechanismus halten.

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{„CT/SLOT_FREE“; Op; Info; „CtID=$CT.CTID; SlotNo=$X“; noLog; noDisp} rufen
- TUC_KON_256{„CT/DISCONNECTED“; Op; Info; „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_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 26: TAB_KON_727 Terminalanzeigen beim Anfordern und Auswerfen von Karten

Kontext	Kartentyp	Terminal-Anzeige
Karte anfordern	EGK	Bitte•***0x0B***eGK•***0x0B***in •***0x0B***SLOT *X*•***0x0B***stecken
	HBA, HBAx, HBA-qSig	Bitte•***0x0B***HBA•***0x0B***in •***0x0B***SLOT *X*•***0x0B***stecken
	SMC-B	Bitte•***0x0B***SMC-B•***0x0B***in •***0x0B***SLOT *X*•***0x0B***stecken
	sonstiger Kartentyp oder kein explizit angegebener Kartentyp	Bitte•***0x0B***Karte•***0x0B***in •***0x0B***SLOT *X*•***0x0B***stecken
Karte auswerfen	EGK	Bitte•***0x0B***eGK•***0x0B***aus •***0x0B***SLOT *X*•***0x0B***entnehmen
	HBA, HBAx, HBA-qSig	Bitte•***0x0B***HBA•***0x0B***aus •***0x0B***SLOT *X*•***0x0B***entnehmen
	SMC-B	Bitte•***0x0B***SMC-B•***0x0B***aus •***0x0B***SLOT *X*•***0x0B***entnehmen
	sonstiger Kartentyp oder kein explizit angegebener Kartentyp	Bitte•***0x0B***Karte•***0x0B***entnehmen

⌫

4.1.4.2 Durch Ereignisse ausgelöste Reaktionen

TIP1-A_4540 Reaktion auf KT-Service-Announcement

Der Konnektor MUSS Service Announcement für das Auffinden von Kartenterminals entsprechend [SICCT] und [gemSpec_KT] unterstützen. d. h. Service Announcement auf UDP Port `CTM_SERVICE_ANNOUNCEMENT_PORT` entgegennehmen.

Erhält er ein solches Dienstbeschreibungspaket, MUSS er

- TUC_KON_054 mit Mode=AutoAdded und IP-Adresse; TCP-Port; MAC-Adresse; Hostname aus dem Dienstbeschreibungspaket, aufrufen
- für das mit der MAC-Adressen in CTM_CT_LIST korrelierende CT-Obejct wenn CT.VALID_VERSION = True ist TUC_KON_050 {CT.CtID, User} aufrufen.

TIP1-A_4541 Reaktion auf KT-Slot-Ereignis – Karte eingesteckt

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

- das meldende Kartenterminal CT in CTM_CT_LIST ermitteln,
- den in der Ereignisnachricht benannten Slot (FU-Nummer) in CT.SLOTS_USED aufnehmen,
- zur weiteren internen Bearbeitung rufe TUC_KON_256{„CT/SLOT_IN_USE“; Op; Info; „CtID=$CT.CTID;SlotNo=<FU-Nummer aus Ereignisnachricht>“; noLog; noDisp} auf.

TIP1-A_4542 Reaktion auf KT-Slot-Ereignis – Karte entfernt

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

- das meldende Kartenterminal CT in CTM_CT_LIST ermitteln,
- den in der Ereignisnachricht benannten Slot (FU-Nummer) aus CT.SLOTS_USED entfernen,
- zur weiteren internen Bearbeitung rufe TUC_KON_256{„CT/SLOT_FREE“; Op; Info; „CtID=$CT.CTID;SlotNo=<FU-Nummer aus Ereignisnachricht>“; noLog; noDisp} auf.

TIP1-A_4543 KT-Statusanpassung bei Beginn eines Updatevorgangs

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

- Setze CT = CTM_CT_LIST(CTID-Parameter des Ereignisses)
- CT.CORRELATION für das Kartenterminal merken und auf „aktualisierend“ setzen
- Für jeden in CT.SLOTS_USED gelisteten Slot X zur weiteren internen Bearbeitung TUC_KON_256{„CT/SLOT_FREE“; Op; Info; „CtID=$CT.CTID;SlotNo=$CT.SLOTS_USED[X]“; noLog; noDisp} aufrufen

⌦ **TIP1-A_4544 KT-Statusanpassung bei Ende eines Updatevorgangs**

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

- Setze CT = CTM_CT_LIST(CTID-Parameter des Ereignisses)
- CT.CORRELATION auf den beim „KSR/UPDATE/START“ gemerkten Wert setzen
- Aktualisiere Gerätedaten durch Aufruf TUC_KON_055 „Befülle CT-Object“ {CTID}
- Wenn CT.VALID_VERSION = True Rufe TUC_KON_050 „Beginne Kartenterminalsitzung“ {CTID, User}
- Wenn CT.VALID_VERSION = False und CT.CORRELATION = “aktiv“ setze CT.CORRELATION=“gepairt“⌫

4.1.4.3 Interne TUCs, nicht durch Fachmodule nutzbar

4.1.4.3.1 TUC_KON_050 „Beginne Kartenterminalsitzung“

⌦ **TIP1-A_4545 TUC_KON_050 „BeginneKartenterminalsitzung“**

Der Konnektor MUSS den technischen Use Case “Beginne Kartenterminalsitzung” gemäß TUC_KON_050 umsetzen.

Tabelle 27: 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	• Neustart des Konnektors • nach dem Setzen eines Kartenterminals auf „aktiv“ • im Rahmen eines erneuten Verbindungsversuchs
Vorbedingungen	CtID ist in CTM_CT_LIST vorhanden
Eingangsdaten	CtID Benutzerrolle (gültig sind: User und Admin)
Komponenten	Konnektor, Kartenterminal
Ausgangsdaten	keine
Nachbedingungen	• TLS-Kanal und SICCT-Session mit gewünschter Benutzerrolle aufgebaut, wenn CT.CORRELATION >= "gepairt" • TLS-Kanal und SICCT-Session mit leerem Username, Password und Session ID aufgebaut, wenn CT.CORRELATION <= “zugewiesen“ • Steck-Ereignisse für alle im KT befindlichen Karten ausgelöst, wenn CT.CORRELATION>=“gepairt“

Element	Beschreibung
Standardablauf	Setze CT = CTM_CT_LIST(CtID) 1. Wenn CT.IS_PHYSICAL = Nein: prüfen, ob Benutzerrolle = User Wenn CT.CONNECTED = Ja: TUC endet erfolgreich Nein: - Verbindung zu HSM in Slot 1 aufbauen - weiter mit Schritt 9 2. Wenn CT.CONNECTED = Ja prüfen, ob CT.ACTIVEROLE = Benutzerrolle Ja: TUC endet erfolgreich Nein: weiter ab Schritt 6 (halten der TLS-Verbindung und nur Wechsel der Session) 3. Aufbau einer TLS-Verbindung mit dem Kartenterminal unter Verwendung von ID.SAK.AUT. Dabei Prüfung des KT-Zertifikats mittels TUC_KON_037{C.SMKT.AUT; not_required; ; true ; oid_smkt_aut; digitalSignature&keyEncipherment; id-kp-serverAuth; ; NONE 4. Wenn CT.CORRELATION <="zugewiesen": a. Öffne eine Cardterminal Session mit dem Kartenterminalkommando SICCT INIT CT SESSION (siehe [SICCT#5.10]) mit leerem Username, Password und Session ID b. Nur Verbindung in niedriger Korrelation, daher setze CT.CONNECTED = Nein, um fachliche Nutzung des KT zu verhindern c. beende TUC erfolgreich 5. Prüfe, ob das Zertifikat aus der TLS-Verbindung mit den zum Kartenterminal gespeicherten Referenzdaten (CT.SMKT_AUT) übereinstimmt. 6. Parallelisierung a. Generierung eines zufälligen Werts (Challenge) mit mindestens 16 Byte Länge gemäß [gemSpec_Krypt#2.2] (siehe [gemSpec_KT#DO_KT_0004]), b. Öffnen einer Cardterminal Session mit dem Kartenterminalkommando SICCT INIT CT SESSION (siehe [SICCT#5.10]) mit: - CtID als Adressat - Wenn Benutzerrolle = User dann mit leerem Username, Password und Session ID

Element	Beschreibung
	Wenn Benutzerrolle = Admin dann mit leerer Session ID aber mit CT.ADMIN_USERNAME und CT.ADMIN_PASSWORD 7. Senden der Challenge mittels Kartenterminalkommando EHEALTH TERMINAL AUTHENTICATE (siehe [gemSpec_KT#3.7.2]) in der Ausprägung VALIDATE mit: - Kartenterminal als Empfänger - und mit der in Schritt a generierten Challenge im Shared Secret Challenge DO 8. Prüfe Antwort des Kartenterminals, ob korrekte Signatur der Challenge gemäß [gemSpec_KT#SEQ_KT_0002] Schritt 4-5, ausgeführt mit `CT.SHARED_SECRET` 9. Setze: a. CT.ACTIVEROLE auf Benutzerrolle b. CT.CONNECTED = Ja 10. Wenn TLS-Verbindung neu aufgebaut werden musste, rufe TUC_KON_256{„CT/CONNECTED“; Op; Info; „CtID=$CT.CTID, Hostname=$CT.HOSTNAME“} 11. Ermittle alle im KT gesteckten Karten und befülle entsprechend CT.SLOTS_USED Für jeden in CT.SLOTS_USED gelisteten Slot X zur weiteren internen Bearbeitung TUC_KON_256{„CT/SLOT_IN_USE“; Op; Info; „CtID=$CT.CTID;SlotNo=$CT.SLOTS_USED[X]“; noLog; noDisp} rufen
Varianten/Alternativen	Keine.
Fehlerfälle	Fehler in den folgenden Schritten des Ablaufs führen zu: Aufruf von TUC_KON_256 mit folgenden Parametern {"CT/TLS_ESTABLISHMENT_FAILURE"; $ErrorType; $Severity; „CTID=$CT.ID; Name=$CT.HOSTNAME; Error=$Fehlercode; Bedeutung=$Fehlertext“} Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes (→1): Admin-Rolle für logische KTs nicht möglich (hätte bei korrekter Implementierung nicht stattfinden dürfen), Fehlercode: 4032 (→1): Verbindungsaufbau zu HSM fehlgeschlagen, Fehlercode: 4032 (→3): Fehler im TLS-Verbindungsaufbau bzw. Zertifikatsprüfung, Fehlercode: 4028 und setze CT.CONNECTED auf „Nein“ (→5): Präsentiertes Zertifikat nicht das aus dem Pairing, Fehlercode: 4029 und setze CT.CORRELATION auf „gepairt“ und setze CT.CONNECTED auf „Nein“ und terminiere TLS-Verbindung (→6b): Hinterlegte KT-Admin-Credentials fehlerhaft, Fehlercode: 4030 und in die User-Session zurückzuwechseln (damit das KT für den normalen Fachbetrieb weiterhin zur Verfügung steht) (→8): Prüfung auf Nachweis SharedSecret fehlgeschlagen, Fehlercode

Element	Beschreibung
	4029 und setze CT.CORRELATION auf „gepairt“ und setze CT.CONNECTED auf „Nein“ und terminiere TLS-Verbindung
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	Abbildung 8: PIC_KON_110 Aktivitätsdiagramm zu BeginneKartenterminalsitzung

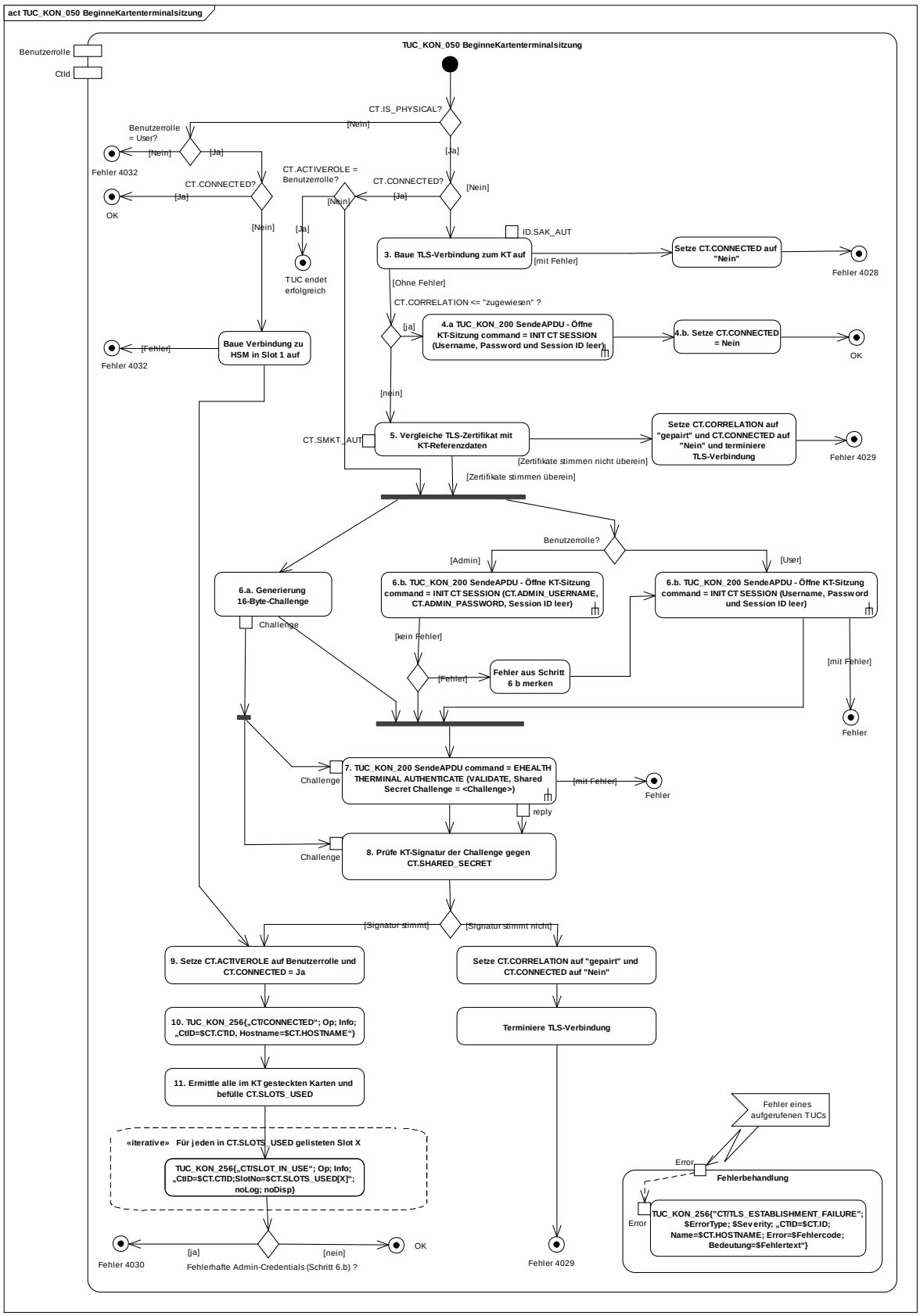


Abbildung 8: PIC_KON_110 Aktivitätsdiagramm zu BeginneKartenterminalsitzung

Tabelle 28: TAB_KON_523 Übersicht Fehlercodes für „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

⌫

4.1.4.3.2 TUC_KON_054 „Kartenterminal hinzufügen“

⌦ **TIP1-A_4546 TUC_KON_054 „Kartenterminal hinzufügen“**

Der Konnektor MUSS den technischen Use Case TUC_KON_054 „Kartenterminal hinzufügen“ umsetzen.

Tabelle 29: 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	• Ein empfangenes KT Service Announcement ohne vorheriges Service Discovery • Manuelles Hinzufügen eines KT-Eintrags
Vorbedingungen	• Entweder ist das KT noch nicht in CTM_CT_LIST enthalten • oder das KT ist unter anderer IP / anderem Hostnamen schon gelistet
Eingangsdaten	• Mode (AutoAdded, ManuallyAdded, ManuallyModified) • IP-Adresse (IPv4) • TCP-Port (optional) • MAC-Adresse (optional) • Hostname (optional)
Komponenten	Konnektor, Kartenterminal
Ausgangsdaten	• Keine
Nachbedingungen	• Das Kartenterminal ist mit allen Gerätekenndaten in CTM_CT_LIST vorhanden
Standardablauf	1. Sofern optionale Parameter nicht übergeben wurden oder Mode<>AutoAdded, ermittle Port, MAC und Hostname via Service Discovery als UDP/IP-Unicast an IP-Adresse und Port

Element	Beschreibung
	`CTM_SERVICE_DISCOVERY_PORT` 2. Finde CT in CTM_CT_LIST über MAC-Adresse 3. Wenn MAC-Adresse nicht in CTM_CT_LIST: a) Erzeuge neuen CT-Object-Eintrag in CTM_CT_LIST und ▪ Generiere eindeutige CT.CTID ▪ setzte CT.MAC_ADRESS auf MAC-Adresse ▪ Setze CT.CORRELATION =„bekannt“ b) Wenn Mode= ManuallyAdded setze CT.CORRELATION =„zugewiesen“ 4. Befülle: CT.IP_ADRESS, CT.Hostname, CT.TCP_PORT 5. Wenn CT.CORRELATION>=“zugewiesen“ rufe TUC_KON_055 „Befülle CT-Object“
Varianten/Alternativen	Keine
Fehlerfälle	Fehler im Ablauf (Standardablauf oder Varianten) führen in den folgend ausgewiesenen Schritten zu einem Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes: (→1) Keine Antwort innerhalb `CTM_SERVICE_DISCOVERY_TIMEOUT`, Fehlercode: 4033 (→1) Ermittelte MAC-Adresse weicht von übergebener MAC-Adresse ab, Fehlercode: 4035 (→1) Ermittelter Hostname-Adresse weicht von übergebenem Hostname ab, Fehlercode: 4036 (→2) Wenn Mode=ManuallyModified und nicht gefunden, Fehlercode: 4037 (→4) Nur wenn Mode= ManuallyAdded: Hostname bereits in CTM_CT_LIST vorhanden (Eindeutigkeit verletzt), Fehlercode: 4034 Zusätzlich im Abbruchfall: • Aufruf von TUC_KON_256 {"CT/ERROR"; $ErrorType; $Severity; „IP=$IP-Adresse; Name=$HOSTNAME; Error=$Fehlercode; Bedeutung=$Fehlertext“} • Keine Veränderung an CTM_CT_LIST
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	keine

Tabelle 30: TAB_KON_525 Übersicht Fehlercodes für „Kartenterminal hinzufügen“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			

Fehlercode	ErrorType	Severity	Fehlertext
4033	Technical	Error	Kartenterminal antwortet nicht, Zufügen fehlgeschlagen
4034	Technical	Error	Kartenterminal mit gleichem Hostname bereits in der Liste der Kartenterminals vorhanden. Bitte Hostname des Kartenterminals ändern.
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

⌫

4.1.4.3.3 TUC_KON_053 „Paire Kartenterminal“

⌦ **TIP1-A_4548 TUC_KON_053 „Paire Kartenterminal“**

Der Konnektor MUSS den technischen Use Case “Paire Kartenterminal” gemäß TUC_KON_053 umsetzen.⌫

Tabelle 31: 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	• KT ist in CTM_CT_LIST vorhanden • CT.CORRELATION = „zugewiesen“ • CT.IS_PHYSICAL = Ja
Eingangsdaten	• CtID
Komponenten	Konnektor, Kartenterminal
Ausgangsdaten	• Keine
Nachbedingungen	• CT.CORRELATION = „aktiv“, wenn Pairing erfolgreich • CT.CORRELATION = „zugewiesen“, wenn Pairing nicht erfolgreich • CT.CONNECTED = „Ja“, wenn Pairing erfolgreich
Standardablauf	Setze CT = CTM_CT_LIST(CtID) 1. Prüfe `CT.VALID_VERSION=True` 2. Aufbau einer TLS-Verbindung mit dem Kartenterminal unter Verwendung von ID.SAK.AUT. Dabei:

Element	Beschreibung
	a. Speichern des präsentierten KT-Zertifikats in CT.SMKT_AUT b. Prüfung des KT-Zertifikats mittels TUC_KON_037{C.SMKT.AUT; not_required; ; true ; oid _smkt_aut; digitalSignature&keyEncipherment; id-kp-serverAuth; ; NONE 3. Der Konnektor entnimmt den Fingerprint dem KT-Zertifikat und stellt dies dem Administrator im Dialog der Managementschnittstelle dar. Der Konnektor fordert den Administrator auf, den Fingerprint zu akzeptieren oder zurückzuweisen. 4. Wenn der Administrator den Fingerprint bestätigt, 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` b. und eröffnet der Konnektor mit dem Kartenterminalkommando SICCT INIT CT SESSION (siehe [SICCT#5.10]) mit • CtID als Adressat • und mit leerem Username, Password und Session ID eine Cardterminal Session. 5. Der Konnektor sendet mittels Kartenterminalkommando EHEALTH TERMINAL AUTHENTICATE (siehe [gemSpec_KT#3.7.2]) in der Ausprägung CREATE mit • CtID als Empfänger • und mit dem in Schritt a generierten Schlüssel im Shared Secret DO und der Display Message `„KT:$CT.MAC_ADRESS MIT KON:$MGM_KONN_HOSTNAME PAIREN OK?“`, wobei die MAC-Adresse mit Trenner im folgenden Format dargestellt werden: „AABBCC:DDEEFF“ werden muss das Shared Secret an das Kartenterminal. 6. Der Konnektor prüft ob in der Antwort des Kartenterminals eine korrekte Signatur des Shared Secrets gemäß [gemSpec_KT#SEQ_KT_0001] Schritt 7, ausgeführt mit dem Schlüssel zum Zertifikat CT.SMKT_AUT vorliegt. 7. `CT.CORRELATION` wird auf „gepairt“ gesetzt 8. TLS-Verbindung, die zum Pairen diente, beenden und zuvor mit CtID als Adressat das Kartenterminalkommando SICCT CLOSE CT SESSION senden 9. Automatischer Zustandsübergang `CT.CORRELATION` = „gepairt“ nach „aktiv“ (implizite Aktion des Administrators durch Aufruf von TUC_KON_053). 10. „Arbeits“-TLS-Verbindung neu aufbauen durch Aufruf

Element	Beschreibung
	TUC_KON_050 {CtID, User}
Varianten/Alternativen	(→4): weist der Administrator den Fingerprint in Schritt 3 ab, wird TUC_KON_053 nach Ausführung folgender Aktivitäten beendet: 4.1.a) Löschen von CT.SMKT_AUT 4.1.b) Aufruf von TUC_KON_200 mit CtID und dem Kartenterminalkommando SICCT CLOSE CT SESSION 4.1.c) Abbau der TLS-Verbindung, Setzen von CT.CONNECTED auf „Nein“
Fehlerfälle	Fehler in den folgenden Schritten des Ablaufs führen zu: a) Aufruf von TUC_KON_256 mit folgenden Parametern {"CT/ERROR"; $ErrorType; $Severity; „CtID=$CtID; Name=$CT.HOSTNAME; Error=$Fehlercode; Bedeutung=$Fehlertext“; noDisp} b) Löschen von CT.SMKT_AUT, CT.SHARED_SECRET c) Direkte Anzeige der Fehlermeldung für den Administrator d) Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes (→1) Version des KT wird nicht unterstützt, Fehlercode: 4042 (→2) Fehler im TLS Verbindungsaufbau bzw. Zertifikatsprüfung, Fehlercode: 4040 (→4b) Fehler in SICCT INIT CT SESSION, Fehlercode: 4041 (→5) Fehler in EHEALTH TERMINAL AUTHENTICATE, Fehlercode: 4041 (→6) Signaturprüfung fehlgeschlagen, Fehlercode: 4041
Zugehörige Diagramme	Siehe PIC_KON_057

Tabelle 32: TAB_KON_113 Übersicht Fehler TUC_KON_053 „PaireKartenterminal“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4040	Security	Error	Fehler beim Versuch eines Verbindungsaufbau zu KT
4041	Technical	Error	Fehler im Pairing, SICCT-Fehler: <SICCT-Fehler>
4042	Technical	Error	Die Version des Kartenterminals wird nicht unterstützt

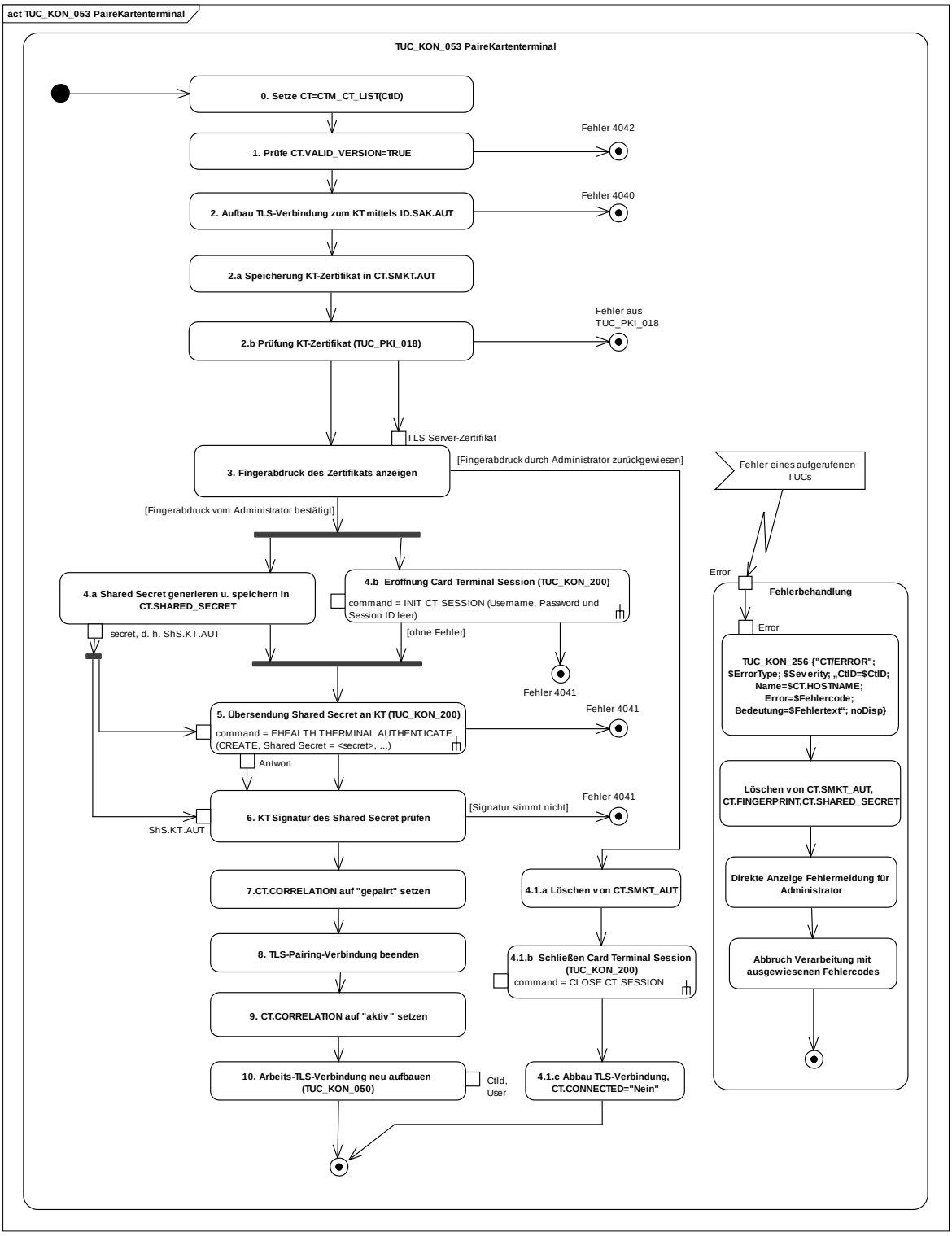


Abbildung 9: PIC_KON_057 Aktivitätsdiagramm zu PaireKartenterminal

4.1.4.3.4 TUC_KON_055 „Befülle CT-Object“

⌦ TIP1-A_4985 TUC_KON_055 „Befülle CT-Object“

Der Konnektor MUSS den technischen Use Case TUC_KON_055 „Befülle CT-Object“ umsetzen.

Tabelle 33: 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	• TUC_KON_054 • Ereignis „KSR/UPDATE/END“ mit „Target=KT“ • Verändern von CT.CORRELATION auf „zugewiesen“
Vorbedingungen	• CtID ist in CTM_CT_LIST vorhanden
Eingangsdaten	• CtID
Komponenten	Konnektor, Kartenterminal
Ausgangsdaten	• Supported (Boolean, True, wenn die Version des KT unterstütz wird)
Nachbedingungen	• Die Gerätekenndaten des Kartenterminals in CTM_CT_LIST sind aktualisiert
Standardablauf	Setze CT = CTM_CT_LIST(CtID) 1. Wenn CT.CONNECTED=Nein: Rufe TUC_KON_050 {CtID, User} 2. Folgende CT.Werte via SICCT GET STATUS ermitteln und befüllen: o CT.SLOTCOUNT o CT.PRODUCTINFORMATION 3. Finde Eintrag in `CTM_SUPPORTED_KT_VERSIONS` anhand der ersten beiden Stellen (Haupt- und Nebenversionsnummer) aus CT.PRODUCTINFORMATION/ProductTypeVersion:: Eintrag gefunden: Die dritte Stelle der KT-Version ist im Vergleich zur dritten Stelle im gefundenen Eintrag: >=: Setze Result = True <: Setze Result = False Kein Eintrag gefunden: Setze Result = False 4. Setze CT.VALID_VERSION auf Result 5. Wenn Verbindung in (1) aufgebaut wurde trenne Verbindung 6. Liefere Result zurück
Varianten/Alternativen	Keine
Fehlerfälle	-> 2) Kartenterminal antwortet mit einer spezifischen Fehlermeldung, Fehlercode <gemäß [SICCT]>

Element	Beschreibung
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	keine

⌫

4.1.4.4 Interne TUCs, auch durch Fachmodule nutzbar

4.1.4.4.1 TUC_KON_051 „Mit Anwender über Kartenterminal interagieren“

⌦ **TIP1-A_4547 TUC_KON_051 „Mit Anwender über Kartenterminal interagieren“**

Der Konnektor MUSS den technischen Use Case “Mit Anwender über Kartenterminal interagieren” gemäß TUC_KON_051 umsetzen.

Tabelle 34: TAB_KON_112 - TUC_KON_051 „Mit Anwender über Kartenterminal interagieren“

Element	Beschreibung
Name	TUC_KON_051 „Mit Anwender über Kartenterminal interagieren“
Beschreibung	Der TUC ermöglicht es, Meldungen an das Display eines Kartenterminals zu senden oder Eingaben vom Benutzer über das PIN-Pad eines Kartenterminals abzufragen (unter Anzeige einer Meldung).
Auslöser	Fachmodul im Konnektor oder anderer technischer Use Case ruft diesen Use Case auf.
Vorbedingungen	• KT ist in CTM_CT_LIST vorhanden • CT.CORRELATION = „aktiv“ • CT.CONNECTED = Ja • CT.IS_PHYSICAL = Ja
Eingangsdaten	• CtID • Data (Text zur Darstellung am KT, Länge durch KT begrenzt); optional bei Mode = OutputErase, sonst mandatory • Mode (Input, OutputWait, OutputConfirm, OutputKeep, OutputErase) • InputLength (nur bei Mode=Input, 00 für „beliebig“ lang) • WaitTimer (in Sekunden, nur bei Mode=OutputWait)
Komponenten	Konnektor, Kartenterminal
Ausgangsdaten	• Bei Input und OutputConfirm: Nutzertastendruck OK / ABBRUCH • Bei Input: Zifferneingabe des Benutzers
Nachbedingungen	Wenn Mode=OutputKeep bleibt Data auf dem Display des KT stehen
Standardablauf	Setze CT = CTM_CT_LIST(CtID) 1. Mode= a. <u>Input:</u> Der Konnektor MUSS via SICCT INPUT am CT zur

Element	Beschreibung
	Eingabe auffordern. Dabei MUSS die KT-Ansteuerung so erfolgen, dass: • Data zur Anzeige gebracht wird • maximal InputLength Ziffern akzeptiert werden. Bei 00 werden 1-n Zeichen akzeptiert (n=Maximalwert, definiert durch KT) • die eingegebenen Zeichen am Display angezeigt werden • die Eingabe explizit mit OK oder Abbruch beendet werden muss b. <u>OutputWait:</u> Der Konnektor MUSS via SICCT OUTPUT am CT Data zur Anzeige bringen. Der TUC wartet WaitTimer Sekunden, dann MUSS der Konnektor die Anzeige des KT leeren. c. <u>OutputConfirm:</u> Der Konnektor MUSS via SICCT <u>INPUT</u> am CT Data zur Anzeige bringen und auf eine Bestätigung durch den Nutzer warten (zulässig: OK, Abbruch) d. <u>OutputKeep:</u> Der Konnektor MUSS via SICCT OUTPUT am CT Data zur Anzeige bringen. Die Anzeige bleibt erhalten, bis das KT neue Informationen am Display darstellen muss / soll. e. <u>OutputErase:</u> Der Konnektor MUSS via SICCT OUTPUT am CT Data die Anzeige leeren.
Varianten/Alternativen	• 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.
Fehlerfälle	Fehler in den folgenden Schritten des Ablaufs führen zu Aufruf von TUC_KON_256 mit folgenden Parametern {"CT/ERROR"; $ErrorType; $Severity; „CtID=$CtID; Name=$CT.HOSTNAME; Error=$Fehlercode; Bedeutung=$Fehlertext“} (→2) Display und PinPad des Kartenterminals sind aktuell belegt (PIN, Eingabe, andere Ausgabe etc.), Fehlercode: 4039 (→2) Kartenterminal antwortet mit einer spezifischen Fehlermeldung, Fehlercode <gemäß [SICCT]>
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 35: TAB_KON_114 Übersicht Fehler 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:			
4038	Security	Error	Zur Anzeige vorgesehene Meldung enthält eine nicht erlaubte Zeichenfolge
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.

Tabelle 36: 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	• CtID • SlotID • CardType (optional) • DisplayMsg (optional, Text zur Darstellung am KT, Länge durch KT begrenzt) • TimeOut (in Sekunden)
Komponenten	Konnektor, Kartenterminal
Ausgangsdaten	Informationsobjekt der Karte
Nachbedingungen	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].
Standardablauf	Setze CT = CTM_CT_LIST(CtID) 1. Falls DisplayMsg nicht explizit angegeben muss sie gemäß Anforderung [TIP1-A_5408] erstellt werden. 2. Der Konnektor MUSS das Kommando `SICCT REQUEST ICC` an das Kartenterminal CT senden. Die verfügbaren Optionen (`Optical Signal, Acoustic Signal`) können herstellerspezifisch bzw. über die Konfigurationsschnittstelle

Element	Beschreibung
	des Konnektors eingestellt werden. Die Message wird als Eingabeaufforderung mitgegeben. 3. 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. 4. Es wird ein Informationsobjekt der Karte, die sich in dem angegebenen Slot befindet zurückgegeben.
Varianten/Alternativen	Die Ausgabe einer Display-Nachricht entfällt, wenn der adressierte Slot bereits eine gesteckte Karte beinhaltet.
Fehlerfälle	Fehler in den folgenden Schritten des Ablaufs führen zu Aufruf von TUC_KON_256 mit folgenden Parametern {"CT/ERROR"; $ErrorType; $Severity; „CtID=$CtID; Name=$CT.HOSTNAME; Error=$Fehlercode; Bedeutung=$Fehlertext“} (→2) Display des Kartenterminals ist aktuell belegt, Fehlercode: 4039 (→2) Fehler beim Zugriff auf das Kartenterminal, Fehlercode: 4044 (→2) Ungültige Kartenterminal-ID: Fehlercode: 4096 (→2) Ungültige Kartenslot-ID: Fehlercode: 4097 (→2) Kartenterminal nicht aktiv, Fehlercode: 4221 (→2) Kartenterminal ist nicht verbunden, Fehlercode: 4222 (→2) Kartenterminal antwortet mit einer spezifischen Fehlermeldung, Fehlercode <gemäß [SICCT]> (→3) Falscher Kartentyp, Fehlercode: 4051
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 37: TAB_KON_724 Übersicht Fehler 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
4096	Technical	Error	Ungültige Kartenterminal-ID
4097	Technical	Error	Ungültige Kartenslot-ID
4221	Technical	Error	Kartenterminal nicht aktiv
4222	Technical	Error	Kartenterminal ist nicht verbunden

⌫

4.1.4.4.3 TUC_KON_057 „Karte auswerfen“

⌦ **TIP1-A_5410 TUC_KON_057 „Karte auswerfen“**

Der Konnektor MUSS den technischen Use Case “Karte auswerfen” gemäß TUC_KON_057 umsetzen.

Tabelle 38: 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	• CtID • SlotID • DisplayMsg (optional, Text zur Darstellung am KT, Länge durch KT begrenzt) • TimeOut (in Sekunden)
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	Setze CT = CTM_CT_LIST(CtID) 1. Der Konnektor prüft, dass entweder die Karte nicht reserviert ist oder der Aufrufer im Besitz des Karten-Locks ist. 2. Falls DisplayMsg nicht explizit angegeben muss sie gemäß Anforderung [TIP1-A_5408] erstellt werden. 3. Der Konnektor MUSS das Kommando `SICCT EJECT ICC` 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.
Varianten/Alternativen	Auch im Falle, dass nach der internen Buchführung des Konnektors in einem angegebenen Slot des Kartenterminals keine Karte steckt, MUSS der Konnektor das SICCT-Kommando SICCT EJECT ICC 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 mit folgenden Parametern {"CT/ERROR"; $ErrorType;

Element	Beschreibung
	$Severity; „CtID=$CtID; Name=$CT.HOSTNAME; Error=$Fehlercode; Bedeutung=$Fehlertext“} (→1) Die Karte ist fremdreserviert, Fehlercode 4093 (→3) Display des Kartenterminals ist aktuell belegt, Fehlercode: 4039 (→3) Fehler beim Zugriff auf das Kartenterminal, Fehlercode: 4044 (→3) Karte deaktiviert, aber nicht entnommen, Fehlercode: 4203 (→3) Ungültige Kartenterminal-ID: Fehlercode: 4096 (→3) Ungültige Kartenslot-ID: Fehlercode: 4097 (→3) Kartenterminal nicht aktiv, Fehlercode: 4221 (→3) Kartenterminal ist nicht verbunden, Fehlercode: 4222 (→3) Kartenterminal antwortet mit einer spezifischen Fehlermeldung, Fehlercode <gemäß [SICCT]>
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 39: TAB_KON_796 Übersicht Fehler 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
4096	Technical	Error	Ungültige Kartenterminal-ID
4097	Technical	Error	Ungültige Kartenslot-ID
4221	Technical	Error	Kartenterminal nicht aktiv
4222	Technical	Error	Kartenterminal ist nicht verbunden

⌫

4.1.4.5 Operationen an der Außenschnittstelle

⌦ **TIP1-A_5411 Basisdienst Kartenterminaldienst**

Der Konnektor MUSS Clientsystemen den Basisdienst Kartenterminaldienst anbieten.

Tabelle 40: TAB_KON_722 Basisdienst Kartenterminaldienst

Name	`CardTerminalService`	
Version (KDV)	wird im Produktsteckbrief des Konnektors definiert	
Namensraum	Siehe Anhang D	
Namensraum-Kürzel	`CT für Schema und CTW für WSDL`	
Operationen	**Name**	**Kurzbeschreibung**
	`RequestCard`	Karte anfordern
	`EjectCard`	Karte auswerfen
WSDL	`CardTerminalService.wsdl`	
Schema	`CardTerminalService.xsd`	

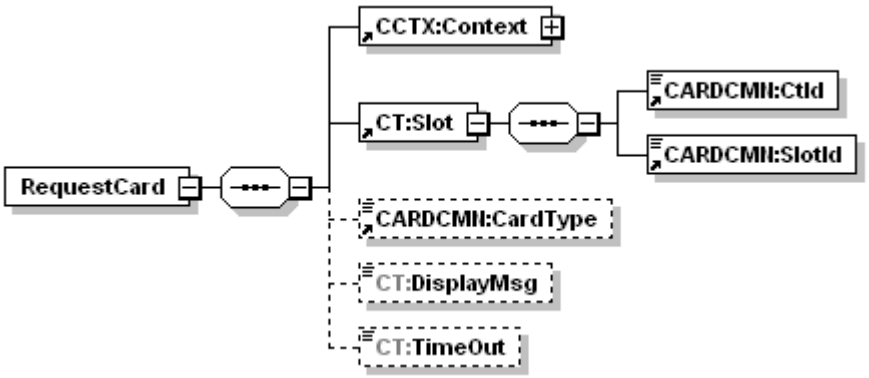
⌫

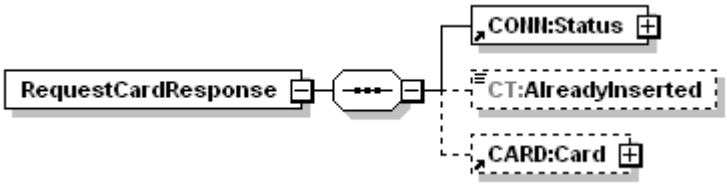
4.1.4.5.1 RequestCard

⌦ **TIP1-A_5412 Operation RequestCard**

Der Konnektor MUSS an der Außenschnittstelle eine Operation RequestCard, wie in Tabelle TAB_KON_716 Operation RequestCard beschrieben, anbieten.

Tabelle 41: TAB_KON_716 Operation RequestCard

Name	**`RequestCard`**	
Beschreibung	Liefert die Information zu einer Karte, die in dem Slot eines Kartenterminals steckt oder innerhalb einer bestimmten Zeit (Timeout) gesteckt wird.	
Aufrufparameter		
	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 5 sec verwenden. Optional KANN dieser Default-Wert im Konnektor konfigurierbar sein.
Rückgabe		
	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)
Vorbedingung	keine	
Nachbedingung	keine	

Der Ablauf der Operation RequestCard ist in Tabelle TAB_KON_717 Ablauf RequestCard beschrieben.

Tabelle 42: TAB_KON_717 Ablauf RequestCard

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 Zugriffs-berechtigung“	Prüfung der Zugriffsberechtigung durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId, $ctId; needCardSesssion=false; allWorkplaces=false } Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.
3.	TUC_KON_056 „Karte anfordern“	Anforderung der Karte vom Kartenterminal durch Aufruf TUC_KON_056($slot.ctId, $slot.slotid, $cardType, $displayMsg, $TimeOut)

Tabelle 43: TAB_KON_718 Übersicht Fehler Operation „RequestCard“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4000	Technical	Error	Syntaxfehler
4202	Technical	Error	Timeout. Es wurde keine Karte innerhalb der angegebenen Zeitspanne gesteckt.
4058	Security	Error	Aufruf nicht zulässig

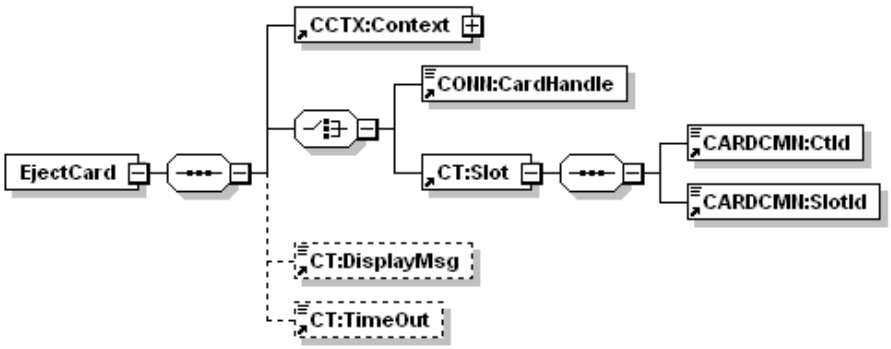
⌫

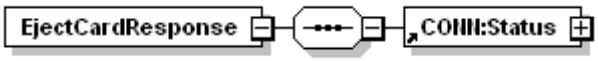
4.1.4.5.2 EjectCard

⌦ **TIP1-A_5413 Operation EjectCard**

Der Konnektor MUSS an der Außenschnittstelle eine Operation EjectCard, wie in Tabelle TAB_KON_719 Operation EjectCard beschrieben, anbieten.

Tabelle 44: 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.	
Aufrufparameter		
	Name	Beschreibung
	Context	MandantId, CsId, WorkplaceId verpflichtend
	`CONN: CardHandle`	Adressiert die Karte, die ausgeworfen werden soll. Unterstützt werden die Kartentypen EGK, KVK, HBAx, SM-B und UNKNOWN.
	`CT:Slot`	Adressiert alternativ den Slot eines Kartenterminals, aus dem die Karte ausgeworfen werden soll. Die Adressierung erfolgt über die Identifikation des Kartenterminal `CARDCMN:CtId` und die Nummer des Slots `CARDCMN:SlotId`.
	`CT: 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 5000 msec verwenden. Optional KANN dieser Default-Wert im Konnektor konfigurierbar sein.
Rückgabe		
	Name	Beschreibung
	`Status`	Enthält den Ausführungsstatus der Operation (siehe 3.5.2)
Vorbedingung	keine.	
Nachbedingung	keine.	

Der Ablauf der Operation EjectCard ist in Tabelle TAB_KON_720 Ablauf EjectCard beschrieben.

Tabelle 45: 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 Zugriffs-berechtigung“	Ist $cardHandle vorgegeben, so wird $ctId als Id des Kartenterminals bestimmt, in dem die Karte steckt. Prüfung der Zugriffsberechtigung durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; $ctId; needCardSesssion=false; allWorkplaces=false } Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.
3.	TUC_KON_057 „Karte auswerfen“	Veranlassen des Kartenauswurfs am Kartenterminal durch Aufruf TUC_KON_057($slot.ctId, $slot.slotid, Card($cardHandle).cardType, $displayMsg, $TimeOut)

Tabelle 46: TAB_KON_721 Übersicht Fehler 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

⌫

4.1.4.6 Betriebsaspekte

4.1.4.6.1 Allgemeine Betriebsaspekte

⌦ TIP1-A_4549 Initialisierung Kartenterminaldienst

Während des Bootvorgangs, nach dem Einlesen der persistierten Informationen des Kartenterminaldienstes MUSS der Konnektor für jedes Kartenterminal CT in CTM_CT_LIST:

- die zugehörigen Attribute aktualisieren
- TUC_KON_050 „Beginne Kartenterminalsitzung“ {CT.CtID, User} für jedes CT mit `CT.CORRELATION` = „aktiv“ aufrufen⌫

⌦ TIP1-A_4550 Konfigurationsparameter des Kartenterminaldienstes

Die Managementschnittstelle MUSS es einem Administrator ermöglichen Konfigurationsänderungen gemäß Tabelle TAB_KON_527 vorzunehmen:

Tabelle 47 TAB_KON_527 Konfigurationswerte eines Kartenterminalobjekts

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
`CTM_SERVICE_DISCOVERY_PORT`	`Portnummer`	Der Administrator MUSS die Portnummer eingeben können, auf der die KTs im lokalen Netz auf Dienstanfragen hören. 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. Default-Wert=3
`CTM_SERVICE_ANNOUNCEMENT_PORT`	`Portnummer`	Der Administrator MUSS die Portnummer eingeben können, auf der der Konnektor auf Dienstbeschreibungspakete hört. 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. Default-Wert=10, 0=Deaktiviert
`CTM_KEEP_ALIVE_INTERVAL`	`X Sekunden`	Intervall in Sekunden in den Keep-Alive Nachrichten an das Kartenterminal gesendet werden Der Administrator MUSS diesen Wert im vorgegebenen Bereich anpassen können. Wertebereich: 1 -10 Default-Wert=10
`CTM_KEEP_ALIVE_TRY_COUNT`	`Anzahl der Versuche`	Anzahl von aufeinander folgenden, nicht beantworteten Keep-Alive Nachrichten, nachdem ein Timeout der TLS-Verbindung festgestellt wird Der Administrator MUSS diesen Wert im

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
		vorgegebenen Bereich anpassen können. Wertebereich: 3 -10 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). Wertebereich: 1-60 Default-Wert=10

⌫

⌦ **TIP1-A_4986 Informationsparameter des Kartenterminaldienstes**

Die Managementschnittstelle MUSS es einem Administrator ermöglichen die Informationsparameter gemäß Tabelle TAB_KON_528 einzusehen:

Tabelle 48 TAB_KON_528 Informationsparamter des Kartenterminaldienstes

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
CTM_SUPPORTED_KT_VERSIONS	Liste von KT-Produkttyp-versionen	Der Administrator MUSS die Liste der vom Konnektor unterstützten modellunabhängigen KT-Produkttypversionen einsehen können.

⌫

4.1.4.6.2 Kartenterminals pflegen

Im Folgenden werden die Administratorinteraktionen beschrieben, die zum Hinzufügen, Pairen, Bearbeiten und Löschen von Kartenterminals innerhalb der CTM_CT_LIST angeboten werden müssen. Eine Aktualisierung der Kartenterminals mit neuer Firmware wird in Kapitel 4.3.9 beschrieben.

⌦ **TIP1-A_4551 Einsichtname von Kartenterminaleinträgen**

Die Managementschnittstelle MUSS es einem Administrator ermöglichen die Liste der verwalteten und neu entdeckten Kartenterminals einzusehen (CTM_CT_LIST).⌫

⌦ **TIP1-A_4552 Manueller Verbindungsversuch zu Kartenterminals**

Die Managementschnittstelle MUSS es einem Administrator ermöglichen zu jedem CT-Object-Eintrag in CTM_CT_LIST mit CT.CONNECTED=Nein und CT.CORRELATION=aktiv einen manuellen Verbindungsaufbau über TUC_KON_050 {CtID, User} auszulösen.⌫

⌦ **TIP1-A_4553 Einsichtname in und Aktualisierung der Kartenterminaleinträge**

Die Managementschnittstelle MUSS es einem Administrator ermöglichen zu jedem CT-Object-Eintrag in CTM_CT_LIST die Werte gemäß Tabelle TAB_KON_529 einsehen zu können:

Zu jedem Eintrag MUSS der Administrator TUC_KON_055 „Befülle CT-Object“ auslösen können.

Tabelle 49 TAB_KON_529 Anzeigewerte zu einem Kartenterminalobjekt

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
Gerätekenndaten		
CT.CTID	Identifier	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.PRODUCTINFORMATION	Inhalt ProductInformation.xsd	Die Herstellerinformationen zum Kartenterminal gemäß [gemSpec_OM]
CT.VALID_VERSION	Boolean	True, wenn die Version des Kartenterminals durch den Konnektor unterstütz wird
Pairingdaten		
CT.SMKT_AUT	X.509-Cert	C.SMKT.AUT-Zertifikat des Kartenterminals, gespeichert im Rahmen des Pairings
Verbindungsdaten		
CT.CORRELATION	bekannt zugewiesen gepairt aktiv aktualisierend	Der Korrelationsstatus zum Konnektor: • bekannt (über Service Announcement/Service Discovery gelernte Kartenterminals), • zugewiesen (durch den Administrator aus dem Bereich der bekannten Kartenterminals oder manuell konfigurierte Kartenterminals), • gepairt (Pairing erfolgreich aber noch nicht zum Verbindungsaufbau freigegeben) • aktiv (durch Administrator zum Verbindungsaufbau freigegeben), • 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"),
CT.CONNECTED	Ja / Nein	Der Verfügbarkeitsstatus des Kartenterminals (Ja =

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
		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

⌫

⌦ **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.

Tabelle 50 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_4555 Manuelles Hinzufügen eines Kartenterminals**

Die Managementschnittstelle MUSS es einem Administrator ermöglichen für neue Kartenterminals CT-Objects manuell in CTM_CT_LIST aufzunehmen.

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

- IP-Adresse (Eingabe verpflichtend)
- TCP-Port (Eingabe optional)
- MAC-Adresse (Eingabe optional)
- Hostname (Eingabe optional)

Bestätigt der Administrator seine Eingaben, MUSS TUC_KON_054 mit Mode=ManuallyAdded und allen eingegebenen Parametern aufgerufen werden.⌫

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

⌦ **TIP1-A_4556 Pairing mit Kartenterminal durchführen**

Die Managementschnittstelle MUSS es einem Administrator ermöglichen alle Kartenterminals mit CT.CORRELATION = „zugewiesen“ in einer Liste einzusehen und für einen ausgewählten Eintrag mit CT.VALID_VERSION=True TUC_KON_053 auslösen zu können.⌫

⌦ **TIP1-A_4557 Ändern der Korrelationswerte eines Kartenterminals**

Die Managementschnittstelle MUSS es einem Administrator ermöglichen zu einem Kartenterminal aus CTM_CT_LIST für KTs mit CT.IS_PHYSICAL=Ja den Wert für CT.CORRELATION nach folgenden Mustern zu ändern:

- CT.CORRELATION = „bekannt“
  Das Kartenterminal gilt als nicht durch den Konnektor verwaltet.
  - → „zugewiesen“:
    Ein (per Service Announcement entdecktes) Kartenterminal dem Konnektor zuweisen.
    Folgende Schritte MUSS der Konnektor für diesen Zustandswechsel zuvor erfolgreich durchlaufen:
    - Rufe TUC_KON_055 **„**Befülle CT-Object**“**
    - Prüfen, ob CT.HOSTNAME bereits für ein anderes Kartenterminal in CTM_CT_LIST verwendet wird. Wenn ja MUSS dieser Änderungsversuch fehlschlagen (Prinzip der Eindeutigkeit verletzt). Der Administrator MUSS eine entsprechende Fehlermeldung erhalten.
- CT.CORRELATION = „zugewiesen“
  Das Kartenterminal gilt als durch den Konnektor verwaltet.
  - → „bekannt“
- CT.CORRELATION = „gepairt“
  Verbundenheit zwischen Kartenterminal und gesteckter gSMC-KT wurde nachgewiesen

- o → „zugewiesen“:
  Die Werte CT.SMKT_AUT, CT.SHARED_SECRET werden gelöscht
- o → „aktiv“:
  Wechsel nur möglich, wenn CT.VALID_VERSION=True. Dann Aufruf von TUC_KON_050 „Beginne Kartenterminalsitzung“ {CT.CtID, User}

- CT.CORRELATION = „aktiv“
  Das Kartenterminal kann fachlich genutzt werden
  - o → „gepairt“:
    Eventuelle TLS-Verbindung wird beendet, CT.CONNECTED auf Nein gesetzt.⌫

4.1.4.6.3 Import der Kartenterminal-Informationen

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

⌦ **TIP1-A_5011 Import von Kartenterminal-Informationen**

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

1. Die Liste der zu importierenden Kartenterminals MUSS dem Administrator angezeigt werden. Er MUSS die Möglichkeit erhalten, einzelne Kartenterminals aus dieser Liste zu löschen.
2. Erst nach Bestätigung durch den Administrator werden die Kartenterminalinformationen in die Kartenterminalverwaltung übernommen.
3. Sofern die Kartenterminal-Konfiguration in einen Konnektor mit neuer Identität importiert werden soll (neuer Konnektor oder neuer privater Schlüssel und neues Zertifikat C.SAK.AUT auf der gSMC-K), muss die neue Identität des Konnektors allen importierten Kartenterminals bekannt gemacht werden (Wartungs-Pairing, siehe auch [gemSpec_KT#2.5.2.4]).
   a) Dazu baut der Konnektor unter der Nutzung von C.SAK.AUT eine temporäre TLS-Verbindung auf und sendet das eHealth-Kartenterminal-Kommando EHEALTH TERMINAL AUTHENTICATE in der Variante “ADD” an jedes in der Liste aufgeführte Kartenterminal. Mit dem Kommando und P2=03 holt sich der Konnektor eine Challenge.
   b) Der eigentliche Austausch bzw. die Aufnahme des neuen Zertifikates erfolgt im KT erst, nachdem diese Challenge mit dem Kommando EHEALTH TERMINAL AUTHENTICATE im Modus P2=04 vom Konnektor korrekt beantwortet wurde. Dieses Kommando sowie die Erzeugung der Challenge-Antwort wird in [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 Sicherheitssicht 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 51 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

ReferenzID	Belegung	Zustandswerte
CARD.ICCSN		ICCSN der Karte (sofern auslesbar),
CARD.TYPE		Typ der Karte gemäß Tabelle 1: 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.KeyInfo
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 (subject.commonName)
CARD.KVNR	String	Versichertennummer
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 a) C2C:KeyRef, Role oder b) CHV: PINRef	Liste von erreichten Sicherheitszuständen. Jeder einzelne Sicherheitszustand kann entweder über C2C gegen KeyRef (mit einer bestimmten Rolle) oder Card Holder Verification (CHV) gegen eine referenzierte PIN erreicht worden sein.
CARDSESSION.MANDANTID		Mandant-ID

ReferenzID	Belegung	Zustandswerte
CARDSESSION.CSID		Clientsystem-ID
CARDSESSION.USERID		Nutzer-ID
CARDSESSION.AUTHBY	Referenz auf CardSession	Kartensitzung, über die diese Karte freigeschaltet wurde (nur für eGK und SM-B belegt)

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.

Bei Karten der Generation 2

- MUSS der Konnektor für eGKs nur die Elemente des Objektsystems verwenden, die bereits in Generation 1+ definiert waren.
- MUSS der Konnektor die ELC-basierten Geräte-CV-Zertifikate unterstützen.⌫

⌦ 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.⌫

⌦ 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.CARDSESSION). 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 Terminal-Anzeigen für PIN-Operationen**

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

Tabelle 52: TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal

Karte/ Kontext	PIN-Referenz	I/O	Terminal-Anzeige
eGK	PIN.CH	I	Eingabe•***0x0B***Versicherten-***0x0B***PIN•***0x0B***für•*ANW* ***0x0F***PIN.eGK:
HBAx	PIN.CH	I	Eingabe•***0x0B***Freigabe-PIN•***0x0B***HBA ***0x0F***PIN.HBA:
	PIN.QES	I	*#UVW-XYZ*•***0x0B***Eingabe•***0x0B***Signatur-PIN•***0x0B***HBA ***0x0F***PIN.QES:
SMC-B	PIN.SMC	I	Eingabe•***0x0B***PIN•SMC-B•***0x0B***SLOT:*X* ***0x0F***PIN.SMC:
	PIN.CONF	I	Eingabe•***0x0B***SMC-B•PIN•CONF•***0x0B***SLOT:*X* ***0x0F***PIN.CONF:
ANDERE	BELIEBIG	I	Herstellerspezifisch
Erfolgreiche PIN-Eingabe	ALLE	O	PIN•***0x0B***erfolgreich•***0x0B***verifiziert!
Fehlerhafte PIN-Eingabe	ALLE	O	PIN•***0x0B***falsch•***0x0B***oder•***0x0B***gesperrt!
PUK-Eingabe	eGK PUK.CH	I	Eingabe•***0x0B***Versicherten-***0x0B***PUK ***0x0F***PUK.eGK:
	HBAx PUK.CH	I	Eingabe•***0x0B***Freigabe-PUK•***0x0B***HBA ***0x0F***PUK.HBA:
	HBAx PUK.QES	I	Eingabe•***0x0B***Signatur-PUK•***0x0B***HBA ***0x0F***PUK.QES:
	SMC-B PUK.SMC	I	Eingabe•***0x0B***PUK•SMC-B•***0x0B***SLOT:*X* ***0x0F***PUK.SMC:

Karte/ Kontext	PIN-Referenz	I/O	Terminal-Anzeige
	SMC-B PUK.CONF	I	Eingabe•***0x0B***SMC-B•PUK•CONF•***0x0B***SLOT:*X* ***0x0F***PUK.CONF:
	SMC-B PUK.NET	I	Eingabe•***0x0B***SMC-B•PUK•NET•***0x0B***SLOT:*X* ***0x0F***PUK.NET:
Erfolgreiche PUK-Eingabe	ALLE	O	PIN•***0x0B***erfolgreich•***0x0B***entsperrt!
Fehlerhafte PUK-Eingabe	ALLE	O	PUK•***0x0B***falsch•***0x0B***oder•***0x0B***gesperrt!
Eingabe einer neuen PIN	eGK PIN.CH	I	Eingabe• ***0x0Bn***eue•***0x0B***Versicherten-***0x0B***PIN• ***0x0B***(6-8 Ziffern) ***0x0F***PIN.eGK:
	HBAx PIN.CH	I	Eingabe•***0x0B***neue•***0x0B***Freigabe-PIN•***0x0B***HBA•***0x0B***(5-8 Ziffern) ***0x0F***PIN.HBA:
	HBAx PIN.QES	I	Eingabe• ***0x0B***neue•***0x0B***Signatur-PIN•***0x0B***HBA•***0x0B***(6-8 Ziffern) ***0x0F***PIN.QES:
	SMC-B PIN.SMC	I	Eingabe•***0x0B***neue•***0x0B***PIN SMC-B• ***0x0B***SLOT:*X*•***0x0B***(6-8 Ziffern) ***0x0F***PIN.SMC:
	SMC-B PIN.CONF	I	Eingabe•***0x0B***neue•***0x0B***SMC-B•PIN.CONF•***0x0B*** SLOT:*X* ***0x0F***PIN.CONF:
Eingabe einer Transport-PIN	eGK PIN.CH	I	Eingabe•***0x0B***Transport-***0x0B***Versicherten-***0x0B***PIN ***0x0F***T-PIN.eGK:
	HBAx PIN.CH	I	Eingabe•***0x0B***Transport-***0x0B***PIN•***0x0B***HBA ***0x0F***T-PIN.HBA:
	HBAx PIN.QES	I	Eingabe•***0x0B***Transport-***0x0B***PIN•***0x0B***HBA ***0x0F***T-PIN.QES:
	SMC-B PIN.SMC	I	Eingabe•***0x0B***Transport-***0x0B***PIN•SMC-B•***0x0B***SLOT:*X* ***0x0F***T-PIN.SMC:
Wiederholung einer neuen PIN	eGK PIN.CH	I	Eingabe•***0x0B***Versicherten-***0x0B***PIN•***0x0B*** wiederholen! ***0x0F***PIN.eGK:
	HBAx	I	Eingabe•***0x0B***für•HBA•***0x0B***wiederholen!

Karte/ Kontext	PIN-Referenz	I/O	Terminal-Anzeige
	PIN.CH		***0x0F***PIN.HBA:
	HBAx PIN.QES	I	Eingabe•***0x0B***für•HBA•***0x0B***wiederholen! ***0x0F***PIN.QES:
	SMC-B PIN.SMC	I	Eingabe•***0x0B***PIN.SMC•***0x0B***SLOT:*X*• ***0x0B***wiederholen! ***0x0F***PIN.SMC:
	SMC-B PIN.CONF	I	Eingabe•***0x0B***PIN.CONF•***0x0B***SLOT:*X*•***0x0B*** wiederholen! ***0x0F***PIN.CONF:
Ungleichheit bei der Wiederholung der Eingabe der neuen PIN	ALLE	O	PINs•***0x0B*** nicht•***0x0B***identisch!•***0x0B***Abbruch!

⌫

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

- ANW wird durch das Anwendungskürzel ersetzt (siehe TUC_KON_012)
- Zu PIN.SMC/PIN.CONF: "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 in als "•" dargestellt
- UVW-XYZ: zeigt die Jobnummer an (siehe Kapitel 4.1.8.1.6)
- #: Beginn der Job-Nummer zur Verifizierung des korrekten Kartenterminals
- Weitere Details zur Gestaltung der Job-Nummer finden sich im Kapitel 4.1.8.1.6.
- 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 Remote-PIN-KT von workplaceID (PinInputKT). Dabei wendet der Konnektor das folgende Verfahren an:

⌦ **TIP1-A_5012 Remote-PIN-Verfahren**

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

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

Für die Interaktion mit dem Anwender MÜSSEN die Display Messages entsprechend Tabelle 52: TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal verwendet werden.

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

1. Aufruf TUC_KON_005 „Card-to-Card authentisieren“ mit eigens für diesen Zweck erzeugten Cardsession sowohl für die „Sendekarte“ im PinInputKT (gSMC-KT oder SM-B) sowie der Zielkarte. AuthMode ist „gegenseitig+TC“
2. Der Benutzer wird zur Eingabe der PIN am PinInputKT aufgefordert.
3. Im PinInputKT verschlüsselt die gSMC-KT die eingegebene PIN mit dem zuvor erzeugten Sessionkey.
4. Die verschlüsselte PIN wird in das zur intendierten PIN-Operation passende Kommando eingebettet (PIN verifizieren, ändern oder entsperren - wird durch den eigentlichen PIN-TUC festgelegt) und das Kommando vom Konnektor an die Zielkarte zur Entschlüsselung und Verifikation übergeben. Dabei MUSS die Übertragung im gleichen Logischen Kanal wie die SM Vereinbarung erfolgen.
5. Das Result der Zielkarte wird an den Aufrufer zurückgegeben

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

4.1.5.2 Durch Ereignisse ausgelöste Reaktionen

⌦ **TIP1-A_4562 Reaktion auf „Karte entfernt“**

Empfängt der Kartendienst das Ereignis „CT/SLOT_FREE“, so MUSS der Konnektor:

- das über die im Ereignis gemeldeten Parameter CtID und SlotNo in CM_CARD_LIST adressierte CardObject CARD identifizieren
- für dieses CardObject folgendes Ereignis absetzen:
  TUC_KON_256{„CARD/REMOVED“; Op; Info; <Params>} wobei <Params> mit folgenden Werten belegt werden MUSS:

- o „CardHandle=$CARD.CARDHANDLE;
- o Type=$CARD.TYP;
- o CardVersion=$CARD.VER;
- o ICCSN=$CARD.ICCSN
- o CtID=$CARD.CTID
- o SlotID=$CARD.SLOTID
- o InsertTime=$CARD.INSERTTIME
- o CardHolderName=$CARD.CARDHOLDERNAME
- o KVNR=$CARD.KVNR“

- das zugehörige CardObject aus CM_CARD_LIST entfernen ⌫

⌦ **TIP1-A_4563 Reaktion auf „Karte gesteckt“**

Empfängt der Kartendienst das Ereignis „CT/SLOT_IN_USE“, so MUSS der Konnektor für die Karte, die über die im Ereignis gemeldeten Parameter CtID und SlotNo adressiert ist, über TUC_KON_001 ein neues CardObject in CM_CARD_LIST anlegen. ⌫

4.1.5.3 Interne TUCs, nicht durch Fachmodule nutzbar

4.1.5.3.1 TUC_KON_001 „Karte öffnen“

⌦ **TIP1-A_4565 TUC_KON_001 „Karte öffnen“**

Der Konnektor MUSS den technischen Use Case “Karte öffnen“ gemäß TUC_KON_001 umsetzen.

Tabelle 53: 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 Karten kann erst im Anschluss unter Verwendung des erzeugten CardHandles verwendet werden.
Auslöser	• Der Kartenterminaldienst meldet das Belegen eines KT-Slots
Vorbedingungen	• In CtID/SlotNo steckt eine Karte
Eingangsdaten	• CtID • SlotNo
Komponenten	Karte, Kartenterminal, Konnektor
Ausgangsdaten	Keine

Element	Beschreibung
Standardablauf	1. Prüfe, ob unter CtID und SlotNo ein Eintrag in CM_CARD_LIST vorhanden ist. Wenn bereits ein Eintrag vorhanden ist, lösche diesen. 2. Erzeuge neuen Card-Object-Eintrag in CM_CARD_LIST und a) Generiere CARD.CARDHANDLE. mit folgenden Anforderungen: ▪ Das CardHandle MUSS innerhalb CM_CARD_LIST eindeutig sein ▪ Ein ungültig gewordenes CardHandle DARF innerhalb von 48h NICHT als neues CardHandle vergeben werden b) Befülle CARD.CTID und CARD.SLOTNO mit den Eingangsdaten c) Ermittle und befülle (soweit durch Karte unterstützt) die folgenden Daten: ▪ CARD.ICCSN ▪ CARD.TYPE (mögliche Werte siehe Tabelle 1: TAB_KON_500 Wertetabelle Kartentypen) ▪ CARD.CARDVERSION ▪ CARD.INSERTTIME (=aktuelle Systemzeit) ▪ CARD.CARDHOLDERNAME (aus X.509-AUT-Zertifikat) ▪ CARD.KVNR (nur für eGK, aus C.CH.AUT) ▪ CARD.CERTEXPIRATIONDATE (=validity aus X.509-AUT-Zertifikat) X.509-AUT-Zertifikat bezeichnet für eGK das C.CH.AUT-Zertifikat, für HBA das C.HP.AUT-Zertifikat und für SMC-B das C.HCI.AUT-Zertifikat. 3. Rufe TUC_KON_256{„CARD/INSERTED“; Op; Info; <Params>} mit <Params> belegt aus dem CARD-Object: „CardHandle=$; CardType=$; CardVersion=$; ICCSN=$;CtID=$; SlotID=$;InsertTime=$; CardholderName=$; KVNR=$; CertExpirationDate=$“
Varianten/Alternativen	a) 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.
Fehlerfälle	(-> 2c) Karte/Kartenterminal antwortet mit einer spezifischen Fehlermeldung, Fehlercode <gemäß [gemSpec_COS]/[SICCT]> Alle anderen Fehler beim Zugriff auf die Karte führen dazu, dass keine weiteren Zugriffsversuche unternommen werden. Es wird jedoch trotzdem die Karte in die interne Liste des Konnektors aufgenommen. Konnte bis zu diesem Zeitpunkt Card.Type noch nicht bestimmt werden, wird UNKNOWN verwendet. Schritt 3 wird in jedem Fall aufgerufen.
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

⌫

4.1.5.4 Interne TUCs, auch durch Fachmodule nutzbar

4.1.5.4.1 TUC_KON_026 „Liefere CardSession“

TIP1-A_4566 TUC_KON_026 „Liefere CardSession“

Der Konnektor MUSS den technischen Use Case “ Liefere CardSession“ gemäß TUC_KON_26 umsetzen.

Tabelle 54: 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	• Indirekter Aufruf über durch Clientsysteme ausgeführte Operationen. • Aufruf durch Fachmodul
Vorbedingungen	Keine
Eingangsdaten	• MandantId • clientSystemId • cardHandle • userId (nur für CardType = HBAx, da aber verpflichtend)
Komponenten	Konnektor
Ausgangsdaten	• CardSession
Standardablauf	1. Ermittle Card in CM_CARD_LIST über CardHandle 2. Prüfe dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt. 3. Ermittle CardSession in Card.CARDSESSION_LIST über MandantId, clientSystemId und userId
Varianten/Alternativen	(→2) Karte bereits reserviert, Fehlercode 4093 (→3) Wenn keine CardSession für diese Parameter vorhanden: 1. erzeuge neue CardSession in Card. CARDSESSION_LIST 2. Befülle CardSession.MANDANTID, .CSID und .USERID mit Übergabeparametern
Fehlerfälle	Keine
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

4.1.5.4.2 TUC_KON_012 „PIN verifizieren“

⌦ TIP1-A_4567 TUC_KON_012 „PIN verifizieren“

Der Konnektor MUSS den technischen Use Case “PIN verifizieren“ gemäß TUC_KON_012 umsetzen.

Tabelle 55: 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. Remote-PIN-Eingabe wird dabei automatisch unterstützt.
Auslöser	• Aufruf des Use Case durch Basisdienste des Konnektors • Aufruf des Use Cases durch ein Fachmodul im Konnektor • Aufruf der Operation VerifyPin des CardService (siehe 4.1.5.5.1) durch das Clientsystem.
Vorbedingungen	Keine
Eingangsdaten	• CardSession (Kartensitzung der Karte, deren PIN verifiziert werden soll) • workplaceID • PinRef (laut Kartenspec) • AppName (Name der zugreifenden Fachanwendung, z. B. „VSDM“, max. 9 Zeichen) • VerificationTyp (Art der PIN-Verifikation): o Mandatorisch: PIN wird immer verifiziert. o Sitzung: PIN wird nicht erneut verifiziert, falls dies für die CardSession zuvor bereits geschehen ist und der dadurch erreichte Sicherheitszustand nicht zurückgesetzt wurde.
Komponenten	Karte, Kartenterminal, Konnektor
Ausgangsdaten	Ergebnis der PIN-Verifikation: [OK / REJECTED / BLOCKED / ERROR]
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt. 3. Wenn PinTyp(PinRef)=PIN.QES oder VerificationTyp=Mandatorisch →6. 4. Wenn PinRef in CARDSESSION.AUTHSTATE vorhanden: a. Result = OK b. → 8 5. Prüfe TUC_KON_022 “Liefere PIN-Status“ = „verifizierbar“ 6. Ermittle PinInputKT: Wenn Card.ctId ein dem Arbeitsplatz(workplaceID) lokal zugeordnetes Kartenterminal ist (siehe Relation [6], Kapitel 4.1.1.1) a. Setze PinInputKT = Card.CtID b. sonst „lokales Kartenterminal, das für die Remote-PIN-Eingabe zu verwenden ist“: PinInputKT =

Element	Beschreibung
	Arbeitsplatz(workplaceId).remote-PIN-KT(mandantId) 7. Atomare Operation: PIN verifizieren inkl. Eventing und Ergebnisvermerk a. Rufe TUC_KON_256 {„CARD/PIN/VERIFY_STARTED"; Op; Info; „CardHandle=$; CardType=$; ICCSN=$;CtID=$; SlotID=$; PinRef=$PinRef; PinInputCtID=$PinInputKT"; noLog} b. Pin-Verifikation über „Perform Verification" ([SICCT]) mit Display Messages entsprechend Tabelle 52: TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal, bei eGK mit PIN.CH ersetze „ANW" durch AppName in Display Message. Wenn PinInputKT=Card.CtID dann PIN Verifikation direkt an Card.CtID, ansonsten Remote-PIN-Eingabe gemäß (TIP1-A_5012) c. Setze Result in Abhängigkeit von Ergebnis Perform Verification: - Result = OK für erfolgreiche Prüfung - Result = ERROR für Nutzer-Abbruch oder Bearbeitungsfehler (siehe Fehlerfälle) - Result = REJECTED für falsche PIN - Result = BLOCKED für gesperrte PIN d. Rufe TUC_KON_256 {„CARD/PIN/VERIFY_FINISHED"; Op; Info; „CardHandle=$; CardType=$; ICCSN=$;CtID=$; SlotID=$; PinRef=$PinRef; PinInputCtID=$PinInputKT; Result=$Result"; noLog} e. Befülle CARDSESSION.AUTHSTATE mit PinRef und Ergebnis der PIN-Prüfung 8. Liefere Result zurück
Varianten/Alternativen	Keine
Fehlerfälle	Schritt 7 wird in jedem Fall 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 oder BLOCKED abgebildet. * Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094 (→2) Karte ist fremd reserviert, Fehlercode 4093 (→5) Rückgabewert= - verifiziert, Fehlercode 4001 - transportgeschützt (Transport-PIN oder Leer-PIN), Fehlercode 4065 - gesperrt, Fehlercode 4063 (→6b) kein Remote-PIN-KT zugeordnet, Fehlercode 4092 (→7) Timeout bei PIN Eingabe: Fehlercode 4043. (→7) Abbruch durch Nutzer: Fehlercode 4049. (→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

Element	Beschreibung
	bricht der Use Case mit Fehler 4060 ab. (→7b) Karte/Kartenterminal antwortet mit einer spezifischen Fehlermeldung, Fehlercode <gemäß [gemSpec_COS]/[SICCT]> Zusätzliche Fehlerfälle ergeben sich aus der Remote-PIN-Eingabe gemäß (TIP1-A_5012)
Nichtfunktionale Anforderungen	
Zugehörige Diagramme	Abbildung 10: PIC_KON_111 Aktivitätsdiagramm zu „PIN verifizieren“

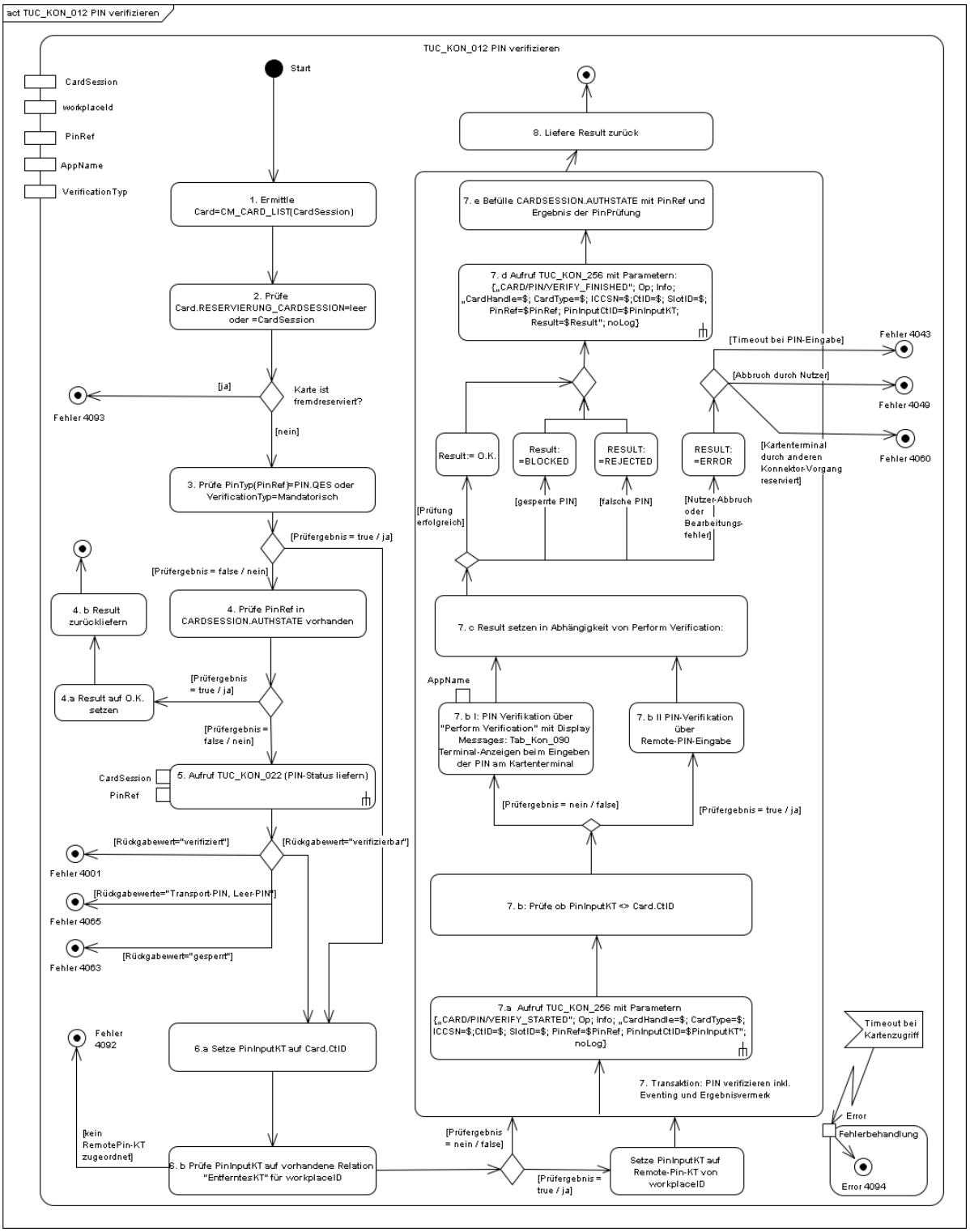


Abbildung 10: PIC_KON_111 Aktivitätsdiagramm zu „PIN verifizieren“

Tabelle 56: TAB_KON_089 Übersicht Fehler

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
4060	Technical	Error	Ressource belegt
4063	Security	Error	PIN bereits gesperrt (BLOCKED)
4065	Technical	Warning	PIN ist transportgeschützt, Änderung erforderlich
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

⌫

4.1.5.4.3 TUC_KON_019 „PIN ändern“

⌦ **TIP1-A_4568 TUC_KON_019 „PIN ändern“**

Der Konnektor MUSS den technischen Use Case “PIN ändern“ gemäß TUC_KON_019 umsetzen.

Tabelle 57: 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	• Aufruf der Operation ChangePin des CardService (siehe 4.1.5.5.2) durch das Clientsystem. • Aufruf durch Fachmodul
Vorbedingungen	keine
Eingangsdaten	• CardSession • workplaceID • PinRef (laut Kartenspec.) • Sup.CardSession (der Karte, die für die Card-to-Card-Authentisierung bei Änderung der PIN einer eGK der Generation 1 verwendet werden soll) (optional)
Komponenten	Karte, Kartenterminal, Konnektor
Ausgangsdaten	Keine

Element	Beschreibung
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt. 3. Prüfe TUC_KON_022 “Liefere PIN-Status“ {CardSession; PinRef}<>gesperrt 4. Wenn Card.TYP=eGK UND Card.Version=Generation1 UND PIN-Status in {Transport-PIN, Leer-PIN}", Aufruf TUC_KON_005 „Card-to-Card authentisieren“ {Sup.CardSession; CardSession; einseitig}. Falls keine Sup.CardSession angegeben ist, kann die CardSession der für den Mandanten verwalteten SMC-B verwendet werden. 5. Ermittle PinInputKT: Wenn Card.ctId ein dem Arbeitsplatz(workplaceID) lokal zugeordnetes Kartenterminal ist (siehe Relation [6], Kapitel 4.1.1.1) a. Setze PinInputKT = Card.CtID b. sonst „lokales Kartenterminal, das für die Remote-PIN-Eingabe zu verwenden ist“: PinInputKT = Arbeitsplatz(workplaceId).remote-PIN-KT(mandantId) 6. Atomare Operation: PIN ändern inkl. Eventing und Ergebnisvermerk a. Rufe TUC_KON_256 {„CARD/PIN/CHANGE_STARTED“; Op; Info; „CardHandle=$; CardType=$; ICCSN=$;CtID=$; SlotID=$; PinRef=$PinRef; PinInputCtID=$PinInputKT“; noLog} b. Pin-Änderung über „MODIFY VERIFICATION DATA“ ([SICCT]) mit Display Messages entsprechend Tabelle 52: TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal. Wenn PinInputKT=Card.CtID dann PIN-Änderung direkt an Card.CtID, ansonsten Remote-PIN-Eingabe gemäß (TIP1-A_5012) Dabei Unterstützung normaler PIN-Änderung als auch Umsetzens eines Transportschutzes (alle Variante gemäß Kartenspec sind zu unterstützen) c. Setze Result in Abhängigkeit von Ergebnis MODIFY VERIFICATION DATA: - Result = OK für erfolgreiche Änderung - Result = ERROR für Nutzer-Abbruch oder Bearbeitungsfehler (siehe Fehlerfälle) - Result = REJECTED für falsche PIN-Eingaben - Result = BLOCKED für gesperrte PIN d. Rufe TUC_KON_256 {„CARD/PIN/CHANGE_FINISHED“; Op; Info; „CardHandle=$; CardType=$; ICCSN=$;CtID=$; SlotID=$; PinRef=$PinRef; PinInputCtID=$PinInputKT; Result=$Result“; noLog} e. Entferne PinRef aus CARDSESSION.AUTHSTATE
Varianten/Alternativen	Keine
Fehlerfälle	Schritt 6 wird in jedem Fall mit allen Teilschritten durchlaufen. Ein als Fehler ausgewiesener Zustand führt erst nach Schritt 6e zum Abbruch des TUCs. * Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094 (→2) Karte ist fremd reserviert, Fehlercode 4093

Element	Beschreibung
	(→3) PIN.STAT=Blocked: Fehlercode 4063 (→4) Sup.CardSession benötigt aber leer, Fehlercode 4071 (→5b) kein Remote-PIN-KT zugeordnet, Fehlercode 4092 (→6b) alte PIN falsch eingegeben: Fehlercode 4061 (→6b) neue PIN zu kurz/lang: Fehlercode 4068 (→6b) zweite neue PIN<> erste neue PIN: Fehlercode 4067 (→6b) Timeout bei PIN Eingabe: Fehlercode 4043. (→6b) Abbruch durch Nutzer: Fehlercode 4049. (→6b) PIN.STAT=Blocked: Fehlercode 4082 (→6b) Ist das Kartenterminal oder Teile davon (PIN-Pad, Display) durch einen anderen Vorgang reserviert: Fehlercode 4060 (→6b) kein PIN-Pad am Kartenterminal verfügbar: Fehlercode 4066 (→6b) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode <Kartenfehlercode gemäß [gemSpec_COS]> Zusätzliche Fehlerfälle ergeben sich aus der Remote-PIN-Eingabe gemäß(TIP1-A_5012)
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 58: TAB_KON_093 Übersicht Fehler 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
4060	Technical	Error	Ressource belegt
4061	Security	Warning	Falsche alte PIN, verbleibende Eingabeversuche <x>
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 hat nicht die zulässige Länge
4071	Technical	Error	Keine Karte für C2C Auth gesetzt
4082	Security	Error	PIN durch diese Fehleingabe blockiert (BLOCKED)
4092	Technical	Error	Remote-PIN-KT benötigt aber für diesen Arbeitsplatz nicht definiert
4093	Technical	Error	Karte wird in einer anderen Kartensitzung exklusiv verwendet

© gematik – öffentlich

Fehlercode	ErrorType	Severity	Fehlertext
4094	Technical	Error	Timeout beim Kartenzugriff aufgetreten

⌫

4.1.5.4.4 TUC_KON_021 „PIN entsperren“

⌦ TIP1-A_4569 TUC_KON_021 „PIN entsperren“

Der Konnektor MUSS den technischen Use Case “PIN entsperren“ gemäß TUC_KON_011 umsetzen.

Tabelle 59: 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. Remote-PIN-Eingabe wird dabei automatisch unterstützt.
Auslöser	• Aufruf der Operation UnblockPin des CardService (siehe 4.1.5.5.4) durch das Clientsystem.
Vorbedingungen	
Eingangsdaten	• CardSession (der Karte, deren PIN entsperrt werden soll) • workplaceID • PinRef (nach Kartenspec) • setNewPin (true/false) - Angabe, ob eine neue PIN gesetzt oder die aktuelle weiterverwendet werden soll. Default = false
Komponenten	Karte, Kartenterminal, Konnektor
Ausgangsdaten	Keine
Standardablauf	1. Ermittle Card = CM_CARD_LIST(Target.CardHandle) 2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lockbesitzt. 3. Wenn TUC_KON_022 “Liefere PIN-Status“ {CardSession; PinRef}<>“gesperrt“ dann beende TUC erfolgreich 4. Ermittle PinInputKT: Wenn Card.ctId ein dem Arbeitsplatz(workplaceID) lokal zugeordnetes Kartenterminal ist (siehe Relation [6], Kapitel 4.1.1.1) a. Setze PinInputKT = Card.CtID b. sonst „lokales Kartenterminal, das für die Remote-PIN-Eingabe zu verwenden ist“: PinInputKT = Arbeitsplatz(workplaceId).remote-PIN-KT(mandantId) 5. Atomare Operation: PIN entsperren inkl. Eventing und Ergebnisvermerk a. Rufe TUC_KON_256 {„CARD/PIN/CHANGE_STARTED“; Op; Info; „CardHandle=$; CardType=$; ICCSN=$;CtID=$;

Element	Beschreibung
	SlotID=$; PinRef=$PinRef; PinInputCtID=$PinInputKT“; noLog} b. PIN-Entsperrung mit Display Messages entsprechend Tabelle 52: TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal. Wenn PinInputKT=Card.CtID, dann PIN-Änderung direkt an Card.CtID, ansonsten Remote-PIN-Eingabe gemäß (TIP1-A_5012) - Für PinRef=PIN.QES über „PERFORM VERIFICATION“ [SICCT] - Für PinRef<>PIN.QES über „MODIFY VERIFICATION DATA“ [SICCT]. Das mit dem SICCT-Kommando als Command-To-Perform mitgesandte „Reset Retry Counter“ wird entsprechend dem Wert von setNewPIN parametrisiert. c. Setze Result in Abhängigkeit von Ergebnis Perform bzw. Modify Verification: - Result = OK für erfolgreiche Entsperrung - Result = ERROR für Nutzer-Abbruch oder Bearbeitungsfehler (siehe Fehlerfälle) - Result = REJECTED für falsche PUK - Result = BLOCKED für gesperrte PUK d. Rufe TUC_KON_256 {„CARD/PIN/CHANGE_FINISHED“; Op; Info; „CardHandle=$; CardType=$; ICCSN=$;CtID=$; SlotID=$; PinRef=$PinRef; PinInputCtID=$PinInputKT; Result=$Result“; noLog}
Varianten/Alternativen	Keine
Fehlerfälle	Schritt 6 wird in jedem Fall mit allen Teilschritten durchlaufen. Ein als Fehler ausgewiesener Zustand führt erst nach Schritt 6d zum Abbruch des TUCs. * Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094 (→2) Karte wird in einer anderen Kartensitzung exklusiv verwendet, Fehlercode 4093 (→5b) kein Remote-PIN-KT zugeordnet, Fehlercode 4092 (→6b) PUK falsch eingegeben: Fehlercode 4062 (→6b) blockierte PUK: Fehlercodes 4064 (→6b) neue PIN zu kurz/lang: Fehlercode 4068 (→6b) zweite neue PIN<> erste neue PIN: Fehlercode 4067 (→6b) Timeout bei PIN Eingabe: Fehlercode 4043. (→6b) Abbruch durch Nutzer: Fehlercode 4049. (→6b) PIN.STAT=Blocked: Fehlercode 4063 (→6b) Ist das Kartenterminal oder Teile davon (PIN-Pad, Display) durch einen anderen Vorgang reserviert: Fehlercode 4060 (→6b) Karte/Kartenterminal antwortet mit einer spezifischen Fehlermeldung, Fehlercode <gemäß [gemSpec_COS]/[SICCT]> (→6b) Ungültige PIN-Referenz; Fehlercode 4072.

Element	Beschreibung
	Zusätzliche Fehlerfälle ergeben sich aus der Remote-PIN-Eingabe gemäß (TIP1-A_5012)
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 60: TAB_KON_193 Übersicht Fehler 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
4062	Security	Warning	Falsche PIN (hier: PUK) verbleibende Eingabeversuche <x>
4063	Security	Error	PIN bereits gesperrt (BLOCKED)
4064	Security	Error	Alte PIN bereits blockiert (hier: PUK)
4067	Security	Error	Neue PIN nicht identisch
4068	Security	Error	Neue PIN mit falscher Länge
4072	Technical	Error	Ungültige PIN-Referenz Die PIN-Referenz (PinRef) ist für die Karte nicht bekannt.
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

⌫

4.1.5.4.5 TUC_KON_022 „Liefere PIN-Status“

⌦ **TIP1-A_4570 TUC_KON_022 „**Liefere PIN-Status**“**

Der Konnektor MUSS den technischen Use Case “Liefere PIN-Status“ gemäß TUC_KON_022 umsetzen.

Tabelle 61 TAB_KON_532 – TUC_KON_022 „Liefere PIN-Status“

Element	Beschreibung
Name	TUC_KON_022 „Liefere PIN-Status“

Element	Beschreibung
Beschreibung	Dieser Use Case prüft den Zustand eines PIN-Objekts einer Karte im Kontext einer CardSession.
Auslöser	• Aufruf des Use Case im Rahmen von technischen Use Cases der Basisdienste des Konnektors • Aufruf des Use Cases durch ein Fachmodul im Konnektor • Aufruf der Operation GetPinStatus des CardService (siehe 4.1.5.5.1) durch das Clientsystem.
Vorbedingungen	
Eingangsdaten	• CardSession • PinRef (laut Kartenspec.)
Komponenten	Karte, Kartenterminal, Konnektor
Ausgangsdaten	• PINStatus (verifiziert, verifizierbar, Transport-PIN, Leer-PIN, gesperrt) • LeftTries (Anzahl der verbleibenden Versuche für die Verifikation der PIN)
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lockbesitzt. 3. PinRef in CardSession.AUTHSTATE vorhanden: a) Ja: PINStatus = verifiziert b) Nein: Aufruf der Kartenoperation „GET PIN STATUS“, Antwort der Karte wird ausgewertet: a. ´90 00´ (NoError: Verifiziert oder Verifikation nicht erforderlich (PIN deaktiviert)): PINStatus = verifizierbar (da nicht in dieser CardSession verifiziert) b. ´62 C1´: PINStatus = Transport-PIN c. ´62 C7´: PINStatus = Leer-PIN d. ´63 Cx´: PINStatus = verifizierbar (mit 1<= x <= 3); LeftTries=x e. ´63 C0´: PINStatus = gesperrt; LeftTries=0 f. Antwortet die Karte mit einer Fehlermeldung, bricht der TUC ab. Liefere LeftTries nur in den Fällen d und e zurück.
Varianten/Alternativen	
Fehlerfälle	* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094 (→3b) PinRef nicht gefunden: Fehlercode 4072
Zugehörige Diagramme	keine

Tabelle 62: TAB_KON_091 Übersicht Fehler 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 Die PIN-Referenz (PinRef) ist für die Karte nicht

Fehlercode	ErrorType	Severity	Fehlertext
			bekannt.
4094	Technical	Error	Timeout beim Kartenzugriff aufgetreten

⌫

4.1.5.4.6 TUC_KON_023 „Karte reservieren“

⌦ **TIP1-A_4571 TUC_KON_023 „Karte reservieren“**

Der Konnektor MUSS den technischen Use Case “Karte reservieren“ gemäß TUC_KON_023 umsetzen.

Tabelle 63: TAB_KON_533 - TUC_KON_023 “Karte reservieren”

Element	Beschreibung
Name	TUC_KON_023 “Karte reservieren”
Beschreibung	Reservierung der Karte
Auslöser	• Aufruf des Use Case im Rahmen von technischen Use Cases der Basisdienste des Konnektors • Aufruf des Use Cases durch ein Fachmodul im Konnektor
Vorbedingungen	Keine
Eingangsdaten	• CardSession • DoLock (Ja / Nein)
Komponenten	Konnektor
Ausgangsdaten	Keine
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Wenn DoLock A: =Ja: i. Prüfe, dass der zur CardSession gehörenden Karte kein Lock eines anderen Aufrufers zugeordnet ist ii. Dem Aufrufer wird ein Lock auf die zur CardSession gehörende Karte zugeordnet. Dieses wird nicht explizit als Ausgangsdatum modelliert. Der Aufrufer hat das Lock durch die implizite Zuordnung und muss es nicht verwalten. B: =Nein: i. Prüfe, dass der Aufrufer für die zur CardSession gehörende Karte ein Lock hat. ii. Das der Karte zugeordnete Lock wird gelöscht.
Varianten/Alternativen	Keine
Fehlerfälle	(→2Ai) Karte bereits reserviert, Fehlercode 4093 (→2Bi) Karte nicht durch Aufrufer reserviert, Fehlercode 4001

Element	Beschreibung
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	keine

Tabelle 64: TAB_KON_534 Übersicht Fehler 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

⌫

4.1.5.4.7 TUC_KON_005 „Card-to-Card authentisieren“

Die C2C-Authentisierung erfolgt konform zu den in [gemSpec_COS#15] festgelegten Authentisierungsprotokollen.

Definition Quellkarte/Zielkarte:

Bei einseitiger Card-to-Card-Authentisierung ohne Aushandlung eines Session Key ist die Quellkarte diejenige, die ihre Rolle bzw. Identität gegenüber der anderen Karte nachweist, z. B. der HBA bei der Freischaltung einer SMC.

Bei gegenseitiger Card-to-Card-Authentisierung ohne Aushandlung eines Session Key erfolgen nach einander zwei einseitige Card-to-Card-Authentisierungen mit vertauschten Rollen. Quell- und Zielkarte habe daher für den Gesamtablauf keine nähere Bedeutung.

Bei Card-to-Card-Authentisierung mit Aushandlung eines Session Key ist die Quellkarte diejenige, die die SM-APDUs produzieren kann, also die SMC (-KT,-B oder -K).

Die Zielkarte ist jeweils die Karte, die nicht die Quellkarte ist.

⌦ **TIP1-A_4572 TUC_KON_005 „Card-to-Card authentisieren“**

Der Konnektor MUSS den technischen Use Case “Card-to-Card authentisieren“ gemäß TUC_KON_005 umsetzen.

Die Card-to-Card-Authentisierung zwischen zwei Karten, bei der eine Karte der Generation 1 angehört MUSS das RSA-Verfahren verwenden.

Die Card-to-Card-Authentisierung zwischen zwei Karten der Generation 2 MUSS das Verfahren der elliptischen Kurven verwenden.

Tabelle 65: 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	• Aufruf des Use Case im Rahmen von technischen Use Cases der Basisdienste des Konnektors • Aufruf des Use Cases durch ein Fachmodul im Konnektor
Vorbedingungen	Wert von Source_CARDSESSION.AUTHSTATE: wenn Quellkarte a) ein HBA ist: CHV; PIN.CH, verifiziert b) eine SMC-B ist: CHV; PIN.SMC verifiziert oder C2C:PrK.SMC.AUTR_CVC.xxxx (Karte durch AuthorizeSMC freigeschaltet; xxxx in {R2048, E256})
Eingangsdaten	• Source_CardSession (Quellkarte) • Target_CardSession (Zielkarte) • AuthMode (gemäß Tabelle 66)
Komponenten	Karten, Konnektor, Kartenterminal
Ausgangsdaten	Keine
Standardablauf	1. Ermittle sCard = CM_CARD_LIST(Source_CardSession) 2. Ermittle tCard = CM_CARD_LIST(Target_CardSession) 3. Prüfe, dass der Quellkarte entweder kein Lock zugeordnet ist oder der Aufrufer im Besitz auf das Lock der Quellkarte ist. Prüfe, dass der Zielkarte entweder kein Lock zugeordnet ist oder der Aufrufer im Besitz auf das Lock der Zielkarte ist. 4. Prüfe Aufrufparameter auf erlaubte Kombination gemäß Tabelle 67 5. Prüfe das Ausstellungsdatum (CED) des CV-Zertifikats der Quellkarte gegen die aktuelle Zeit 6. Prüfe Key-Referenz gemäß Tabelle 67 = xxx.E256 oder aktuelles System Datum ist < 01.01.2018 7. Prüfe PinRef/KeyRef in sCard.CARDSESSION.AUTHSTATE und tCard.CARDSESSION.AUTHSTATE für adressierte Schlüssel wie in Zugriffsbedingung der Karten definiert vorhanden 8. Durchführung der Authentisierung gemäß Tabelle 66 mit Key-Referenzen gemäß Tabelle 67 9. Ergänze Target_CardSession.AUTHSTATE mit tKeyRef und Rolle aus sKeyRef
Varianten/Alternativen	(→8) Wenn der für die CA-Zertifikatsprüfung zu selektierende CVC-Root-Key auf der Zielkarte nicht vorhanden ist (Returncode des Karten-kommandos „MANAGE SECURITY ENVIRONMENT“ ist ’6A 88’), muss der Konnektor: • das oder die passenden Cross-CV-Zertifikate aus dem Truststore auswählen • mit dem Kartenkommando „PSO Verify Certificate“ jedes ausgewählte Cross-CV-Zertifikat durch die Zielkarte prüfen lassen. Dadurch wird der im Cross-CV-Zertifikat enthaltene öffentliche Schlüssel an die Zielkarte übertragen. Die Zielkarte speichert den

Element	Beschreibung
	darin enthaltenen neuen CVC-Root-Key. • neuen CVC-Root-Key auf der Zielkarte selektieren • Standardablauf der C2C-Authentisierung fortsetzen (→8) Wenn tCard.TYPE=EGK und AuthMode=gegenseitig, dann Echtheitsprüfung der eGK durch den Konnektor: a) Durchführen der Authentisierung gemäß Tabelle 66 mit Key-Referenzen gemäß Tabelle 67 aber mit AuthMode=einseitig b) Konnektor liest EF.C.CA_eGK.CS c) Konnektor ruft TUC_KON_037 "Prüfe Zertifikat" { C.CA_eGK.CS; PuK.RCA.CS.R2048 (für G1+) } bzw. { C.CA_eGK.CS .E256; PuK.RCA.CS.E.256 (für G2) } auf, um das CA-Zertifikat zu verifizieren. Der benötigte Root-Key für G2 befindet sich im Truststore des Konnektors. Der Root-Key für G1+ befindet sich in der Datei EF.PuK.RCA.CS.R2048 auf der gSMC-K. d) Konnektor extrahiert PuK.CA_eGK.CS aus EF.C.CA_eGK.CS e) Konnektor liest EF.C.eGK.AUT_CVC (G1+) bzw. EF.C.eGK.AUT_CVC.E256 (G2) f) Konnektor ruft TUC_KON_037 "Prüfe Zertifikat" {C.eGK.AUT_CVC; PuK.CA_eGK.CS} auf g) Konnektor erzeugt Zufallszahl h) Konnektor aktiviert den PrK.eGK.AUT_CVC (G1+) bzw. PrK.eGK.AUT_CVC.E256 (G2) und stellt abhängig von der Version der eGK den Algorithmus auf der eGK ein (MSE Set) i) Konnektor sendet Konkatenation aus Zufallszahl und CARD.ICCSN mit dem Befehl „PSO INTERNAL AUTHENTICATE" an die eGK j) Konnektor wertet das von der Karte erhaltene Chiffrat aus
Fehlerfälle	* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094 (→3) Eine Karte ist fremd reserviert, Fehlercode 4093 (→5) Zertifikat der Quellkarte fehlerhaft. Ausstellungsdatum liegt in der Zukunft; Fehlercode 4233 (→6) eGK G1+ ausgealtert, Fehlercode 4192 (→7) Nötige PIN, bzw. KeyRef ist nicht verifiziert, Fehlercode 4085 (→8) Je nachdem, welche Karte den Fehler verursachte, wird zum ursprünglichen Fehler (Fehlercode gemäß [gemSpec_COS]) im Error-Trace (welcher an erster Stelle im Falle des HBA z. B. bereits ein Fehler bezüglich PIN-Verifikation enthalten kann) noch ein weiterer mit Code 4056 oder 4057 hinzugefügt. Kann der Fehler nicht eindeutig einer der beiden Karten zugeordnet werden (z. B. Fehlen von Cross-CVCs im Konnektor oder inkompatible Versionen der Karten), so wird Error-Code 4048 verwendet. (→8) das benötigte Cross-CV-Zertifikat ist nicht vorhanden, Fehlercode 4228
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	Keine

Tabelle 66: TAB_KON_673 AuthMode für C2C

AuthMode	Definition des Ablaufs
einseitig	Externe oder Interne Authentisierung ([gemSpec_COS#15.1] oder [gemSpec_COS#15.2, passend zu den Zugriffsregeln der beteiligten CVC)
gegenseitig	Card-2-Card-Authentisierung ohne Sessionkey-Aushandlung ([gemSpec_COS#15.3])
gegenseitig+TC	Card-2-Card-Authentisierung mit Sessionkey-Aushandlung zur Etablierung eines Trusted Channels ([gemSpec_COS#15.4])

Tabelle 67: TAB_KON_674 Erlaubte Parameterkombinationen und resultierende CV-Zertifikate für C2C

Quellkarte	Zielkarte	AuthMode	sKeyRef	tKeyRef	Fachlicher UseCase
HBA oder SM-B	SM-B	einseitig	{HPC.AUTR_CVC.E256 und HPC.AUTR_CVC.R2048 | SMC.AUTR_CVC.E256 und SMC.AUTR_CVC.R2048}		Freischaltung SM-B
HBA oder SM-B	eGK G1+	einseitig	{HPC.AUTR_CVC.R2048 | SMC.AUTR_CVC.R2048}		Freischaltung eGK
HBA oder SM-B	eGK G1+	gegenseitig	{HPC.AUTR_CVC.R2048 | SMC.AUTR_CVC.R2048}	eGK.AUT_CVC.R2048	Freischaltung eGK mit Echtheitsprüfung eGK
HBA oder SM-B	eGK G2	einseitig	{HPC.AUTR_CVC.E256 | SMC.AUTR_CVC.E256}		Freischaltung eGK
HBA oder SM-B	eGK G2	gegenseitig	{HPC.AUTR_CVC.E256 | SMC.AUTR_CVC.E256}	eGK.AUT_CVC.E256	Freischaltung eGK mit Echtheitsprüfung eGK
SMC-K	HBA	gegenseitig +TC	SMC.AUTD_CVC.E256	HPC.AUTD_SUK_CVC.E256	DTBS-Übertragung bei QES
SMC-KT	HBA	gegenseitig +TC	SMC.AUTD_RPS_CVC.E256	HPC.AUTD_SUK_CVC.E256	Remote-PIN
SMC-KT	SM-B	gegenseitig +TC	SMC.AUTD_RPS_CVC.E	SMC.AUTD_RPE_CVC.E	Remote-PIN

Quellkarte	Zielkarte	AuthMode	sKeyRef	tKeyRef	Fachlicher UseCase
			256	256	

Tabelle 68: TAB_KON_535 Übersicht Fehler 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, 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 exklusiv verwendet
4094	Technical	Error	Timeout beim Kartenzugriff aufgetreten
4192	Security	Error	C2C mit eGK G1+ ab 01.01.2018 nicht mehr gestattet
4233	Security	Error	Ausstellungsdatum des Zertifikats liegt in der Zukunft;
4228	Technical	Error	Das benötigte Cross-CV-Zertifikat ist nicht vorhanden

⌫

4.1.5.4.8 TUC_KON_202 „LeseDatei“

⌦ **TIP1-A_4573 TUC_KON_202 „LeseDatei“**

Der Konnektor MUSS den technischen Use Case “LeseDatei“ gemäß TUC_KON_202 umsetzen.

Tabelle 69: TAB_KON_218 – TUC_KON_202 „LeseDatei“

Element	Beschreibung
Name	TUC_KON_202 „LeseDatei“
Beschreibung	Transparente Datei oder Teile davon lesen
Auslöser	• Aufruf des Use Case im Rahmen von technischen Use Cases der Basisdienste des Konnektors • Aufruf des Use Cases durch ein Fachmodul im Konnektor
Vorbedingungen	Die Karte MUSS den nötigen Sicherheitszustand für den Zugriff besitzen
Eingangsdaten	• CardSession • FileIdentifier • SFI (Short File Identifier, optional) • Verzeichnis der Karte, in dem sich die Datei befindet • Offset- und Längenangaben, um den Zugriff auf Teile einer Datei einzuschränken (optional)
Komponenten	Karte, Kartenterminal, Konnektor

Element	Beschreibung
Ausgangsdaten	Gelesene Daten
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lockbesitzt. 3. Prüfe PinRef / KeyRef in CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden 4. Selektiere Verzeichnis und Datei 5. Lese Daten über Kartenkommando “READ BINARY“ unter Berücksichtigung von Offset- und Längenangaben 6. Daten werden an den Aufrufer zurückgegeben
Varianten/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	* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094 (→2) Karte ist fremd reserviert, Fehlercode 4093 (→3) Nötige PIN ist nicht verifiziert, Fehlercode 4085 (->4-5) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode <Kartenfehlercode gemäß [gemSpec_COS]>
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 70: TAB_KON_536 Übersicht Fehler TUC_KON_202 „Lese Datei“

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

⌫

4.1.5.4.9 TUC_KON_203 „SchreibeDatei“

⌦ **TIP1-A_4574 TUC_KON_203 „SchreibeDatei“**

Der Konnektor MUSS den technischen Use Case “SchreibeDatei“ gemäß TUC_KON_203 umsetzen.

Tabelle 71: TAB_KON_219 – TUC_KON_203 „SchreibeDatei“

Element	Beschreibung
Name	TUC_KON_203 „SchreibeDatei“

Element	Beschreibung
Beschreibung	Daten in transparente Datei schreiben
Auslöser	• Aufruf durch Fachmodul
Vorbedingungen	Die Karte MUSS den nötigen Sicherheitszustand für den Zugriff besitzen
Eingangsdaten	• CardSession • FileIdentifier • SFI (Short File Identifier, optional) • Verzeichnis der Karte, in dem sich die Datei befindet • Offset- und Längenangaben, um den Zugriff auf Teile einer Datei einzuschränken (optional) • Zu schreibende Daten
Komponenten	Karte, Kartenterminal, Konnektor
Ausgangsdaten	Keine
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Prüfe Card.TYPE <> KVK 3. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lockbesitzt. 4. Prüfe PinRef / KeyRef in CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden 5. Selektiere Verzeichnis und Datei 6. Schreibe Daten über Kartenkommando “UPDATE BINARY“ unter Berücksichtigung von Offset- und Längenangaben
Varianten/Alternativen	keine
Fehlerfälle	* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094 (→2) Scheibzugriff auf KVK nicht gestattet, Fehlercode 4085 (→3) Karte ist fremd reserviert, Fehlercode 4093 (→4) Nötige PIN ist nicht verifiziert, Fehlercode 4085 (→5) Verzeichnis oder Datei existiert nicht, Fehlercode 4087 (->4-5) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode <Kartenfehlercode gemäß [gemSpec_COS]> (→6) Ausgewählte Datei ist nicht transparent, Fehlercode 4089
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 72: TAB_KON_537 Übersicht Fehler 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
4087	Technical	Error	Datei nicht vorhanden

Fehlercode	ErrorType	Severity	Fehlertext
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

⌫

4.1.5.4.10 TUC_KON_209 „LeseRecord“

⌦ **TIP1-A_4575 TUC_KON_209 „LeseRecord“**

Der Konnektor MUSS den technischen Use Case “LeseRecord“ gemäß TUC_KON_209 umsetzen.

Tabelle 73: TAB_KON_538 – TUC_KON_209 „LeseRecord“

Element	Beschreibung
Name	TUC_KON_209 „LeseRecord"
Beschreibung	Daten aus linearer Datei lesen
Auslöser	• Aufruf durch Fachmodul
Vorbedingungen	Die Karte MUSS den nötigen Sicherheitszustand für den Zugriff besitzen
Eingangsdaten	• CardSession • FileIdentifier • SFI (Short File Identifier, optional) • Verzeichnis der Karte, in dem sich die Datei befindet • RecordNummer
Komponenten	Karte, Kartenterminal, Konnektor
Ausgangsdaten	Inhalt des Records
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lockbesitzt 3. Prüfe PinRef / KeyRef in CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden 4. Selektiere Verzeichnis, Datei und Record 5. Lese Daten über Kartenkommando “READ RECORD“ unter Berücksichtigung von Recordnummer 6. Rückgabe der Daten an den Aufrufer
Varianten/Alternativen	keine

Element	Beschreibung
Fehlerfälle	* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094 (→2) Karte ist fremd reserviert, Fehlercode 4093 (→3) Nötige PIN ist nicht verifiziert, Fehlercode 4085 (→4) Verzeichnis oder Datei oder Record existiert nicht, Fehlercode 4087 (→5) Wenn Karte WrongFileType liefert, Fehlercode 4089 (→4-5) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode <Kartenfehlercode gemäß [gemSpec_COS]>.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 74: TAB_KON_539 Übersicht Fehler TUC_KON_209 „Lese Record“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4085	Security	Error	Zugriffsbedingungen nicht erfüllt
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

⌫

4.1.5.4.11 TUC_KON_210 „SchreibeRecord“

⌦ **TIP1-A_4576 TUC_KON_210 „SchreibeRecord“**

Der Konnektor MUSS den technischen Use Case “SchreibeRecord“ gemäß TUC_KON_210 umsetzen.

Tabelle 75: TAB_KON_224 – TUC_KON_210 „SchreibeRecord“

Element	Beschreibung
Name	TUC_KON_210 „SchreibeRecord"
Beschreibung	Daten in lineare Datei schreiben
Auslöser	• Aufruf durch Fachmodul
Vorbedingungen	Die Karte MUSS den nötigen Sicherheitszustand für den Zugriff besitzen

Element	Beschreibung
Eingangsdaten	• CardSession • FileIdentifier • SFI (Short File Identifier, optional) • Verzeichnis der Karte, in dem sich die Datei befindet • Recordnummer • Zu schreibende Daten
Komponenten	Karte, Kartenterminal, Konnektor
Ausgangsdaten	keine
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Prüfe Card.TYPE <> KVK 3. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lockbesitzt. 4. Prüfe PinRef / KeyRef in CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden 5. Selektiere Verzeichnis, Datei und Record 6. Schreibe Daten über Kartenkommando “UPDATE BINARY“ unter Berücksichtigung von Recordnummer
Varianten/Alternativen	keine
Fehlerfälle	* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094 (→2) Scheibzugriff auf KVK nicht gestattet, Fehlercode 4085 (→3) Karte ist fremd reserviert, Fehlercode 4093 (→4) Nötige PIN ist nicht verifiziert, Fehlercode 4085 (→5) Verzeichnis, Datei oder Record existiert nicht, Fehlercode 4087 (→4-5) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode <Kartenfehlercode gemäß [gemSpec_COS]>
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 76: TAB_KON_540 Übersicht Fehler TUC_KON_210 „Schreibe Record“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4094	Technical	Error	Timeout beim Kartenzugriff aufgetreten
4086	Technical	Error	Verzeichnis deaktiviert
4087	Technical	Error	Datei nicht vorhanden
4091	Technical	Error	Record nicht vorhanden
4088	Technical	Error	Datensatz zu groß
4085	Security	Error	Zugriffsbedingungen nicht erfüllt
4093	Technical	Error	Karte wird in einer anderen Kartensitzung exklusiv

Fehlercode	ErrorType	Severity	Fehlertext
			verwendet

⌫

4.1.5.4.12 TUC_KON_214 „FügeHinzuRecord“

⌦ **TIP1-A_4577 TUC_KON_214 „FügeHinzuRecord“**

Der Konnektor MUSS den technischen Use Case “FügeHinzuRecord“ gemäß TUC_KON_214 umsetzen.

Tabelle 77: TAB_KON_228 – TUC_KON_214 „FügeHinzuRecord“

Element	Beschreibung
Name	TUC_KON_214 „FuegeHinzuRecord"
Beschreibung	Daten in lineare Datei anfügen
Auslöser	• Aufruf durch Fachmodul • TUC_KON_006
Vorbedingungen	Die Karte MUSS den nötigen Sicherheitszustand für den Zugriff besitzen
Eingangsdaten	• CardSession • FileIdentifier • SFI (Short File Identifier, optional) • Verzeichnis der Karte, in dem sich die Datei befindet • Zu schreibende Daten
Komponenten	Karte, Kartenterminal, Konnektor
Ausgangsdaten	keine
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Prüfe Card.TYPE <> KVK 3. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lockbesitzt. 4. Prüfe PinRef / KeyRef in CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden 5. Selektiere Verzeichnis, Datei und Record 6. Schreibe Daten über Kartenkommando “APEND RECORD“
Varianten/Alternativen	keine
Fehlerfälle	* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094 (→2) Schreibzugriff auf KVK nicht gestattet, Fehlercode 4085 (→3) Karte ist fremd reserviert, Fehlercode 4093 (→4) Nötige PIN ist nicht verifiziert, Fehlercode 4085 (→5) Verzeichnis, Datei oder Record existiert nicht, Fehlercode 4087 (→5-6) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode <Kartenfehlercode gemäß [gemSpec_COS]>

Element	Beschreibung
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 78: TAB_KON_541 Übersicht Fehler 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
4087	Technical	Error	Datei nicht vorhanden
4093	Technical	Error	Karte wird in einer anderen Kartensitzung exklusiv verwendet
4094	Technical	Error	Timeout beim Kartenzugriff aufgetreten

⌫

4.1.5.4.13 TUC_KON_215 „SucheRecord“

⌦ TIP1-A_4578 TUC_KON_215 „SucheRecord“

Der Konnektor MUSS den technischen Use Case “SucheRecord“ gemäß TUC_KON_215 umsetzen.

Tabelle 79: TAB_KON_229 – TUC_KON_215 „SucheRecord“

Element	Beschreibung
Name	TUC_KON_215 „SucheRecord"
Beschreibung	Daten in linearer Datei suchen
Auslöser	• Aufruf durch Fachmodul
Vorbedingungen	Die Karte MUSS den nötigen Sicherheitszustand für den Zugriff besitzen
Eingangsdaten	• CardSession • FileIdentifier • SFI (Short File Identifier, optional) • Verzeichnis der Karte, in dem sich die Datei befindet • SuchMuster • Recordnummer bei der Suche beginnen soll (optional)
Komponenten	Karte, Kartenterminal, Konnektor
Ausgangsdaten	Liste: Nummern der Records, die dem SuchMuster entsprechen

Element	Beschreibung
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lockbesitzt. 3. Prüfe PinRef / KeyRef in CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden 4. Selektiere Verzeichnis, Datei und Record 5. Sende Kartenkommando “SEARCH RECORD“ mit SuchMuster unter Berücksichtigung von Recordnummer 6. Liefere Antwort der Karte zurück
Varianten/Alternativen	Keine
Fehlerfälle	* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094 (→2) Karte ist fremd reserviert, Fehlercode 4093 (→3) Nötige PIN ist nicht verifiziert, Fehlercode 4085 (→4) Verzeichnis, Datei oder Record existiert nicht, Fehlercode 4087 (→4-5) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode <Kartenfehlercode gemäß [gemSpec_COS]>
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 80: TAB_KON_542 Übersicht Fehler TUC_KON_215 „SucheRecord“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4094	Technical	Error	Timeout beim Kartenzugriff aufgetreten
4086	Technical	Error	Verzeichnis deaktiviert
4087	Technical	Error	Datei nicht vorhanden
4085	Security	Error	Zugriffsbedingungen nicht erfüllt
4093	Technical	Error	Karte wird in einer anderen Kartensitzung exklusiv verwendet

⌫

4.1.5.4.14 TUC_KON_018 „eGK-Sperrung prüfen“

⌦ **TIP1-A_4579 TUC_KON_018 „eGK-Sperrung prüfen“**

Der Konnektor MUSS den technischen Use Case “eGK-Sperrung prüfen“ gemäß TUC_KON_018 umsetzen.

Tabelle 81: TAB_KON_110 - TUC_KON_018 „eGK-Sperrung prüfen“

Element	Beschreibung

Element	Beschreibung
Name	TUC_KON_018 „eGK-Sperrung prüfen“
Beschreibung	Es wird geprüft, ob das Aut-Zertifikat im DF.ESIGN gültig ist und DF.HCA (Health Care Application) der eGK gesperrt ist.
Auslöser	Aufruf durch Fachmodul im Konnektor
Vorbedingungen	keine
Eingangsdaten	• CardSession
Komponenten	Konnektor, Kartenterminal, eGK
Ausgangsdaten	Karte gesperrt: ja/nein Status: • DF.HCA gesperrt: ja/nein • Ergebnis der Offline-Prüfung des C.CH.AUT-Zertifikats: gültig/ungültig • Sperrstatus des C.CH.AUT-Zertifikats: gut/gesperrt/nicht ermittelbar
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lock besitzt. 3. Selektiere DF.HCA : a) Wenn die Karte ’90 00’ zurückmeldet, war das Selektieren möglich: DF.HCA gesperrt = nein b) In allen anderen Fällen war das Selektieren nicht fehlerfrei möglich: DF.HCA gesperrt = ja 4. Rufe Cert = TUC_KON_216 „LeseZertifikat“ {CardSession; C.CH.AUT} 5. Bestimme per Aufruf von TUC_KON_037 “Prüfe Zertifikat“ das Ergebnis des Offline-Prüfung des C.CH.AUT-Zertifikats (gültig/ungültig) sowie den Sperrstatus des C.CH.AUT-Zertifkats (gut/gesperrt/nicht ermittelbar). 6. Die Karte ist gesperrt = ja, wenn DF.HCA gesperrt = ja oder Ergebnis der Offline-Prüfung des C.CH.AUT-Zertifik. = ungültig oder Sperrstatus des C.CH.AUT-Zertifkats = gesperrt. In allen anderen Fällen ist die Karte gesperrt = nein.
Varianten/Alternativen	keine
Fehlerfälle	(→2) Karte ist fremd reserviert, Fehlercode 4093
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 82: TAB_KON_239 Übersicht Fehler TUC_KON_018 „eGK-Sperrung prüfen“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere			

Fehlercode	ErrorType	Severity	Fehlertext
Fehlercodes auftreten:			
4093	Technical	Error	Karte wird in einer anderen Kartensitzung exklusiv verwendet

4.1.5.4.15 TUC_KON_006 „Datenzugriffsaudit eGK schreiben“

TIP1-A_4580 TUC_KON_006 „Datenzugriffsaudit eGK schreiben“

Der Konnektor MUSS den technischen Use Case “Datenzugriffsaudit eGK schreiben“ gemäß TUC_KON_006 umsetzen.

Tabelle 83: 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	• CardSession (einer eGK) • Sup.CardSession (HBA/SMC, die für den eGK-Zugriff verwendet wird) • DATA.TYP (siehe gem_Spec_Karten_Fach_TIP#4.1 – Tabelle 11: Tab_Karten_Fach_TIP_010 ab_StrukturEF.Logging – Struktur der Rekords der Datei EF.Logging] • Type of Access (siehe [gemSpec_Karten_Fach_TIP#4.1] – Tabelle 11: Tab_Karten_Fach_TIP_010 ab_StrukturEF.Logging – Struktur der Rekords der Datei EF.Logging)
Komponenten	eGK, HBA/SMC, Konnektor, Kartenterminal
Ausgangsdaten	Keine
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Prüfe Card.TYPE = EGK 3. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lockbesitzt. 4. Wenn KeyRef in CARDSESSION.AUTHSTATE für DF.HCA.EF.LOGGING nicht mit passender Role vorhanden: Rufe TUC_CON_005 „Card-to-Card authentisieren“ Sup.CardSession 5. Erzeuge Logging.Daten nach [gem_Spec_Karten_Fach_TIP#4.1 – Tabelle 11: Tab_Karten_Fach_TIP_010 ab_StrukturEF.Logging – Struktur der Rekords der Datei EF.Logging] 6. Rufe TUC_KON_214 „FügeHinzuRecord“ {CardSession; DF.HCA.EF.LOGGING ;; Logging.Daten}
Varianten/Alternativen	Keine
Fehlerfälle	(→2) Protokoll nur für eGK gestattet, Fehlercode 4251 (→3) Karte ist fremd reserviert, Fehlercode 4093
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 84: TAB_KON_238 Übersicht Fehler TUC_KON_006 „Datenzugriffsaudit eGK 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

⌫

4.1.5.4.16 TUC_KON_218 „Signiere“

⌦ TIP1-A_4581 TUC_KON_218 „Signiere“

Der Konnektor MUSS den technischen Use Case “Signiere“ gemäß TUC_KON_218 umsetzen.

Tabelle 85: 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	• Aufruf der Operation ChangePin des CardService durch das Clientsystem. • Aufruf durch Fachmodul
Vorbedingungen	Zugriffsbedingung für referenzierten Schlüssel MUSS erfüllt sein
Eingangsdaten	• CardSession • PinRef • KeyRef • AlgorithmusID • DTBS (Zu signierende Daten)
Komponenten	Karte, Kartenterminal, Konnektor
Ausgangsdaten	CHIFFRAT (Signierte Daten)
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lockbesitzt. 3. Prüfe PinRef in CARDSESSION.AUTHSTATE vorhanden: 4. Setze KeyRef und AlgorithmusID der Karte 5. Sende „PSO: COMPUTE DS“ mit DTBS an Karte 6. Gebe CHIFFRAT an Aufrufer zurück
Varianten/Alternativen	Keine
Fehlerfälle	* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094

Element	Beschreibung
	(→2) Karte ist fremd reserviert, Fehlercode 4093 (→3) Nötige PIN ist nicht verifiziert, Fehlercode 4085 (→5) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode <Kartenfehlercode gemäß [gemSpec_COS]>
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 86: TAB_KON_543 Übersicht Fehler 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

⌫

4.1.5.4.17 TUC_KON_219 „Entschlüssele“

⌦ TIP1-A_4582 TUC_KON_219 „Entschlüssele“

Der Konnektor MUSS den technischen Use Case “Entschlüssele“ gemäß TUC_KON_219 umsetzen.

Tabelle 87: 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	• Aufruf durch Fachmodul
Vorbedingungen	Zugriffsbedingung für referenzierten Schlüssel muss erfüllt sein
Eingangsdaten	• CardSession • PinRef • KeyRef • AlgorithmusID • Zu entschlüsselnde Daten (Chiffrat)
Komponenten	Karte(n), Kartenterminal, Konnektor
Ausgangsdaten	Entschlüsselte Daten
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession)

Element	Beschreibung
	2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lockbesitzt. 3. Prüfe PinRef in CARDSESSION.AUTHSTATE vorhanden: 4. Selektiere DF, in dem der private Schlüssel (KeyRef) liegt, falls er noch nicht selektiert ist. 5. Setze Schlüssel (KeyRef) und AlgorithmusID. 6. Sende Chiffrat mittels Kommandos PSO: DECIPHER. 7. Sende entschlüsselten Daten an Aufrufer
Varianten/Alternativen	Keine
Fehlerfälle	* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094 (→2) Karte ist fremd reserviert, Fehlercode 4093 (→3) Nötige PIN ist nicht verifiziert, Fehlercode 4085 (→5) Schlüssel nicht vorhanden, Fehlercode 4079 (→ 6) Fehler im Chiffrat: Fehlercode 4069 (→4, 6) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode <Kartenfehlercode gemäß [gemSpec_COS]>
Varianten/Alternativen	Keine
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 88: TAB_KON_210 Übersicht Fehler 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 Chiffrat 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

⌫

4.1.5.4.18 TUC_KON_200 „SendeAPDU“

⌦ **TIP1-A_4583 TUC_KON_200 „SendeAPDU“**

Der Konnektor MUSS den technischen Use Case “SendeAPDU“ gemäß TUC_KON_200 umsetzen.

Tabelle 89: 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. Kartenterminal und das Empfangen der Antwort.
Auslöser	• Aufruf durch Fachmodul
Vorbedingungen	Zugriffsbedingungen für das Kommando müssen in der Karte erfüllt sein und Karte muss für exklusiven Zugriff reserviert worden sein
Eingangsdaten	• CARDSESSION, alternativ CtID • APDU Parameter{CLA, Ins, P1,P2, Data (optional) Le(optional)
Komponenten	Karte(n), Kartenterminal, Konnektor
Ausgangsdaten	• Antwort (Response-APDU) der Chipkarte
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Prüfe, dass der Aufrufer für die zur CardSession gehörenden Karte ein Lock hat. 3. Kommando-APDU wird über das Kartenterminal an die Zielkarte gesendet 4. die Antwort (Response-APDU) der Zielkarte wird an den Aufrufer zurückgegeben.
Varianten/Alternativen	• Soll Secure Messaging verwendet werden, MUSS vorher TUC_KON_023 „Karte reservieren“ aufgerufen werden • Wenn es sich um ein reines Kartenterminalkommando handelt, d. h. ohne Beteiligung einer Karte, wird als Parameter das Kartenterminal per CtID adressiert, Schritte 1 und 2 entfallen, und die Antwort wird in Schritt 4 direkt vom Kartenterminal erzeugt.
Fehlerfälle	* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094 (→2) Der Aufrufer ist nicht im Besitz des Karten-Locks, Fehlercode 4232 (→3) Kommunikationsfehler mit dem Kartenterminal: Fehlercode 4044. (→3) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode <Kartenfehlercode gemäß [gemSpec_COS]>
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 90: TAB_KON_216 Übersicht Fehler 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
4232	Technical	Error	Der Aufrufer besitzt nicht das Karten-Lock
4094	Technical	Error	Timeout beim Kartenzugriff aufgetreten

⌫

4.1.5.4.19 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 91: 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	• CtID • SlotNo Alternativ: • CardSession
Komponenten	Karte, Kartenterminal, Konnektor
Ausgangsdaten	Keine
Standardablauf	1. Setze Card = CM_CARD_LIST(CtID, SlotNo) bzw. Card = CM_CARD_LIST(CardSession) 2. Der Konnektor prüft, dass entweder die Karte nicht reserviert ist oder der Aufrufer im Besitz des Karten-Locks ist. 3. Breche eventuell parallel laufenden TUC_KON_005 ab 4. Sende reset an Karte 5. Lösche alle Sicherheitszustände aus `CARDSESSION.AUTHSTATE` und den Inhalt von `CARDSESSION.AUTHBY`.
Varianten/Alternativen	Keine
Fehlerfälle	* Karte antwortet nicht innerhalb von CARD_TIMEOUT_CARD Sekunden, Fehlercode 4094 (→2) Der Aufrufer ist nicht im Besitz des Karten-Locks, Fehlercode 4232 (→4) Karte antwortet mit einer spezifischen Fehlermeldung, Fehlercode <Kartenfehlercode gemäß [gemSpec_COS]>
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 92: TAB_KON_544 Übersicht Fehler 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

⌫

4.1.5.4.20 TUC_KON_216 „LeseZertifikat“

⌦ **TIP1-A_4585 TUC_KON_216 „LeseZertifikat“**

Der Konnektor MUSS den technischen Use Case “LeseZertifikat“ gemäß TUC_KON_216 umsetzen.

Tabelle 93: 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	• Aufruf der Operation ReadCardCertificate des Zertifikatsdienstes durch das Clientsystem. • Aufruf durch Fachmodul • Aufruf im Rahmen von technischen Use Cases der Basisdienste des Konnektors
Vorbedingungen	
Eingangsdaten	• CardSession • FileIdentifier
Komponenten	Karte, Kartenterminal, Konnektor
Ausgangsdaten	• Zertifikat
Standardablauf	1. Ermittle Card = CM_CARD_LIST(CardSession) 2. Prüfe, dass der Karte entweder kein Lock zugeordnet ist oder der Aufrufer das Karten-Lockbesitzt. 3. Prüfe CARDSESSION.AUTHSTATE für adressierte Datei wie in Zugriffsbedingung der Karte definiert vorhanden 4. Rufe TUC_KON_202 „LeseDatei“ {CardSession, FileIdentifier} 5. Das Zertifikat wird an den Aufrufer zurückgegeben.
Varianten/Alternativen	Keine
Fehlerfälle	(→2) Karte ist fremd reserviert, Fehlercode 4093
Nichtfunktionale Anforderungen	
Zugehörige Diagramme	

Tabelle 94: TAB_KON_209 Übersicht Fehler 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

⌫

4.1.5.5 Operationen an der Außenschnittstelle

⌦ **TIP1-A_4586 Basisanwendung Kartendienst**

Der Konnektor MUSS für Clients eine Basisanwendung Kartendienst mit den Operationen AuthorizeSMC, VerifyPin, ChangePin, UnblockPin, GetPinStatus an der Außenschnittstelle anbieten.

Tabelle 95: TAB_KON_038 Basisanwendung Karten- und Kartenterminaldienst

Name	`CardService`	
Version (KDV)	wird im Produktsteckbrief des Konnektors definiert	
Namensraum	Siehe Anhand D	
Namensraum-Kürzel	`CARD für Schema und CARDW für WSDL`	
Operationen	**Name**	**Kurzbeschreibung**
	`AuthorizeSMC`	SM-B mit HBA freischalten
	`VerifyPin`	PIN prüfen
	`ChangePin`	PIN ändern
	`UnblockPin`	PIN entsperren
	`GetPinStatus`	PIN-Status ermitteln
WSDL	`CardService.wsdl`	
Schema	`CardService.xsd`	

⌫

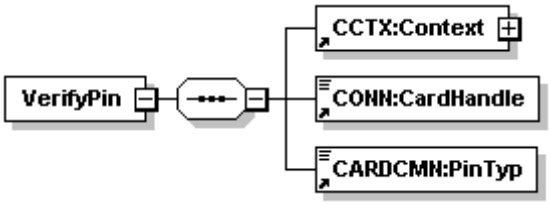
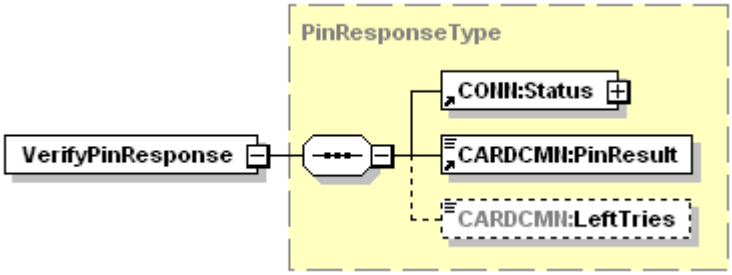
4.1.5.5.1 VerifyPin

⌦ **TIP1-A_4587 Operation VerifyPin**

Der Konnektor MUSS an der Außenschnittstelle eine Operation VerifyPin, wie in Tabelle 96: TAB_KON_047 Operation VerifyPin beschrieben, anbieten.

Tabelle 96: TAB_KON_047 Operation VerifyPin

Name	**`VerifyPin`**
Beschreibung	Stößt die sichere Eingabe einer PIN am PIN-Pad des Kartenterminals für eine

	Karte an. Das Ergebnis der PIN-Prüfung gibt Auskunft darüber, ob die PIN richtig oder falsch war oder aufgrund zu vieler Fehlversuche blockiert ist. 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. 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 die PIN-Eingabe ergibt sich dabei aus der im Aufrufkontext angegebenen Mandanten-ID und Arbeitsplatz-ID Diese Operation entspricht dem Aufruf von 4.1.5.4.2 TUC_KON_012 „PIN verifizieren“. Dort sind auch die Display Messages definiert, die bei PIN-Eingabe am Kartenterminal anzuzeigen sind (Tabelle 52: 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.
Aufrufparameter	
	Name: `Context` — Beschreibung: MandantId, CsId, WorkplaceId verpflichtend; UserId verpflichtend für HBAx
	Name: `CardHandle` — Beschreibung: Adressiert die Karte, für die die PIN verifiziert werden soll. 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.
	Name: `PinTyp` — Beschreibung: Gibt an, welche PIN der Karte verifiziert werden soll. Erlaubte Belegung von PINTYP in Abhängigkeit der durch Cardhandle referenzierten Karte: • HBAx: PIN.CH • SM-B: PIN.SMC
Rückgabe	

	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 `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`	PIN ist mit Transportschutz versehen
	`LeftTries`	Im Falle von `Result=REJECTED` wird hier die Anzahl der verbleibenden möglichen Versuche zurückgegeben.	
Vorbedingung	Keine		
Nachbedingung	keine		

Der Ablauf der Operation VerifyPin ist in Tabelle 97: TAB_KON_738 Ablauf VerifyPin beschrieben.

Tabelle 97: 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 {$context.mandantId; $context.clientsystemId; $context.workplaceId; $context.userId; $cardHandle} 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 { MandantId, CsId, CardHandle, UserId }
4.	TUC_KON_012 „PIN verifizieren“	Verifiziere PIN über TUC_KON_012 { CardSession, WorkplaceID, PinRef(PinType), “” (Leerstring), “mandatorisch”}
5.	Verifikationsergebnis auswerten	Wenn TUC_KON_012 mit Fehler 4065 abbricht, wird dies als erfolgreicher Operationsdurchlauf mit PinResult= TRANSPORT_PIN abgefangen. Wenn TUC_KON_012 den Returncode BLOCKED liefert, wird dies als erfolgreicher Operationsdurchlauf mit PinResult=

Nr.	Aufruf Technischer Use Case oder Interne Operation	Beschreibung
		NOWBLOCKED zurückgegeben. Wenn TUC_KON_012 mit Fehler 4063 abbricht, wird dies als erfolgreicher Operationsdurchlauf mit PinResult= WASBLOCKED zurückgegeben

Tabelle 98: TAB_KON_545 Übersicht Fehler Operation „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.

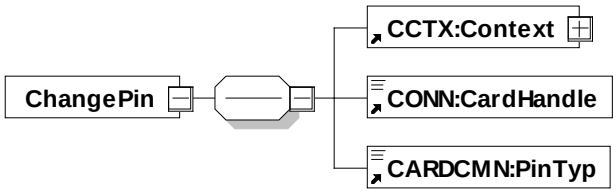
⌫

4.1.5.5.2 ChangePin

⌦ TIP1-A_4588 Operation ChangePin

Der Konnektor MUSS an der Außenschnittstelle eine Operation ChangePin, wie in Tabelle 99: TAB_KON_049 Operation ChangePin beschrieben, anbieten.

Tabelle 99: TAB_KON_049 Operation ChangePin

Name	`ChangePin`	
Beschreibung	Ändert eine PIN einer Karte. Alte und neue PIN werden dabei 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. Diese Operation entspricht dem Aufruf TUC_KON_019 „PIN ändern“.	
Aufrufparameter		
	Name	Beschreibung
	`Context`	MandantId, CsId, WorkplaceId verpflichtend; UserId optional
	`CardHandle`	Adressiert die Karte, für die die PIN geändert werden soll. Unterstützt werden die Kartentypen EGK, HBAx und SM-B. Wird

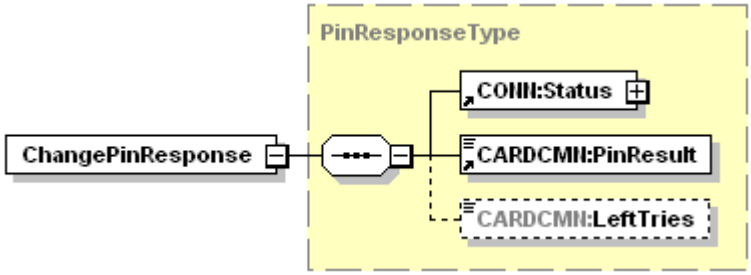
		die Operation mit einem nicht unterstützten Kartentypen aufgerufen, so MUSS der Konnektor die Bearbeitung mit dem Fehler 4209 abbrechen.
	`PinTyp`	Gibt an, welche PIN der Karte geändert werden soll. Erlaubte Belegung von PinTyp in Abhängigkeit der durch CardHandle referenzierten Karte: • eGK: PIN.CH • HBAx: PIN.CH • HBAx: PIN.QES • SM-B: PIN.SMC
Rückgabe		
	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 oder NewPIN1<>NewPIN2 oder NewPIN hatte falsche Länge. Die Anzahl der verbleibenden Versuche ist im Element `LeftTries`
		`WASBLOCKED` / PIN war zum Aufrufzeitpunkt bereits gesperrt
		`NOWBLOCKED` / PIN ist durch aktuellen Fehlversuch gesperrt
Vorbedingung	Ist CardType=eGK und PinTyp=PIN.CH und die Kartengeneration G1+, dann MUSS vorher * im Fall der Leer-Pin_1 eine C2C-Authentisierung mit einem der Profile C{2,3,4,8,9,10} erfolgen (Ablauf dann ohne Old PIN Eingabe) * im Fall nicht Leer-Pin_1 eine C2C-Authentisierung mit einem der Profile C{1,2,3,4,8,9,10} erfolgen (Ablauf dann mit Old PIN Eingabe)	
Nachbedingung	keine	

Tabelle 100: TAB_KON_546 Ablauf ChangePin

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 {$context.mandantId; $context.clientsystemId; $context.workplaceId; $context.userId; $cardHandle} 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 { MandantId, CsId, CardHandle, UserId }
4.	TUC_KON_019 „PIN ändern“	Ändere PIN über TUC_KON_019 { CardSession, WorkplaceId, PinRef(PinType)}

Tabelle 101: TAB_KON_547 Übersicht Fehler Operation „ChangePin“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4072	Technical	Error	Ungültige PIN-Referenz PinRef
4074	Technical	Error	Formatfehler der übergebenen alten PIN oder fehlende alte PIN.
4075	Technical	Error	Formatfehler in der übergebenen neuen PIN
4209	Technical	Error	Kartentyp %CardType% wird durch diese Operation nicht unterstützt.

⌫

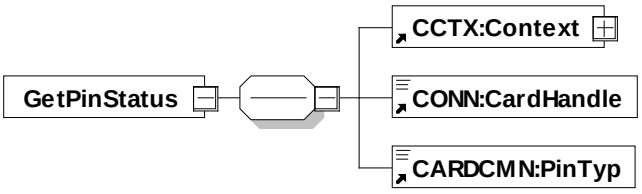
4.1.5.5.3 GetPinStatus

⌦ TIP1-A_4589 Operation GetPinStatus

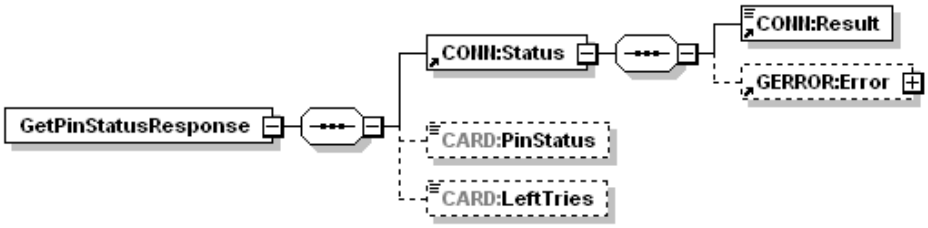
Der Konnektor MUSS an der Außenschnittstelle eine Operation GetPinStatus, wie in Tabelle 102: TAB_KON_051 Operation GetPinStatus beschrieben, anbieten.

Tabelle 102: TAB_KON_051 Operation GetPinStatus

Name	`GetPinStatus`
Beschreibung	Diese Operation gibt Auskunft über den PIN-Zustand einer Karte. Für transportgeschützte PIN gibt die Operation die Art des Transportschutzes an. 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.
Aufrufparameter	

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. Erlaubte Belegung von PINTYP in Abhängigkeit der durch Cardhandle referenzierten Karte: • eGK: PIN.CH • HBAx: PIN.CH, PIN.QES • SM-B: PIN.SMC

Rückgabe	

Name	Beschreibung
`Status`	Enthält den Ausführungsstatus der Operation siehe 3.5.2
`PinStatus`	Status der PIN. Die folgenden Werte sind verpflichtend: VERIFIED: Bereits verifiziert (in `CARDSESSION.AUTHSTATE` vorhanden) TRANSPORT_PIN Transport-PIN EMPTY_PIN: Leer-PIN BLOCKED: gesperrt VERIFIABLE Echt-PIN oder NoError (Verifikation nicht erforderlich)
`LeftTries`	Bei einer Echt-PIN wird hier die Anzahl der verbleibenden möglichen Versuche für die Verifikation der PIN zurückgegeben, bei einer gesperrten PIN 0.

Vorbedingung	keine

Nachbedingung	keine

Tabelle 103: TAB_KON_548 Ablauf GetPinStatus

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 {$context.mandantId; $context.clientsystemId; $context.workplaceId; $context.userId; $cardHandle} 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 { MandantId, CsId, CardHandle, UserId }
4.	TUC_KON_022 „Liefere PIN-Status“	Emittle PIN-Status über TUC_KON_022 { CardSession, PinRef(PinType) }

Tabelle 104: TAB_KON_549 Übersicht Fehler Operation „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
4044	Technical	Error	Fehler beim Zugriff auf das Kartenterminal
4045	Technical	Error	Fehler beim Zugriff auf die Karte
4046	Technical	Error	Kartenapplikation existiert nicht
4047	Technical	Error	Karten-Handle ungültig
4072	Technical	Error	Ungültige PIN-Referenz `PinRef`
4209	Technical	Error	Kartentyp %CardType% wird durch diese Operation nicht unterstützt.

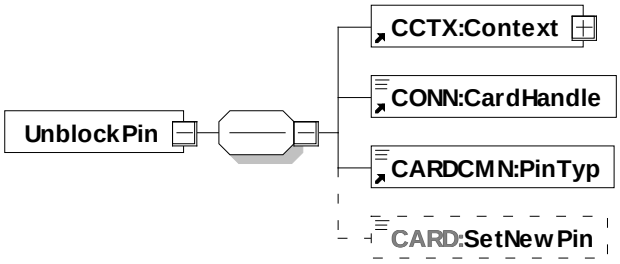
⌫

4.1.5.5.4 UnblockPin

TIP1-A_4590 Operation UnblockPin

Der Konnektor MUSS an der Außenschnittstelle eine Operation UnblockPin, wie in Tabelle 105: TAB_KON_053 Operation UnblockPin beschrieben, anbieten.

Tabelle 105: TAB_KON_053 Operation UnblockPin

Name	`UnblockPin`	
Beschreibung	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. Um diese Aktion durchführen zu können, muss eine PUK (auch als Resetting Code bezeichnet) präsentiert werden. 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. Diese Operation entspricht dem Aufruf von TUC_KON_021 „PIN entsperren“.	
Aufrufparameter		
	Name	Beschreibung
	Context	MandantId, CsId, WorkplaceId verpflichtend; UserId
	`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. Erlaubte Belegung von PINTYP in Abhängigkeit der durch Cardhandle referenzierten Karte: eGK: PIN.CH HBAx: PIN.CH HBAx: PIN.QES SM-B: PIN.SMC
	`SetNewPin`	Dieses Flag zeigt an, ob eine neue PIN gesetzt werden soll. Wird dieses Flag nicht angegeben, so wird FALSE angenommen. Das Flag ist notwendig, um bei Eingabe am PIN-Pad eindeutig

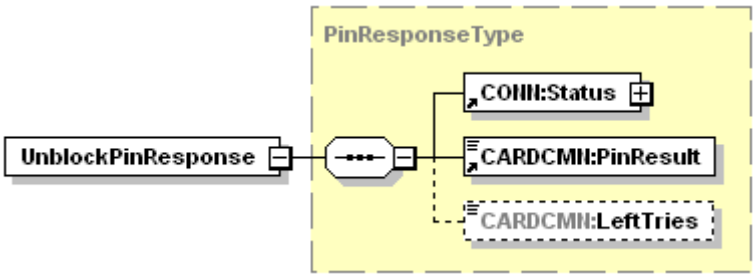
		festzulegen, ob eine neue PIN gesetzt werden soll.	
Rückgabe			
	Name	Beschreibung	
	`LeftTries`	Im Falle von `REJECTED` wird hier die Anzahl der verbleibenden möglichen Versuche für die Eingabe der PUK zurückgegeben.	
	`Status`	Enthält den Ausführungsstatus der Operation siehe 3.5.2	
	`PinResult`	Wert	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
Vorbedingungen	keine		
Nachbedingungen	keine		

Tabelle 106: 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 „Prüfe Zugriffs-berechtigung“	Prüfung der Zugriffsberechtigung durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; $context.userId; $cardHandle}
3.	TUC_KON_026 „Liefere CardSession“	Ermittle CardSession über TUC_KON_026 { MandantId, CsId, CardHandle, UserId }
4.	TUC_KON_021 „PIN entsperren“	Rücksetzen des Fehlbedienungszählers über TUC_KON_021 { CardSession, WorkplaceID, PinRef(PinType), SetNewPin }

Tabelle 107: TAB_KON_551 Übersicht Fehler Operation „UnblockPin“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4209	Technical	Error	Kartentyp %CardType% wird durch diese Operation nicht unterstützt.

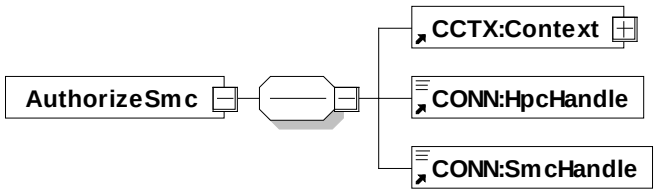
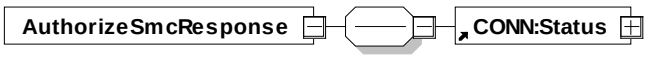
⌫

4.1.5.5.5 AuthorizeSMC

⌦ TIP1-A_4591 Operation AuthorizeSMC

Der Konnektors MUSS an der Außenschnittstelle eine Operation AuthorizeSMC, wie in Tabelle 108: TAB_KON_055 Operation AuthorizeSMC beschrieben, anbieten.

Tabelle 108: TAB_KON_055 Operation AuthorizeSMC

Name	`AuthorizeSMC`	
Beschreibung	Führt eine einseitige C2C-Authentifizierung ohne Aufbau eines Trusted Channel (SM) zwischen HBA und SMC durch. Durch diesen Vorgang wird der Sicherheitsstatus der SMC geändert. Er erlaubt danach die Nutzung des CVC-Zertifikates der SMC für die Authentisierung gegenüber der eGK, wobei die Authentifizierung einseitig durchgeführt wird.	
Aufrufparameter		
	Name	Beschreibung
	`Context`	MandantId, CsId, WorkplaceId verpflichtend; UserId verpflichtend für HBA
	`HpcHandle`	Adressiert den HBA oder die SM-B, die zur Authentisierung verwendet werden soll (Quellkarte).
	`SmcHandle`	Adressiert die SM-B, die für den Zugriff auf die eGK freigeschaltet werden soll (Zielkarte). Folgende Kombinationen von Quellkarte und Zielkarte sind erlaubt, alle anderen führen zu Fehlercode 4051: HpcHandle SmcHandle HBA SM-B SM-B SM-B
Rückgabe		
	Name	Beschreibung

	`Status`	Enthält den Ausführungsstatus der Operation siehe 3.5.2
Vorbedingung	Für PinTyp=PIN.CH und CardType=HBA MUSS PINStatus=verifiziert sein.	
Nachbedingung	keine	

Tabelle 109: TAB_KON_552 Ablauf AuthorizeSMC

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 jeweils einen Aufruf für HpcHandle $hpcHandle TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; $context.userId; $hpcHandle} und einen für das SmcHandle $smcHandle TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; $smcHandle} 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 { MandantId, CsId, CardHandle, UserId }
4.	TUC_KON_005 „Card to Card authentisieren“	Schalte SMC frei über TUC_KON_005 { HpcCardHandle,HBAKeyRef, HBACVCRef,SmcCardHandle, SMCKeyRef, SMCCVCRef, CardSession, , , WorkplaceId}

Tabelle 110: TAB_KON_553 Übersicht Fehler Operation „AuthorizeSMC“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4051	Technical	Error	Falscher Kartentyp

⌫

4.1.5.6 Betriebsaspekte

⌦ TIP1-A_4592 Konfigurationswerte des Kartendienstes

Der Konnektor MUSS es einem Administrator ermöglichen, Konfigurationsänderungen gemäß Tabelle TAB_KON_554 vorzunehmen.

Tabelle 111: 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. Der Administrator MUSS diesen Wert in einer Range von 1-30 Sekunden einstellen können. Default-Wert: 2 Sekunden

⌫

4.1.5.6.1 TUC_KON_025 "Initialisierung Kartendienst"

⌦ TIP1-A_4593 TUC_KON_025 „Initialisierung Kartendienst“

Der Konnektor MUSS den technischen Use Case „Initialisierung Kartendienst“ gemäß TUC_KON_025 umsetzen.

Tabelle 112: 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, SlotNo} 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	1. Rufe TUC_KON_001 „Karte öffnen“ 2. Wiederhole, bis für alle gesteckten Karten ein Eintrag in CM_CARD_LIST existiert.
Varianten/Alternativen	keine
Fehlerfälle	keine
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

⌫

4.1.5.6.2 Kartenübersicht und PIN-Management

⌦ TIP1-A_5110 Übersicht über alle verfügbaren Karten

Die Administrationsoberfläche MUSS dem Administrator eine Übersichtseite anbieten, die alle in CM_CARD_LIST enthaltenen Karten listet.

In dieser Übersichtsseite muss zu jeder Karte dargestellt werden:

- CT(CARD.CTID).HOSTNAME
- CARD.SLOTNO
- CARD.TYPE
- CARD.INSERTTIME
- CARD.CARDHOLDERNAME

Ferner MÜSSEN auf Verlangen des Administrators zu jeder Karte neben den obigen Informationen auch folgende Details angezeigt werden:

- CARD.ICCSN
- CARD.CARDVERSION
- CARD.CERTEXPIRATIONDATE⌫

⌦ **TIP1-A_5111 PIN-Management der SM-Bs für den Administrator**

Über die Administrationsoberfläche MUSS der Administrator für jede in der Übersichtsseite angezeigte Karte vom Typ SM-B die folgenden TUCs auslösen können.

Für diese MUSS er einen der gemäß Kapitel 4.1.1.6 [TIP1-A_4526] definierten Mandanten auswählen können. Es wird eine CardSession im Kontext der Administration erzeugt: CS = TUC_KON_026 {MandantID, „intern“, CARD.CardHandle, „“}

- TUC_KON_012 „PIN verifizieren“ {CS, „Konnektor“, PIN.SMC, „“, Mandatorisch}
- TUC_KON_019 „PIN ändern“ {CS, „Konnektor“, PIN.SMC}
- TUC_KON_021 „PIN entsperren“ {CS, „Konnektor“, PIN.SMC}
- TUC_KON 022 „Liefere PIN-Status“ {CS, PIN.SMC}⌫

4.1.6 Systeminformationsdienst

Der Systeminformationsdienst stellt Basisdiensten, Fachmodulen und Clientsystemen sowohl aktiv (Push-Mechanismus) wie passiv (Pull-Mechanismus) Informationen zur Verfügung. Dabei erhebt er selbst keine Daten, sondern dient nur als zentraler Mechanismus, der von anderen Basisdiensten und Fachmodulen zur Verteilung und Bereitstellung derer Informationen verwendet werden kann.

Innerhalb des Systeminformationsdienstes werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): „EVT“
- Konfigurationsparameter: „EVT_“

Push-Mechanismus

Der Push-Mechanismus des Systeminformationsdienstes hat die Aufgabe, Ereignisse von internen Ereignisquellen im Konnektor (z. B. von anderen Basisdiensten wie Kartendienst, Kartenterminaldienst oder Fachmodulen) an alle Basisdienste und Fachmodule sowie an die bei ihm registrierten Ereignisempfänger (Clientsysteme) weiterzuleiten.

Die Namen der Ereignisse, die Topics, sind als Baumstruktur organisiert und werden mittels „/"-getrennter Liste angegeben (z. B. „Auslöser/Ereigniskategorie1/…/Ereignis1"). Die konkreten Topics werden innerhalb der einzelnen Funktionsmerkmale kontextbezogen definiert und im Anhang in einer zentralen Liste übersichtlich dargestellt.

Clientsysteme können sich für den Empfang bestimmter Ereigniskategorien (Topics) anmelden. Der Systeminformationsdienst übernimmt dementsprechend bei der Verteilung der Ereignisse auch eine Filterfunktion für die weiterzuleitenden Ereignisse.

Die Zustellung der Ereignisse erfolgt unidirektional über eine Netzschnittstelle, deren Kommunikationsendpunkt („Ereignissenke") vom Clientsystem realisiert werden muss. Zur Übertragung der Daten wird ein konnektoreigenes Protokoll cetp (Connector Event Transport Protocol) verwendet.

Pull-Mechanismus

Der Pull-Mechanismus des Systeminformationsdienstes hat die Aufgabe sowohl Zustandswerte als auch statische Informationen des Konnektors selbst als auch von den über ihn verwalteten Ressourcen durch Fachmodule und Clientsysteme abrufbar zu machen. Dabei können entweder Listen von Ressourcen oder Details zu einzelnen Ressourcen abgerufen werden.

Die folgenden Unterkapitel regeln:

- Das Kommunikationsprotokoll cetp
- Die Struktur der Ereignisnachricht
- Die TUCs für die Ereignisverteilung (PUSH)
- Die TUCs und Operationen der Außenschnittstelle für den Abruf von Informationen (PULL)
- Einstellungen, die der Administrator zur Steuerung des Verhaltens vornehmen kann.

4.1.6.1 Funktionsmerkmalweite Aspekte

⌦ **TIP1-A_4594 Richtung bei Verbindungsaufbau des Systeminformationsdienstes**

Der Konnektors MUSS zur Übertragung von Ereignissen eine cetp-Verbindung zu der Ereignissenke des Clientsystems aufbauen, die das Clientsystem beim Operationsaufruf Subscribe per `EventTo` angegeben hatte.⌫

⌦ **TIP1-A_5536 Connector Event Transport Protocol über TCP**

Der Konnektor MUSS das Anwendungsprotokoll cetp (Connector Event Transport Protocol) ausschließlich über das Transportprotokoll TCP (gebenenfalls TLS gesichert) anbieten. ⌫

TIP1-A_4595 Gesicherte Übertragung von Ereignissen

Der Konnektors MUSS zur Übertragung der Ereignisse eine gesicherte Verbindung (TLS) verwenden, wenn ANCL_TLS_MANDATORY=Enabled.

Für die Übermittlung der Ereignisse wurde ein leichtgewichtiges Protokoll gewählt, da vom Clientsystem keine Antwort auf Anwendungsebene erwartet wird.

TIP1-A_4596 Nachrichtenaufbau und -kodierung des Systeminformations-dienstes

Der Konnektors MUSS Ereignisse an Ereignissenken mittels Nachrichten verteilen, die gemäß TAB_KON_030 „Ereignisnachricht“ aufgebaut sind. Der Konnektor MUSS die Nachrichten mit der Zeichenkette „CETP“ beginnen, gefolgt von der Länge der folgenden Ereignisnachricht. Das vier Byte lange Längenfeld MUSS in der Byte-Reihenfolge Big-Endian codiert werden (das hochwertigste Byte wird als erstes übertragen).

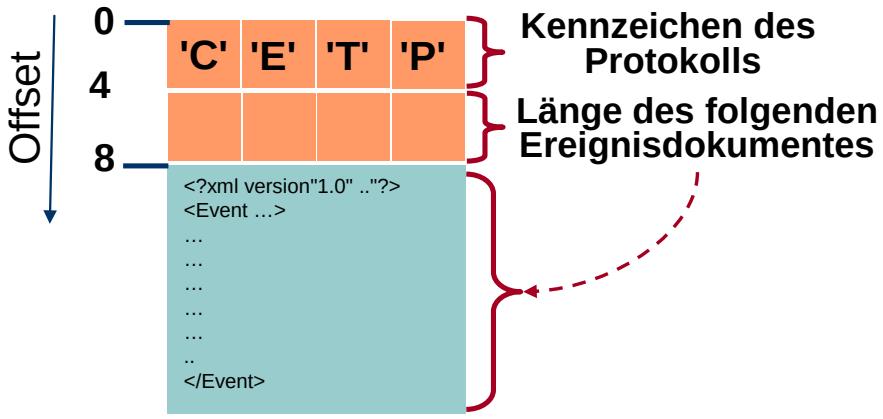
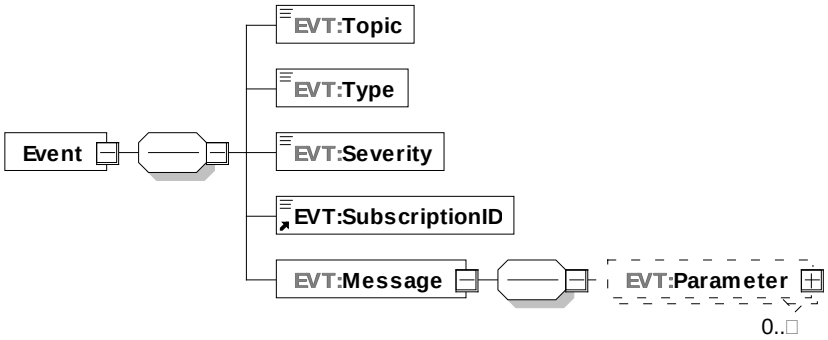
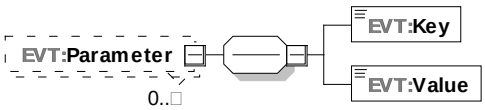


Abbildung 11: PIC_KON_022 Grundsätzlicher Aufbau der Ereignisnachricht

Tabelle 113: 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.
	

		
	`Topic`	Topic der Ereignisnachricht
	`Type`	Typ der Ereignisnachricht (Security, Operation, Infrastructure)
	`Severity`	Schwere der Ereignisnachricht (Info, Warning, Error, Fatal)
	`SubscriptionID`	Identifikator der Anmeldung, der vom Konnektor bei der Operation `Subscribe` für die Anmeldung des jeweiligen Clientsystems vergeben wurde.
	`Message`	Dieses Element enthält die Ereignisnachricht. Der Inhalt ist abhängig vom Topic und wird mittels „Key-Value“-Parametern übertragen.
	`Message/Parameter/Key`	Name des Parameters (String)
	`Message/Parameter/Value`	Wert des Parameters (String)
Hinweise	Das XML-Dokument MUSS UTF-8-codiert sein.	

⌫

4.1.6.2 Durch Ereignisse ausgelöste Reaktionen

Keine.

4.1.6.3 Interne TUCs, nicht durch Fachmodule nutzbar

Keine.

4.1.6.4 Interne TUCs, auch durch Fachmodule nutzbar

4.1.6.4.1 TUC_KON_256 „Systemereignis absetzen“

⌦ **TIP1-A_4598 TUC_KON_256 „Systemereignis absetzen“**

Der Konnektor MUSS für den PUSH-Mechanismus des Systeminformationsdienstes den technischen Use Case TUC_KON_256 “Systemereignis absetzen” umsetzen.

Tabelle 114: 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 `Subscribe`). Zusätzlich wird, bei

Element	Beschreibung
	gesetztem Flag, das Ereignis durch den Protokollierungsdienst protokolliert.
Auslöser	Aufruf durch Basisdienst oder Fachmodul
Vorbedingungen	Fall Topic = „BOOTUP/BOOTUP_COMPLETE": Zu allen URLs von clientseitigen Endpunkten, wie sie bei der `Subscribe`-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 zu ihrer TerminationTime auch über Bootups hinweg gespeichert werden und nach Ablauf ihrer TerminationTime gelöscht werden. Sämtliche Subscriptions müssen vor dem Versenden des Events gelöscht sein.

Eingangsdaten	• Event (zu versendendes Ereignis)
	  o Topic
	  o Typ (Op = Operation, Sec = Security, Infra = Infrastructur)
	  o Schwere (Info = Information, Warn = Warning, Err = Error, Fatal)
	  o Parameter
	• Schalter „Schreibe Protokolleintrag" (doLog/noLog; optional; default = doLog)
	• Schalter „An Clientsysteme versenden" (doDisp/noDisp; optional; default = doDisp)
	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.

Komponenten	Konnektor
Ausgangsdaten	Keine
Nachbedingungen	

Standardablauf	Für das übergebene Ereignis werden folgende Schritte durchgeführt:
	1. Das Ereignis wird an alle Basisdienste und Fachmodule des Konnektors gesendet.
	2. Wenn der Schalter „Schreibe Protokolleintrag" den Wert „doLog" hat, erfolgt der Aufruf von TUC_KON_271 { $event }.
	3. Falls der Schalter „An Clientsysteme versenden" nicht gesetzt ist (noDisp) wird an dieser Stelle abgebrochen.
	4. Das für den Versand an Clientsysteme benötigte XML-Dokument des Ereignisses wird aufgebaut (Element „Event" gemäß EventService.XSD).
	5. Setze $subscriptionList = Liste der Clientsystem-Abonnements, die durch die Operationen Subscribe/Unsubscribe gepflegt werden und deren TerminationTime > Systemzeit. 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.
	  a. Filtern nach Topics:
	  für jede $subscription in $subscriptionList {

Element	Beschreibung

	wenn $event.topic nicht mit $subscription.topic
	beginnt oder übereinstimmt,
	dann entferne $subscription aus $subscriptionList
	}
	
	b. Filtern nach Zugriffberechtigung:
	für jede $subscription in $subscriptionList {
	wenn
	TUC_KON_000 {
	$subscription.context.mandantId;
	$subscription.context.clientsystemId;
	$subscription.context.workplaceId;
	$parameter.value für $parameter.key = „ctId“
	$parameter.value für [$parameter.key = „cardHandle“;
	doNotNeedCardSession;
	allWorkplaces
	}
	mit einem Fehler abgeschlossen wird,
	dann entferne $subscription aus $subscriptionList
	}
	
	c. Filtern nach XPath-Filter in Subscription ([XPATH]):
	für jede $subscription in $subscriptionList {
	wenn der XPath-Ausdruck $subscription.filter
	angewandt auf das als XML-Dokument dargestellte
	Ereignis ein leeres Ergebnis liefert,
	dann entferne $subscription aus $subscriptionList
	}
	
	6. Versenden:
	für jede $subscription in $subscriptionList {
	versende das Ereignis an $subscription.eventTo
	}
	Für das versendete Ereignis wird keine Antwort durch das Clientsystem erwartet

Varianten/Alternativen	Fall Topic = „BOOTUP/BOOTUP_COMPLETE":
	
	5. Setze $urlList = Liste der URLs von clientseitigen Endpunkten, wie sie bei der `Subscribe`-Operation angegeben wurden und deren TerminationTime > Systemzeit ist.
	
	6. Versenden:
	für jede $url in $urlList {
	versende das Ereignis an $url
	}
	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.

Fehlerfälle	Fehler in den folgenden Schritten des Standardablaufs führen zum Abbruch mit den ausgewiesenen Fehlercodes: (→5c) Fehler bei der Auswertung des XPath-Ausdrucks: Fehlercode: 4095, nur für die jeweilige Abonnement-Prüfung.

Element	Beschreibung
Fachliche Fehlermeldung	Keine
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Abbildung 12: PIC_KON_112 Aktivitätsdiagramm zu „Systemereignis absetzen“

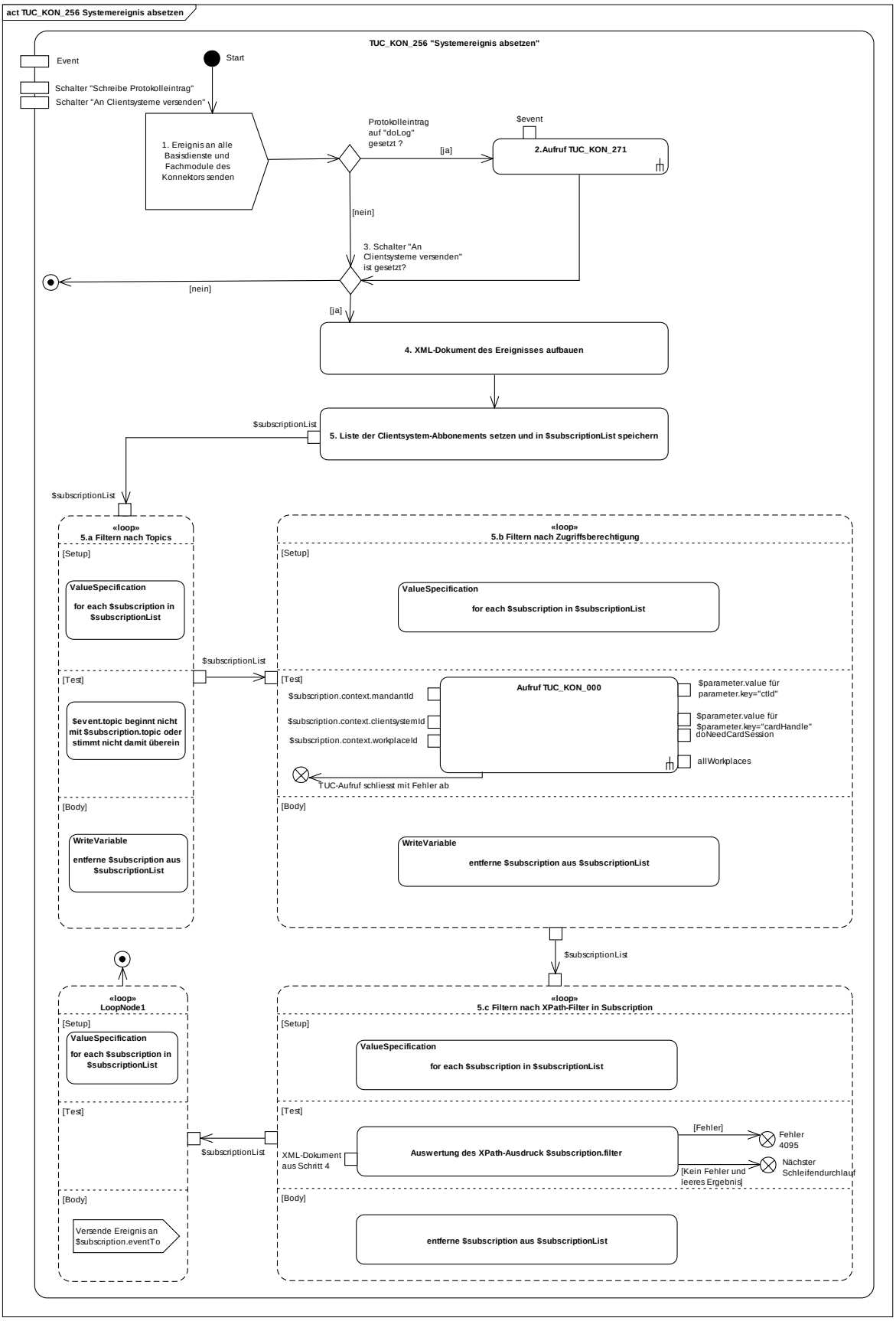


Abbildung 12: PIC_KON_112 Aktivitätsdiagramm zu „Systemereignis absetzen“

Tabelle 115: TAB_KON_557 Übersicht Fehler 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 116: 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 `GetCardTerminals`) oder ein Fachmodul
Vorbedingungen	Keine
Eingangsdaten	• Arbeitsplatz ID (Optional) • Clientsystem ID • Mandanten ID
Komponenten	Konnektor, Kartenterminal
Ausgangsdaten	• 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.
Nachbedingungen	• Der Zustand der Kartenterminals bleibt unverändert
Standardablauf	1. Erstellen der Liste aller Kartenterminals, auf die der angegebene Mandant und das angegebene Clientsystem zugreifen dürfen (siehe Zugriffsberechtigungsdienst) a. Wurde der optionale Parameter Arbeitsplatz 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 Kartenterminal bezeichneten KT. b. Fehlt dieser Parameter, so werden alle Kartenterminals in die Liste aufgenommen, die sowohl dem Clientsystem als auch dem Mandanten zugeordnet sind.

Element	Beschreibung
	2. Rückgabe der Liste der in Schritt 1 erstellten Liste mit Angaben zu jedem Kartenterminal gemäß Schema „`Eventservice.xsd`“.
Varianten/Alternativen	Keine
Fehlerfälle	Keine
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

⌫

4.1.6.4.3 TUC_KON_253 „Liefere Karten_Liste“

⌦ **TIP1-A_4600 TUC_KON_253 “Liefere Karten_Liste”**

Der Konnektor MUSS den technischen Use Case TUC_KON_253 “Liefere Karten_Liste” umsetzen.

Tabelle 117: 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	• Arbeitsplatz ID (Optional) • Clientsystem ID • Kartenterminal ID (Optional) • Mandanten ID • CardType (Optional)
Komponenten	Konnektor, Kartenterminal, Karte
Ausgangsdaten	• Liste der gesteckten Karten, auf die der Mandant und das Clientsystem von dem Arbeitsplatz aus zugreifen dürfen (siehe Zugriffsberechtigungsdienst). Falls Kartenterminal angegeben, nur Karten die im entsprechenden Kartenterminal stecken.
Nachbedingungen	• Der Zustand der Kartenterminals und Karten bleibt unverändert
Standardablauf	1. Erstellen der Liste aller Karten, auf die der angegebene Mandant und das angegebene Clientsystem zugreifen dürfen (siehe Zugriffsberechtigungsdienst). a. Wurde eine Kartenterminal ID übergeben, dann nur Karten berücksichtigen, die dem dadurch referenziertem

Element	Beschreibung
	Kartenterminal zugeordnet sind. b. Wurde der optionale Parameter Arbeitsplatz ID übergeben, so werden nur die Karten in die Liste aufgenommen, auf die von diesem Arbeitsplatz aus zugegriffen werden darf (siehe „Zugriffsberechtigung Ressourcen“). c. Wurde der optionale Parameter CardType übergeben, so werden nur die Karten in die Liste aufgenommen, die dem Kartentyp in CardType entsprechen. 2. Rückgabe der Liste der in Schritt 1 erstellten Liste mit Angaben zu jeder Karte gemäß Schema „`Eventservice.xsd`“.
Varianten/Alternativen	Keine
Fehlerfälle	Fehler in den folgenden Schritten des Standardablaufs führen zum Abbruch mit den ausgewiesenen Fehlercodes: (→1 a) Ungültige Kartenterminal ID: Fehlercode: 4096
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 118: TAB_KON_560 Übersicht Fehler TUC_KON_253 „Liefere Karten_Liste“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4096	Technical	Error	Ungültige Kartenterminal ID

⌫

4.1.6.4.4 TUC_KON_254 „Liefere Ressourcendetails“

⌦ **TIP1-A_4602 TUC_KON_254 “Liefere Ressourcendetails”**

Der Konnektor MUSS den technischen Use Case TUC_KON_254 “Liefere Ressourcendetails” umsetzen.

Tabelle 119: 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, HSM) oder dem Konnektor
Auslöser	Aufruf durch ein Clientsystem (Operation `GetRessourceInformation`) oder ein Fachmodul
Vorbedingungen	Keine
Eingangsanforderung	Keine

Element	Beschreibung
Eingangsdaten	• Clientsystem ID • Mandanten ID • Arbeitsplatz ID (Optional) • Kartenterminal ID (Optional) • Kartenslot ID (Optional und nur in Kombination mit Kartenterminal ID) • CardHandle (Optional) • Iccsn (Optional) • HSM ID (Optional)
Komponenten	Konnektor, Kartenterminal, Karte, HSM
Ausgangsdaten	• Informationsobjekt einer Ressource (Kartenterminal, Karte, HSM)
Nachbedingungen	• Der Zustand der Kartenterminals, Karten und HSM bleibt unverändert
Standardablauf	1. Falls eine Kartenterminal ID, aber keine Kartenslot ID übergeben wurde, wird ein Informationsobjekt des Kartenterminals zurückgegeben. 2. Falls eine Kartenterminal ID und eine Kartenslot ID oder eine Iccsn oder ein CardHandle übergeben wurde, wird ein Informationsobjekt der Karte, die sich in dem angegebenen Slot befindet bzw. die über die Iccsn oder das CardHandle identifiziert werden kann, zurückgegeben. 3. Falls eine HSM ID übergeben wurde, wird ein Informationsobjekt der HSM zurückgegeben. 4. Wurde weder eine Iccns, ein CardHandle, eine Kartenterminal ID, eine Kartenslot ID noch eine HSM ID übergeben, so wird ein Informationsobjekt des Konnektors zurückgegeben. Vor der Rückgabe der Informationen über eine Ressource wird geprüft, ob der angegebene Mandant und das angegebene Clientsystem darauf zugreifen dürfen (siehe Zugriffsberechtigungsdienst). Wurde zusätzlich der optionale Parameter Arbeitsplatz ID übergeben, so wird auch geprüft, ob die Ressource diesem Arbeitsplatz zugeordnet ist. Die Rückgabe der Informationen erfolgt gemäß dem Schema „`Eventservice.xsd`“.
Varianten/Alternativen	Keine
Fehlerfälle	Fehler in den folgenden Schritten des Standardablaufs führen zum Abbruch mit den ausgewiesenen Fehlercodes: (→1) Ungültige Kartenterminal ID: Fehlercode: 4096 (→2) Ungültige Kartenterminal ID: Fehlercode: 4096 (→2) Ungültige Kartenslot ID: Fehlercode: 4097 (→2) Keine Karte im angegebenen Slot: Fehlercode: 4098 (→2) Keine Karte mit angegebener Iccsn gefunden: Fehlercode: 4099 (→2) Karten-Handle ungültig: Fehlercode: 4101 (→3) Ungültige HSM ID: Fehlercode: 4100

Element	Beschreibung
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 120: TAB_KON_562 Übersicht Fehler TUC_KON_254 „Liefere Ressourcendetails“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4096	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
4100	Technical	Error	Ungültige HSM-ID
4101	Technical	Error	Karten-Handle ungültig

⌫

4.1.6.5 Operationen an der Außenschnittstelle

⌦ TIP1-A_4603 Basisanwendung Systeminformationsdienst

Der Konnektor MUSS für Clients eine Basisanwendung Systeminformationsdienst anbieten.

Tabelle 121 TAB_KON_029 Basisanwendung Systeminformationsdienst

Name	`EventService`	
Version	wird im Produktsteckbrief des Konnektors definiert	
Namensraum	Siehe Anhand D	
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
	`Unsubsribe`	Abmelden von der Zustellung von Ereignissen

	`RenewSubscriptions`	Gültigkeit bestehender Subscriptions verlängern
	`GetSubscriptions`	Abfrage der angemeldeten Zustellungen von Ereignissen
WSDL	`EventService.wsdl`	
Schema	`EventService.xsd`	

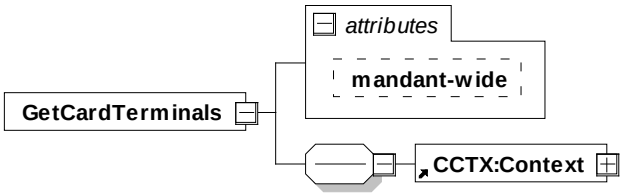
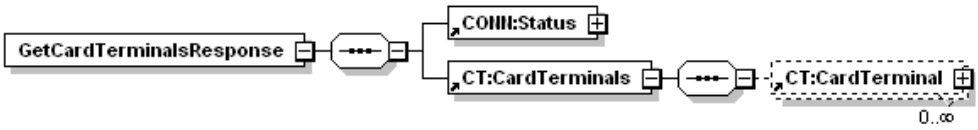
⌫

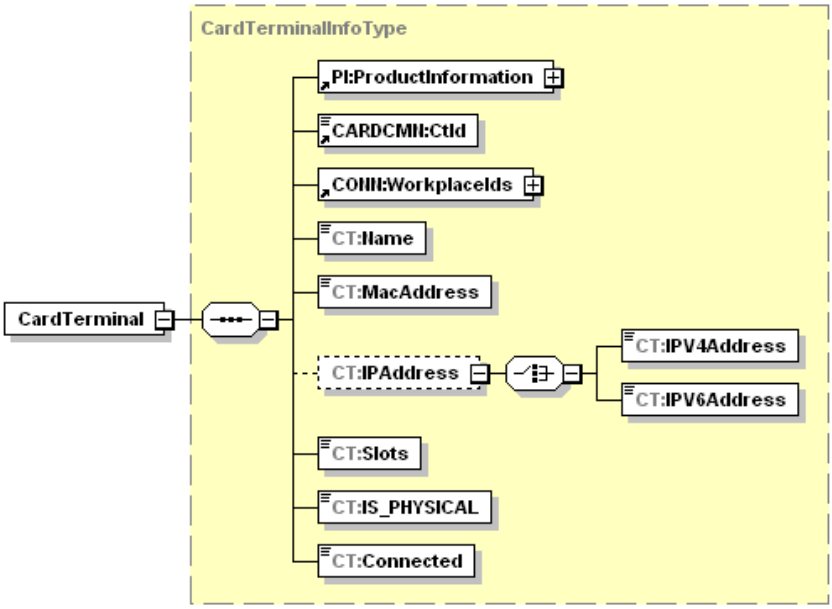
4.1.6.5.1 GetCardTerminals

TIP1-A_4604 Operation GetCardTerminals

Der Konnektors MUSS an der Außenschnittstelle eine Operation GetCardTerminals, wie in Tabelle TAB_KON_563 „Operation GetCardTerminals“ beschrieben, anbieten.

Tabelle 122: TAB_KON_563 Operation GetCardTerminals

Name	`GetCardTerminals`	
Beschreibung	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.	
Aufrufparameter		
	Name	Beschreibung
	`@mandant-wide`	Wenn „true“, werden alle Kartenterminals zurückgegeben, auf die der Mandant zugreifen darf. Wenn „false“ (Standardbelegung), werden nur Kartenterminals zurückgegeben, auf die von dem im Aufrufkontext spezifizierten Arbeitsplatz zugriffen werden darf.
	`Context`	Aufrufkontext
Rückgabe		
	Name	**Beschreibung**
	`Status`	Ergebnis der Operation



Die Liste der Kartenterminals

Name	Beschreibung
`ProductInformation`	Produkt Informationen gemäß [gemSpec_OM] und dem Schema „`ProductInformation.xsd`“ zu formatieren.
`CtId`	Eineindeutige 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 (siehe auch TAB_KON_522 Parameterübersicht des Kartenterminaldienstes)
`Connected`	Zeigt an, ob dieses Kartenterminal aktuell verfügbar ist. TRUE – ist verfügbar FALSE – ist nicht verfügbar

Vorbedingungen	Keine
Nachbedingungen	Der Zustand der Kartenterminals bleibt unverändert.
Hinweise	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 Tabelle 123: TAB_KON_564 Ablauf GetCardTerminals beschrieben:

Tabelle 123: TAB_KON_564 Ablauf GetCardTerminals

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 {$context.mandantId; $context.clientsystemId; $context.workplaceId; doNotNeedCardSession; @mandant-wide} Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.
3.	TUC_KON_252 „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.

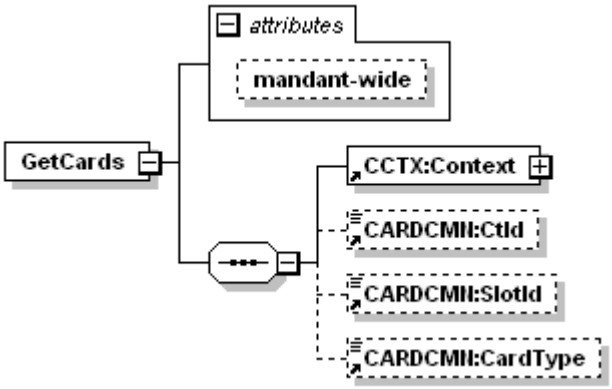
⌫

4.1.6.5.2 GetCards

TIP1-A_4605 Operation GetCards

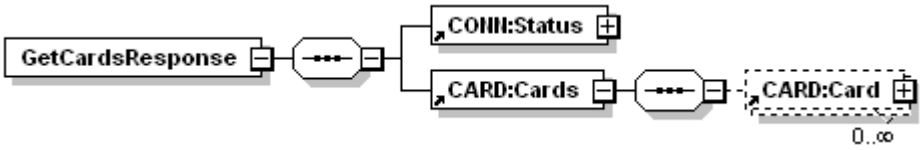
Der Konnektors MUSS an der Außenschnittstelle eine Operation GetCards, wie in Tabelle TAB_KON_565 „Operation GetCards“ beschrieben, anbieten und MUSS dabei Kartentypen aus Tabelle 1: TAB_KON_500 Wertetabelle Kartentypen unterscheiden.

Tabelle 124: Tab_KON_565 Operation GetCards

Name	`GetCards`
Beschreibung	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.
Aufrufparameter	

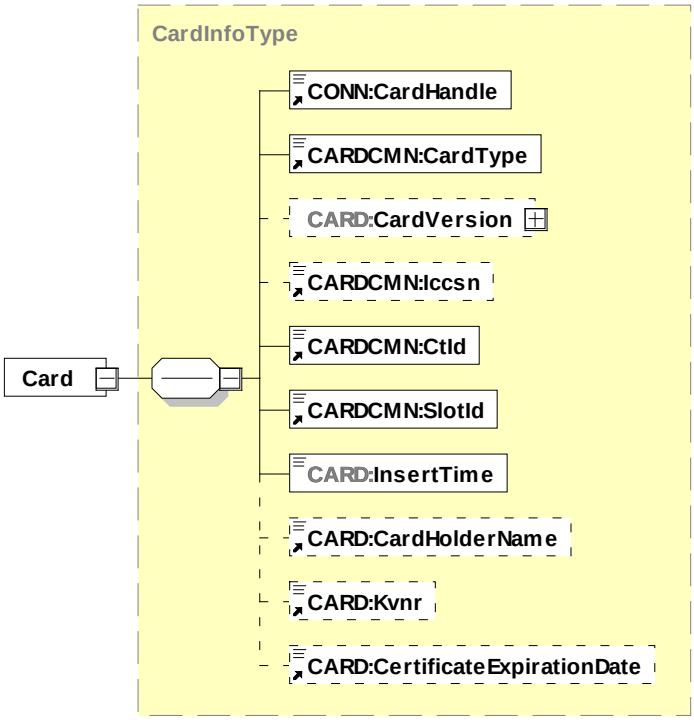
	Name	Beschreibung
	`@mandant-wide`	Wenn „true“, werden alle Karten zurückgegeben, auf die der Mandant zugreifen darf. Wenn „false“ (Standardbelegung), werden nur Karten zurückgegeben, auf die von dem im Aufrufkontext spezifizierten Arbeitsplatz zugriffen werden darf.
	`Context`	Aufrufkontext
	`CtId`	Identifikation des Kartenterminals. Wenn angegeben, werden nur die Karten zurückgeliefert, die in diesem Kartenterminal verfügbar sind.
	`SlotId`	Nummer des Slots, beginnend bei 1.
	`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.

Antwort

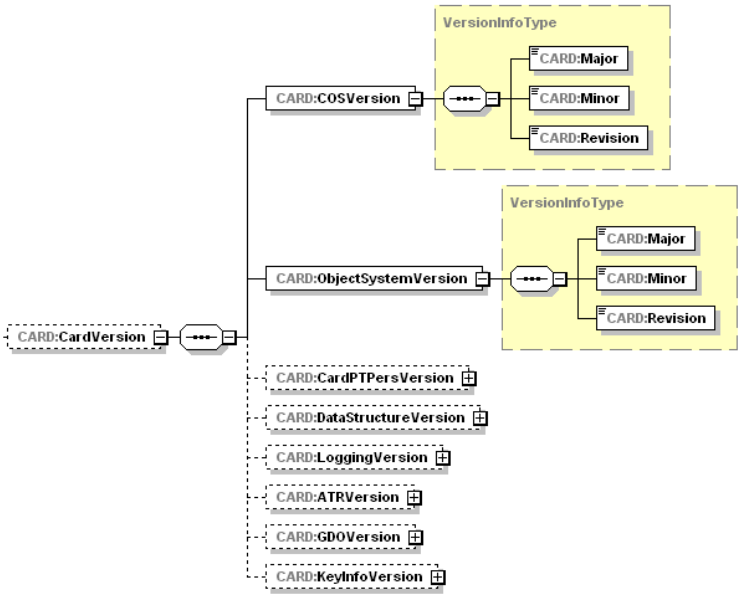


Name	Beschreibung
`Status`	Ergebnis der Operation

Im Element `Cards` wird die Liste der gesteckten Karten zurückgegeben. Für jede Karte wird dabei ein `Card`-Element angegeben. Leere Slots der Kartenterminals sind in dieser Liste nicht enthalten.



Name	Beschreibung
`CardHandle`	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. 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.
	`CardType`	Erkannter Typ der Karte. Siehe Tabelle 1: TAB_KON_500 Wertetabelle Kartentypen
	`CardVersion`	 Der Konnektor MUSS in CardVersion bei eGK, HBAx und SM-B/SMC-KT die Versionsinformationen aus /MF/EF.Version übergeben. Bei KVK und unbekannten Karten MUSS das Element weggelassen werden.
	`Iccsn`	Falls auslesbar, die ICC-Serial-Number der Karte
	`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. Die Zeit wird mit dem Datentyp `DateTime` in folgendem Format angegeben: `yyyy-mm-ddThh:mm:ss+hh:mm` Es ist also – gemäß ISO 8601 [ISO8601] – die lokale Zeit und die Differenz zur UTC anzugeben.
	`CardHolderName`	Name des Karteninhabers bzw. der Institution (subject.commonName). 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.
	`CertificateExpirationDate`	Ablaufdatum des Zertifikates (AUT bzw. OSIG). Bei KVK und unbekannten Karten MUSS das Element weggelassen werden.
Vorbe-dingungen	Keine.	
Nachbe-dingungen	Der Zustand der Karten und der Kartenterminals bleibt unverändert.	
Hinweise	Der Aufruf darf nur den im Konnektor verwalteten aktuellen Zustand der Karte liefern und keine Abfragen an die Kartenterminals absetzen.	

Der Ablauf der Operation GetCards ist in Tabelle 125: TAB_KON_566 Ablauf GetCards beschrieben:

Tabelle 125: 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 Zugriffs-berechtigung“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; doNotNeedCardSession; @mandant-wide} 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.

Die Fehlerfälle der Operation GetCards sind in Tabelle 126 TAB_KON_567 Fehlerfälle GetCards dargestellt:

Tabelle 126 TAB_KON_567 Fehlerfälle GetCards

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weiteren Fehlercodes auftreten:			
4000	Technical	Error	Syntaxfehler

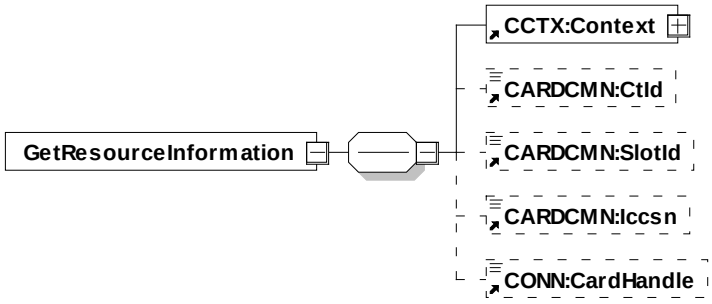
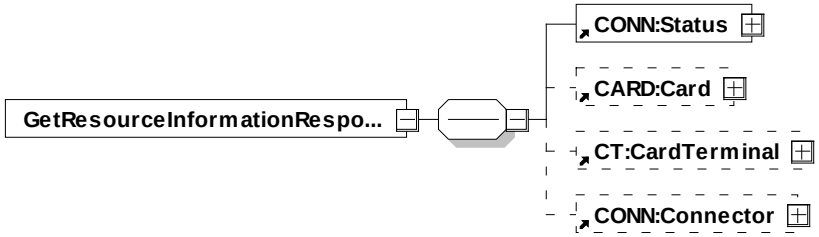
⌫

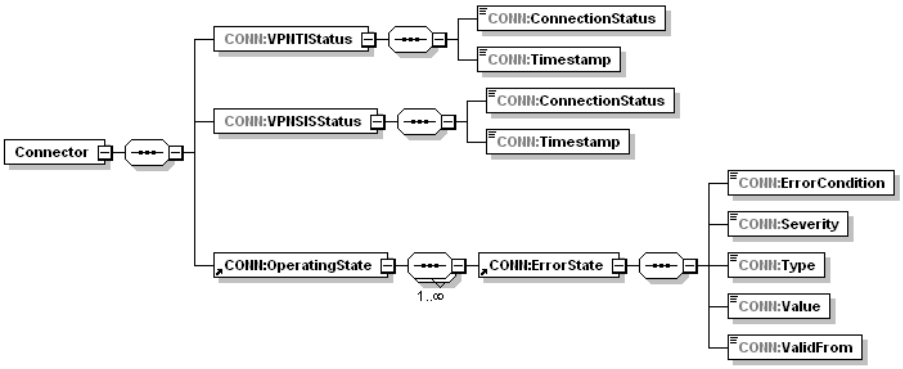
4.1.6.5.3 GetResourceInformation

TIP1-A_4607 Operation GetResourceInformation

Der Konnektors MUSS an der Außenschnittstelle eine Operation GetResourceInformation, wie in Tabelle TAB_KON_568 „Operation GetResourceInformation“ beschrieben, anbieten.

Tabelle 127: TAB_KON_568 Operation GetResourceInformation

Name	`GetResourceInformation`	
Beschreibung	Gibt Informationen zu einer Ressource (Karte, KT) oder dem Konnektor selbst zurück	
Aufrufparameter		
	Name	**Beschreibung**
	`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.	
Rückgabe		
	Name	**Beschreibung**
	`Status`	Ergebnis der Operation
	`Card`	Informationen einer Karte (siehe `GetCards`)
	`CardTerminal`	Informationen eines Kartenterminals (siehe `GetCardTerminals`)

	`Connector`	Informationen zum Konnektor
		
	`VPNTIStatus`	
	`VPNTIStatus/ConnectionStatus`	Status der VPN-Verbindung zur TI (Online oder Offline)
	`VPNTIStatus/Timestamp`	Zeitstempel der letzten Statusänderung
	`VPNSISSTatus`	
	`VPNSISStatus/ConnectionStatus`	Status der VPN-Verbindung des SIS (Online oder Offline)
	`VPNSISStatus/Timestamp`	Zeitstempel der letzten Statusänderung
	`OperatingState`	Betriebszustand (siehe Kapitel 3.3)
	`OperatingState/ErrorState`	Eine Zeile der Fehlerzustandsliste gemäß Tabelle 3 TAB_KON_503 Betriebszustand_Fehlerzustandsliste
	`OperatingState/ErrorState/ErrorCode`	Fehlercode gemäß Tabelle 3 TAB_KON_503 Betriebszustand_Fehlerzustandsliste
	`OperatingState/ErrorState/Severity`	Schwere des Fehlerzustandes gemäß Tabelle 3 TAB_KON_503 Betriebszustand_Fehlerzustandsliste
	`OperatingState/ErrorState/Type`	Fehlertyp gemäß Tabelle 3 TAB_KON_503 Betriebszustand_Fehlerzustandsliste
	`OperatingState/ErrorState/Value`	Fehlerzustandswert
	`OperatingState/ErrorState/ValidFrom`	Zeitstempel der letzten Änderung des Fehlerzustands
Vorbedingung		
Nachbedingung	Der Zustand der Ressource bleibt unverändert.	
Hinweise		

Der Ablauf der Operation GetResourceInformation ist in Tabelle 128: TAB_KON_569 Ablauf GetResourceInformation beschrieben:

Tabelle 128: TAB_KON_569 Ablauf GetResourceInformation

Nr.	Aufruf Technischer Use Case oder Interne Operation	Beschreibung
1.	checkArguments	Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Insbesondere wird geprüft, dass eine `SlotId` nur in Verbindung mit einer `CtId` übergeben werden kann (Abfrage einer Karte). Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.

2.	TUC_KON_000 „Prüfe Zugriffs-berechtigung“	Die Prüfung erfolgt,
		falls ein CardHandle übergeben wurde, durch den Aufruf
		
		TUC_KON_000 {
		mandantId = $context.mandantId;
		clientSystemId = $context.clientsystemId;
		workplaceId = $context.workplaceId;
		ctId = $ctId;
		cardHandle = $cardHandle;
		needCardSession = false
		allWorkplaces = false
		}
		
		und, falls kein CardHandle übergeben wurde, durch den Aufruf
		
		TUC_KON_000 {
		mandantId = $context.mandantId;
		clientSystemId = $context.clientsystemId;
		workplaceId = $context.workplaceId;
		ctId = $ctId;
		needCardSession = false
		allWorkplaces = true
		}
		
		Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.

3.	TUC_KON_254 „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.

Die Fehlerfälle der Operation GetResourceInformation sind in Tabelle 129 TAB_KON_570 Fehlerfälle GetResourceInformation dargestellt:

Tabelle 129 TAB_KON_570 Fehlerfälle GetResourceInformation

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weiteren Fehlercodes auftreten:			
4000	Technical	Error	Syntaxfehler

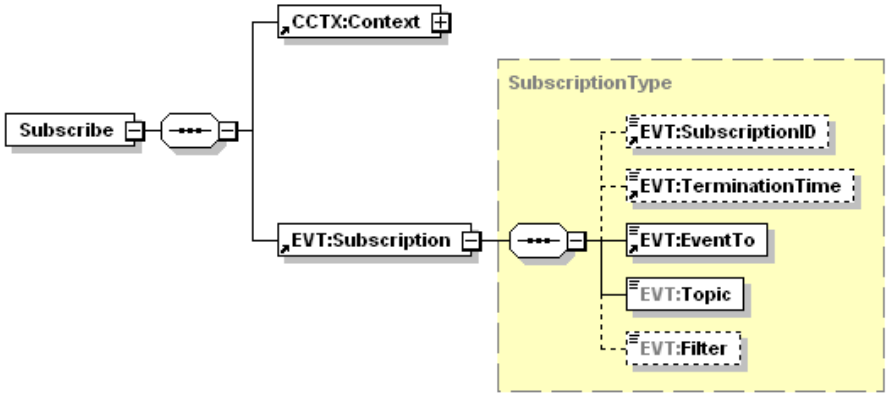
⌫

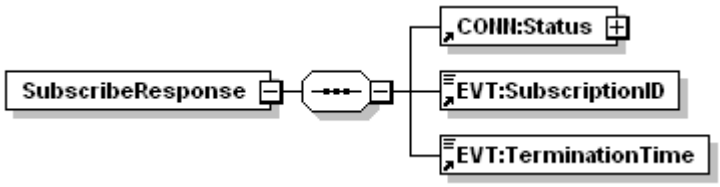
4.1.6.5.4 Subscribe

⌦ **TIP1-A_4608 Operation Subscribe**

Der Konnektors MUSS an der Außenschnittstelle eine Operation Subscribe, wie in Tabelle TAB_KON_571,Operation Subscribe‘ beschrieben, anbieten.

Tabelle 130: TAB_KON_571 Operation Subscribe

Name	`Subscribe`	
Beschreibung	Clientsysteme melden mit dieser Operation ihr Interesse an bestimmten Topics (Ereignissen) an.	
Aufrufparameter		
	Name	**Beschreibung**
	`Context`	Aufrufkontext
	`EventTo`	URL des Endpunkts, wo die Ereignisse zugestellt werden sollen. Syntax: `cetp://host:port` *`host`*: IP-Adresse oder FQDN des Clientsystems. *`port`*: 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, der beim Versand von Ereignissen in TUC_KON_256 ausgewertet wird. Ermöglicht die Filterung auf Basis der Elemente einer XML-Ereignisnachricht)

Rückgabe		
	Name	**Beschreibung**
	`Status`	Ergebnis der Operation
	`Subscription-ID`	Ein Identifikator, der die Anmeldung für die Topics eindeutig identifiziert. Bei den Operationen `Unsubscribe`, `GetSubscription` und `RenewSubsriptions` MUSS dieser `SubscriptionID` angegeben werden.
	`TerminationTime`	Maximaler Gültigkeitszeitpunkt der Subscription. Sie MUSS auf Systemzeit + 25 h gesetzt werden.
Vorbedingung	Das Clientsystem muss die Ereignissenke realisieren.	
Nachbedingung	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 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. 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.	
Hinweise		

Der Ablauf der Operation Subscribe ist in Tabelle 131: TAB_KON_572 Ablauf Subscribe beschrieben:

Tabelle 131: TAB_KON_572 Ablauf Subscribe

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“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; doNotNeedCardSession; allWorkplaces} Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.

Nr.	Aufruf Technischer Use Case oder Interne Operation	Beschreibung
3	saveSubscription	Die Anmeldung wird im Konnektor hinterlegt. Vorgehalten werden folgende Daten: • SubscriptionId (wird generiert) • TerminationTime (Systemzeit + 25h) • MandantId • ClientsystemId (falls im Aufrufkontext angegeben) • WorkplaceId (falls im Aufrufkontext angegeben) • Ereignissenke (Feld `EventTo`) • Abonnierter Topic (Feld `Topic`) • Filterausdruck (Feld `Filter`) Bei der Übernahme wird eine eindeutige `SubscriptionId` generiert, die dem aufrufenden System zurückgegeben wird, falls die Subscription noch nicht existiert. Existiert sie bereits, wird die bestehende `SubscriptionID` zurückgegeben.

Die Fehlerfälle der Operation Subscribe sind in Tabelle 132 TAB_KON_573 Fehlerfälle Subscribe dargestellt:

Tabelle 132 TAB_KON_573 Fehlerfälle Subscribe

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weiteren Fehlercodes auftreten:			
4000	Technical	Error	Syntaxfehler

⌫

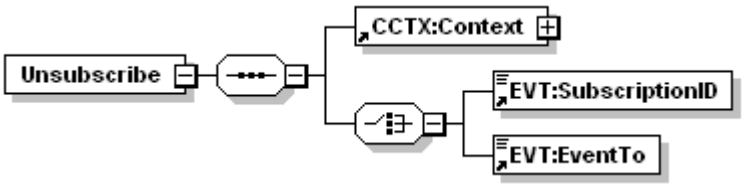
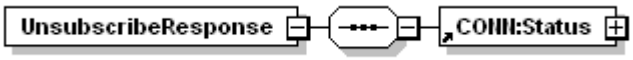
4.1.6.5.5 Unsubscribe

⌦ **TIP1-A_4609 Operation Unsubscribe**

Der Konnektors MUSS an der Außenschnittstelle eine Operation Unsubscribe, wie in Tabelle TAB_KON_574 „Operation Unsubscribe“ beschrieben, anbieten.

Tabelle 133: TAB_KON_574 Operation Unsubscribe

Name	`Unsubscribe`
Beschreibung	Löscht eine Anmeldung mit dem angegebenen `SubscriptionID` oder alle Anmeldungen zu einem Endpunkt `EventTo`.

Aufrufparameter		
	Name	**Beschreibung**
	`Context`	Aufrufkontext
	`Subscription-ID`	Der Identifikator, der bei der `Subscribe`-Operation geliefert wurde.
	`EventTo`	URL des clientseitigen Endpunkts, wie er bei der `Subscribe`-Operation angegeben wurde.
Rückgabe		
	Name	**Beschreibung**
	`Status`	Ergebnis der Operation
Vorbedingung	Die Anmeldung `Subscribe` muss vor dieser Operation aufgerufen worden sein.	
Nachbedingung	Der Konnektor entfernt aus seiner Verwaltung die Subscription zur `Subscription-ID` bzw. alle Subscriptions zur clientseitigen URL des Endpunkts `EventTo`.	
Hinweise	Keine	

Der Ablauf der Operation Unsubscribe ist in Tabelle 134: TAB_KON_575 Ablauf Unsubscribe beschrieben:

Tabelle 134: TAB_KON_575 Ablauf Unsubscribe

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“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; doNotNeedCardSession; allWorkplaces} Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.
3.	removeSubscription	Entfernen der Subscriptions aus der Liste aller Subscriptions.

Die Fehlerfälle der Operation Unsubscribe sind in Tabelle 135 TAB_KON_576 Fehlerfälle Unsubscribe dargestellt:

Tabelle 135 TAB_KON_576 Fehlerfälle 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 SubscriptionId

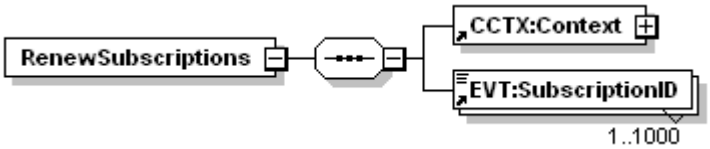
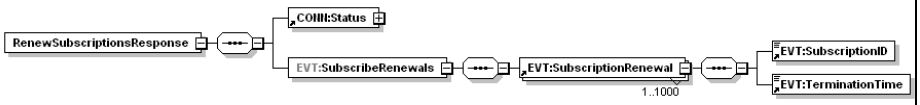
⌫

4.1.6.5.6 RenewSubscriptions

TIP1-A_5112 Operation RenewSubscriptions

Der Konnektors MUSS an der Außenschnittstelle eine Operation RenewSubscriptions, wie in Tabelle 136 TAB_KON_792 „Operation RenewSubscriptions“ beschrieben, anbieten.

Tabelle 136: TAB_KON_792 Operation RenewSubscriptions

Name	`RenewSubscriptions`	
Beschreibung	Verlängert die Gültigkeit einer Liste von Anmeldungen, die jeweils per `SubscriptionID` identifiziert werden.	
Aufrufparameter		
	Name	**Beschreibung**
	`Context`	Aufrufkontext
	`Subscription-ID`	Der Identifikator, der bei der `Subscribe`-Operation geliefert wurde.
Rückgabe		
	Name	**Beschreibung**
	`Status`	Ergebnis der Operation
	`Subscription ID`	Ein Identifikator, der die Anmeldung für die Topics eindeutig identifiziert. Bei den Operationen `Unsubscribe`, `GetSubscription` und `RenewSubsriptions` MUSS diese `SubscriptionID` angegeben werden.
	`TerminationTime`	Maximaler Gültigkeitszeitpunkt der Subscription. Sie MUSS auf Systemzeit + 25 h gesetzt werden.
Vorbedingung		
Nachbedingung	Der Konnektor speichert jede neu vergebene `TerminationTime` in seiner Verwaltung der Subscriptions.	

Hinweise	Keine

Der Ablauf der Operation `RenewSubscriptions` ist in Tabelle 137: TAB_KON_793 Ablauf RenewSubscriptions beschrieben:

Tabelle 137: TAB_KON_793 Ablauf RenewSubscriptions

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“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; doNotNeedCardSession; allWorkplaces} Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.
3.	renewSubscriptions	Es wird eine neue `SubscribeRenewals`-Liste angelegt. Alle Subscriptions, deren `TerminationTime` kleiner als die Systemzeit sind, muss der Konnektor aus der Verwaltung entfernen. Für jede `SubscriptionID`, die in der Verwaltung der Subscriptions existiert und deren `TerminationTime` größer als die Systemzeit ist, wird eine neue `TerminationTime = Systemzeit + 25h` bestimmt. Diese wird zusammen mit der `SubscriptionID` als `SubscribeRenewal` der `SubscribeRenewals`-Liste hinzugefügt.

Die Fehlerfälle der Operation RenewSubscriptions sind in Tabelle 138 TAB_KON_794 Fehlerfälle RenewSubscriptions dargestellt:

Tabelle 138 TAB_KON_794 Fehlerfälle RenewSubscriptions

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weiteren Fehlercodes auftreten:			
4000	Technical	Error	Syntaxfehler

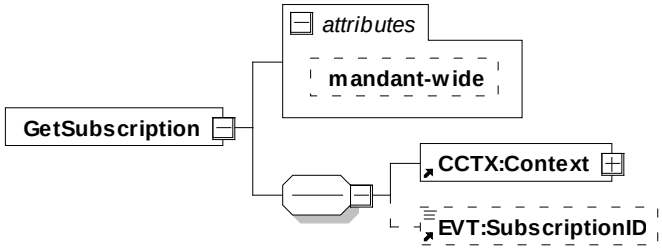
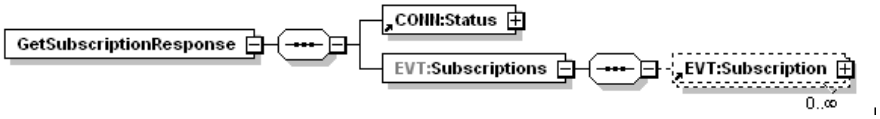
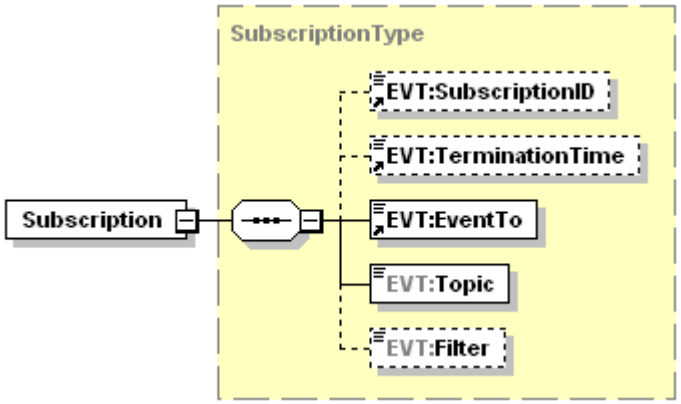
⌫

4.1.6.5.7 GetSubscription

⌦ **TIP1-A_4610 Operation GetSubscription**

Der Konnektor MUSS an der Außenschnittstelle eine Operation GetSubscription, wie in Tabelle TAB_KON_577 „Operation GetSubscription“ beschrieben, anbieten.

Tabelle 139: TAB_KON_577 Operation GetSubscription

Name	`GetSubscription`	
Beschreibung	Gibt die Liste der angemeldeten Topics zurück	
Aufrufparameter		
	Name	**Beschreibung**
	`@mandant-wide`	Wenn „true“, werden alle Subscriptions des Mandanten zurückgegeben. Wenn „false“ (Standardbelegung) werden nur Subscriptions zurückgegeben, die dem im Aufrufkontext spezifizierten Arbeitsplatz zugeordnet sind.
	`Context`	Aufrufkontext
	`Subscription-ID`	Der Identifikator, der bei der `Subscribe`-Operation geliefert wurde.
Rückgabe		
	Name	**Beschreibung**
	`Status`	Ergebnis der Operation
	`Subscriptions`	Die Liste Subscriptions (vgl. Operation `Subscribe`)
		
	`Subscription`	Angefordertes Subscription-Element
	`Subscription/SubscriptionID`	Identifikator der Subscription
	`Subscription/TerminationTime`	Maximaler Gültigkeitszeitpunkt der Subscription.
	`Subscription/`	URL des Endpunkts, wo die Ereignisse zugestellt

	`EventTo`	werden sollen (Ereignissenke)
	`Subscription/Topic`	Angemeldeter Topic
	`Subscription/Filter`	Filterausdruck (falls vorhanden)
Vorbedingung	Keine	
Nachbedingung	Die Liste der Subscriptions bliebt unverändert	
Hinweise	Keine	

Der Ablauf der Operation GetSubscription ist in Tabelle 140: TAB_KON_578 Ablauf GetSubscription beschrieben:

Tabelle 140: TAB_KON_578 Ablauf GetSubscription

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“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; doNotNeedCardSession; allWorkplaces} Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.
3.	getSubscriptions	Rückgabe der Subscription, die durch `SubscriptionId` identifiziert wird. Wurde keine `SubscriptionId` angegeben, werden alle Subscriptions zurückgegeben, die dem angegebenen Clientsystem zugeordnet werden können.

Die Fehlerfälle der Operation GetSubscription sind in Tabelle 141 TAB_KON_579 Fehlerfälle GetSubscription dargestellt:

Tabelle 141 TAB_KON_579 Fehlerfälle 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

⌫

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 142 TAB_KON_580 Konfigurationswerte des Systeminformationsdienstes (Administrator)

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
EVT_MAX_TRY	Nummer	Der Administrator MUSS die Anzahl der Fehlversuche bzgl. Anmeldeversuchen bzw. Ereigniszustellungen festlegen können, welche bei Erreichen zu einem Auto-Unsubscribe führen.

⌫

⌦ **TIP1-A_4612 Maximale Anzahl an Subscriptions**

Der Konnektor MUSS eine Mindestmenge von 999 Subscriptions insgesamt unterstützen. Der Konnektorhersteller kann jedoch die Anzahl der maximal möglichen Subscriptions (insgesamt und/oder pro Ziel) festlegen. ⌫

⌦ **TIP1-A_4613 Initialisierung Subscriptions-Liste beim Bootup**

Der Konnektor MUSS beim Bootup mit einer leeren Liste an Subscriptions starten. ⌫

4.1.7 Verschlüsselungsdienst

Der Verschlüsselungsdienst bietet Schnittstellen zum hybriden und symmetrischen Ver- und Entschlüsseln von Dokumenten an.

Der Verschlüsselungsdienst bietet für alle Alle_DocFormate die hybride und symmetrische Ver- und Entschlüsselung nach dem Cryptographic Message Syntax (CMS) Standard an [RFC5652].

Darüber hinaus werden folgende formaterhaltende Ver-/Entschlüsselungsmechanismen unterstützt:

- hybride Ver-/Entschlüsselung von XML-Dokumenten nach der W3C Recommendation „XML Encryption Syntax and Processing“ [XMLEnc]
- hybride Ver-/Entschlüsselung von MIME-Dokumenten nach dem S/MIME-Standard [S/MIME]

Der Konnektor muss bezüglich der zur Ver- und Entschlüsselung von Dokumenten verwendeten Verfahren und Algorithmen die Vorgaben in [gemSpec_Krypt#3.1.4] sowie in [gemSpec_Krypt#3.1.5] erfüllen.

4.1.7.1 Funktionsmerkmalweite Aspekte

⌦ **TIP1-A_4614 Missbrauchserkennung Verschlüsselungsdienst**

Der Konnektors MUSS zur Unterstützung von Missbrauchserkennungen die in Tabelle TAB_KON_581 gelisteten Operationen als Einträge in EVT_MONITOR_OPERATIONS berücksichtigen.

Tabelle 143 TAB_KON_581 Verschlüsselungsdienst-Operationen für EVT_MONITOR_OPERATIONS

Operationsname	OK_Val	NOK_Val	Alarmwert (Default-Grenzwert 10 Minuten-Σ)
`EncryptDocument`	1	5	101
`DecryptDocument`	1	5	101

⌫

⌦ TIP1-A_4615 Enc: Keine Onlineprüfung bei Logischer Separation

Ist MGM_LOGICAL_SEPARATION=Enabled DARF der Verschlüsselungsdienst des Konnektors für Aufrufe an der Außenschnittstelle NICHT den Revocationsstatus von Zertifikaten prüfen, d. h. der Konnektor muss sich nach Außen verhalten, als wäre MGM_LU_ONLINE=Disabled.⌫

⌦ 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].⌫

4.1.7.2 Durch Ereignisse ausgelöste Reaktionen

Keine.

4.1.7.3 Interne TUCs, nicht durch Fachmodule nutzbar

Keine.

4.1.7.4 Interne TUCs, auch durch Fachmodule nutzbar

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.

4.1.7.4.1 TUC_KON_070 “Daten hybrid verschlüsseln”

⌦ TIP1-A_4616 TUC_KON_070 “Daten hybrid verschlüsseln”

Der Konnektor MUSS den technischen Use Case TUC_KON_070 “Daten hybrid verschlüsseln” umsetzen.

Tabelle 144: TAB_KON_739 - TUC_KON_070 „Daten hybrid verschlüsseln“

Element	Beschreibung
Name	TUC_KON_070 ”Daten hybrid verschlüsseln”

Element	Beschreibung
Beschreibung	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. 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. Es werden die folgenden formaterhaltenden Verschlüsselungsverfahren für die genannten Dokumententypen unterstützt: • XML mit [XMLEnc] • MIME mit [S/MIME] Des Weiteren ist für alle unterstützten Dokumentformate (`Alle_DocFormate`) die Verschlüsselung mit CMS [RFC5652] möglich.
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	• Zu verschlüsselndes Dokument (Document) • X.509v3-Zertifikate oder öffentliche Schlüssel (EncryptionCertificates or EncryptionKeys) Unterstützte Karten sind SM-B, HBAx und eGK. • Verschlüsselungsverfahren (EncryptionType) Angabe zum einzusetzenden Verschlüsselungsverfahren (CMS, XMLEnc oder S/MIME). • CardSession (Kartensitzung) und Zertifikatsreferenz (falls ein Zertifikat von einer Karte gelesen werden soll) Unterstützte Karten sind SM-B, HBAx und eGK. Darüber hinaus werden die folgenden, vom Dokumentformat und dem Verschlüsselungsverfahren abhängigen Eingangsdaten benötigt: <u>Bei Verschlüsselung von XML-Dokumenten mit XMLEnc</u>: • Festlegung der zu verschlüsselnden Teile des Dokumentes durch Spezifikation eines XPath-Ausdruckes (XML-Elements). • Angabe, ob die KeyInfo in das XML-Dokument eingebettet oder separat an den Aufrufer zurückgegeben werden soll
Komponenten	Konnektor, Kartenterminal, Karte
Ausgangsdaten	• Verschlüsseltes Dokument • Verschlüsselte symmetrische Schlüssel (wenn diese nicht im verschlüsselten Dokument enthalten sind) • OCSP-Checked (True/False, default=True) <u>Bei Verschlüsselung von XML-Dokumenten mit XMLEnc:</u> • KeyInfo (falls nicht ins Dokument eingebettet)
Standardablauf	1. Das Verschlüsselungsverfahren wird anhand des Eingangsparameters EncryptionType gewählt. 2. <u>Nur für XMLEnc</u>: 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

Element	Beschreibung
	ineinander verschachtelt sein. Sind die zu verschlüsselnden XML-Elemente ineinander verschachtelt, so wird Fehler 4104 gemeldet. 3. Falls ein Zertifikat von einer Karte gelesen werden soll, wird TUC_KON_216 “Lese Zertifikat” aufgerufen. Tritt dabei ein Fehler auf, bricht der TUC ab. 4. Falls Zertifikate übergeben oder referenziert wurden, werden diese durch Aufruf von TUC_KON_037 „Zertifikat prüfen“ geprüft. Als Parameter des TUC-Aufrufs gilt für Zertifikate aus CERT_IMPORTED_CA_LIST: {Zertifikat; not_required; ; true; ; keyEncipherment; NONE} für alle anderen Zertifikate gilt: {[C.CH.ENC / C.CH.ENCV / C.HCI.ENC / C.HP.ENC]; not_required; ; true; (oid_egk_enc / oid_egk_encv / oid_smc_b_enc / oid_hba_enc); (keyEncipherment / keyEncipherment / keyEncipherment / keyEncipherment&dataEncipherment); OCSP}. Sollte die Prüfung eines der Zertifikate dieses als nicht gültig ausweisen, bricht der TUC ab. Wenn eine Online-Prüfung des Status der Zertifikate nicht erfolgen konnte wird OCSP-Checked=False gesetzt, der Fehlercode 1028 (Warning CHECK_REVOCATION_FAILED gemäß gemSpec_PKI#Tab_PKI_274) in die GERROR:Trace-Struktur aufgenommen und mit Schritt 5 fortgefahren. 5. Die öffentlichen Schlüssel werden aus den Zertifikaten extrahiert, falls sie nicht direkt übergeben wurden. Falls ein Schlüssel keinen der zugelassenen Verschlüsselungsalgorithmen gemäß [gemSpecKrypt#3.5.2] erlaubt, wird Fehler 4200 gemeldet. 6. Der Konnektor generiert einen symmetrischen Schlüssel. Dabei muss der symmetrische Schlüssel den Kriterien aus [gemSpec_Krypt#2.4] entsprechen. 7. Der Konnektor verschlüsselt das Dokument oder Teile des Dokuments mit dem generierten symmetrischen Schlüssel. a. <u>CMS:</u> Es MÜSSEN die Vorgaben aus [gemSpec_Krypt#3.5.1] beachtet werden. b. <u>XMLEnc</u>: Alle zu verschlüsselnden XML-Elemente werden mit demselben symmetrischen Schlüssel verschlüsselt. Dabei MÜSSEN die Vorgaben aus [gemSpec_Krypt#3.1.4] beachtet werden. 8. Der symmetrische Schlüssel wird asymmetrisch für die einzelnen Identitäten verschlüsselt. Dabei müssen die Vorgaben aus [gemSpec_Krypt#3.1.5; 3.5.2] beachtet werden. 9. Das Zieldokument wird erstellt. <u>XMLEnc</u>: 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 145: TAB_KON_073 Vorgaben zum Format verschlüsselter XML-Dokumente. Die verschlüsselten Datenelemente (EncryptedData) werden erstellt. EncryptedData ersetzt jeweils das zu verschlüsselnde Element des XML-Dokuments. In [COMMON_PKI] wird die Verwendung des Attributs Type in EncryptedData ausgeschlossen; diese Spezifikation

Element	Beschreibung
	sieht jedoch dessen Verwendung für verschlüsselte XML-Bestandteile (element, content) wie in [XMLEnc] beschrieben vor. Der Namespace von EncryptedData ist als http://www.w3.org/2001/04/xmlenc# anzugeben. Für das Element EncryptedData wird das Sub-Element EncryptionMethod mit Angaben zum Verschlüsselungsalgorithmus als obligatorisch vorgegeben, ebenso die Elemente KeyInfo und CipherData. Das Element EncryptedData/KeyInfo hat den Namespace "http://www.w3.org/2000/09/xmldsig#". 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 base64-kodiert und darin DER-kodiert abgelegt. 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. <u>CMS:</u> Format und Inhalt des verschlüsselten Dokuments SOLL [COMMON_PKI#Part 3] folgen. Die Content Type Option „Enveloped-data Content Type“ ist 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. ContentType = OID {… pkcs-7 envelopedData} = 1.2.840.113549.1.7.3 10. Der symmetrische Schlüssel wird aktiv gelöscht (überschrieben).
Varianten/Alternativen	<u>Falls das Verschlüsselungszertifikat</u> übergeben wurde, entfällt das Lesen des Zertifikats von der Karte. <u>Zur Rückgabe der Hybridschlüssel</u> MUSS auch die Variante vorgesehen werden, dass die Hybridschlüssel („KeyInfo“) nicht eingebettet im Zieldokument zurückgegeben werden, sondern separat. <u>Im Fall des Verschlüsselungsverfahrens S/MIME</u> 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. Der S/MIME-Vorbereitungsschritt bereitet das übergebene MIME-Dokument gemäß [S/MIME#3.1] auf die nachfolgende CMS-Verschlüsselung durch

Element	Beschreibung
	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. Im S/MIME-Nachbereitungsschritt wird das im Standardablauf erzeugt CMS-Objekt in eine MIME-Nachricht vom Typ „application/pkcs7-mime“ eingebettet. Sämtliche Header-Felder der Nachricht MÜSSEN in die Header-Felder der S/MIME-Nachricht übernommen werden. Es MUSS ein neues message-id Element für den S/MIME-Header generiert werden. "MIME-Version: 1.0" MUSS definiert sein. Das Feld "Subject" MUSS mit "Subject: Verschlüsselte Nachricht" überschrieben werden. Die Codierung des verschlüsselten Inhalts der Nachricht MUSS in "base64" erfolgen. Entsprechend ist das zugehörige Header-Feld zu füllen: "Content-Transfer-Encoding: base64". Das Feld "Content-Type:" ist als "application/pkcs7-mime" zu definieren. Die weiteren Attribute dieses Feldes sind: • "smime-type=enveloped-data;" • "name=$dateiname", wobei $dateiname auf ".p7m" endet. Das Feld "Content-Disposition" definiert den Inhalt der Nachricht als Dateianhang: "Content-Disposition: attachment; filename=$dateiname" <u>Zu Schritten 5 und 8 für TI-fremde X.509-Zertifikate</u> 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. 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] erfüllen.
Fehlerfälle	Siehe Tabelle 146: TAB_KON_740 Übersicht Fehlercodes für „Daten hybrid verschlüsseln“. Wenn im Ablauf des TUCs ein anderer Fehler als die in Tabelle 146 beschriebenen Fehler auftritt, wird Fehler 4105 gemeldet.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	Abbildung 13: PIC_KON_058 Aktivitätsdiagramm „Daten hybrid verschlüsseln“

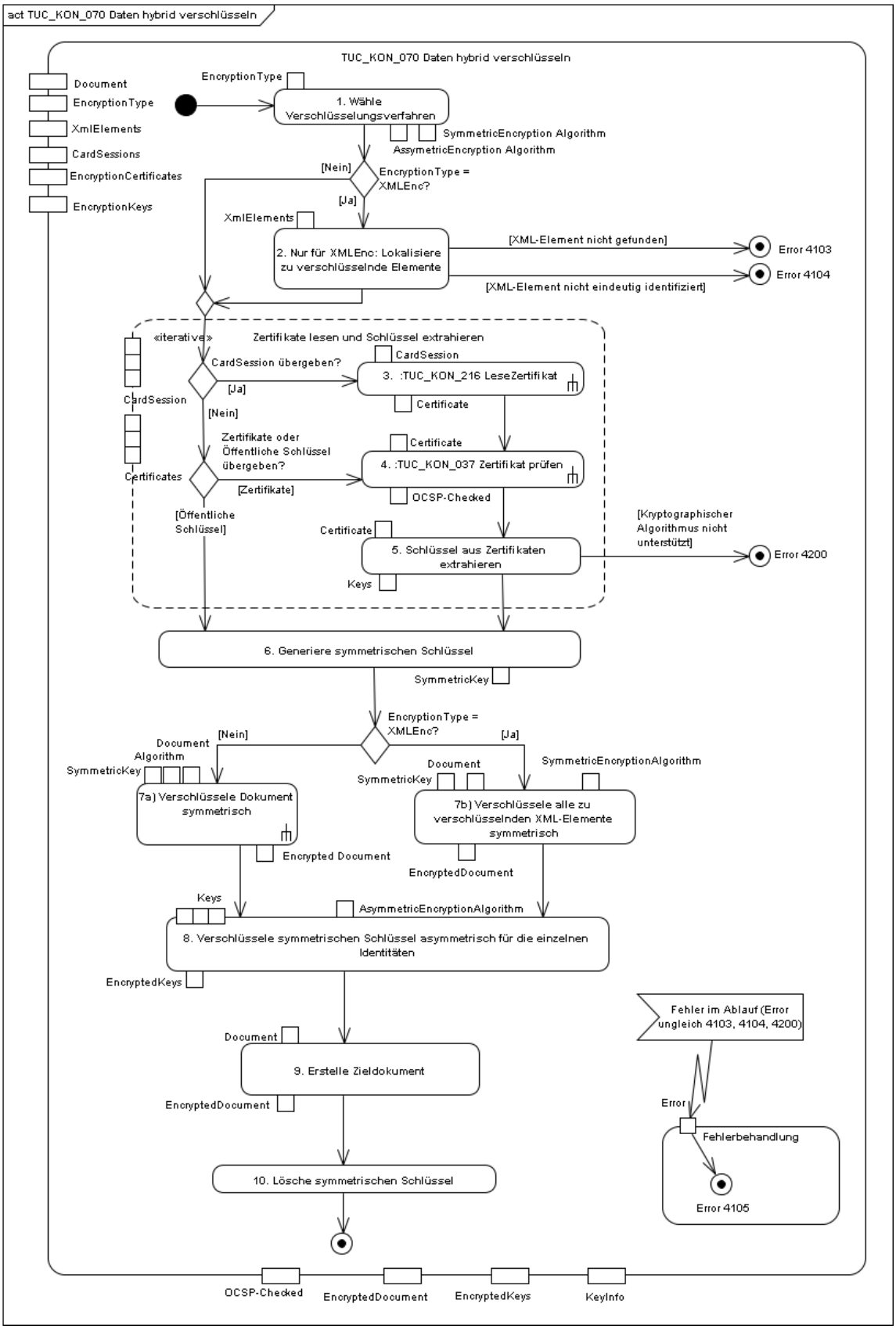


Abbildung 13: PIC_KON_058 Aktivitätsdiagramm „Daten hybrid verschlüsseln“

Tabelle 145: 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.

Tabelle 146: TAB_KON_740 Übersicht Fehlercodes für „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

⌫

4.1.7.4.2 TUC_KON_071 “Daten hybrid entschlüsseln”

⌦ TIP1-A_4617 TUC_KON_071 „Daten hybrid entschlüsseln“

Der Konnektor MUSS den technischen Use Case TUC_KON_071 “Daten hybrid entschlüsseln” umsetzen.

Tabelle 147: 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. 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.

Element	Beschreibung
	Ein konform zu TUC_KON_070 hybrid verschlüsseltes Dokument liegt vor. 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	• Zu entschlüsselndes Dokument (EncryptedDocument) • CardSession (Kartensitzung) mit Referenz auf den privaten Schlüssel (KeyReference) Unterstützt werden SM-B, HBAx und eGK mit der jeweiligen C.ENC-KeyReference. • Verschlüsselungszertifikat bzw. eine Referenz auf das Zertifikat auf obiger Karte passend zur Schlüsselreferenz (optional). • Hybrid verschlüsselter symmetrischer Schlüssel (optional, falls nicht in EncryptedDocument enthalten) Darüber hinaus werden die folgenden, vom Dokumentformat und dem Verschlüsselungsverfahren abhängigen Eingangsdaten benötigt: Bei XML-Dokumenten: • Angabe der zu entschlüsselnden Teile des XML-Dokuments (XmlElements)
Komponenten	Konnektor, Kartenterminal, Karte
Ausgangsdaten	Unverschlüsseltes Dokument Bei XML-Dokumenten: Das EncryptedData-Element ist durch das entschlüsselte ersetzt.
Standardablauf	1. Das Verfahren zum Entschlüsseln wird entsprechend dem Format des übergebenen zu entschlüsselnden Dokuments (EncryptedDocument) gewählt. Der Konnektor SOLL beim asymmetrischen Anteil der Entschlüsselung hybrid verschlüsselter Dokumente sowohl RSAES-OAEP als auch RSAES-PKCS1-v1-5 unterstützen. 2. XMLEnc: 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. Ist in einem EncryptedData Element ein vom Konnektor nicht unterstützter Mechanismus spezifiziert, wird Fehler 4201 gemeldet. 3. Falls erforderlich, wird TUC_KON_216 „Lese Zertifikat“ aufgerufen, um das Zertifikat von der Karte zu lesen. 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.) 4. XMLEnc: 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. 5. Es wird TUC_KON_219 „Entschlüssele“ aufgerufen, um den symmetrischen Schlüssel mit Hilfe des angegebenen privaten Schlüssels zu entschlüsseln. 6. Mit dem symmetrischen Schlüssel wird der unverschlüsselte

Element	Beschreibung
	Dateninhalt wiederhergestellt. 6.1 XMLEnc: Das EncrpytedData Element wird durch die entschlüsselten Daten ersetzt. 7. Der symmetrische Schlüssel wird aktiv gelöscht (überschrieben).
Varianten/Alternativen	Zu 6.: Zur Unterstützung von Bestandssystemen werden, neben den für den symmetrischen Teil der hybriden Verschlüsselung vorgeschriebenen kryptographischen Algorithmen, für den symmetrischen Teil der hybriden Entschlüsselung auch folgende Algorithmen unterstützt (siehe [gemSpec_Krypt#3.5.1]): o AES-128 GCM o AES-192 GCM
Fehlerfälle	Siehe Tabelle 148: TAB_KON_142 Übersicht Fehlercodes für „Daten hybrid entschlüsseln“. Wenn im Ablauf des TUCs ein anderer Fehler als die in Tabelle 148: TAB_KON_142 Übersicht Fehlercodes für „Daten hybrid entschlüsseln“ beschriebenen Fehler auftritt, wird Fehler 4107 gemeldet.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	Abbildung 14: PIC_KON_059 Aktivitätsdiagramm „Daten hybrid entschlüsseln“

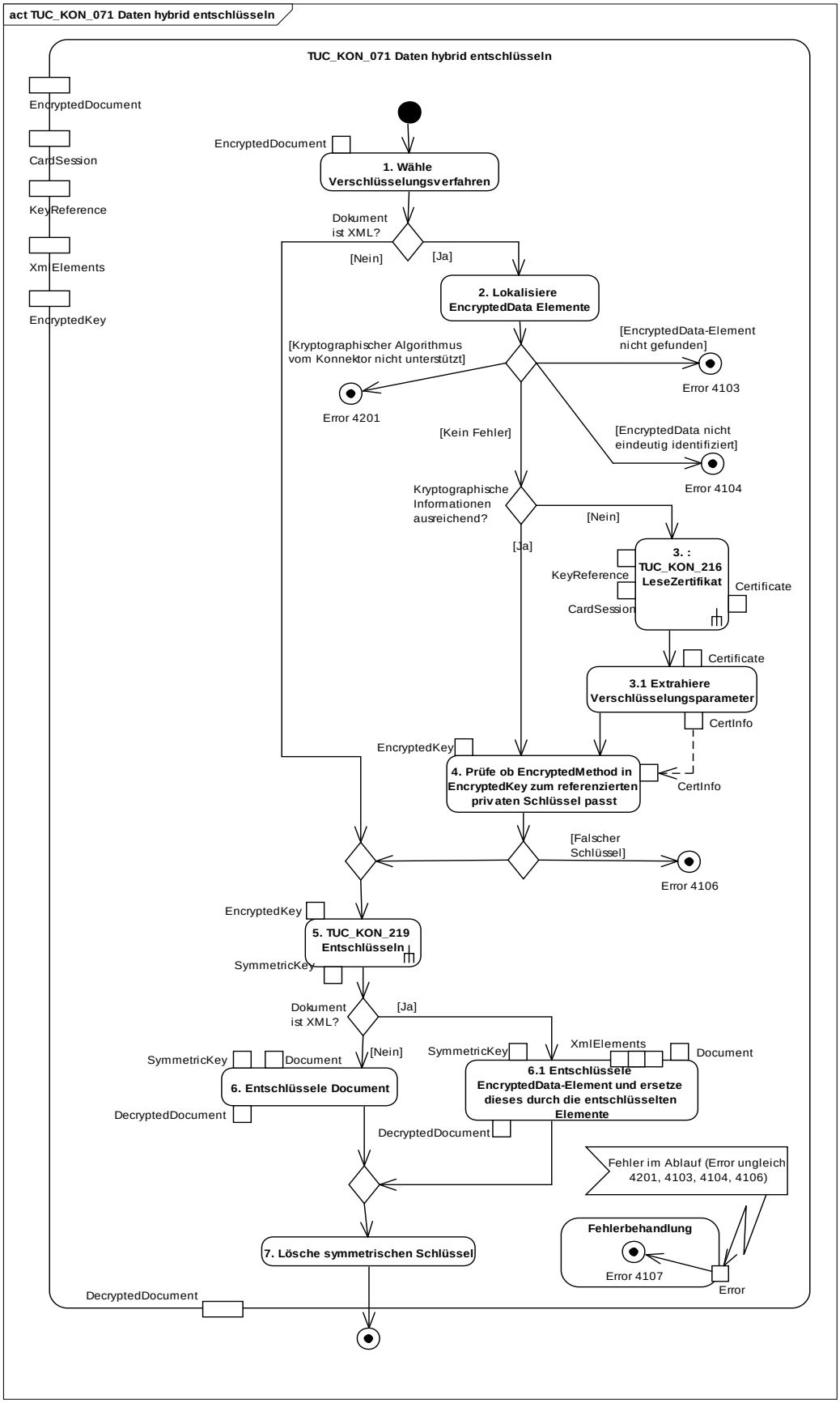


Abbildung 14: PIC_KON_059 Aktivitätsdiagramm „Daten hybrid entschlüsseln“

Tabelle 148: TAB_KON_142 Übersicht Fehlercodes für „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 149: 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	• Zu verschlüsselndes Dokument. • Symmetrischer Schlüssel (optional)
Komponenten	Konnektor
Ausgangsdaten	• Verschlüsseltes Dokument • Erzeugter symmetrischer Schlüssel (optional)

Element	Beschreibung
Standardablauf	1. Wurde kein symmetrischer Schlüssel übergeben, so wird ein Schlüssel erzeugt. Die Qualität des Schlüssels muss den Vorgaben in [gemSpec_Krypt#2.2] genügen. 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. ContentType = OID{… pkcs-7 encryptedData} = 1.2.840.113549.1.7.6 Die symmetrische Verschlüsselung binärer Daten erfolgt nach den Vorgaben gemäß [gemSpec_Krypt#3.6]. Falls die Verschlüsselung fehlschlägt, wird Fehler 4108 gemeldet. 3. Das verschlüsselte Dokument und der symmetrische Schlüssel (falls dieser erzeugt wurde) werden zurückgeliefert.
Varianten/Alternativen	keine
Fehlerfälle	Siehe Standardablauf.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 150: TAB_KON_742 Übersicht Fehlercodes für „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

⌫

4.1.7.4.4 TUC_KON_073 “Daten symmetrisch entschlüsseln”

⌦ TIP1-A_4619 TUC_KON_073 “Daten symmetrisch entschlüsseln”

Der Konnektor MUSS den technischen Use Case TUC_KON_073 “Daten symmetrisch entschlüsseln“ umsetzen.

Tabelle 151: 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

Element	Beschreibung
Vorbedingungen	keine
Eingangsdaten	• Verschlüsseltes Dokument • Symmetrischer Schlüssel
Komponenten	Konnektor
Ausgangsdaten	• Entschlüsseltes Dokument
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

Tabelle 152: TAB_KON_744 Übersicht Fehlercodes für „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

⌫

4.1.7.5 Operationen an der Außenschnittstelle

⌦ TIP1-A_4620 Basisdienst Verschlüsselungsdienst

Der Konnektor MUSS für Clients einen Basisdienst Verschlüsselungsdienst anbieten.

Tabelle 153: TAB_KON_745 Basisdienst Verschlüsselungsdienst

Name	`EncryptionService`
Version (KDV)	wird im Produktsteckbrief des Konnektors definiert
Namensraum	Siehe Anhand D
Namensraum-Kürzel	`CRYPT` für Schema und `CRYPTW` für WSDL

Operationen	Name	Kurzbeschreibung
	`EncryptDocument`	Dokument hybrid verschlüsseln
	`DecryptDocument`	Dokument hybrid entschlüsseln
WSDL	`EncryptionService.wsdl`	
Schema	`EncryptionService.xsd`	

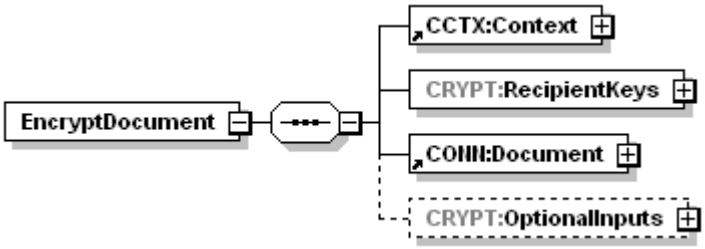
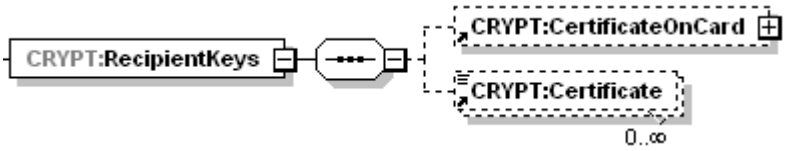
⌫

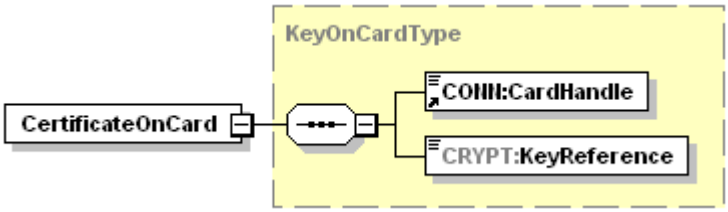
4.1.7.5.1 EncryptDocument

⌦ TIP1-A_4621 Operation EncryptDocument

Der Basisdienst Verschlüsselungsdienst des Konnektors MUSS an der Clientschnittstelle eine Operation EncryptDocument anbieten**.**

Tabelle 154: TAB_KON_071 Operation EncryptDocument

Name	`EncryptDocument`
Beschreibung	Diese Operation verschlüsselt ein übergebenes Dokument hybrid. Es werden die Dokumententypen `Alle_DocFormate` 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. Es werden die folgenden Karten unterstützt: HBAx und SM-B. Die Operation EncryptDocument DARF das Verschlüsseln mit der eGK NICHT unterstützen. Bei XML-Dokumenten werden ein oder mehrere XML-Elemente des Dokumentes verschlüsselt. Für alle übrigen Dokumenttypen wird immer das gesamte Dokument verschlüsselt.
	
	Name: Beschreibung
	`Context`: Aufrufkontext: - MandantID, ClientSystemID, WorkplaceId verpflichtend - UserID verpflichtend bei HBAx, bei SM-B nicht ausgewertet
	

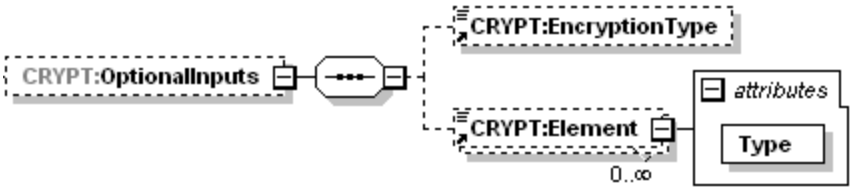


Das RecipientKeys-Element identifiziert die Empfänger der zu verschlüsselnden Nachricht über X.509-Zertifikate (öffentliche Schlüssel). Quelle für die Zertifikate kann eine gesteckte Karte sein, die per CertificateOnCard-Element referenziert wird, oder der Aufrufer, der X.509-Zertifikate im Certificate-Element übergibt.

`CardHandle`	Identifiziert die zu verwendende Karte mit dem (öffentlichen) Schlüssel. Ist das Element nicht vorhanden, so werden nur Zertifikate per Element `Certificate` übergeben.
`KeyReference`	Der Wert dieses Parameters ist in Tabelle 157: TAB_KON_747 KeyReference für Encrypt-/DecryptDocument spezifiziert. Ist der Parameter nicht angegeben oder leer, gilt der Default-Wert C.ENC.
`Certificate`	`Certificate` ist ein Base64-kodiertes XML-Element, in dem das Zertifikat, das den asymmetrischen Schlüssel enthält (öffentlicher Schlüssel), DER-kodiert übergeben wird. 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 spezifizierte 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

`EncryptionType`	Zu wählendes Verschlüsselungsverfahren, wobei folgende URI vorgesehen sind: • XMLEnc: „http://www.w3.org/TR/xmlenc-core1/" • CMS: „urn:ietf:rfc:5652" • S/MIME: "urn:ietf:rfc:5751"

		Im Fall XMLEnc wird ein Base64-codiertes XML-Dokument im Element `CONN:Document/CONN:Base64XML` übergeben. In den Fällen CMS und S/MIME wird ein Base64-codiertes Binär-Dokument im Element `CONN:Document/dss:Base64Data` übergeben oder ein XML-Dokument im Element `CONN:Document/CONN:Base64XML`. Ist der Parameter EncryptionType nicht gesetzt, dann gilt folgendes Default-Verhalten: Für ein im Element `CONN:Document/CONN:Base64XML` übergebenes XML-Dokument wird als Verschlüsselungsverfahren [XMLEnc] angewandt, und für ein im Element `CONN:Document/dss:Base64Data` übergebenes Dokument wird das Verschlüsselungsverfahren CMS angewandt. Im Fall S/MIME ist das in `CONN:Document/dss:Base64Data` übergebene Dokument eine MIME-Nachricht.
	Element	

	`Element`	Dieses möglicherweise mehrfach auftretende Element ist nur relevant für XML-Dokumente.
		XPath Ausdruck, der das Element ermittelt, welches verschlüsselt werden soll. Der Ausdruck darf nur ein Element-Node des XML-Dokumentes als Ergebnis liefern. Dieses Element wird verschlüsselt.
		
		Das XML-Attribut Type kann einen der Werte
		`http://www.w3.org/2001/04/xmlenc#Element`
		`http://www.w3.org/2001/04/xmlenc#Content`
		annehmen. Gemäß XMLEnc steuert der Parameter, ob das gesamte Element oder nur sein Content verschlüsselt wird.
		Wird der Parameter weggelassen, so wird das Root-Element, d. h. das gesamte Dokument verschlüsselt. In diesem Fall ist Type
		`http://www.w3.org/2001/04/xmlenc#Element`
		anzusetzen.
		
		Sind mehrere Elemente angegeben, so darf keines der Elemente unter den angegebenen Elementen Vorfahren haben, was sicherstellt, dass keine zu signierenden Dokumententeile überlappen.

	UnprotectedProperties	
	`CRYPT:UnprotectedProperties`	Dieses optionale Element wird im CMS-Fall (EncryptionType = urn:ietf:rfc:5652) ausgewertet. Die Elemente `./UnprotectedProperties/Property/Value/CMSAt`

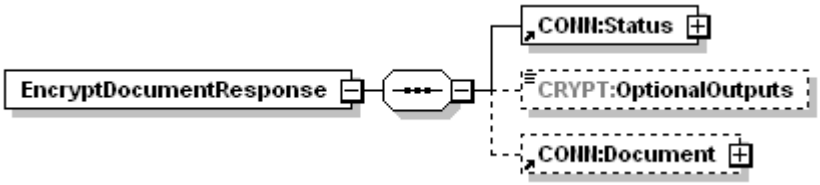
		`tribute` müssen base64/DER-kodiert ein vollständiges ASN.1-`Attribute` enthalten, definiert in [CMS#6.1.EnvelopedData Type]. Es muss bei der Erstellung des CMS-Containers unter `UnprotectedAttributes` aufgenommen werden. Das zugehörige Element `./UnprotectedProperties/Property/Identifier` wird nicht ausgewertet.
Rückgabe		
	`Status`	Enthält den Ausführungsstatus der Operation.
	`CRYPT:OptionalOutputs`	Kann – in zukünftigen Versionen der Spezifikation – optionale Ausgabeparameter enthalten.
	`CONN:Document`	Enthält das verschlüsselte Dokument in base64-codierter Form, wenn die Verschlüsselung erfolgreich durchgeführt wurde. Im Fall XMLEnc wird das Base64-codierte verschlüsselte XML-Dokument im Element `CONN:Document/CONN:Base64XML` zurückgegeben. Im Fall CMS wird das Base64-codierte Binär-Dokument im Element `CONN:Document/dssBase64Data` zurückgegeben. Im Fall S/MIME wird die Base64-codierte S/MIME-Nachricht im Element `CONN:Document/dssBase64Data` zurückgegeben. Das Attribut `CONN:Document/dssBase64Data/ @MimeType` wird auf „application/pkcs7-mime“ gesetzt. Die S/MIME-Nachricht hat Content-Transfer-Encoding: base64.
Fehler	Bei Auftreten eines Fehlers im Standardablauf werden Fehlercodes entsprechend TAB_KON_141 gemeldet.	
Vorbedingungen	Keine	
Nachbedingungen	Keine	

Tabelle 155: TAB_KON_141 Übersicht Fehlercodes für „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

Fehlercode	ErrorType	Severity	Fehlertext
4058	Security	Error	Aufruf nicht zulässig

Der Ablauf der Operation EncryptDocument ist in Tabelle 156: TAB_KON_746 Ablauf EncryptDocument beschrieben:

Tabelle 156: TAB_KON_746 Ablauf EncryptDocument

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 Zugriffs-berechtigung“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; $context.userId; $cardHandle} 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 { MandantId, CsId, CardHandle, UserId }
4.	TUC_KON_070 „Daten hybrid verschlüsseln“	Die hybride Verschlüsselung wird durchgeführt. Tritt hierbei ein Fehler auf, bricht die Operation ab. Die KeyInfo, d.h. die Liste der Hybridschlüssel inklusive des bei ihrer Erzeugung verwendeten Zertifikates, sind dabei in das Dokument einzubetten.

⌫

Die folgende Tabelle führt zulässige Werte für den Parameter KeyReference auf:

Tabelle 157: TAB_KON_747 KeyReference für Encrypt-/DecryptDocument

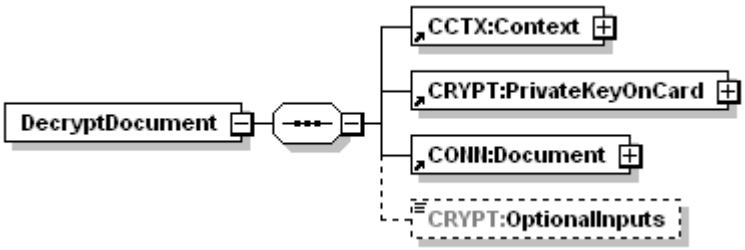
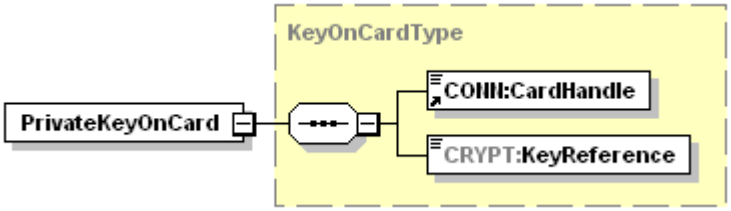
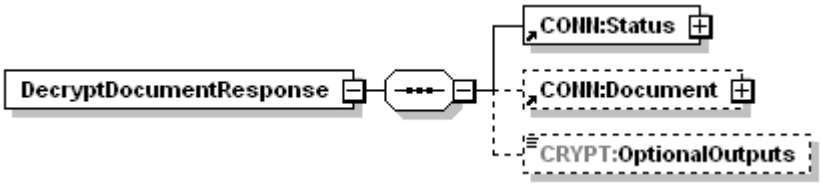
Karte	KeyReference	Zertifikat (Encrypt)	Schlüssel (Decrypt)
HBA-VK	C.ENC	EF.C.HP.ENC in DF.ESIGN	PrK.HP.ENC in DF.ESIGN
HBA	C.ENC1.R2048	EF.C.HP.ENC1.R2048 in DF.ESIGN	PrK.HP.ENC1.R2048 in DF.ESIGN
SM-B	C.ENC1.R2048	EF.C.HCI.ENC1.R2048 in DF.ESIGN	PrK.HCI.ENC1.R2048 in DF.ESIGN

4.1.7.5.2 DecryptDocument

⌦ TIP1-A_4622 Operation DecryptDocument

Der Basisdienst Verschlüsselungsdienst des Konnektors MUSS an der Clientschnitt-stelle eine Operation DecryptDocument anbieten.

Tabelle 158: TAB_KON_075 Operation DecryptDocument

Name	`DecryptDocument`	
Beschreibung	Diese Operation entschlüsselt ein hybrid verschlüsseltes Dokument. Es werden die Dokumententypen `Alle_DocFormate` unterstützt. Für die Entschlüsselung wird ein asymmetrischer Schlüssel zu einem X.509v3-Zertifikat genutzt. Dieses Zertifikat und der Schlüssel müssen von einer Karte kommen.	
Aufrufparameter		
	Name	Beschreibung
	`Context`	Aufrufkontext: - MandantId, ClientSystemId, WorkplaceId verpflichtend - UserId verpflichtend bei HBAx, bei SM-B nicht ausgewertet
		
	`PrivateKeyOnCard`	Identifiziert die zu verwendende Karte mit dem (privaten) Schlüssel. 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 157: TAB_KON_747 KeyReference für Encrypt-/DecryptDocument spezifiziert. Ist der Parameter nicht angegeben oder leer, gilt der Default-Wert C.ENC.
	`CONN:Document`	Enthält das base64-codierte Dokument, das entschlüsselt werden soll.
	`CRYPT:OptionalInputs`	Kann – in zukünftigen Versionen der Spezifikation – optionale Ausgabeparameter enthalten.
Rückgabe		

	`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
Fehler	Bei Auftreten eines Fehlers im Standardablauf werden Fehlercodes entsprechend TAB_KON_145 gemeldet.	
Vorbedingungen	Keine	
Nachbedingungen	Keine	

Tabelle 159: TAB_KON_145 Übersicht Fehlercodes für „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

Der Ablauf der Operation DecryptDocument ist in Tabelle 160: TAB_KON_076 Ablauf DecryptDocument beschrieben:

Tabelle 160: TAB_KON_076 Ablauf DecryptDocument

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 Zugriffs-berechtigung“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; $context.userId; $cardHandle} 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 { MandantId, CsId, CardHandle, UserId }
4.	TUC_KON_071 Daten hybrid entschlüsseln	Die Entschlüsselung wird durchgeführt.

⌫

4.1.7.6 Betriebsaspekte

keine

4.1.8 Signaturdienst und SAK

Der Signaturdienst bietet Clientsystemen und Fachmodulen eine Schnittstelle zum Signieren von Dokumenten und Prüfen von Dokumentensignaturen und zum Signieren von Binärstrings zum Zweck der externen Authentisierung.

Innerhalb des Signaturdienstes werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): *keine Events vorhanden*
- Konfigurationsparameter: „SAK_“

4.1.8.1 Funktionsmerkmalweite Aspekte

4.1.8.1.1 Dokumentensignatur

Der Signaturdienst umfasst die Funktionalität der nicht-qualifizierten elektronischen Signatur (nonQES) mit der SM-B, sowie die qualifizierte elektronische Signatur (QES) mit dem HBA und den HBA-Vorläuferkarten HBA-qSig und ZOD_2.0 (=HBAx).

In der Abbildung fachlicher Abläufe kann es nötig sein, ein Dokument mehrfach parallel zu signieren, oder existierende Signaturen gegenzusignieren. Der Konnektor unterstützt **parallele Signaturen** (QES und nonQES). Ebenso unterstützt er Gegensignaturen (QES und nonQES), die jeweils alle bestehenden Signaturen gegensignieren. Die angebotene Möglichkeit des Gegensignierens bezieht sich dabei auf das Signieren aller vorhandenen parallelen Signaturen, während ein Gegensignieren von Gegensignaturen nicht angeboten wird. Zwei Arten der Gegensignatur werden unterstützt, eine **dokumentinkludierende Gegensignatur**, bei der das Dokument und alle Signaturen gegensignieren werden, sowie eine **dokumentexkludierende Gegensignatur**, bei der alle Signaturen gegensignieren werden, aber nicht der fachliche Inhalt des Dokumentes selbst.

⌦ TIP1-A_4623 Unterstützte Signaturverfahren nonQES

Der Signaturdienst MUSS für die Erstellung und Prüfung von nicht-qualifizierten elektronischen Signaturen (nonQES) für die `nonQES_DocFormate` die Signaturverfahren entsprechend Tabelle 161: TAB_KON_582 – Signaturverfahren unterstützen. ⌫

⌦ TIP1-A_4627 Unterstützte Signaturverfahren QES

Der Signaturdienst MUSS für die Erstellung und Prüfung von qualifizierten elektronischen Signaturen (QES) für die `QES_DocFormate` die Signaturverfahren entsprechend Tabelle 161: TAB_KON_582 – Signaturverfahren unterstützen.⌫

Tabelle 161: TAB_KON_582 – Signaturverfahren Dokumentensignatur

Signaturformat	Standard	Dokumentformate	QES/ nonQES	Bemerkung
XMLDSig (XAdES)	[RFC3275] [XMLDSig] [XAdES] [RFC6931]	XML	QES, nonQES	Hierdurch können abgesetzte (detached), umschließende (enveloping) und eingebettete (enveloped) Signaturen erzeugt werden.
CMS (CAdES)	[RFC5652] [CAdES]	`QES_DocFormate` `nonQES_DocFormate`	QES, nonQES	Hierdurch können abgesetzte (detached) und umschließende (enveloping) Signaturen erzeugt werden.
PDF/A (PAdES)	[PAdES-3]	PDF/A	QES, nonQES	Hierdurch können CMS-basierte Signaturen in PDF/A-Dokumente eingefügt und dadurch eingebettete Signaturen erzeugt werden.
S/MIME	[RFC5751]	`nonQES_DocFormate`	nonQES	Es werden MIME-Nachrichten signiert.

Zu den Begriffen detached, enveloping und enveloped Signaturen siehe beispielsweise auch [HüKo06#Abs. 4.3.3. und 4.3.1.5].

⌦ TIP1-A_5446 Zusätzliche Signaturverfahren für Dokumentensignaturprüfung

Der Signaturdienst MUSS für die Signaturprüfung zusätzlich zu den in „TAB_KON_582 – Signaturverfahren Dokumentensignatur“ geforderten Signaturverfahren auch die Signaturverfahren in „TAB_KON_585 – Zusätzliche Signaturverfahren für Dokumentensignaturprüfung“ unterstützen. ⌫

Tabelle 162: TAB_KON_585 – Zusätzliche Signaturverfahren für Dokumentensignaturprüfung

Signaturformat	Standard	SignatureScheme	QES/ nonQES
PKCS#1 (V2.1)	[RFC3447]	RSASSA-PSS RSASSA-PKCS1-v1_5	QES, nonQES

⌦ TIP1-A_5447 Einsatzbereich der Signaturvarianten

Der Signaturdienst MUSS für die Erstellung und Prüfung von nicht-qualifizierten elektronischen Signaturen (nonQES) und qualifizierten elektronischen Signaturen (QES) die Vorgaben zum Einsatzbereich gemäß Tabelle TAB_KON_778 umsetzen. ⌫

Tabelle 163: TAB_KON_778 - Einsatzbereich der Signaturvarianten für XAdES, CAdES und PAdES

Signaturvarianten				Einsatzbereich		
Signa-turver-fahren	Signatur-variante	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	Ja	Nein	Ja
XAdES	enveloped	gesamtes Input XML-Dokument (= Root-Element mit Subelementen)	Als direktes Child des Root-Elements	Ja	Ja	Ja
XAdES	enveloped	ausgewähltes nicht Root-Element mit Subelementen im Input XML-Dokument	Als direktes Child des ausgewählten Elements	Ja	Nein	Ja
XAdES	enveloping	gesamtes Input XML-Dokument (= Root-Element mit Subelementen)	Im Dokument, das Root-Element umschließend	Ja	Ja	Ja
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

Signaturvarianten				Einsatzbereich		
Signa-turver-fahren	Signatur-variante	WAS wird signiert?	WO wird die Signatur abgelegt?	nonQES	QES Außen-schnittstelle	QES Fachmodul-schnittstelle
CAdES	enveloping	gesamtes Binär-Dokument	innerhalb des CMS-Dokuments	Ja	Ja	Ja
PAdES	-	gesamtes PDF-Dokument	Im PDF-Dokument	Ja	Ja	Ja

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.

⌦ 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 Signaturdienst 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 Signaturdienst bei der Erstellung von qualifizierten elektronischen Signaturen (QES) die Default-Signaturverfahren entsprechend TAB_KON_583 – Default-Signaturverfahren wählen. ⌫

Tabelle 164: 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 Signaturdienst (nonQES)

Der Konnektor MUSS zur Unterstützung von Missbrauchserkennungen die in Tabelle TAB_KON_584 gelisteten Operationen als Einträge in EVT_MONITOR_OPERATIONS berücksichtigen.

Tabelle 165: 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

⌫

⌦ TIP1-A_4626 Sig: Keine Onlineprüfung bei Logischer Separation

Ist MGM_LOGICAL_SEPARATION=Enabled DARF der Signaturdienst und die SAK des Konnektors für Aufrufe an der Außenschnittstelle NICHT den Revocationsstatus von Zertifikaten prüfen, d. h. der Konnektor muss sich nach Außen verhalten, als wäre MGM_LU_ONLINE=Disabled.⌫

⌦ TIP1-A_4629 Unterstützte Karten QES-Erstellung

Der Signaturdienst MUSS für die QES-Erstellung die Kartentypen HBA, HBA-qSig und ZOD_2.0 unterstützen. ⌫

⌦ TIP1-A_4630 QES-Erstellung und Prüfung mit SAK

Der Signaturdienst MUSS zur Erstellung und Prüfung qualifizierter elektronischer Signaturen eine integrierte Signaturanwendungskomponente (SAK) gemäß § 17 Abs. 2 des Signaturgesetzes verwenden. ⌫

⌦ TIP1-A_5436 XML Dokument nach Entfernen der Signatur unverändert

Der Konnektor MUSS die Operation SignDocument für XML-Dokumente so implementieren, dass das Dokument nach Entfernen der Signatur, insbesondere

auch einer Teilsignatur, als Ganzes unverändert ist, wobei zwei XML-Dokumente als identisch zu betrachten sind, wenn sie gemäß Canonical XML 1.1 gleich sind [CanonXML1.1].⌫

4.1.8.1.2 Signaturrichtlinien

Für die sicher anzuzeigenden XML-Dokumentenformate DF_SV_XML werden pro Dokumentenformat die nachfolgenden Aspekte über eine Signaturrichtlinie gekapselt festgelegt:

- Gültigkeitsbereich der Signaturrichtlinie
- XML-Schemas für die Typkonformitätsprüfung (im Konnektor zu hinterlegen)
- XSL-Datei für die Transformation des XML-Dokuments in ein PDF-Dokument zum Zweck der sicheren Anzeige (im Konnektor zu hinterlegen)
- XSL-Datei für die Bildung eines Kurztext ShortTextSecure aus dem XML-Dokument (im Konnektor zu hinterlegen)
- Constraints für den Aufruf der Schnittstelle SignDocument und VerifyDocument, die zur sicherheitstechnischen Härtung der Schnittstelle dienen.

⌦ **TIP1-A_5538 Signaturrichtlinien für Dokumentenformate DF_SV_XML**

Der Konnektor MUSS die Signaturrichtlinien für die Dokumentenformate DF_SV_XML umsetzen.

Die umzusetzenden Signaturrichtlinien lauten

- o SR_DF_SV_VHITG_ARZTBRIEF
  für das Dokumentenformat DF_SV_VHITG_ARZTBRIEF
- o SR_DF_SV_EGES
  für das Dokumentenformat DF_SV_EGES

Der Konnektor MUSS den für jede Signaturrichtlinie definierten Bezeichner (URI) bei der Signatur als SigPolicyId im Feld SignaturePolicyIdentifier einbetten. Bei der Signaturprüfung MUSS der Konnektor über eine etwaig vorhandene SigPolicyId die Signaturrrichtlinie identifizieren.

Die gemäß AdES erforderliche Hash-Referenz über die Policy (SigPolicyHash) MUSS Schema-konform leer gelassen werden. Bei der Signaturprüfung DARF die Hash-Referenz über die Policy NICHT geprüft werden.⌫

Die Signaturrichtlinen sind im Anhang C definiert.

Die Signaturrichtlinie SR_DF_SV_VHITG_ARZTBRIEF basiert auf den Vorgaben zu VHiTG-Arztbrief [CDA]/[CDA-Sig] und eArztbrief [eAB].

Die Signaturrichtlinie SR_DF_SV_EGES basiert auf den Vorgaben zu eGesamtaufstellung [eGES].

4.1.8.1.3 Signaturzeitpunkt

Bezogen auf den vom Konnektor für die Signaturprüfung anzunehmenden Signaturerstellungszeitpunkt (als Interpretation des "angegebenen Zeitpunkts" lt. SigV §15 (2) 2.b) werden in dieser Spezifikation die Bezeichner Ermittelter_Signaturzeitpunkt und Benutzerdefinierter_Zeitpunkt verwendet.

Ermittelter_Signaturzeitpunkt: Vom Konnektor ermittelter Zeitpunkt, zu dem eine Signatur geprüft wird. Es werden folgende Signaturzeitpunkte ermittelt:

1. `Ermittelter_Signaturzeitpunkt_Eingebettet:`
   in der Signatur eingebetteter Zeitpunkt (falls vorhanden)
2. `Ermittelter_Signaturzeitpunkt_Qualifiziert:`
   qualifizierter Zeitstempel über die Signatur (falls vorhanden)
3. `Ermittelter_Signaturzeitpunkt_System:`
   Systemzeit des Konnektors bei Signaturprüfung

Anmerkung: Bei vom Konnektor selbst erstellten Signaturen ist immer ein in der Signatur eingebetteter Zeitpunkt vorhanden, jedoch kein qualifizierter Zeitstempel, da in der TI keine qualifizierten Zeitstempel ausgestellt werden.

Benutzerdefinierter_Zeitpunkt: Vom Benutzer beim Aufruf der Signaturprüfoperation als Parameter an den Konnektor übergebener Zeitpunkt, zu dem eine Signatur geprüft werden soll.

4.1.8.1.4 Extended Trusted Viewer Service

Nach § 17 Abs. 2 des Signaturgesetzes (SigG) [SigG01] bzw. § 15 Abs. 2 Satz 1c, 2a und 2b der Signaturverordnung (SigV) [SigV01] benötigt eine Signaturanwendungskomponente (SAK) für die Erstellung und Prüfung einer qualifizierten elektronischen Signatur eine vertrauenswürdige Anzeigekomponente, welche auch als Trusted Viewer bezeichnet wird.

Neben diesen klassischen Aufgaben einer vertrauenswürdigen Darstellungskomponente umfasst die Funktionalität des Extended Trusted Viewer (xTV) auch das Benutzer-Interface für die Administration der Signaturanwendungskomponente, insbesondere die Verwaltung der signaturkartenbezogenen Konfiguration bei der Stapelsignatur.

⌦ TIP1-A_4631 Bereitstellung des xTV

Der Konnektor MUSS eine vertrauenswürdige Anzeigekomponente (Extended Trusted Viewer) bereitstellen. Der Extended Trusted Viewer (xTV) MUSS auf dem Arbeitsplatz des Benutzers einsetzbar sein.

Wird der xTV auf dem Betriebssystem des Clientsystems realisiert, so MUSS die Realisierung die folgenden Betriebssysteme in ihren 32- und 64-Bit-Varianten unterstützen:

- Windows XP
- Windows Server
- Windows Vista
- Windows 7
- Windows 8

- Linux / Unix (Alle Derivate mit Unterstützung von Java Laufzeitumgebung ab Version 1.6)
- MAC OS X (Java Laufzeitumgebung ab Version 1.6)

Es MUSS eine Terminal-Server-Lösung unterstützt werden.⌫

Da die vertrauenswürdige Anzeigekomponente sinnvoll nur am Arbeitsplatz des Benutzers eingesetzt werden kann, existiert im Kontext dieser Spezifikation eine Aufteilung der SAK in eine Komponente auf dem Konnektor und eine Komponente am Arbeitsplatz des Benutzers. Diese beiden Bestandteile werden im Folgenden als **SAK_{KON}** und **Extended Trusted Viewer (xTV)** bezeichnet.

Die genaue Ausprägung der xTV ist nicht vorgeschrieben. Es ist sowohl eine Software-Komponente möglich, die – zusammen mit dem Clientsystem (bzw. Teilen davon) - auf dem Arbeitsplatzsystem des Benutzers läuft, als auch eine separate Hardware-Komponente.

Ebenso lässt diese Spezifikation den Betrieb einer xTV als Software-Komponente auf einem (Terminal-)Server zu, wobei die Inhalte der xTV durch diese Software-Komponente dann zu den Arbeitsplätzen (Terminals) sicher übertragen werden müssen. Das Informationsmodell des Konnektors unterstützt eine solche Lösung durch die Wahl der Kardinalitäten zwischen den Klassen der xTV und dem Arbeitsplatz.

Die Aufteilung in SAK_{KON} und xTV ist auch für den Fall valide, dass beide Komponenten innerhalb der gleichen Systemgrenzen agieren.

Die Zulassungsbedingungen der gematik für den Konnektor erfordern eine CommonCriteria-Evaluierung entsprechend der zugehörigen Schutzprofile. Für die SAK_{KON} und den xTV ist das Schutzprofil für den AK relevant. Beide Bestandteile der SAK müssen zusammen geprüft und evaluiert werden. Eine Zulassung durch die gematik durch die Ausstellung einer Herstellererklärung – wie sie das SigG § 17 Abs. 4 erlaubt, ist nicht möglich.

⌦ TIP1-A_5150 Anzeige definierter Dokumentenformate im xTV

Der Konnektor MUSS bei der QES-Erstellung und QES-Prüfung grundsätzlich alle gemäß QES_DocFormate definierten Dokumentformate im xTV anzeigen können.

Der Konnektor MUSS Dokumentformate gemäß QES_DF_SecView sicher im xTV anzeigen können.

Bei der Anzeige von QES_DF_NonSecView Dokumenten MUSS der xTV zusätzlich eine Warnung anzeigen, dass das aktuell angezeigte Dokument nicht sicher angezeigt werden kann.⌫

⌦ TIP1-A_4632 Integritätsprüfung Extended Trusted Viewer (xTV)

Der Konnektor MUSS die SAK mit ihren Bestandteilen xTV und SAKKON auf Integrität prüfen und das Ergebnis zuverlässig anzeigen, bevor er den xTV als Anzeigekomponente der SAK verwendet. Wenn diese Integritätsprüfung nicht erfolgreich war, MUSS der Konnektor eine etwaig bestehende Verbindung zwischen xTV und SAK_KON abbauen und DARF den xTV NICHT als Anzeigekomponente der SAK nutzen. ⌫

⌦ TIP1-A_4633 xTV muss Veränderungen sichtbar machen

Der Extended Trusted Viewer MUSS dem Benutzer Veränderungen an der SAK (durch Manipulation oder Updates) über die gesamte Lebensdauer des Konnektors (ab Erstevaluierung der SAK) sichtbar machen.⌫

Für die Anzeige von XML-Dokumenten am xTV stellt der Konnektor einen XSL-basierten Transformationsprozess bereit.

⌦ TIP1-A_5531 xTV - PDF Anzeige von XML-Dokumenten

Der Konnektor MUSS für die Anzeige von XML-Dokumenten einen XSL-Transformationsprozess bereitstellen, der aus einem XML-Dokument und einem XSL-Stylesheet ein PDF-Dokument für die Anzeige erzeugt. Der Konnektor KANN einen Transformationsprozess auf der Basis mehrerer anstelle eines Stylesheets anbieten.

Der Konnektor MUSS für die Anzeige eines Kurztextes zu einem XML-Dokument einen Transformationsprozess bereitstellen, der aus einem XML-Dokument und einem XSL-Stylesheet einen Kurztext erzeugt.

Der Konnektor MUSS XSL-FO, XSLT, XPath unterstützen.

Der Konnektor MUSS XSLT in der Version 2.0 [XSLT], Conformance „basic XSLT processor“, unterstützen.

Der Konnektor MUSS XPath in der Version 2.0 [XPath] unterstützen.

Der Konnektor SOLL XSL-FO gemäß XSL 1.1 [XSL] unterstützen. Von den XSL-FO-Objects und -Properties aus XSL 1.1 MÜSSEN mindestens die in Tabelle TAB_KON_801 aufgeführten unterstützt werden. ⌫

⌦ TIP1-A_5532 xTV - HTML/CSS Anzeige von XML-Dokumenten

Der Konnektor KANN die Anzeige von XML-Dokumenten per HTML/CSS anbieten. Hierfür muss er einen XSL basierten Transformationsprozess bereitstellen, der aus einem XML-Dokument und einem oder mehreren Stylesheet die HTML/CSS-Darstellung erzeugt. Die Anforderungen an die Stylesheets müssen dann so dokumentiert werden, dass eine Erstellung der Stylesheets auf Basis der Dokumentation erfolgen kann.⌫

4.1.8.1.5 *Schnittstelle zwischen SAK (im Konnektor) und xTV*

Zur Kommunikation zwischen den beiden SAK-Komponenten werden hier die Anforderungen an eine Schnittstelle beschrieben, welche vom Konnektorhersteller bereitzustellen ist. Die Inhalte und der Kommunikationsablauf bei der Verwendung dieser Schnittstelle sind herstellerspezifisch, der Nachweis der angemessenen Sicherheit muss im Rahmen der CC-Evaluierung erbracht werden.

⌦ TIP1-A_4634 TLS zwischen SAK im Konnektor und xTV

Die Verbindung zwischen xTV und SAK im Konnektor MUSS mit TLS mit gegenseitiger Authentisierung abgesichert werden. Zur Authentisierung der SAK im Konnektor MUSS die X.509-Identität ID.SAK.AUT genutzt werden. Der xTV MUSS sich mit einer herstellerspezifischen X.509-Identität gegenüber der SAK im

Konnektor authentisieren. Die SAK im Konnektor MUSS jeder Instanz des xTV auf den einzelnen Arbeitsplätzen eine eigene xTV-Identität zuordnen. ⌫

Umsetzungshinweise finden sich in Anhang I.

4.1.8.1.6 *Job-Nummer*

Da die eHealth-Kartenterminals dezentral über eine Netzwerkschnittstelle am Konnektor betrieben werden, fehlt die Möglichkeit zur direkten physischen und vom Anwender kontrollierbaren Zuordnung eines solchen Terminals zu einem Arbeitsplatz, auf dem sich das Clientsystem befindet.

Daher ist es bei einer fehlerhaften Zuordnung eines eHealth-Kartenterminals zu einem Arbeitsplatz möglich, dass die PIN-Eingabeaufforderung – beispielsweise zu einem Signaturauftrag – an ein entferntes Kartenterminal weitergeleitet wird. Diese fehlerhafte Zuordnung kann durch einen Fehler des Clientsystems oder den Versuch eines Angriffes hervorgerufen werden.

⌦ TIP1-A_4639 Generierung von Job-Nummern für PIN-Eingaben

Um Fehler- und Angriffsmöglichkeiten auszuschließen, MUSS der Konnektor bei bestimmten PIN-Verifikationen vor der Aufforderung zur PIN-Eingabe an einem eHealth-Kartenterminal eine hinreichend eindeutige Nummer – die Job-Nummer – generieren, welche den Auftrag kennzeichnet, für dessen Verarbeitung die PIN-Eingabe erfolgen soll. Bei welchen PIN-Verifikationen dies der Fall ist, kann den PIN-Prompts in Tabelle 52: TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal entnommen werden.⌫

⌦ TIP1-A_4640 Anzeige der Job-Nummern für PIN-Eingaben

Diese Job-Nummer MUSS vom Konnektor sowohl im Extended Trusted Viewer auf dem Arbeitsplatz als auch im Display des eHealth-Kartenterminals neben der PIN-Eingabeaufforderung angezeigt werden.⌫

⌦ TIP1-A_4992 Guidance zur Job-Nummer

Das Handbuch des Konnektors MUSS den Benutzer über den korrekten Gebrauch der Job-Nummer informieren. Es MUSS ihm verdeutlichen, dass er seine PIN über die Tastatur des eHealth-Kartenterminals nur eingeben darf, wenn am xTV und am Display des Kartenterminals die gleiche Job-Nummer angezeigt wird. Stimmen die beiden Nummern nicht überein, so soll der Benutzer seine PIN nicht eingeben und stattdessen weitergehende Schritte zur Klärung des aufgetretenen Fehlverhaltens einleiten. ⌫

⌦ TIP1-A_4642 Ableitung der Job-Nummer von einem Zufallswert

Zur hinreichend eindeutigen Kennzeichnung des Vorganges MUSS eine Job-Nummer von einem Zufallswert abgeleitet sein, wobei die Vorgaben an einen solchen Zufallswert beachtet werden MÜSSEN [gemSpec_Krypt#2.2].⌫

⌦ TIP1-A_4643 Beschaffenheit der Job-Nummer

Zur Wahrung der Benutzerfreundlichkeit MUSS eine Reduzierung der Job-Nummer auf eine Länge von sechs Zeichen erfolgen. Diese sechs Zeichen MÜSSEN in zwei Zeichengruppen mit je drei Zeichen, getrennt durch einen Bindestrich (0x2D), dargestellt werden. Die erste Zeichengruppe MUSS ausschließlich Buchstaben

beinhalten, die zweite Zeichengruppe MUSS aus Ziffern bestehen. Die Länge der resultierenden, reduzierten Job-Nummer ist sieben und wird durch den Umfang der darstellbaren Zeichen auf dem Display des eHealth-Kartenterminals beschränkt.⌫

⌦ TIP1-A_4644 Job-Nummer über 1.000 Vorgänge eindeutig

Der Konnektor MUSS sicherstellen, dass eine einmal angezeigte Job-Nummer nicht innerhalb der nächsten 1.000 Vorgänge erneut Anwendung findet.⌫

⌦ TIP1-A_4645 Zeichen der Job-Nummer

Die einzelnen Zeichen der Job-Nummer MÜSSEN 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 Job-Nummer verwendet werden. ⌫

Beispiele für eine Job-Nummer sind ABC-475 und HZF-696.

Die Einbettung der Job-Nummer in den Nachrichtentext für den Bildschirm des Kartenlesers wird in Tabelle 52: TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal beschrieben.

4.1.8.1.7 Externe Authentisierung

⌦ TIP1-A_5437 Signaturverfahren für externe Authentisierung

Der Signaturdienst MUSS das Signaturverfahren PKCS#1 entsprechend TAB_KON_780 – Signaturverfahren Externe Authentisierung unterstützen. ⌫

Tabelle 166: TAB_KON_780 – Signaturverfahren Externe Authentisierung

Signaturformat	Standard	Dokumentformate	QES/ nonQES	Bemerkung
PKCS#1 (V2.1)	[RFC3447]	Binär	nonQES	Dieses Signaturformat DARF NUR in Verbindung mit dem zur Authentisierung vorgesehenen Schlüssel des HBAx und des SM-B genutzt werden. Die Nutzung ist auf Dokumente (Hash) von maximal 512 bit Länge beschränkt.

⌦ TIP1-A_5149 PKCS#1-Schnittstelle nur für Authentisierung mit HBAx und SM-B nutzen

Der Hersteller des Konnektors MUSS den Anwender (Clientsystem) im Handbuch des Konnektors geeignet und ausreichend darüber informieren, dass das Signaturformat PKCS#1 nur zu Authentisierungszwecken mit dem Authentisierungsschlüssel des HBAx und des SM-B verwendet werden darf.⌫

4.1.8.2 Durch Ereignisse ausgelöste Reaktionen

keine

4.1.8.3 Interne TUCs, nicht durch Fachmodule nutzbar

Abbildung 15: 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.

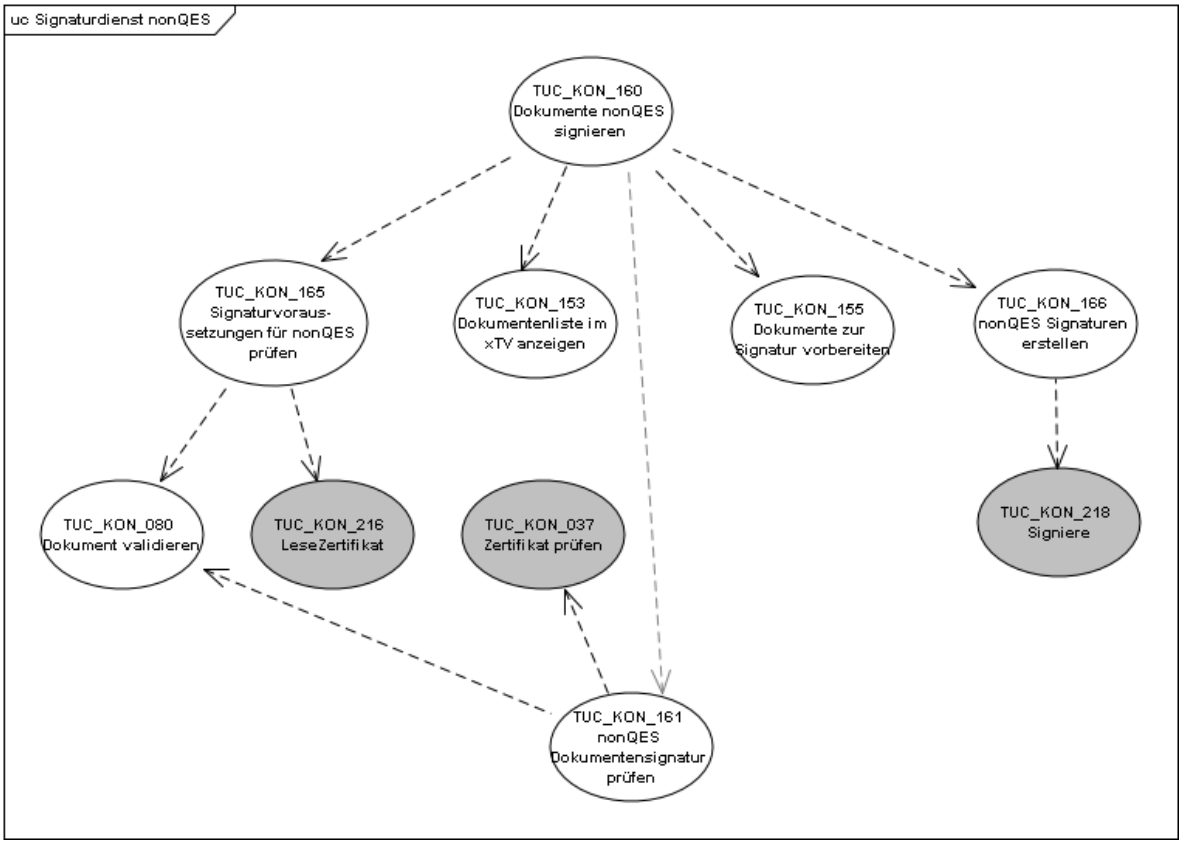


Abbildung 15: PIC_KON_103 Use Case Diagramm Signaturdienst (nonQES)

Abbildung 16: PIC_KON_104 Use Case Diagramm Signaturdienst (QES) beschreibt die Aufrufbeziehungen der QES-TUCs des Signaturdienstes.

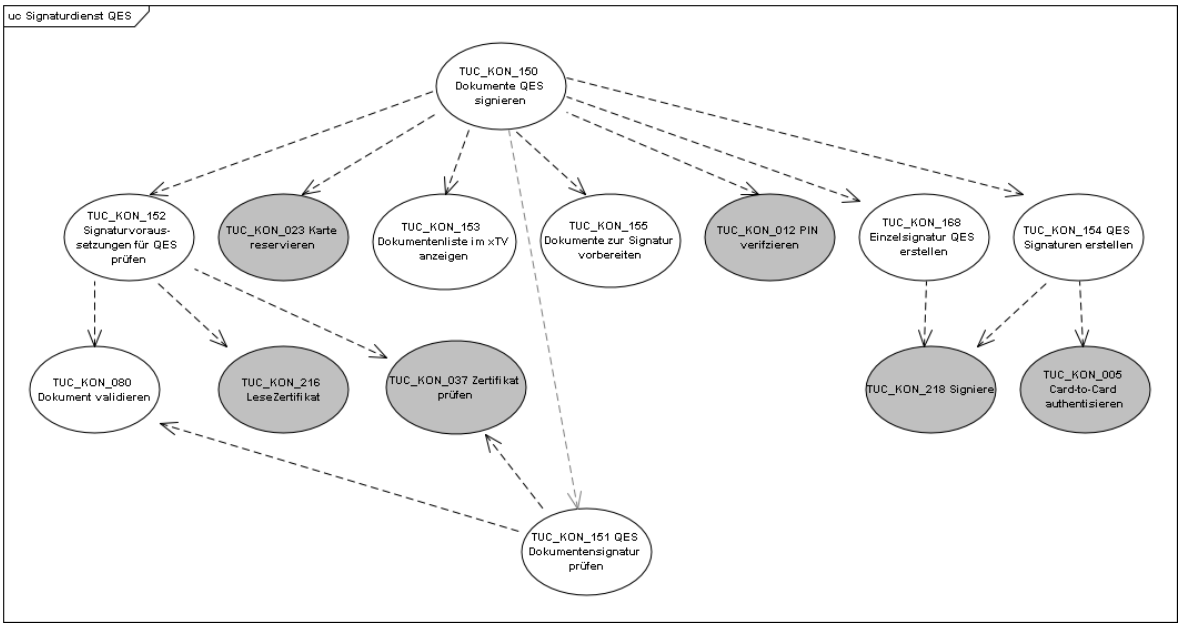


Abbildung 16: PIC_KON_104 Use Case Diagramm Signaturdienst (QES)

4.1.8.3.1 TUC_KON_155 "Dokumente zur Signatur vorbereiten"

⌦ TIP1-A_4646 TUC_KON_155 „Dokumente zur Signatur vorbereiten"

Der Konnektor MUSS den technischen Use Case TUC_KON_155 "Dokumente zur Signatur vorbereiten" umsetzen.

Tabelle 167: 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	• Signaturart (QES / nonQES) • Zu signierendes Dokument bzw. zu signierende Dokumente und pro Dokument: o Formatangabe für das zu signierende Dokument • weitere optionale Eingabeparameter zur Steuerung der Details bei der zu erstellenden Signatur (siehe Operation SignDocument, Parameter dss:OptionalInputs) • Signaturzertifikat • ocspResponses – optional (Liste der OCSP-Responses, die bei der Signaturerstellung in die

Element	Beschreibung
	Signatur eingebettet werden.)
Komponenten	Konnektor
Ausgangsdaten	Aufbereitetes zu signierendes Dokument bzw. aufbereitete zu signierende Dokumente
Standardablauf	<u>XMLDSig (XAdES)</u> Entsprechend den Regeln für die QES und die nonQES werden zunächst weitere Signatureigenschaften zum jeweiligen Dokument in Form von `QualifyingProperties` (siehe [XAdES]) hinzugefügt. Falls erforderlich, werden die Attributzertifikate hinzugefügt. Die Systemzeit des Anwendungskonnektors muss in das XML-Element `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. Das Signaturzertifikat wird im Element `ds:KeyInfo/ds:X509Data` gespeichert. Im Fall Signaturart=QES müssen 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) und das/die XSL-Stylesheet(s) für die Erzeugung der sicheren Anzeige und des Kurztextes. Für diese werden Referenzen (Hash + URI) in die Signatur eingebettet. Die URI ist im Fall übergebener XML-Schemata/XSL-Stylesheets der übergebene RefURI-Parameter. Die URI ist im Fall der im Konnektor im Rahmen einer Signaturrichtlinie hinterlegten XML-Schemata/XSL-Stylesheets die URI der Signaturrichtlinie ergänzt um den Dateinamen mit Pfad, wie in der Signaturrichtlinie festgelegt. (Beispiel: URI für Schemadatei CDA.xsd der Signaturrichtlinie SR_DF_SV_VHITG_ARZTBRIEF lautet: urn:gematik:fa:sak:cda:r1:v1:fa:sak:cda:schemas:CDA.xsd) Das Einbetten der Referenzen erfolgt über das XML-Element ds:object/ds:manifest (XMLDSig) mit eingebetteten XML-Elementen ds:Reference, 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 `DataObjectFormat/Description`-Element gemäß [XAdES] (Abschnitt 7.2.5) eingebettet werden. Falls durch den Aufrufparameter `SIG:IncludeRevocationInfo` angefordert werden die für die Offline-Prüfung notwendigen OCSP-Antworten im Sinne vom ES-X-L vom Konnektor in die Signatur eingebettet: Die base-64 kodierte OCSP-Response wird im Feld `QualifyingProperties/UnsignedProperties/UnsignedSignaturePropertiesRevocationValues/OCSPValues/EncapsulatedOCSPValue` (selbst DER-kodiert) gespeichert. Für die QES-Prüfung werden die OCSP-Sperrinformationen zur gesamten Vertrauenskette benötigt (EE-Zertifikat, CA-Zertifikat, Root-CA-Zertifikat), im nonQES-Fall nur die für das EE-

Element	Beschreibung
	Zertifikat. CMS (CAdES) Die Einbettung der Referenzen auf XML-Schemata und XSL-Stylesheets erfolgt über signedAttrs innerhalb SignerInfo. Dort wird das manifest-Element, wie im Fall XAdES beschrieben, als ASN.1 UTF8-String eingebettet. Der ShortTextClientsystem muss in die Signatur in das `content-hints.ContentDescription`-Attribut gemäß [CAdES] (Abschnitt 5.10.3) eingebettet werden. Ist die Einbettung von OCSP-Responses gefordert, werden die für die Offline-Prüfung notwendigen OCSP-Antworten im Attribut revocation-values abgelegt. PDF/A (PAdES) Der ShortTextClientsystem muss bei einer PDF-Signatur in das `Reason`-Feld eingebettet werden. OCSP-Responses sind im Document Security Store gemäß [PAdES-4] einzubetten. Es sind die Vorgaben zum Signaturprofil gemäß Tabelle TAB_KON_779 „Profilierung der Signaturformate“ zu erfüllen. Die aufbereiteten zu signierenden Dokumente werden an den Aufrufer zurückgegeben.
Varianten/Alternativen	keine
Fehlerfälle	Das Verhalten des TUCs bei einem Fehlerfall ist in TAB_KON_124 Übersicht Fehlercodes für „nonQES Dokumentensignatur prüfen“ beschrieben. „PDF/A (PAdES)“ Es ist nicht genügend Speicherplatz im PDF-Dokument verfügbar: 4205
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 168: TAB_KON_586 Übersicht Fehlercodes für „Dokumente zur Signatur vorbereiten“

Fehlercode	ErrorType	Severity	Fehlertext
4205	Technical	Error	Es ist nicht genügend Speicherplatz im PDF-Dokument verfügbar.

⌫

4.1.8.3.2 TUC_KON_165 „Signaturvoraussetzungen für nonQES prüfen“

⌦ TIP1-A_4647 TUC_KON 165 „Signaturvoraussetzungen für nonQES prüfen“

Der Konnektor MUSS den technischen Use Case “Signaturvoraussetzungen für nonQES prüfen” umsetzen.

Tabelle 169: TAB_KON_749 - TUC_KON_165 „Signaturvoraussetzungen für nonQES prüfen“

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	• Zu signierende Dokumente • optionale Eingabeparameter zur Steuerung der Details der Signaturerstellung • CardSession Signaturkarte • zu verwendende Identität (Zertifikatsreferenz)
Komponenten	Konnektor, Kartenterminal, Signaturkarte
Ausgangsdaten	• Prüfergebnis • Signaturzertifikat
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“. Wird der Aufruf von TUC_KON_080 mit einem Fehler beendet, wird die Prüfung im laufenden TUC mit diesem Fehler abgebrochen. 2. Durch Aufruf von TUC_KON_216 „Lese Zertifikat“ wird das Signaturzertifikat von der Signaturkarte gelesen.
Varianten/Alternativen	Keine
Fehlerfälle	Keine
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 170: TAB_KON_587 Übersicht Fehlercodes für „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 171: 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	Liste der zu signierenden Dokumente CardSession Signaturkarte zu verwendende Identität (Zertifikatsreferenz)
Komponenten	Konnektor, Kartenterminal, Signaturkarte
Ausgangsdaten	Signierte Dokumente
Standardablauf	Die folgenden Schritte werden für jedes Dokument der Liste durchgeführt. 1. 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. 2. Für das zu signierende Dokument werden die DTBS zur Signatur an die Signaturkarte übermittelt (Aufruf von TUC_KON_218 „Signiere“). 3. Die erstellte Signatur wird mathematisch geprüft. 4. Der ermittelte Signaturwert wird in die zuvor vorbereitete XML-Signatur eingefügt.
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

Tabelle 172: TAB_KON_120 Übersicht Fehlercodes für „nonQES Signaturen erstellen“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4120	`Security`	Error	Kartenfehler

⌫

4.1.8.3.4 TUC_KON_152 "Signaturvoraussetzungen für QES prüfen"

⌦ TIP1-A_4649 TUC_KON_152 „Signaturvoraussetzungen für QES prüfen“

Der Konnektor MUSS den technischen Use Case TUC_KON_152 “Signaturvoraussetzungen für QES prüfen” umsetzen.

Tabelle 173: 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 `QES_DocFormate` unterstützt.
Auslöser	TUC_KON_150 „Dokumente QES signieren“
Vorbedingungen	keine
Eingangsdaten	• Zu signierende Dokumente • optionale Eingabeparameter zur Steuerung der Details der Signaturerstellung • CardSession Signaturkarte • zu verwendende Identität (Zertifikatsreferenz) • includeRevocationInfo [Boolean] - optional; Default: true (Dieser Parameter steuert die Einbettung von OCSP-Responses in die Signatur. true: Die Sperrinformationen werden in ocspResponses zurückgegeben.)
Komponenten	Konnektor, Kartenterminal, Signaturkarte
Ausgangsdaten	• Prüfergebnis • Signaturzertifikat • ocspResponses - optional / nur wenn includeRevocationInfo = true (Liste der OCSP-Responses, die beim Aufruf von TUC_KON_037 „Zertifikat prüfen“ zurückgegeben werden)
Standardablauf	1. Abhängig von seinem Typ werden für jedes Dokument die typabhängigen Dokumentvalidierungsschritte einschließlich der Prüfung auf sichere Anzeigbarkeit 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. 2. Durch Aufruf von TUC_KON_216 „Lese Zertifikat“ werden das Signaturzertifikat und – falls für mindestens ein Dokument benötigt – die Attributzertifikate von der Signaturkarte gelesen.

Element	Beschreibung
	3. Das Signaturzertifikat und die Attributzertifikate werden durch Aufruf von TUC_KON_037 „Zertifikat prüfen“{C.HP.QES; required; true; oid_hba_qes; OCSP; getOCSPResponses = includeRevocationInfo ; Liste der Attributzertifikate} geprüft.
Varianten/Alternativen	keine
Fehlerfälle	Keine
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 174: TAB_KON_588 Übersicht Fehlercodes für „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_153 “Dokumentenliste im xTV anzeigen”

⌦ TIP1-A_4650 TUC_KON_153 “Dokumentenliste im xTV anzeigen”

Der Konnektor MUSS den technischen Use Case TUC_KON_153 “Dokumentenliste im xTV anzeigen” umsetzen.

Tabelle 175: TAB_KON_131 - TUC_KON_153 „Dokumentenliste im xTV anzeigen“

Element	Beschreibung
Name	TUC_KON_153 „Dokumentenliste im xTV anzeigen“
Beschreibung	Die zu signierenden Dokumente werden als Liste in der vertrauenswürdigen Anzeigekomponente (Extended Trusted Viewer) dargestellt. Der Anwender kann sich die Dokumenteninhalte und Zertifikate von der Signaturkarte anzeigen lassen und gegebenenfalls Dokumente vom weiteren Signaturvorgang ausschließen. Die Anforderungen an die Funktionalität des Extended Trusted Viewers sind im Kapitel 4.1.8.1.4 beschrieben.
Auslöser	TUC_KON_150 „Dokumente QES signieren“, TUC_KON_160 „Dokumente nonQES signieren“
Vorbedingungen	keine
Eingangsdaten	• Zu signierendes Dokument bzw. zu signierende Dokumente und pro Dokument: o ShortTextClientsystem (der vom Clientsystem übergebene ShortText)

Element	Beschreibung
	o SecurelyDisplayable (yes/no) o Zu signierende Eigenschaften, insbesondere Attributzertifikate Für XML-Dokumente: o XmlSchemas (optional) o XslStylesheets (optional) • Zertifikate von der Signaturkarte • ExtendedTrustedViewer • TvMode (Confirmed / Unconfirmed)
Komponenten	Konnektor, Extended Trusted Viewer
Ausgangsdaten	• Liste der zu signierenden Dokumente
Standardablauf	1. Bestimmung Kurztext (ShortText) Für jedes zu signierende Dokument für das eine Bildungsvorschrift für den Kurztext ShortTextSecure existiert, wird der Kurztext ShortTextSecure bestimmt. 2. Die Liste der zu signierenden Dokumente wird im Extended Trusted Viewer angezeigt. Für jedes Dokument der Liste wird der Kurztext ShortTextClientsystem zur Identifikation des Dokuments angezeigt. Ist der ShortTextSecure nicht gleich dem ShortTextClientsystem wird zusätzlich der ShortTextSecure angezeigt. Die Anzeige MUSS so gestaltet sein, dass der Benutzer die Quelle des Kurztextes nachvollziehen kann. Für QES verpflichtend: Die Anzeige MUSS so gestaltet sein, dass der Benutzer nachvollziehen kann, ob die Kurztextanzeige „sicher“ ist. Die Anzeige des Kurztextes ist „sicher“, wenn ein ShortTextSecure bestimmt werden konnte. Dokumente, die nicht sicher anzeigbar sind („SecurelyDisplayable = no“), werden in der Liste mit der entsprechenden Warnmeldung versehen. Ebenso wird die sichere Anzeigbarkeit („SecurelyDisplayable = yes“) für die entsprechenden Dokumente der Liste kenntlich gemacht. Des Weiteren wird im Extended Trusted Viewer die von der SAK generierte Job-Nummer für den Signaturvorgang angezeigt. 3. Auf Wunsch des Nutzers werden die Inhalte einzelner Dokumente, die Zertifikate und alle zu signierenden Eigenschaften dargestellt. Je nach Modus für die Anzeige im Extended Trusted Viewer verhält sich die SAK folgendermaßen: o Confirmed-TvMode (Bestätigungsmodus): Die SAK wartet, bis der Nutzer den Signaturvorgang im Extended Trusted Viewer bestätigt hat. Eine Deselektion einzelner Dokumente des Stapels durch den Nutzer ist möglich. Der Nutzer hat – bis er die Signaturerstellung autorisiert hat – die Möglichkeit, sich den Inhalt einzelner zu signierender Dokumente anzusehen und den Signaturvorgang gegebenenfalls am Kartenterminal abzubrechen. (siehe auch TIP1-A_4662 xTV-Bestätigungsmodus: Warten bei PIN-Eingabe). o Unconfirmed-TvMode (Ansichtsmodus): Im Vergleich zum Confirmed-Mode wartet die SAK nicht auf eine Bestätigung

Element	Beschreibung
	des Signaturvorgangs durch den Nutzer. Eine Deselektion einzelner Dokumente des Stapels durch den Nutzer ist nicht möglich. Für QES verpflichtend: Falls eines der Dokumente der Liste nicht sicher anzeigbar ist oder bei der Prüfung vorhandener Signaturen aus den bisherigen Prüfschritten das VerificationResult INCONCLUSIVE bzw. INVALID gemäß TAB_KON_593 bestimmt wurde, muss die SAK automatisch in den Confirmed-Mode wechseln und auf die Bestätigung des Nutzers warten. Ist bei der Prüfung einer Signatur das VerificationResult INVALID so muss der xTV den Benutzer darauf hinweisen, dass bei der Prüfung enthaltener Signaturen Fehler aufgetreten sind und die Erstellung der Gegensignatur nicht empfohlen wird. Bei der Anzeige des Dokumenteninhalts eines nicht sicher anzeigbaren Dokuments muss dem Benutzer ein Warnhinweis ausgegeben werden, dass es sich um eine unsichere Anzeige handelt. (siehe auch TIP1-A_4664 SAK im Ansichtsmodus: Allein die PIN-Eingabe am Kartenterminal ist maßgeblich).
Varianten/Alternativen	Der Nutzer kann den gesamten Signaturvorgang während der Anzeige des Extended Trusted Viewers abbrechen. Hierbei sind die gleichen Regeln anzuwenden wie im Fehlerfall.
Fehlerfälle	(-> 1) Kurztext kann nicht gebildet werden: 4121 (-> 3) Fehler bei Anzeige im xTV: 4122 Weitere Fehlerfälle sind Timeout bei der Bestätigung im Extended Trusted Viewer oder Abbruch durch den Benutzer.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 176: TAB_KON_589 Übersicht Fehlercodes für „Dokumentenliste im xTV anzeigen“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4121	`Security`	Error	Kurztext für Listenanzeige konnte nicht gebildet werden
4122	`Security`	Error	Fehler bei Anzeige im xTV

⌫

4.1.8.3.6 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 177: 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. DF.QES ist selektiert. Dem zum Kartenkontext gehörigen Arbeitsplatz muss ein Extended Trusted Viewer zugeordnet sein. PIN.QES ist initial verifiziert
Eingangsdaten	• Zu signierendes Dokument bzw. zu signierende Dokumente • CardSession (nur HBA erlaubt) • zu verwendende Identität (Zertifikatsreferenz) • WorkplaceId
Komponenten	Konnektor, Extended Trusted Viewer, Kartenterminal, Signaturkarte (HBA)
Ausgangsdaten	• Signierte Dokumente
Standardablauf	1. 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 (Aufruf TUC_KON_005 „Card-to-Card authentisieren“ {gSMC-K; CardSession; „gegenseitig+TC“} Die folgenden Schritte werden für jedes Dokument des Stapels durchgeführt. 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. 3. Für das zu signierende Dokument werden die DTBS zur Signatur im sicheren Kanal an den HBA übermittelt (Aufruf TUC_KON_218 „Signiere“). 4. 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 verifizieren“, nachdem der im Konnektor verwaltetete Sicherheitszustand (CARDSESSION.AUTHSTATE) aktualisiert wurde). Am Display des Kartenterminals wird dabei die Job-Nummer für den

Element	Beschreibung
	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. 5. Die erstellte Signatur wird mathematisch geprüft 6. Der ermittelte Signaturwert wird in den zuvor vorbereiteten Signaturprototypen eingefügt. 7. Nach erfolgreicher Signatur des Dokuments wird eine entsprechende Fortschrittsmeldung im Extended Trusted Viewer aktualisiert.
Varianten/Alternativen	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.
Fehlerfälle	(→3) Fehler im Signaturvorgang führen zum Abbruch des gesamten Signaturvorgangs, Fehlercode 4123 (→4) Fehler bei der PIN-Eingabe führen zum Abbruch des Signaturvorgangs (→5) Fehler in mathematischer Prüfung der Signatur führen zum Abbruch des Signaturvorgangs, Fehlercode 4120 Das Verhalten des TUCs bei einem Fehlerfall, einem Timeout der PIN-Eingabe oder beim Abbruch durch den Benutzer ist in Tabelle 188: TAB_KON_192 Verhalten des Konnektors beim Abbruch einer Stapelsignatur beschrieben.
Sicherheitsanforderungen	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. Die in [gemSpec_Krypt#3.1.2] angegebenen Festlegungen der zu unterstützenden Algorithmen MÜSSEN berücksichtigt werden.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	Abbildung 17: PIC_KON_113 Aktivitätsdiagramm zu „QES Signatur erstellen“

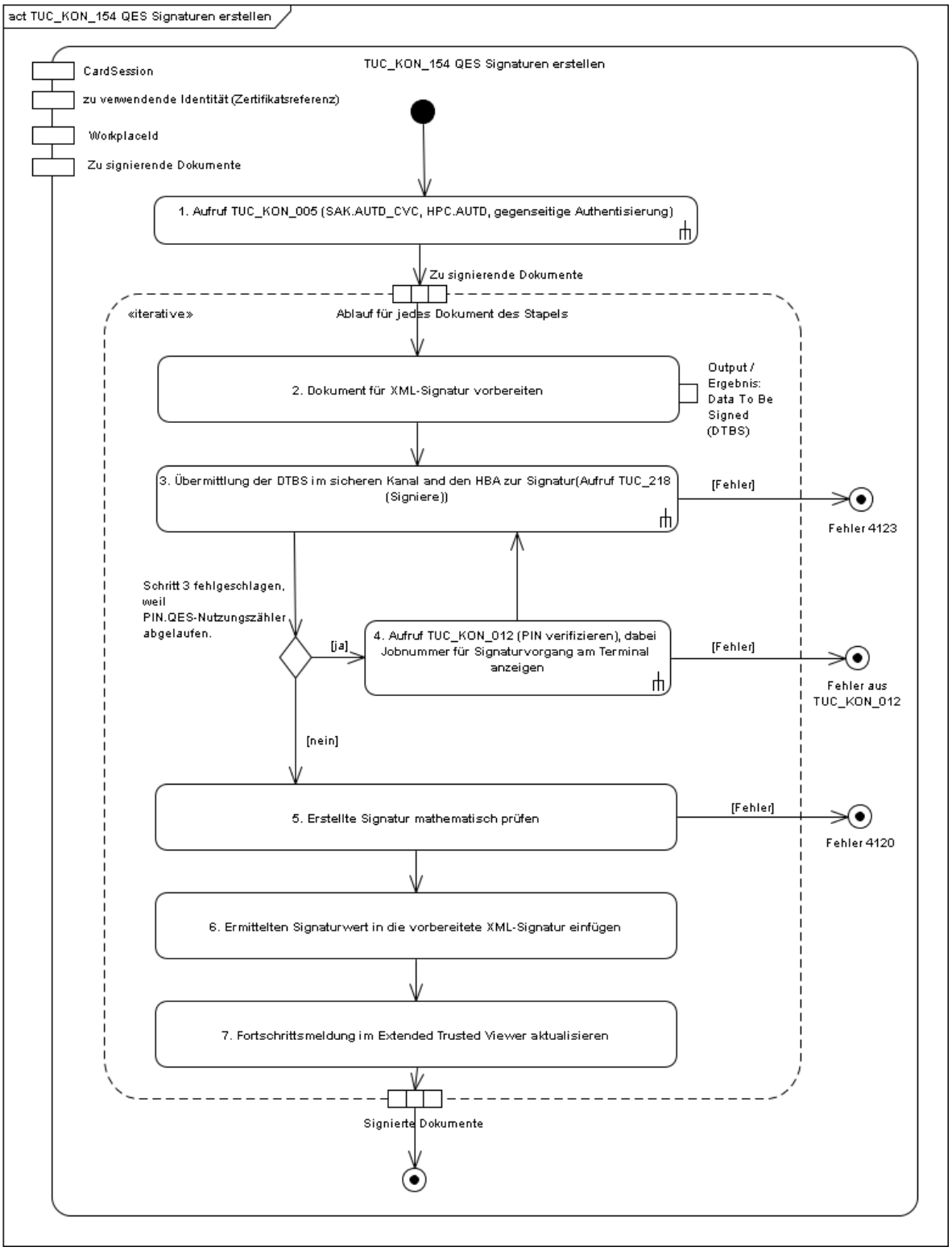


Abbildung 17: PIC_KON_113 Aktivitätsdiagramm zu „QES Signatur erstellen“

Tabelle 178: TAB_KON_126 Übersicht Fehlercodes für „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

⌫

4.1.8.3.7 TUC_KON_168 „Einzelsignatur QES erstellen“

⌦ TIP1-A_4652 TUC_KON_168 „Einzelsignatur QES erstellen“

Der Konnektor MUSS den technischen Use Case TUC_KON_168 „Einzelsignatur QES erstellen“ umsetzen.

Tabelle 179: 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. DF.QES ist selektiert. Dem zum Kartenkontext gehörigen Arbeitsplatz muss ein Extended Trusted Viewer zugeordnet sein.
Eingangsdaten	• Zu signierendes Dokument • CardSession (HBAx) • zu verwendende Identität (Zertifikatsreferenz) • WorkplaceId
Komponenten	Konnektor, Extended Trusted Viewer, Kartenterminal, Signaturkarte (HBAx)
Ausgangsdaten	• Signiertes Dokument
Standardablauf	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. 2. Für das zu signierende Dokument werden die DTBS zur Signatur an den HBAx übermittelt (Aufruf TUC_KON_218 „Signiere“). Jeder Fehler führt zum Abbruch des Signaturvorgangs 3. Die erstellte Signatur wird mathematisch geprüft. 4. Der ermittelte Signaturwert wird in den zuvor gemäß des entsprechenden Signaturformates vorbereiteten Signaturprototypen eingefügt. 5. Nach erfolgreicher Signatur des Dokuments kann eine

Element	Beschreibung
	entsprechende Statusmeldung an den Extended Trusted Viewer gesendet werden.
Varianten/Alternativen	keine
Fehlerfälle	Das Verhalten des TUCs bei einem Fehlerfall, einem Timeout der PIN-Eingabe oder beim Abbruch durch den Benutzer ist in Tabelle 188: TAB_KON_192 Verhalten des Konnektors beim Abbruch einer Stapelsignatur beschrieben. (→3) Fehler in mathematischer Prüfung der Signatur: Abbruch mit 4120
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 180: TAB_KON_590 Übersicht Fehlercodes für „Einzelsignatur QES erstellen“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten.			
4120	`Security`	Error	Kartenfehler

⌫

4.1.8.4 Interne TUCs, auch durch Fachmodule nutzbar

4.1.8.4.1 TUC_KON_160 “Dokumente nonQES signieren”

⌦ TIP1-A_4653 TUC_KON_160 „Dokumente nonQES signieren“

Der Konnektor MUSS den technischen Use Case TUC_KON_160 “Dokumente nonQES signieren” umsetzen.

Tabelle 181: 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	• Zu signierendes Dokument (Document) bzw. zu signierende Dokumente • CardSession (SM-B, HBAx oder bei Aufruf durch Fachmodul auch zusätzlich eGK) • TvMode (Modus für die Anzeige im Extended Trusted Viewer: Confirmed,

Element	Beschreibung
	Unconfirmed, None) • WorkplaceId • Weitere optionale Eingabeparameter (siehe Operation SignDocument, Parameter dss:OptionalInputs)
Komponenten	Konnektor, Extended Trusted Viewer, Kartenterminal, Signaturkarte bzw. HSM-B
Ausgangsdaten	Signierte Dokumente
Standardablauf	Der Konnektor KANN die Schritte 1 bis 4 in einer beliebigen Reihenfolge durchführen. 1. Falls TvMode ungleich None, wird aus dem Arbeitsplatz (WorkplaceId) der Extended Trusted Viewer des Arbeitsplatzes ermittelt. 2. Der Signaturtyp und die Signaturvariante werden für jedes Dokument der Liste entsprechend SignatureType und SignatureVariant festgelegt. Wenn SignatureType oder SignatureVariant nicht übergeben wurden, wird das dem Dokumentformat entsprechende Default-Verfahren gewählt (siehe TAB_KON_583 – Default-Signaturverfahren). 3. Für alle Dokumente des Stapels wird die Zulässigkeit des Kartentyps geprüft. Das für die Signatur zu nutzende Zertifikat wird anhand des Kartentyps implizit ausgewählt. 4. Es werden die Voraussetzungen für die Signatur geprüft. Dies erfolgt durch Aufruf von TUC_KON_165. 5. Für alle Dokumente, die gegenzusignieren sind, wird für alle Parallelsignaturen des Dokuments eine Signaturprüfung nach TUC_KON_161 „nonQES Dokumentensignatur prüfen“ und für alle Parallelsignaturen vom Typ QES des Dokuments eine Signaturprüfung nach TUC_KON_151 “QES-Dokumentensignatur prüfen“ durchgeführt. Bei Fehlern wird nicht abgebrochen. 6. Falls TvMode ungleich None, wird für die Anzeige im Extended Trusted Viewer TUC_KON_153 aufgerufen. 7. Zum Vorbereiten der Dokumente für die Signatur wird TUC_KON_155 „Dokumente zur Signatur vorbereiten“ aufgerufen. 8. Die Signaturen werden durch Aufruf von TUC_KON_166 erstellt. 9. Falls TvMode ungleich None, wird das Ergebnis des Signaturvorgangs im Extended Trusted Viewer angezeigt. Eine Bestätigung durch den Nutzer ist nicht erforderlich. 10. Die signierten Dokumente werden an den Aufrufer zurückgegeben.
Varianten/Alternativen	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. 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. Im S/MIME-Nachbereitungsschritt wird das im Standardablauf erzeugt CMS-Objekt in eine MIME-Nachricht vom Typ „application/pkcs7-mime“ eingebettet.

Element	Beschreibung
	Sämtliche Header-Felder der Nachricht MÜSSEN in die Header-Felder der S/MIME-Nachricht übernommen werden. "MIME-Version: 1.0" MUSS definiert sein. Das Feld "Content-Type:" ist als "application/pkcs7-mime" zu definieren. Die weiteren Attribute dieses Feldes sind: • "smime-type=signed-data;" • "name=$dateiname", wobei $dateiname auf ".p7m" endet. Die Codierung des signierten Inhalts der Nachricht MUSS in "base64" erfolgen. Entsprechend ist das zugehörige Header-Feld zu füllen: "Content-Transfer-Encoding: base64". Das Feld "Content-Disposition" definiert den Inhalt der Nachricht als Dateianhang: "Content-Disposition: attachment; filename=$dateiname"
Fehlerfälle	Fehler in den folgenden Schritten des Standardablaufs führen zum Abbruch mit den ausgewiesenen Fehlercodes: (→1) Fehler beim Ermitteln des Extended Trusted Viewers: Fehlercode 4117. (→ 2) Ungültige Angabe des Signaturverfahrens: Fehlercode 4111 Übergabe eines für die nonQES nicht unterstützten Dokumentformats: Fehlercode 4110 (→ 3) Kartentyp nicht zulässig für Signatur: Fehlercode 4126
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 182: TAB_KON_127 Übersicht Fehlercodes für „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%) Der Parameter Format enthält das übergebene Dokumentformat.
4111	`Technical`	Error	Ungültiger Signaturyp oder Signaturvariante
4117	`Technical`	Error	Kein Extended Trusted Viewer zugeordnet
4126	`Security`	Error	Kartentyp nicht zulässig für Signatur

⌫

4.1.8.4.2 TUC_KON_161 “nonQES Dokumentsignatur prüfen ”

⌦ TIP1-A_4654 TUC_KON_161 “nonQES Dokumentsignatur prüfen”

Der Konnektor MUSS den technischen Use Case TUC_KON_161 “nonQES Dokumentsignatur prüfen” umsetzen.

Tabelle 183: 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 161: 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	• Signiertes Document vom Typ `nonQES_DocFormate` • Signatur (optional, falls detached Signatur). Es werden Parallel- und Gegensignaturen unterstützt. • optionale Eingabeparameter (siehe Operation VerifyDocument, Parameter SIG:OptionalInputs) • X.509-Zertifikat (falls das Zertifikat nicht im signierten Dokument enthalten ist) • Grace Period • TvMode (NONE/UNCONFIRMED) Für XML-Dokumente: • Liste von XML-Schemata (optional) • Liste von XSL-Stylesheets (optional) • includeRevocationInfo [Boolean] – optional; Default = false (Dieser Parameter steuert die Einbettung von OCSP-Responses in die Signatur.)
Komponenten	Konnektor
Ausgangsdaten	• VerificationResult • SIG:OptionalOutput (optional)
Standardablauf	1. **„DocumentValidation“**: Falls die Signatur im Dokument eingebettet ist, wird das signierte Dokument wird validiert (Aufruf TUC_KON_080 „Dokument validieren“ { CheckDisplayability=no }).. Treten dabei Fehler bei Validierung der Typkonformität auf, wird die Prüfung mit einem Fehler abgebrochen. 2. **„CoreValidation“**: 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. <u>XML-Signatur:</u> Die Core Validation erfolgt entsprechend [XMLDSig] Kapitel 3.2 Core Validation. <u>CMS-Signatur:</u> Die Core Validation erfolgt entsprechend Cryptographic Message Syntax (CMS) Kapitel 5.6 Signature Verification Process [RFC5652].

Element	Beschreibung
	PDF-Signatur: Die Core Validation erfolgt entsprechend [PAdES-3] Kapitel 4.6 Signature Validation aus PAdES-BES Part 3. Auch wenn die Validierung fehlschlägt, werden die folgenden Prüfschritte durchgeführt, so dass ein vollständiges Prüfprotokoll erstellt werden kann. 3. **„CheckSignatureCertificate“:** **Teil 1: Signaturzertifikat ermitteln** XML-Signatur: Das Signaturzertifikat ist im XMLDSig Element `ds:KeyInfo/ds:X509Data` gespeichert [XMLDSig] oder wird als Eingangsparameter übergeben. CMS-Signatur: Das Signaturzertifikat für CAdES ist im Feld `certificates` im `SignedData` Container gespeichert [CAdES] oder wird als Eingangsparameter übergeben. PDF-Signatur: Das PDF Signaturzertifikat für PAdES ist im Feld SignedData.certificates entsprechend Kapitel 6.1.1 „Placements of the signing certificate“ [PAdES Baseline Profile] gespeichert oder wird als Eingangsparameter übergeben. **Teil 2: Signaturzeitpunkt bestimmen** Der Signaturzeitpunkt Ermittelter_Signaturzeitpunkt_Eingebettet wird wie folgt selektiert: XML-Signatur: Das XML element `SigningTime` spezifiziert den Signaturzeitpunkt entsprechend Kapitel 7.2.1 XAdES [XAdES]. CMS-Signatur: Das Attribut `SigningTime` spezifiziert den Signaturzeitpunkt entsprechend Kapitel 11.3 CMS [CMS]. PDF-Signatur: Der Signaturzeitpunkt kann dem M Eintrag des Signature Dictionary entnommen werden [PAdES Baseline Profile] Kapitel 6.2.1 Signing time. Der Signaturzeitpunkt `Benutzerdefinierter_Zeitpunkt` liegt gegebenenfalls als Aufrufparameter vor. Der Signaturzeitpunkt `Ermittelter_Signaturzeitpunkt_System` wird ermittelt. **Teil 3: Signaturzertifikatsprüfung:** Bei der folgenden Signaturzertifikatsprüfung sind die Signaturzeitpunkte gemäß [TIP1-A_5544] zu berücksichtigen. Die Signaturzertifikatsprüfung erfolgt durch Aufruf von - Für eGK: (C.CH.AUT / C.CH.AUTN) TUC_KON_037 „Zertifikat prüfen“ {(C.CH.AUT / C.CH.AUTN); not_required; Signaturzeitpunkt; true; (oid_egk_aut / oid_egk_autn); digitalSignature&keyEncipherment; id-kp-clientAuth; ;OCSP }, - Für SM-B: (C.HCI.OSIG)

Element	Beschreibung
	TUC_KON_037 „Zertifikat prüfen“ {C.HCI.OSIG; not_required; Signaturzeitpunkt; true; oid_smc_b_osig; nonRepudiation; id-kp-clientAuth; ; OCSP; OCSP-Response;…; getOCSPResponses = includeRevocationInfo}. Sind OCSP-Responses in der Signatur eingebettet, ist die jüngste OCSP-Response, die für die Zertifikatsprüfung notwendig ist, beim Aufruf von TUC_KON_037 zu übergeben. Sofern der Aufruf von TUC_KON_037 ocspResponsesRenewed zurück gibt, wird die Liste der OCSP-Responses in die Signatur eingebettet. Auch wenn die Zertifikatsprüfung fehlschlägt, werden die folgenden Prüfungen durchgeführt. 4. **„CheckPolicyConstraints“** 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], [CAdES], [CAdES Baseline], [PAdES-3] und [PAdES Baseline] umzusetzen. Dabei sind die Vorgaben aus Tabelle TAB_KON_779 „Profilierung der Signaturformate“ zu erfüllen. 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. 5. Falls TvMode = UNCONFIRMED wird das Ergebnis der Signaturprüfung dem Nutzer im Extended Trusted Viewer angezeigt. Der Nutzer kann sich den Inhalt des signierten Dokuments und der Zertifikate anzeigen lassen. Im Fall TvMode = NONE erfolgt keine Anzeige im Trusted Viewer. 6. Das Prüfergebnis (VerificationResult, OptionalOutput) wird an den Aufrufer zurückgegeben (siehe TAB_KON_754 Übersicht Status für Prüfung einer Dokumentensignatur).
Varianten/Alternativen	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. Im Fall einer PKCS#1-Signatur ist das verwendete Signaturverfahren, RSASSA-PSS bzw. RSASSA-PKCS1-v1_5, aus der Signatur zu bestimmen.

Element	Beschreibung
Fehlerfälle	Das Verhalten des TUCs bei einem Fehlerfall ist in TAB_KON_124 Übersicht Fehlercodes für „nonQES Dokumentensignatur prüfen“ beschrieben. (→2 **„CoreValidation“**) Interner Fehler: 4001, Signatur des Dokument ungültig: 4115. (→3 „**CheckSignatureCertificate“**) Interner Fehler: 4001, Signaturzertifikat ermitteln fehlgeschlagen: 4206. (→4 **„CheckPolicyConstraints“**) Interner Fehler: 4001, Dokument nicht konform zu Regeln für nonQES: 4112.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 184: TAB_KON_124 Übersicht Fehlercodes für „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. Der SignatureValue des Dokuments ist falsch oder für mindestens eine Reference ist der DigestValue falsch.

Das Gesamtergebnis (Result) für die Prüfung einer Dokumentensignatur fasst die Ergebnisse der obigen Einzelschritte zusammen. Welche nicht erfüllten Kriterien einzelner Prüfschritte zu einem Fehler oder einer Warnung führen, wird durch die Regeln für die nonQES (Anhang B) bestimmt.

Tabelle 185: TAB_KON_754 Übersicht Status für Prüfung einer Dokumentensignatur

VerificationResult pro Signatur		
	Wert	**Bedeutung**
	VALID	Die Signatur wurde gemäß den Regeln für die nonQES geprüft und für gültig befunden.
	INVALID	Die Signatur des Dokuments ist ungültig oder aufgrund eines Fehlers konnte die Signaturprüfung nicht durchgeführt werden.

	INCONCLUSIVE	Die Signatur wurde gemäß den Regeln für die nonQES geprüft. Allerdings konnten einige Prüfungen nicht vollständig durchgeführt werden. Einzelheiten finden sich in Result-Detail. Die Prüfungen, die durchgeführt werden konnten, waren erfolgreich.

VerificationResult für gesamtes Dokument	
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

Das VerificationResult pro Signatur MUSS wie folgt zur Belegung des Elements VerficationReport/IndividualReport/Result korrespondieren:

- VALID, wenn
  ResultMajor GLEICH urn:oasis:names:tc:dss:1.0:resultmajor:Success AND
  ResultMinor GLEICH urn:oasis:names:tc:dss:1.0:resultminor:valid:signature:OnAllDocuments
- INVALID, wenn
  ResultMajor UNGLEICH urn:oasis:names:tc:dss:1.0:resultmajor:Success ODER
  (ResultMajor GLEICH urn:oasis:names:tc:dss:1.0:resultmajor:Success UND
  ResultMinor GLEICH urn:oasis:names:tc:dss:1.0:resultminor:invalid:IncorrectSignature)

⌫

⌦ **TIP1-A_5544 nonQES-Signaturprüfung zu angenommenem Signaturzeitpunkt**

Der Konnektor MUSS die nonQES-Signaturprüfung zu den Zeitpunkten

- `Ermittelter_Signaturzeitpunkt_Eingebettet`
- `Benutzerdefinierter_Zeitpunkt`

soweit vorhanden, durchführen und das Prüfergebnis pro Signaturzeitpunkt (im Fall TvMode = UNCONFIRMED) am xTV darstellen. Sind beide Zeitpunkte nicht vorhanden, so ist der Zeitpunkt

- `Ermittelter_Signaturzeitpunkt_System`

zu verwenden.⌫

⌦ **TIP1-A_5545 nonQES-Signaturprüfergebnis bezogen auf Signaturzeitpunkt**

Der Konnektor MUSS zur nonQES-Signaturprüfung ein Prüfergebnis (Ampelwert), der sich auf genau einen angenommenen Signaturzeitpunkt bezieht, als Prüfergebnis an den Aufrufer zurückgeben und zusätzlich (im Fall TvMode = UNCONFIRMED) am xTV darstellen.

Die Auswahl des angenommenen Signaturzeitpunkts, auf den sich das Signaturergebnis bezieht, erfolgt hierarchisch:

- `Benutzerdefinierter_Zeitpunkt`
  falls vorhanden, sonst
- `Ermittelter_Signaturzeitpunkt_Eingebettet`
  falls vorhanden, sonst

- o Ermittelter_Signaturzeitpunkt_System⌫

4.1.8.4.3 TUC_KON_150 "Dokumente QES signieren"

⌦ TIP1-A_4655 TUC_KON_150 „Dokument QES signieren"

Der Konnektor MUSS den technischen Use Case TUC_KON_150 "Dokumente QES signieren" umsetzen.

Tabelle 186: 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	• SignRequests. Jeder SignRequest kapselt: o Zu signierendes Dokument (Document) bzw. zu signierende Dokumente o weitere optionale Eingabeparameter zur Steuerung der Details bei der zu erstellenden Signatur (siehe Operation SignDocument, Parameter dss:OptionalInputs) o includeRevocationInfo [Boolean] - optional; Default: true Dieser optionale Parameter steuert die Einbettung von OCSP-Responses in die Signatur (siehe Operation SignDocument, Parameter SIG:IncludeRevocationInfo) • CardSession (HBAx) • TvMode (Modus für die Anzeige im Extended Trusted Viewer: Confirmed, Unconfirmed) • WorkplaceId
Komponenten	Konnektor, Extended Trusted Viewer, Kartenterminal, Signaturkarte (HBAx)
Ausgangsdaten	Signierte Dokumente
Standardablauf	Der Konnektor KANN die Schritte 1 bis 4 in einer beliebigen Reihenfolge durchführen. 1. Aus dem Arbeitsplatz (WorkplaceId) wird der Extended Trusted Viewer des Arbeitsplatzes ermittelt. 2. Der Signaturtyp und die Signaturvariante werden für jedes Dokument der Liste entsprechend SignatureType und SignatureVariant festgelegt. Wenn SignatureType oder SignatureVariant nicht übergeben wurden, wird das dem Dokumentformat entsprechende Default-Verfahren gewählt (siehe TAB_KON_583 – Default-Signaturverfahren). 3. Für alle Dokumente des Stapels wird die Zulässigkeit des Kartentyps geprüft. Das für die Signatur zu nutzende Zertifikat wird anhand des Kartentyps implizit ausgewählt.

Element	Beschreibung
	4. Es werden die Voraussetzungen für die Signatur geprüft. Dies erfolgt im TUC_KON_152. Wenn includeRevocationInfo=true, dann setze ocspResponses auf Rückgabewert von TUC_KON_152. 5. Für alle Dokumente, die gegensigniert werden, wird für alle Parallelsignaturen des Dokuments eine Signaturprüfung nach TUC_KON_151 „QES Dokumentensignatur prüfen“ durchgeführt. Bei Fehlern wird nicht abgebrochen. 6. Die am Signaturvorgang beteiligten Ressourcen (Extended Trusted Viewer, Signaturkarte sowie PIN Pad und Display des PIN-Eingabe-Kartenterminals) werden für die exklusive Nutzung durch diesen Signaturvorgang reserviert. Die Reservierung der Signaturkarte erfolgt durch Aufruf von TUC_KON_023. 7. Für die Anzeige im Extended Trusted Viewer wird TUC_KON_153 aufgerufen. 8. Zum Vorbereiten der Dokumente für die Signatur wird TUC_KON_155 „Dokumente zur Signatur vorbereiten“ mit ocspResponses aufgerufen. Die Zugriffe auf die Signaturkarte in den Schritten 8 bis 9 müssen im DF.QES erfolgen. 9. Die Signaturerstellung wird durch den Anwender autorisiert. Dies erfolgt durch Aufruf von TUC_KON_012. Wenn nur ein zu signierendes Dokument vorhanden ist und der Einfachsignaturmodus aktiviert ist (siehe Konfigurationsparameter SAK_SIMPLE_SIGNATURE_ MODE), wird Schritt 9a) durchgeführt, ansonsten Schritt 9b). 9a) Die Signatur wird erstellt. Dies erfolgt gemäß TUC_KON_168. 9b) Die Signaturen werden erstellt. Dies erfolgt gemäß TUC_KON_154. 10. Es wird DF.QES verlassen, um den PIN-Status der PIN.QES zurückzusetzen. Der im Konnektor verwaltetete Sicherheitszustand (CARDSESSION.AUTHSTATE) ist zu aktualisieren. 11. Das Ergebnis des Signaturvorgangs wird im Extended Trusted Viewer angezeigt. Eine Bestätigung durch den Nutzer ist nicht erforderlich. 12. Die reservierten Ressourcen (Extended Trusted Viewer, Signaturkarte sowie PIN-Pad und Display des PIN-Eingabe-Kartenterminals) werden wieder freigegeben. Zur Freigabe der Signaturkarte wird TUC_KON_023 aufgerufen. 13. Die signierten Dokumente werden an den Aufrufer zurückgegeben.
Varianten/Alternativen	Der Nutzer kann den Vorgang während der Anzeige des Extended Trusted Viewers (Schritt 7) oder bei der Autorisierung (Schritt 9) 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: (→1) Fehler beim Ermitteln des Extended Trusted Viewers: Fehlercode 4117. (→2) Ungültige Angabe des Signaturtyps oder Signaturvariante: Fehlercode 4111 Übergabe eines für die QES nicht unterstützten Dokumentformats: Fehlercode 4110 (→3) Kartentyp nicht zulässig für Signatur: Fehlercode 4126

Element	Beschreibung
	(→6) Fehler bei der Reservierung von Ressourcen: Fehlercode 4060 (→9b) Karte ist kein HBA, sondern HBA-Vorläuferkarte: Fehlercode 4118 Im Fehlerfall, inklusive Timeout bei der Bestätigung im Extended Trusted Viewer (Fehler 4116) und bei der PIN-Eingabe, oder bei Abbruch durch den Benutzer (Fehler 4049): a) … MUSS DF.QES verlassen werden b) … MUSS der Fehler im Extended Trusted Viewer angezeigt werden, falls möglich (Bestätigung durch den Nutzer erforderlich) c)… MÜSSEN alle reservierten Ressourcen freigegeben werden (PIN Pad, Display und Karte SOLLEN bereits vor der Anzeige des Fehlers im Extended Trusted Viewer freigegeben werden) d) … MUSS der Fehler immer an das Clientsystem zurückgemeldet werden
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	Abbildung 18: PIC_KON_114 Aktivitätsdiagramm zu „Dokument QES signieren“

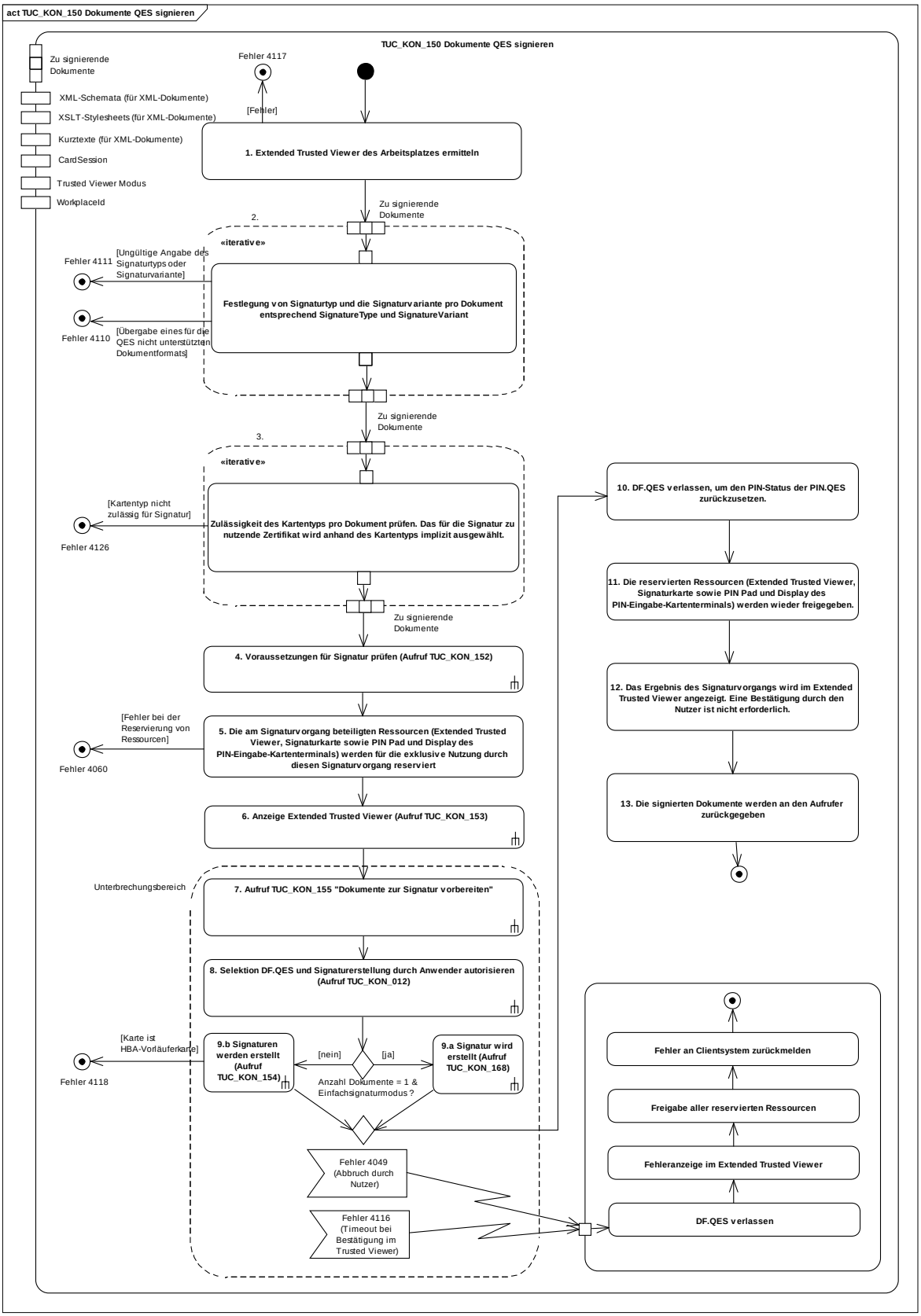


Abbildung 18: PIC_KON_114 Aktivitätsdiagramm zu „Dokument QES signieren“

Tabelle 187: TAB_KON_128 Übersicht Fehlercodes für „Dokument QES signieren“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4117	Technical	Error	Kein Extended Trusted Viewer zugeordnet
4060	Technical	Error	Ressource belegt
4110	Technical	Error	Ungültiges Dokumentformat (%Format%) Der Parameter Format enthält das übergebene Dokumentformat.
4111	Technical	Error	Ungültiger Signaturyp 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
4116	Technical	Warning	Timeout (Benutzer)

⌫

Anforderungen zur XML-Sicherheit:

⌦ TIP1-A_5113 Abwehr von XML-Signature-Wrapping Angriffen

Der Konnektor MUSS XML-Signature-Wrapping-Angriffe (XSW) abwehren.⌫

4.1.8.4.4 Verhalten des xTV bei der Signaturerstellung

⌦ TIP1-A_4656 xTV: Anzeige verpflichtender Parameter bei QES-Signaturerstellung

Bei einer QES-Signaturerstellung MUSS der xTV dem Benutzer

(a) den anstehenden Signaturvorgang,

(b) im Fall einer QES-Signatur die Job-Nummer des Vorganges,

(c) im Fall einer QES-Signatur den Zweck der PIN-Eingabe (dieser Text SOLL dem Display-Text des Kartenterminals entsprechen),

(d) im Fall einer QES-Signatur den Hostnamen des Kartenterminals, an dem die PIN-Eingabe erfolgen soll,

(e) den Ort (entfernt oder lokal), an dem sich der Kartenleser befindet, in den die Karte gesteckt ist,

(f) die Identität des Benutzers, abgeleitet aus dem CN-Feld (Common Name) des Zertifikats auf dem beruhend signiert werden soll,

(g) die Daten auf die sich die Signatur bezieht (einen oder mehrere Kurztexte, welche das zu signierende Dokument kennzeichnet),

(h) das Dokumentenformat

(i) die Signaturart (QES),

(j) den Signaturtyp: XML-Signatur, PDF-Signatur, CMS-Signatur,

(k) die Unterscheidung nach einfacher Dokumentensignatur (ohne Parallel- und Gegensignatur), Parallelsignatur, dokumentinkludierender Gegensignatur und dokumentexkludierende Gegensignatur

(l) im Fall der Gegensignatur das Prüfergebnis für alle Parallelsignaturen

und, sofern der Benutzer es wünscht (per Konfigurationseinstellung),

(m) den Inhalt des Zertifikates auf dem beruhend signiert werden soll,

(n) den Inhalt der zu signierenden Daten
(das zu signierende Dokument, zu signierende Signaturen, zu signierende Attributzertifikate, weitere zu signierende Eigenschaften)

sichtbar machen.⌫

Hinweis zu Punkt k): Die Eigenschaft ergibt sich direkt aus dem Aufrufparameter dss:ReturnUpdatedSignature:

- Parallelsignatur, falls dss:ReturnUpdatedSignature = http://ws.gematik.de/conn/sig/sigupdate/parallel
- Dokumentinkludierende Gegensignatur, falls dss:ReturnUpdatedSignature = http://ws.gematik.de/conn/sig/sigupdate/counter/documentincluding
- Dokumentexkludierende Gegensignatur, falls dss:ReturnUpdatedSignature = http://ws.gematik.de/conn/sig/sigupdate/counter/documentexcluding

⌦ TIP1-A_5404 xTV: Anzeige Kurztext bei Signaturerstellung

Der Konnektor MUSS den vom Clientsystem übergebenen Kurztext (ShortTextClientsystem) zur Identifikation des Dokuments anzeigen.

Der Konnektor MUSS für jedes zu signierende Dokument im XML-Format, für das gemäß DF_SV_XML eine Bildungsvorschrift für den Kurztext existiert, den Kurztext bestimmen (ShortTextSecure).

Der Konnektor MUSS für jedes zu signierende Dokument im TIFF-, Text oder PDF/A-Format den Kurztext ShortTextSecure gemäß der in Tabelle TAB_KON_802 definierten Bildungsvorschrift bestimmen.

Ist der ShortTextSecure nicht gleich dem ShortTextClientsystem, MUSS er ebenfalls als Zusatzinformation angezeigt werden.

Der Konnektor MUSS die Anzeige so gestalten, dass der Benutzer die Quelle des Kurztextes nachvollziehen kann und für QES-Signaturen ob die Kurztextanzeige „sicher“ ist. Die Kurztextanzeige ist „sicher“, wenn der ShortTextSecure korrekt gebildet werden konnte.⌫

⌦ TIP1-A_4657 xTV: Anzeige optionaler Parameter bei QES-Signaturerstellung

Bei einer QES-Signaturerstellung MUSS der xTV dem Benutzer

(a) den Status der identifizierten Algorithmen, wenn die Algorithmen unterstützt und durch die SAK laut des zum Zeitpunkt der SigG-Bestätigung der SAK aktuellen Algorithmenkatalogs [ALGCAT] als geeignet eingestuft werden,

(b) einen aussagekräftigen Hinweis zum verminderten Beweiswert hinsichtlich Authentizität und Integrität der zu erstellenden Signatur, wenn einer der identifizierten und unterstützten Algorithmen zum Zeitpunkt der Signaturerstellung nicht mehr laut Algorithmenkatalog [ALGCAT] als geeignet eingestuft wird,

(c) das Datum, zu dem der Algorithmus seine Eignung verloren hat, wenn einer der identifizierten und unterstützten Algorithmen zum Zeitpunkt der Signaturerstellung nicht mehr laut Algorithmenkatalog [ALGCAT] als geeignet eingestuft wird,

sichtbar machen können.⌫

⌦ TIP1-A_4658 xTV: Anzeige und Deselektion von Daten bei Stapelsignatur

Bei der Erstellung einer Stapelsignatur MUSS der xTV dem Benutzer eine Liste der zu signierenden Daten und Dokumententypen anzeigen. Der xTV MUSS im Bestätigungsmodus dem Benutzer die Möglichkeit einer Deselektion von einzelnen Daten anbieten. Deselektierte Daten MÜSSEN von der Signaturerstellung ausgenommen werden.⌫

⌦ TIP1-A_4659 xTV: Fortschrittsanzeige bei Stapelsignatur

Bei einer Stapelsignatur MUSS der xTV den Fortschritt bei der Signaturerzeugung erkennen lassen.⌫

⌦ TIP1-A_4660 Reihenfolge der Dokumente bei Stapelsignatur

Die zu signierenden Dokumente einer Stapelsignatur MÜSSEN von der SAK im Konnektor in derselben Reihenfolge signiert werden, in der sie in der Liste der zu signierenden Dokumente in dem xTV angezeigt werden.⌫

⌦ TIP1-A_4661 xTV: Kennzeichnung unterschiedlicher Dokumententypen

Sind in der Liste der zu signierenden Daten bei der Erstellung einer Stapelsignatur unterschiedliche Dokumententypen vorhanden, so SOLL der xTV diese Daten so kennzeichnen, dass eine Unterscheidung der vorhandenen Dokumententypen des Stapels optisch leicht möglich ist.⌫

⌦ TIP1-A_4662 xTV-Bestätigungsmodus: Warten bei PIN-Eingabe

Wird der xTV im Bestätigungsmodus betrieben, so MUSS die SAK im Konnektor auf eine Bestätigung des Signaturvorganges durch den xTV warten, bevor auf dem Kartenterminal die Aufforderung zur PIN-Eingabe erscheint. Wird die Bestätigung des Signaturvorganges über den xTV nicht während der Zeitspanne SAK_TIMEOUT_xTV durch den Benutzer erteilt oder treten Kommunikationsprobleme zwischen den beiden SAK-Komponenten auf, so MUSS die SAK im Konnektor die Signaturerzeugung abbrechen.⌫

⌦ TIP1-A_4663 SAK: Möglichkeit zum Abbruch geben

Der xTV MUSS im Bestätigungsmodus dem Benutzer vor Beginn des Signaturvorganges die Möglichkeit zum Abbruch des Vorganges bieten.⌫

Die SAK (Extended Trusted Viewer) wird vom Clientsystem wahlweise für einen Bestätigungs- oder einen Ansichtsmodus für jeden Signaturvorgang parametrisiert (siehe Beschreibung des Parameters TvMode in der Operation SignDocument in Kap.4.1.8.5.1).

⌦ TIP1-A_5151 Kein Standard-xTV-Mode für QES

Der Konnektor DARF für die QES NICHT einen Standardmodus für die Anzeige am Extended Trusted Viewer vorgeben.⌫

⌦ TIP1-A_4664 SAK im Ansichtsmodus: Allein die PIN-Eingabe am Kartenterminal ist maßgeblich

Im Ansichtsmodus ist allein die PIN-Eingabe am Kartenterminal maßgeblich für die Bestätigung des Signaturvorganges. Das zu signierende Dokument muss im xTV solange für den Benutzer sichtbar bleiben, bis entweder die PIN-Eingabe erfolgte oder der Vorgang über das Kartenterminal abgebrochen wurde. Die Deselektion einzelner Daten in dem xTV ist bei einer Stapelsignatur in diesem Modus nicht möglich.⌫

⌦ TIP1-A_4665 SAK im Ansichtsmodus: Muss darin verbleiben wenn alles anzeigbar

Wird der xTV im Ansichtsmodus betrieben und alle Kurztexte aus der Liste der zu signierenden Dokumente können in dem xTV ohne Benutzerinteraktion auf einen Blick angezeigt werden, dann MUSS der xTV im angeforderten Ansichtsmodus verbleiben.⌫

⌦ TIP1-A_4666 SAK im Ansichtsmodus: Muss in Bestätigungsmodus umschalten wenn nicht alles anzeigbar

Passen nicht alle Kurztexte auf den Bildschirmbereich des xTV, so MUSS die SAK im Konnektor automatisch aus dem Ansichtsmodus in den Bestätigungsmodus umschalten.⌫

⌦ TIP1-A_4668 SAK: Bestätigung von Fehlern durch die Benutzer

Nach der Signaturerzeugung MUSS der xTV das Ergebnis der Signaturerzeugung anzeigen. Im Bestätigungs- und Ansichtsmodus MUSS im Fehlerfall eine Bestätigung des Fehlers durch den Benutzer erfolgen. Der Benutzer MUSS sich im Bestätigungsmodus über den xTV die erzeugte(n) Signatur(en) anzeigen lassen können.⌫

4.1.8.4.5 Anforderungen an die Stapelsignatur

Eine Stapelsignatur definiert sich als „Erstellung einer begrenzten Anzahl Signaturen nach den zeitlich unmittelbar aufeinander folgenden Prozessen der Anzeige der zu signierenden Daten und der einmaligen Authentisierung des Signaturschlüssel-Inhabers gegenüber der SSEE“ (siehe [BSI-TR03114]).

⌦ TIP1-A_4669 QES-Stapelsignatur

Der Signaturdienst MUSS die Möglichkeit bieten, Dokumente eines Stapels einzeln qualifiziert elektronisch zu signieren. Der Signaturdienst MUSS als SSEE für die Stapelsignatur den HBA unterstützen. ⌫

⌦ TIP1-A_4670 Secure Messaging für die DTBS

Bei der Stapelsignatur MUSS die SAK die zu signierenden Daten (DTBS) über Secure Messaging vom Konnektor zum HBA übertragen. Dieser Secure Messaging-Kanal MUSS über die gSMC-K zum HBA mittels C.SAK.AUTD_CVC aufgebaut werden.⌫

⌦ TIP1-A_4671 Verhalten des Konnektors beim Abbruch einer Stapelsignatur

Die SAK (xTV und SAK_{KON}) MUSS dem Benutzer während und nach einer PIN-Eingabe die Möglichkeit zum Abbruch einer Stapelsignatur anbieten.

Das geforderte Verhalten des Konnektors beim Abbruch einer Stapelsignatur wird in der folgenden Tabelle beschrieben. Hierbei werden die beiden Punkte „Abbruch, während die erneute PIN-Eingabe angefordert wird“ (Nummer 1 bis 4) und „Abbruch, während der Vorgang der Signaturerstellung läuft“ (Nummer 5 bis 6) unterschieden. Zeile Nummer 7 beschreibt alle sonstigen Fehlerfälle.

Ein Teilstapel einer Stapelsignatur ist durch die maximale Anzahl der Dokumente definiert, welche nach der Eingabe der Signatur-PIN durch den Signaturschlüssel-Inhaber signiert werden kann.

Tabelle 188: 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 beendet: Kein „Fehler“ 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	Abbruchtaste am xTV wird gedrückt	Dieses Ereignis kann nur im Bestätigungsmodus des xTV auftreten. Im Ansichtsmodus ist die Abbruch-„Taste“ im xTV nicht verfügbar. Der Signaturvorgang (Stapel) wird beendet. Kein „Fehler“ Keine weiteren Signaturen des neuen Teilstapels werden erstellt. (Die Weiterverarbeitung bereits erstellter Signaturen des letzten Teilstapels (sofern vorhanden) wird noch abgeschlossen). Es öffnet sich ein Dialogfenster mit der Möglichkeit, die Signaturen der bisher signierten Teilstapel zu verwerfen. Das Dialogfenster muss, sofern keine Benutzerinteraktion erfolgt, nach einem definierten Timeout geschlossen werden. Die Standardaktion bei diesem Dialogfenster MUSS

			„Die Signaturen des/der vorherigen Teilstapel(s) bleiben erhalten und werden an das Clientsystem zurückgegeben.“ sein.
	4	Abbruchtaste am Kartenterminal wird gedrückt	Siehe Verhalten unter Nummer 1
während der Vorgang der Signaturerstellung läuft	5	Abbruchtaste am xTV wird gedrückt	Signaturvorgang (Stapel) wird abgebrochen. Kein „Fehler“ Keine weiteren Signaturen des Stapels werden erstellt. Keine weiteren Signaturen des Teilstapels werden erstellt. Bisher erstellte Signaturen des aktuellen Teilstapels werden verworfen. Es öffnet sich ein Dialogfenster mit der Möglichkeit, die Signaturen der bisher signierten Teilstapel zu verwerfen. Das Dialogfenster muss, sofern keine Benutzerinteraktion erfolgt, nach einem definierten Timeout geschlossen werden. Die Standardaktion bei diesem Dialogfenster MUSS: „Die Signaturen des/der vorherigen Teilstapel(s) werden gelöscht.“ sein.
	6	Abbruchtaste am Kartenterminal wird gedrückt.	Die „Abbruch“-Taste wird nicht von der SAK fortlaufend überwacht → Keine Aktion seitens der SAK.
	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. Schwerer Fehler. Keine weiteren Signaturen des Stapels werden erstellt. Keine weiteren Signaturen des aktuellen Teilstapels werden erstellt. Bisher erstellte Signaturen aller Teilstapel werden verworfen. Es handelt sich um Probleme/Fehlerfälle, die bei typischen Angriffen auftreten können.

⌫

4.1.8.4.6 TUC_KON_151 "QES Dokumentensignatur prüfen"

⌦ TIP1-A_4672 TUC_KON_151 „QES-Dokumentensignatur prüfen“

Der Konnektor MUSS den technischen Use Case TUC_KON_151 “QES-Dokumentensignatur prüfen” umsetzen.

Tabelle 189: 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 161: 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	• QES-signiertes Dokument vom Typ QES_DocFormate • QES Signatur(en). Es werden Parallel- und Gegensignaturen unterstützt. • optionale Eingabeparameter (siehe Operation VerifyDocument, Parameter SIG:OptionalInputs) • X.509-Zertifikate (falls diese nicht im signierten Dokument enthalten sind, sondern nur referenziert werden). • Kurztext (vom Clientsystem übergeben) • WorkplaceId Für XML-Dokumente: • Liste von XML-Schemata (optional) • Liste von XSL-Stylesheets (optional) • includeRevocationInfo [Boolean] - *optional; Default: false* (Dieser optionale Parameter steuert die Einbettung von OCSP-Responses in die Signatur)
Komponenten	Konnektor, Extended Trusted Viewer
Ausgangsdaten	• VerificationResult • SIG:OptionalOutput (optional)
Standardablauf	1. Es wird geprüft, ob dem Arbeitsplatz ein Extended Trusted Viewer zugeordnet ist. 2. **„DocumentValidation“**: Das signierte Dokument wird validiert (Aufruf TUC_KON_080 „Dokument validieren“{ CheckDisplayability=yes }). 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, oder bei der Prüfung auf sichere Anzeigbarkeit 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.) Wird TUC_KON_080 mit einem der Fehler 4193, 4195 abgebrochen, wird die Verarbeitung fortgesetzt. 3. **„CoreValidation“**: 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

Element	Beschreibung
	Schlüssels, des Signaturwertes und des signierten Hashwertes. XML-Signatur: Die Core Validation erfolgt entsprechend [XMLDSig] Kapitel 3.2 Core Validation. CMS-Signatur: Die Core Validation erfolgt entsprechend Cryptographic Message Syntax (CMS) Kapitel 5.6 Signature Verification Process [RFC5652]. PDF-Signatur: Die Core Validation erfolgt entsprechend [PAdES-3] Kapitel 4.6 Signature Validation aus PAdES-BES Part 3. Auch wenn die Validierung fehlschlägt, werden die folgenden Prüfschritte durchgeführt, so dass ein vollständiges Prüfprotokoll erstellt werden kann. 4. **„CheckSignatureCertificate“:** **Teil 1: Signaturzertifikat ermitteln** XML-Signatur: Das Signaturzertifikat ist im XMLDSig Element `ds:KeyInfo/ds:X509Data` gespeichert [XMLDSig] oder wird als Eingangsparameter übergeben. CMS-Signatur: Das Signaturzertifikat für CAdES ist im Feld `certificates` im `SignedData` Container gespeichert [CAdES] oder wird als Eingangsparameter übergeben. PDF-Signatur: Das PDF Signaturzertifikat für PAdES ist im Feld SignedData.certificates entsprechend Kapitel 6.1.1 „Placements of the signing certificate“ [PAdES Baseline Profile] gespeichert oder wird als Eingangsparameter übergeben. **Teil 2: Signaturzeitpunkt bestimmen** Der Signaturzeitpunkt `Ermittelter_Signaturzeitpunkt_Eingebettet` wird wie folgt selektiert: XML-Signatur: Das XML element `SigningTime` spezifiziert den Signaturzeitpunkt entsprechend Kapitel 7.2.1 XAdES [XAdES]. CMS-Signatur: Das Attribut `SigningTime` spezifiziert den Signaturzeitpunkt entsprechend Kapitel 11.3 CMS [CMS]. PDF-Signatur: Der Signaturzeitpunkt kann dem M Eintrag des Signature Dictionary entnommen werden [PAdES Baseline Profile] Kapitel 6.2.1 Signing time. Der Signaturzeitpunkt `Ermittelter_Signaturzeitpunkt_Qualifiziert` wird wie folgt selektiert: XML-Signatur: Das XML element `SignatureTimeStamp` spezifiziert den Signaturzeitpunkt entsprechend Kapitel 4.4.3.1 XAdES [XAdES].

Element	Beschreibung
	CMS-Signatur und PDF-Signatur: Das Attribut `signature-time-stamp` spezifiziert den Signaturzeitpunkt entsprechend Kapitel 6.1 CAdES [CAdES]. Der Signaturzeitpunkt `Benutzerdefinierter_Zeitpunkt` liegt gegebenenfalls als Aufrufparameter vor. Der Signaturzeitpunkt `Ermittelter_Signaturzeitpunkt_System` wird ermittelt. **Teil 3: Signaturzertifikatsprüfung:** Bei der folgenden Signaturzertifikatsprüfung sind die Signaturzeitpunkte gemäß [TIP1-A_5539] zu berücksichtigen. Die Signaturzertifikatsprüfung erfolgt durch Aufruf von TUC_KON_037 „Zertifikat prüfen“ {C.HP.QES; required; Signaturzeitpunkt; true; oid_hba_qes; OCSP; Liste von OCSP-Responses; getOCSPResponses = includeRevocationInfo; Liste der Attributzertifikate}. Sind OCSP-Responses in der Signatur eingebettet, sind die jeweils jüngsten OCSP-Responses, die für die Zertifikatsprüfung notwendig sind beim Aufruf von TUC_KON_037 zu übergeben. Sofern der Aufruf von TUC_KON_037 ocspResponses zurückgibt, wird die Liste der OCSP-Responses in die Signatur eingebettet. Sofern Attributzertifikate in der Signatur vorhanden sind, werden diese beim Aufruf von TUC_KON_037 übergeben. Auch wenn die Zertifikatsprüfung fehlschlägt, werden die folgenden Prüfungen durchgeführt. 5. **„CheckPolicyConstraints“**: 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], [CAdES], [CAdES Baseline], [PAdES-3] und [PAdES Baseline] umzusetzen. Dabei sind die Vorgaben aus Tabelle TAB_KON_779 „Profilierung der Signaturformate“ zu erfüllen. 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. 6. Das Ergebnis der Signaturprüfung wird im Extended Trusted Viewer angezeigt. Um das Dokument zu identifizieren wird der vom Clientsystem übergebene Kurztext verwendet. Der Kurztext muss als Parameter ShortTextClientsystem zusätzlich zum Dokument an den Extended Trusted Viewer übergeben werden. Der Nutzer kann sich den Inhalt des signierten Dokuments und der Zertifikate anzeigen lassen. Zur Anzeige von XML-Dokumenten muss zunächst die XSL-Transformation entsprechend der übergebenen Stylesheets durchgeführt werden. Falls keine Stylesheets übergeben wurden, MUSS es für den Benutzer möglich sein, sich das Dokument im Roh-XML-Format anzeigen zu lassen. Für alle anderen Dokumentformate muss das Vorgehen noch festgelegt werden (siehe Disclaimer). Wird bei der Verifikation einer Signatur eine Warnung (NO_OCSP_CHECK) von der SAK erzeugt, so MUSS der Verifizierende entscheiden, ob er das angebotene

Element	Beschreibung
	Sicherheitsniveau akzeptiert oder diese Warnung als Fehler ansieht und der TUC somit mit Fehler endet. 7. Das Prüfergebnis (VerificationResult, OptionalOutput) wird an den Aufrufer zurückgegeben (siehe TAB_KON_593 Übersicht Status für Prüfung einer Dokumentensignatur).
Varianten/Alternativen	Keine
Fehlerfälle	Das Verhalten des TUCs bei einem Fehlerfall ist in TAB_KON_592 Übersicht Fehlercodes für „QES Dokumentensignatur prüfen“ beschrieben. (→1) Fehler beim Ermitteln des Trusted Viewers: Fehlercode 4117 (→ 3 **„CoreValidation“**) Interner Fehler: 4001, Signatur des Dokuments ungültig: 4115 (→4 **„CheckSignatureCertificate“**) Interner Fehler: 4001, Signaturzertifikat ermitteln ist fehlgeschlagen: 4206. (→5 **„CheckPolicyConstraints“**) Interner Fehler: 4001, Dokument nicht konform zu Regeln für QES: 4124, Dokument nicht konform zu Profilierung der Signaturformate: 4208. (→6): Kein Extended Trusted Viewer zugeordnet: 4122. Kurztext für Listenanzeige konnte nicht gebildet werden: 4121. Fehler bei Anzeige im xTV: 4122.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 190: TAB_KON_592 Übersicht Fehlercodes für „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.
4117	Technical	Error	Kein Extended Trusted Viewer zugeordnet
4121	Security	Error	Kurztext für Listenanzeige konnte nicht gebildet werden
4122	Security	Error	Fehler bei Anzeige im xTV
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

Das Gesamtergebnis (VerificationResult) für die Prüfung einer Dokumentensignatur fasst die Ergebnisse der obigen Einzelschritte zusammen. Welche nicht erfüllten Kriterien einzelner Prüfschritte zu einem Fehler oder einer Warnung führen, wird durch die Regeln für die QES (siehe Anhang B.2) bestimmt.

Tabelle 191: TAB_KON_593 Übersicht Status für Prüfung einer Dokumentensignatur

VerificationResult pro Signatur		
	Wert	**Bedeutung**
	VALID	Die Signatur wurde gemäß den Regeln für die QES geprüft und für gültig befunden.
	INVALID	Die Signatur des Dokuments ist ungültig oder aufgrund eines Fehlers konnte die Signaturprüfung nicht durchgeführt werden. Die Signatur kann auch durch den Anwender im Extended Trusted Viewer abgelehnt worden sein.
	INCONCLUSIVE	Die Signatur wurde gemäß den Regeln für die QES geprüft. Allerdings konnten einige Prüfungen nicht vollständig durchgeführt werden. Einzelheiten finden sich in Result-Detail. Die Prüfungen, die durchgeführt werden konnten, waren erfolgreich.

VerificationResult für gesamtes Dokument	
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

Das VerificationResult pro Signatur MUSS wie folgt zur Belegung des Elements VerficationReport/IndividualReport/Result korrespondieren:

- VALID, wenn
  ResultMajor GLEICH urn:oasis:names:tc:dss:1.0:resultmajor:Success AND
  ResultMinor GLEICH urn:oasis:names:tc:dss:1.0:resultminor:valid:signature:OnAllDocuments
- INVALID, wenn
  ResultMajor UNGLEICH urn:oasis:names:tc:dss:1.0:resultmajor:Success ODER
  (ResultMajor GLEICH urn:oasis:names:tc:dss:1.0:resultmajor:Success UND
  ResultMinor GLEICH urn:oasis:names:tc:dss:1.0:resultminor:invalid:IncorrectSignature)

⌫

⌦ **TIP1-A_5539 QES-Signaturprüfung zu angenommenem Signaturzeitpunkt**

Der Konnektor MUSS die QES-Signaturprüfung zu allen vorhandenen Zeitpunkten durchführen und das Prüfergebnis pro Signaturzeitpunkt am xTV darstellen:

Falls ein `Benutzerdefinierter_Zeitpunkt` vorliegt, für die Zeitpunkte

- `Ermittelter_Signaturzeitpunkt_Eingebettet`
- `Ermittelter_Signaturzeitpunkt_Qualifiziert`
- `Benutzerdefinierter_Zeitpunkt`

Falls ein `Benutzerdefinierter_Zeitpunkt` nicht vorliegt, für die Zeitpunkte

- `Ermittelter_Signaturzeitpunkt_Eingebettet`
- `Ermittelter_Signaturzeitpunkt_Qualifiziert`
- `Ermittelter_Signaturzeitpunkt_System`⌫

⌦ **TIP1-A_5540 QES-Signaturprüfergebnis bezogen auf Signaturzeitpunkt**

Der Konnektor MUSS zur QES-Signaturprüfung ein Prüfergebnis (Ampelwert), der sich auf genau einen angenommenen Signaturzeitpunkt bezieht, am xTV darstellen und als Prüfergebnis an den Aufrufer zurückgeben.

Die Auswahl des angenommenen Signaturzeitpunkts, auf den sich das Signaturergebnis bezieht, erfolgt hierarchisch:

- `Benutzerdefinierter_Zeitpunkt`
  falls vorhanden, sonst
- `Ermittelter_Signaturzeitpunkt_Eingebettet`
  falls vorhanden, sonst
- `Ermittelter_Signaturzeitpunkt_Qualifiziert`
  falls vorhanden, sonst
- `Ermittelter_Signaturzeitpunkt_System`⌫

4.1.8.4.7 Verhalten des xTV bei der Signaturprüfung

⌦ **TIP1-A_4673 xTV: Anzeige verpflichtender Parameter bei Signaturprüfung**

Im Falle der Signaturprüfung MUSS der xTV dem Benutzer

a) die Daten, auf die sich die Signatur bezieht (einen oder mehrere Kurztexte, welche das zu signierende Dokument kennzeichnen),

b) den Inhalt des Zertifikates auf dem beruhend signiert wurde sowie den Inhalt in die Signatur eingefügter Attributzertifikate,

c) das Ergebnis der Signaturprüfung und Zertifikatsprüfung,

d) den Status der bei der Signaturprüfung identifizierten Algorithmen, wenn die Algorithmen unterstützt und durch die SAK laut des zum Zeitpunkt der SigG-Bestätigung der SAK aktuellen Algorithmenkatalogs [ALGCAT] als geeignet eingestuft werden,

e) einen aussagekräftigen Hinweis darauf, ob einer der bei der Signaturprüfung identifizierten Algorithmen nicht durch die SAK unterstützt wird und aus diesem Grund die Signatur nicht geprüft werden kann,

f) einen aussagekräftigen Hinweis zum verminderten Beweiswert hinsichtlich Authentizität und Integrität der Signatur, wenn einer der bei der Signaturprüfung identifizierten und unterstützten Algorithmen zum Zeitpunkt der Signaturerstellung nicht mehr durch die SAK laut Algorithmenkatalog [ALGCAT] als geeignet eingestuft wird,

g) das Datum, zu dem der Algorithmus seine Eignung verloren hat, wenn einer der bei der Signaturprüfung identifizierten und unterstützten Algorithmen zum Zeitpunkt der Signaturerstellung nicht mehr durch die SAK laut Algorithmenkatalog [ALGCAT] als geeignet eingestuft wird,

h) die Signaturart (QES / nonQES),

i) im Fall einer Gegensignatur, die Kennzeichnung als Gegensignatur und den Verweis auf die gegensignierte Signatur,

j) den zur Prüfung der Signatur verwendeten Signaturzeitpunkt als Lokalzeit

und, sofern der Benutzer es wünscht,

k) den Inhalt der signierten Daten
(das signierte Dokument, signierte Signaturen, signierte Attributzertifikate, weitere signierte Eigenschaften)

sichtbar machen. ⌫

⌦ TIP1-A_5405 xTV: Anzeige Kurztext bei Signaturprüfung

Der Konnektor MUSS den vom Clientsystem übergebenen Kurztext (ShortTextClientsystem) zur Identifikation des Dokuments anzeigen.

Der Konnektor MUSS den in der Signatur enthaltenen Kurztext (ShortTextSignature) extrahieren. Ist der ShortTextSignature nicht gleich dem ShortTextClientsystem, MUSS er als Zusatzinformation angezeigt werden.

Der Konnektor MUSS für jedes signierte Dokument im XML-Format, für das gemäß DF_SV_XML eine Bildungsvorschrift für den Kurztext existiert, den Kurztext bestimmen (ShortTextSecure).

Der Konnektor MUSS für jedes signierte Dokument im TIFF-, Text oder PDF/A-Format den ShortTextSecure gemäß der in Tabelle TAB_KON_802 definierten Bildungsvorschrift bestimmen.

Ist der ShortTextSecure nicht gleich dem ShortTextSignature, MUSS er ebenfalls als Zusatzinformation angezeigt werden.

Der Konnektor MUSS die Anzeige so gestalten, dass der Benutzer die Quelle des Kurztextes nachvollziehen kann und ob die Kurztextanzeige „sicher“ ist.

Die Kurztextanzeige ist „sicher“, wenn der ShortTextSignature gleich dem ShortTextSecure ist. ⌫

4.1.8.5 Operationen an der Außenschnittstelle

⌦ TIP1-A_4676 Basisdienst Signaturdienst (nonQES und QES)

Der Konnektor MUSS Clientsystemen den Basisdienst Signaturdienst (nonQES und QES) anbieten.

Tabelle 192: TAB_KON_197 Basisdienst Signaturdienst (nonQES und QES)

Name	`SignatureService`	
Version (KDV)	wird im Produktsteckbrief des Konnektors definiert	
Namensraum	Siehe Anhand D	
Namensraum-Kürzel	`SIG für Schema und SIGW für WSDL`	
Operationen	**Name**	**Kurzbeschreibung**
	`SignDocument`	Dokument signieren
	`VerifyDocument`	Signatur verifizieren
	`ExternalAuthenticate`	Binärstring signieren (nonQES)
	`SetXTVView`	Darstellung von xTV festlegen
	`GetXTVView`	aktuelle Darstellung von xTV erfragen
WSDL	`SignatureService.wsdl`	
Schema	`SignatureService.xsd`	

⌫

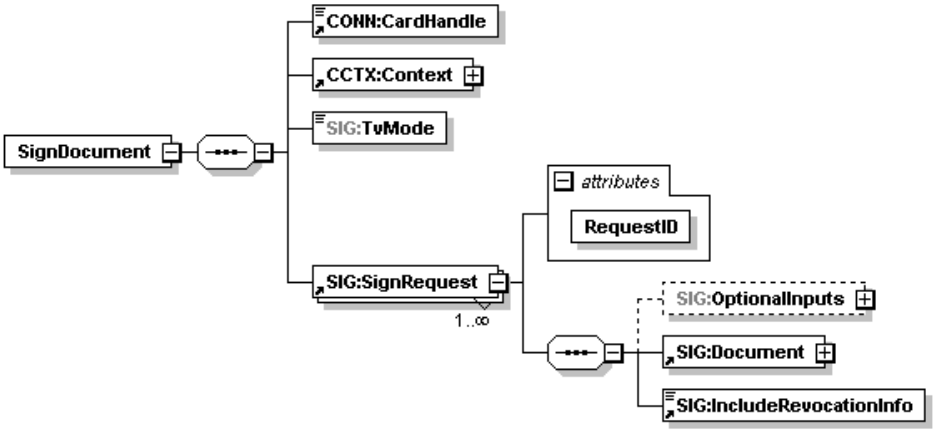
4.1.8.5.1 SignDocument (nonQES und QES)

⌦ TIP1-A_5010 Operation SignDocument (nonQES und QES)

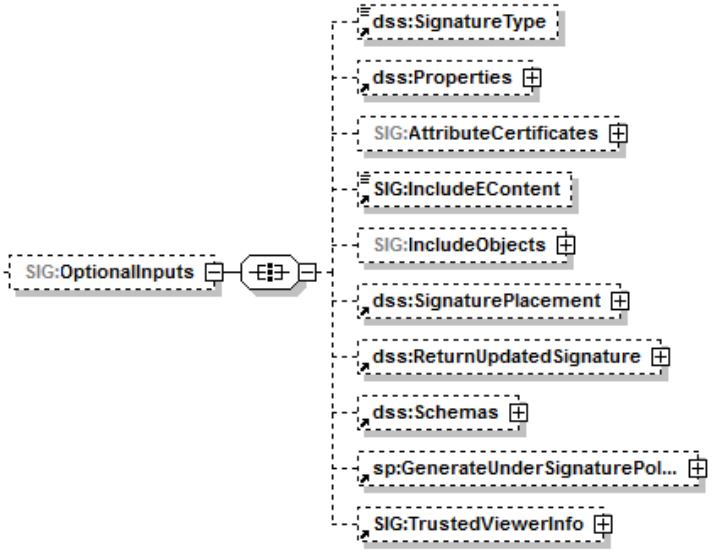
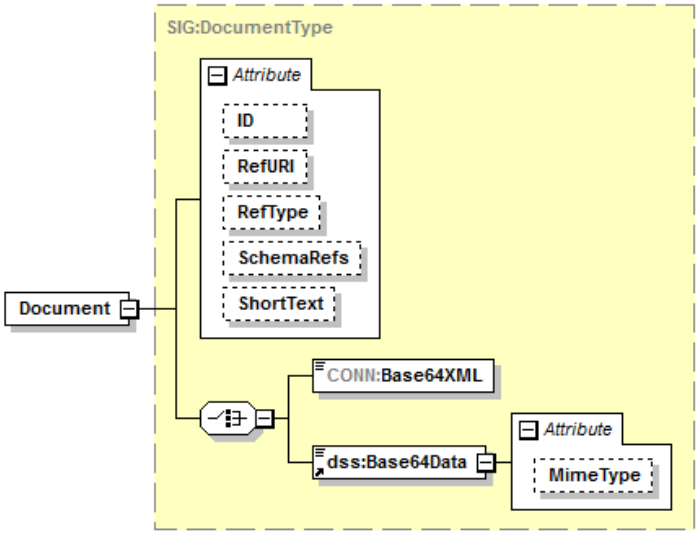
Der Signaturdienst des Konnektors MUSS an der Clientschnittstelle eine an [OASIS-DSS] angelehnte Operation SignDocument anbieten**.**

Tabelle 193: TAB_KON_065 Operation SignDocument (nonQES und QES)

Name	`SignDocument`
Beschreibung	Diese Operation lehnt sich an [OASIS-DSS] an. Sie enthält voneinander unabhängige SignRequests. Jeder SignRequest erzeugt eine Signatur für ein Dokument. Für die qualifizierte elektronische Signatur (QES) werden die `QES_DocFormate` unterstützt. Für nicht-qualifizierte elektronische Signaturen (nonQES) werden die `nonQES_DocFormate` unterstützt. Zur Signaturerzeugung werden Schlüssel und Zertifikate einer Chipkarte benutzt. 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. Bei der Erstellung von XML-Signaturen MUSS Canonical XML 1.1 verwendet werden [CanonXML1.1]. Es soll der Common-PKI-Standard eingesetzt werden, siehe [Common-PKI]. In Summe für die Größe der Dokumente in allen SignRequests innerhalb einer

	SignDocument-Anfrage MUSS der Konnektor eine Gesamtgröße von <= 250 MB unterstützen.
Aufrufparameter	

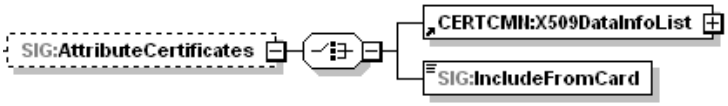
Name	Beschreibung
`CONN:CardHandle`	Identifiziert die zu verwendende Signaturkarte. Die Operation DARF die Signatur mit der eGK NICHT unterstützen. Wird die Operation mit einem nicht unterstützten Kartentypen aufgerufen, so MUSS der Konnektor die Bearbeitung mit dem Fehler 4126 abbrechen.
`CCTX:Context`	<u>Aufrufkontext QES mit HBAx</u>: MandantId, ClientSystemId, WorkplaceId, UserId verpflichtend <u>Aufrufkontext nonQES mit SM-B</u>: MandantId, ClientSystemId, WorkplaceId verpflichtend; UserId nicht ausgewertet
`TvMode`	Legt das Verhalten des Extended Trusted Viewers für den SignRequest-Stapel fest. <u>Erlaubte Werte bei QES und nonQES:</u> CONFIRMED (Bestätigungsmodus): Unter Nutzung des `ShortText`-Attributes aus den `SIG:Document`-Elementen werden die zu signierenden Dokumente im Extended Trusted Viewer angezeigt. Der Benutzer kann Dokumente durch Deselektion von der Signatur ausschließen und sich den Inhalt einzelner zu signierender Dokumente anzeigen lassen. Der Signaturvorgang wird erst nach einer Bestätigung durch den Benutzer im Extended Trusted Viewer gestartet. UNCONFIRMED (Ansichtsmodus): Im Vergleich zum Bestätigungsmodus wird nicht auf eine Bestätigung des Signaturvorgangs durch den Benutzer im Extended Trusted Viewer gewartet. Eine Deselektion einzelner zu signierender Dokumente durch den Benutzer ist nicht möglich. <u>Zusätzlich erlaubter Wert bei der nonQES:</u> NONE Keine Anzeige im Extended Trusted Viewer. (siehe weitere Anzeige gemäß TIP1-A_4656: Anzeige verpflichtender Parameter bei Signaturerstellung).
`SIG:SignRequest`	Ein `SignRequest` kapselt den Signaturauftrag für ein Dokument.

		Das verpflichtende XML-Attribut `RequestID` identifiziert einen `SignRequest` innerhalb eines Stapels von `SignRequests` eindeutig. Es dient der Zuordnung der `SignResponse` zum jeweiligen `SignRequest`.
	`SIG:OptionalInputs`	Enthält optionale Eingangsparameter (angelehnt an `dss:OptionalInputs` gemäß [OASIS-DSS] Section 2.7): 
	`SIG:Document`	 Dieses an das `dss:Document` Element aus [OASIS-DSS] Section 2.4.2 angelehnte Element kann mehrmals auftreten und enthält die zu signierenden Dokumente, wobei die Kindelemente `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,

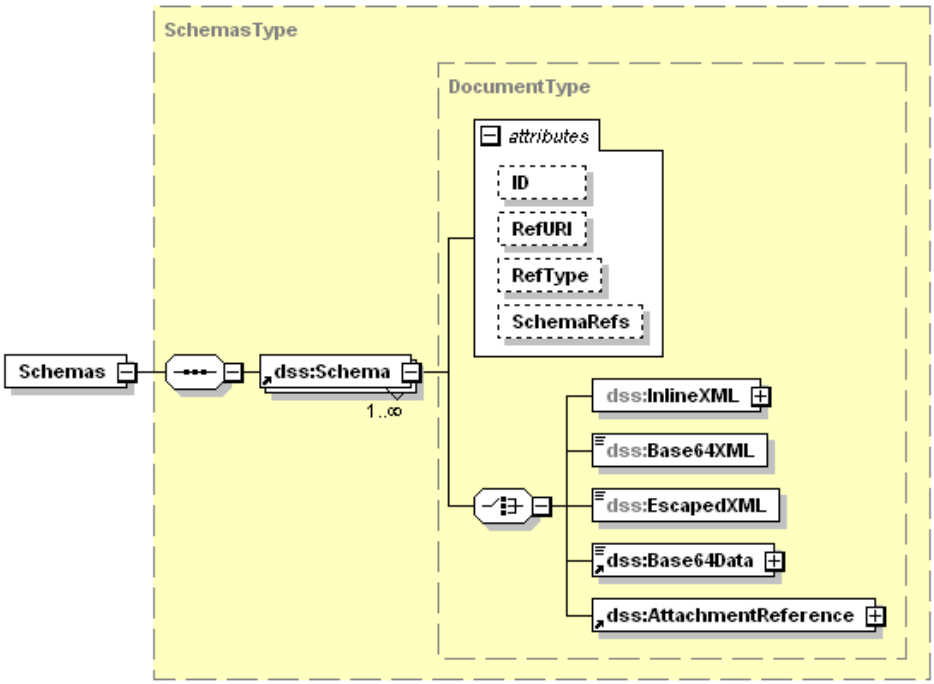
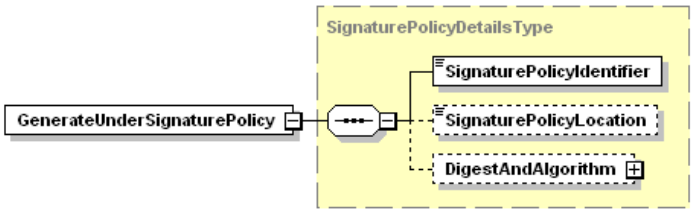
		• ”image/tiff” – für TIFF-Dokumente und • ein beliebiger anderer MIME-Type für nicht näher unterschiedene Binärdaten des spezifizierten Typs. Der MIME-Type „text/plain“ wird interpretiert als „text/plain; charset=iso-8859-15”. Das Element enthält ein Attribut `ShortText`, das für die Anzeige im Extended Trusted Viewer genutzt wird und das dem Benutzer zur Identifikation des zu signierenden Dokuments dient. Es muss für QES-Signaturen bei jedem Aufruf vom Clientsystem übergeben werden, für nonQES-Signaturen ist es optional.
	`SIG:IncludeRevocationInfo`	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. Für nicht-qualifizierte elektronische Signaturen (nonQES) wird diese Funktionalität nicht unterstützt.
	`dss:SignatureType`	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: • **XML-Signatur** Durch Übergabe der URI urn:ietf:rfc:3275 wird die Erstellung von XML-Signaturen gemäß [RFC3275], [XMLDSig] angestoßen. Das zu verwendende Profil ist XAdES-BES ([XAdES]). Die Rückgabe einer solchen Signatur erfolgt als `ds:Signature`-Element. • **CMS-Signatur** Durch Übergabe der URI urn:ietf:rfc:5652 wird eine CMS-Signatur gemäß [RFC5652] angestoßen. Das zu verwendende Profil ist CAdES-BES ([CAdES]). Die Signatur wird als `dss:Base64Signature` mit der oben genannten URI als `Type` zurückgeliefert. • **S/MIME-Signatur** Durch Übergabe der URI “urn:ietf:rfc:5751” wird eine S/MIME-Signatur gemäß [RFC5751] angestoßen. 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 SignedData. Ist das übergebene Dokument keine MIME-Nachricht, so wie der Fehler 4111 (Ungültiger Signaturtyp oder Signaturvariante) zurückgeliefert. • **PDF-Signatur** Durch Übergabe der URI http://uri.etsi.org/02778/3 wird die Erzeugung einer PAdES-Basic Signatur gemäß [PAdES-3]angestoßen, wobei das Dokument mit der

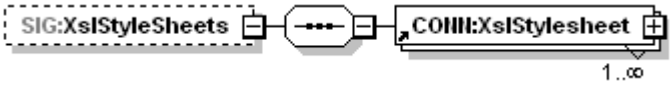
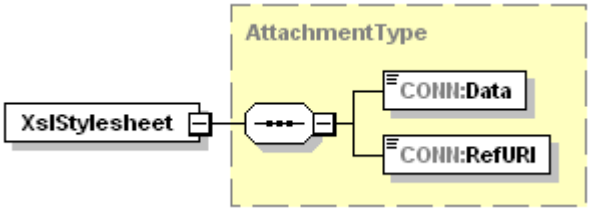
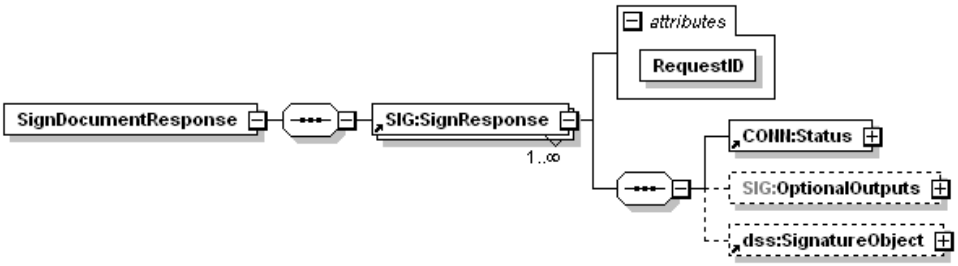
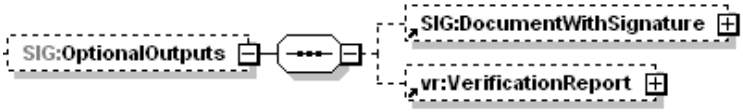
		integrierten Signatur als `dss:Base64Signature` mit der oben genannten URI als `Type` zurückgeliefert wird. Handelt es sich beim übergebenen Dokument nicht um ein `Base64Data`-Element mit MIME-Type „application/pdf-a“, so wird ein Fehler 4111 (Ungültiger Signaturtyp oder Signaturvariante) zurückgeliefert. Andere `SignatureType`-Angaben führen zu einer Fehlermeldung 4111 (Ungültiger Signaturtyp oder Signaturvariante). 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.

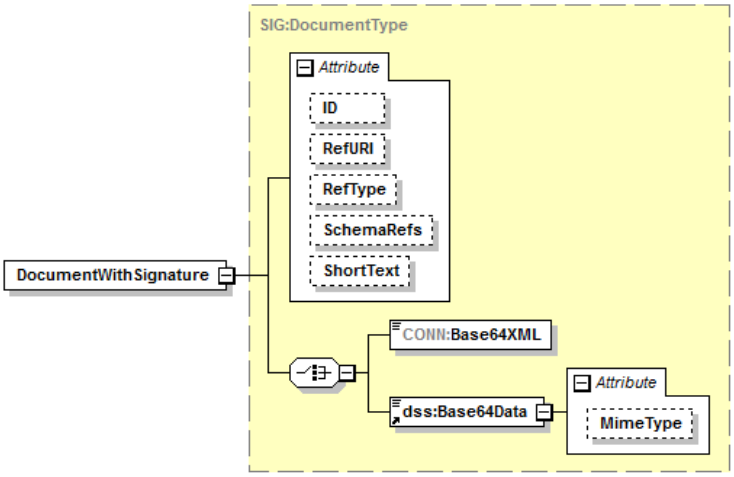
	`dss:Properties`	Durch dieses in [OASIS-DSS] (Abschnitt 3.5.5) definierte Element können zusätzliche signierte und unsignierte Eigenschaften (Properties) bzw. Attribute in die Signatur eingefügt werden.
		
		Im CMS-Fall (SignatureType = urn:ietf:rfc:5652) kann es XML-Elemente
		`./SignedProperties/Property/Value/CMSAttribute`
		und
		`./UnsignedProperties/Property/Value/CMSAttribute`
		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.

	`SIG:AttributeCertificates`	 Dieses Element ist nur im Fall einer QES-Signatur relevant. Ist dieses Element nicht vorhanden, werden keine Attributzertifikate eingebetttet. Ist dieses Element vorhanden, können über das Element CERTCMN:X509DataInfoList Attributzertifikate beim Aufruf in dem Format mitgegeben werden, wie es die Operation ReadCardCertificate des Zertifikatsdienstes liefert. Alternativ kann der Konnektor über das Element SIG:IncludeFromCard angewiesen werden, alle Attributzertifikate zum Basiszertifikat von der signierenden Karte zu lesen. Die Zertifikate bettet der Konnektor für XAdES unter

		`ds:Signature/ds:Object/QualifyingProperties/SignedProperties/SignedSignatureProperties/SignerRole/CertifiedRoles/CertifiedRole` in die Signatur ein und für CAdES und PAdES unter `SignedData/signerInfos/SignerInfo/SignedAttrs/SignerAttributes/CertifiedAttributes`
	`SIG:IncludeEContent`	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. Die Verwendung dieses Parameters bei anderen Signaturtypen führt zu einem Fehler 4111 (Ungültiger Signaturtyp oder Signaturvariante).
	`SIG:IncludeObject`	Dieses Element enthält eine Folge von dss:IncludeObject-Elementen gemäß [OASIS-DSS] (Abschnitt 3.5.6), die bei einer XML-Signatur gemäß [RFC3275] die Liste der Objekte, die in die Signatur einbezogen werden sollen, spezifiziert. Ist eine XML-Signatur angefordert und fehlt dieses Element, so wird jeweils das komplette Dokument signiert. Ist das Element vorhanden und ein anderer Signaturtyp angefordert, so wird der Fehler 4111 (Ungültiger Signaturtyp oder Signaturvariante) zurückgeliefert.
	`dss:SignaturePlacement`	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. Bei anderen Signaturtypen wird das Element ignoriert und eine Warnung (Fehlercode 4197, Parameter SignaturePlacement wurde ignoriert) zurückgeliefert.
	`dss:ReturnUpdatedSignature`	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: • http://ws.gematik.de/conn/sig/sigupdate/parallel Hierdurch wird eine Parallelsignatur zu einer bereits existierenden Signatur erzeugt und entsprechend zurückgeliefert. • http://ws.gematik.de/conn/sig/sigupdate/counter/documentincluding Hierdurch wird eine dokumentinkludierende Gegensignatur für das Dokument und alle vorhandenen parallelen Signaturen erzeugt. • http://ws.gematik.de/conn/sig/sigupdate/counter/documentexcluding Hierdurch wird eine dokumentenexkludierende Gegensignatur für alle vorhandenen parallelen Signaturen erzeugt. Bei anderen `Type`-Attributen wird der Fehler 4111 (Ungültiger Signaturtyp oder Signaturvariante) zurückgeliefert.
	`dss:Schemas`	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.
		
	`dss:Schema`	Dieses Element enthält ein XML-Schema zur Validierung des übergebenen XML-Dokuments. Das Attribut `RefURI` ist verpflichtend. Es kennzeichnet dabei den Namensraum des XML-Schemas entsprechend [OASIS-DSS] (Abschnitt 2.8.5)
	`sp:GenerateUnderSignaturePolicy`	 Über dieses in [OASIS-SP], Kapitel 2.2.1.1.1.1 Optional Input <GenerateUnderSignaturePolicy>, definierte Element wird die für die sichere Anzeige von XML-Dokumenten (DF_SV_XML) erforderliche Singnaturrichlinie ausgewählt. Die im Element `sp:SignaturePolicyIdentifier` übergebene URI identifiziert die Signaturrichtlinie.
	`SIG:TrustedViewerInfo`	Enthält Informationen zur Ansteuerung des Trusted Viewers. Die Struktur dieses Elementes ist nachfolgend erläutert. Für definierte XML-Formate gemäß DF_SV_XML MUSS dieses optionale Element nicht vorhanden sein. Das zur Anzeige zu verwendende Stylesheet wird hierbei vom Konnektor durch das erkannte Format des XML-Dokuments bestimmt.

	`SIG:XslStylesheets`	Liste von XSL-Stylesheets zur Aufbereitung des XML-Dokuments für die Anzeige. Hierbei MUSS das Haupt-Stylesheet als erstes Element übergeben werden.
		
		
	`CONN:XslStylesheet`	Dieses Element enthält ein base64-codiertes Stylesheet im CONN:Data-Element und eine das Stylesheet identifizierende URI im CONN:RefURI-Element.
Rückgabe		
	`SIG:SignResponse`	Eine `SignResponse` kapselt den ausgeführten Signaturauftrag pro Dokument. Die Zuordnung zwischen `SignRequest` und `SignResponse` erfolgt über die RequestID. Für Dokumente, die vom Benutzer durch Deselektion von der Signaturerzeugung ausgeschlossen wurden, wird ebenfalls ein Element SignResponse zurückgegeben.
	`CONN:Status`	Enthält den Status der ausgeführten Operation pro SignRequest.
	`SIG:OptionalOutputs`	Enthält (angelehnt an `dss:OptionalOutputs`) optionale Ausgangsparameter: 

	`SIG:DocumentWithSignature`	 Pro `SignResponse` wird ein Element `SIG:DocumentWithSignature` gemäß [OASIS-DSS] (Abschnitt 3.5.8) zurückgeliefert, in dem das Dokument mit Signatur enthalten ist. Dabei werden die XML-Attribute des Elements `SIG:Document` auf dem zugehörigen `SignRequest` übernommen. Ist die Signatur nicht im Dokument enthalten, wird ein leeres Element `Base64XML` oder `Base64Data` zurückgegeben. Die Signatur wird dann im Element `dss:SignatureObject` abgelegt. Wenn die Signatur im Dokument enthalten ist, wird das signierte Dokument im Feld `Base64XML` bzw. `Base64Data` zurückgeliefert. In diesem Fall MUSS die `dss:SignaturePtr`-Alternative in `dss:SignatureObject` (vgl. [OASIS-DSS] Abschnitt 2.5) dazu genutzt werden, auf die in den Dokumenten enthaltenen Signaturen zu verweisen.
	`vr:VerificationReport`	Dieses in [OASIS-VR] spezifizierte Element wird zurückgeliefert, falls im Rahmen einer Gegensignatur eine Signatur geprüft wurde, und VerificationResult gleich INVALID oder INCONCLUSIVE.
	`dss:SignatureObject`	Enthält im Erfolgsfall die erzeugte Signatur pro `SignRequest` in Form eines `dss:SignatureObject`-Elementes gemäß [OASIS-DSS] (Abschnitt 3.2). Sofern ein übergebenes Dokument durch Deselektion von der Signaturerzeugung ausgeschlossen wurde, wird dies durch ein leeres `dss:SignatureObject/Other/Deselected`-Element angezeigt, ein Abbruch im Confirmed Modus im xTV durch ein leeres `dss:SignatureObject/Other/CanceledAtxTV`-Element, ein Abbruch am KT durch ein leeres `dss:SignatureObject/Other/CanceledAtKT`-Element.
Vorbedingungen	Keine	
Nachbedingungen	Keine	

Der Ablauf der Operation SignDocument ist in Tabelle 194: TAB_KON_756 Ablauf Operation SignDocument (nonQES und QES) beschrieben:

Tabelle 194: TAB_KON_756 Ablauf Operation SignDocument (nonQES und QES)

Nr.	Aufruf Technischer Use Case oder Interne 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 „Prüfe Zugriffs-berechtigung“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; $context.userId; $cardHandle} 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 { MandantId, CsId, CardHandle, UserId }
Im Fall QES wird Schritt 4 ausgeführt. Im Fall nonQES wird Schritt 5 ausgeführt.		
4a)	Prüfe SAK-Modul	Prüfe, ob MGM_LU_SAK=Enabled. Ist dies nicht der Fall, so bricht die Operation mit Fehler 4125 ab.
4b)	TUC_KON_150 „Dokumente QES signieren“	Die QES wird erzeugt. Tritt hierbei ein Fehler auf, bricht die Operation ab.
5)	TUC_KON_160 „Dokumente nonQES signieren“	Die nonQES wird erzeugt. Tritt hierbei ein Fehler auf, bricht die Operation ab.

Tabelle 195: TAB_KON_757 Übersicht Fehler Operation SignDocument (nonQES und QES)

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen TUCs können folgende weiteren Fehlercodes auftreten:			
4000	Technical	Error	Syntaxfehler
4126	Security	Error	Kartentyp nicht zulässig für Signatur
4125	Technical	Error	LU_SAK nicht aktiviert
4197	Technical	Warning	Parameter SignaturePlacement wurde ignoriert

⌫

Die folgende Tabelle führt die zulässigen Zertifikate und Schlüssel für die nonQES auf:

Tabelle 196: TAB_KON_758 Zertifikat und privater Schlüssel je Karte für Sign/VerifyDocument (nonQES)

Karte	Zertifikat (Verify)	Schlüssel (Sign)
SM-B	EF.C.HCI.OSIG.R2048 in DF.ESIGN	PrK.HCI.OSIG.R2048 in DF.ESIGN

Die folgende Tabelle führt zulässige Werte für den Parameter KeyReference für die QES auf:

Tabelle 197: TAB_KON_759 Zertifikat und privater Schlüssel je Karte für Sign/VerifyDocument (QES)

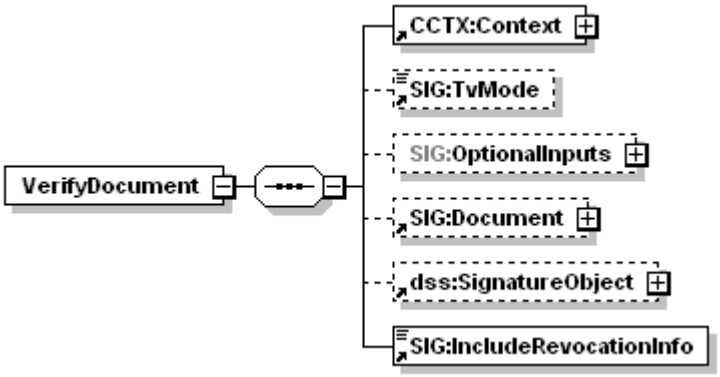
Karte	Zertifikat (Verify)	Schlüssel (Sign)
HBA-VK	EF.C.HP.QES in DF.QES	PrK.HP.QES in DF.QES
HBA	DF.C.HP.QES.R2048 in DF.QES	PrK.HP.QES.R2048 in DF.QES

4.1.8.5.2 VerifyDocument (nonQES und QES)

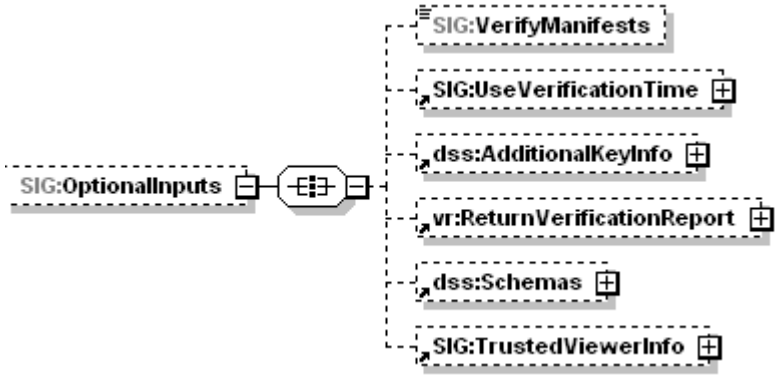
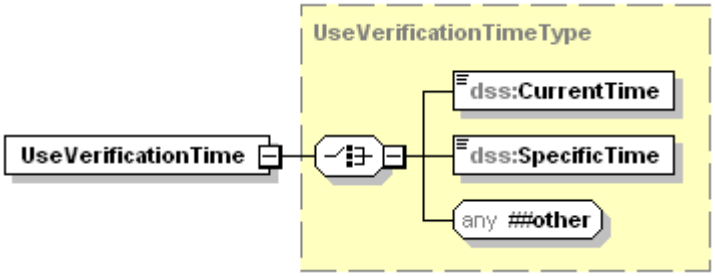
⌦ TIP1-A_5034 Operation VerifyDocument (nonQES und QES)

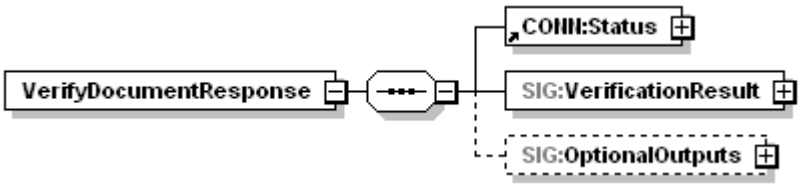
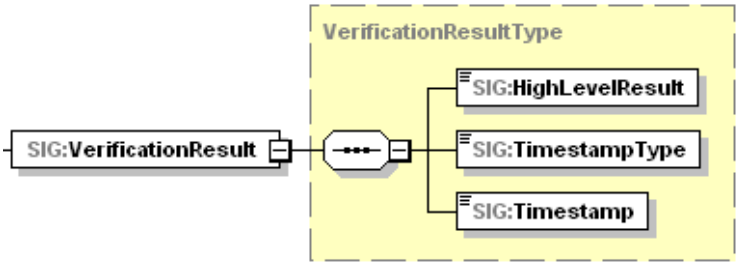
Der Signaturdienst des Konnektors MUSS an der Clientschnittstelle eine an [OASIS-DSS] angelehnte Operation VerifyDocument (nonQES und QES) anbieten**.**

Tabelle 198: TAB_KON_066 Operation VerifyDocument (nonQES und QES)

Name	`VerifyDocument`
Beschreibung	Diese Operation verifiziert die Signatur eines Dokumentes. Der Konnektor MUSS jede konform zur Außenschnittstelle SignDocument erzeugte Signatur durch VerifyDocument prüfen können. Darüber hinaus müssen im Fall QES, falls vorhanden, auch qualifizierte Zeitstempel geprüft werden. Außerdem MÜSSEN die zusätzlich geforderten Signaturverfahren zur Dokumentensignaturprüfung unterstützt werden. Das Ergebnis der Prüfung wird, wenn gefordert, in Form eines standardisierten Prüfberichts in einer `VerificationReport`-Struktur gemäß [OASIS-VR] zurückgeliefert.
Aufrufparameter	

	Name	Beschreibung
	`CCTX:Context`	MandantId, ClientSystemId, WorkplaceId verpflichtend; UserId nicht ausgewertet
	`TvMode`	Der optionale Parameter legt das Verhalten des Extended Trusted Viewers fest. Erlaubter Wert für TvMode bei QES und nonQES: UNCONFIRMED (Ansichtsmodus): Dem Benutzer wird das Ergebnis der Signaturprüfung angezeigt. Der Benutzer kann sich den Inhalt des signierten Dokuments und des Zertifikats im Extended Trusted Viewer anzeigen lassen. Zusätzlich erlaubter Wert für TvMode bei nonQES: NONE Keine Anzeige im Extended Trusted Viewer. (siehe weitere Anzeige gemäß TIP1-A_4673 xTV: Anzeige verpflichtender Parameter bei Signaturprüfung und Beschreibung Parameter TvMode bei Operation SignDocument). default: UNCONFIRMED
	`SIG:OptionalInputs`	Enthält optionale Eingabeparameter (angelehnt an dss:OptionalInputs gemäß [OASIS-DSS] Section 2.7): Die zulässigen optionalen Eingabeparameter sind unten erläutert.
	`SIG: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), das für die Darstellung im Extended Trusted Viewer ein `ShortText`-Attribut enthalten muss. Das ShortText-Attribut dient der Identifikation des jeweiligen Dokuments durch den Benutzer.
	`dss:SignatureObject`	Enthält die zu prüfende Signatur, wenn sie nicht im Dokument selbst eingebettet ist ([OASIS-DSS] Kapitel 4.1). Hierbei werden **XML-Signaturen** als `ds:Signature` Element und alle anderen Signaturen als `dss:Base64Signature` mit entsprechend gesetztem `Type`-Attribut (siehe `SignatureType`, Operationen SignDocument und ExternalAuthenticate) übergeben, wobei die nachfolgenden Werte unterstützt werden müssen: • **CMS-Signatur** urn:ietf:rfc:5652 • **S/MIME-Signatur** urn:ietf:rfc:5751 • **PDF-Signatur** http://uri.etsi.org/02778/3 • **PKCS#1-Signatur** urn:ietf:rfc:3447
	`SIG:IncludeRevocationInfo`	Durch diesen verpflichtenden Schalter kann der Aufrufer die Einbettung von zum Zeitpunkt der Signaturprüfung vorliegenden Sperrinformationen anfordern. Ist bereits eine Sperrinformation eingebettet, so wird die

		neue Sperrinformation zusätzlich eingebettet. Für in einer Gegensignatur enthaltene Signaturen erfolgt keine Einbettung von Sperrinformationen.
		
	`SIG:VerifyManifests`	Durch das in [OASIS-DSS] (Abschnitt 4.5.1) definierte Element kann die Prüfung eines ggf. vorhandenen Manifests angefordert werden.
		
	`SIG:UseVerificationTime`	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.
	`dss:AdditionalKeyInfo`	Durch das in [OASIS-DSS] (Abschnitt 4.5.4) spezifizierte Element kann zusätzliches, für die Prüfung benötigtes, Schlüsselmaterial übergeben werden.
	`vr:ReturnVerificationReport`	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. Die Unterstützung der Konformitätsstufe 3 („Convenient“) wird EMPFOHLEN.
	`dss: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. Zur Struktur dieses Elements siehe Beschreibung des Parameters `dss:Schemas` der Operation SignDocument.
	`SIG:TrustedViewerInfo`	Enthält Informationen zur Ansteuerung des Extended Trusted Viewers. Zur Struktur dieses Elements siehe Beschreibung des Parameters `SIG:TrustedViewerInfo` der Operation SignDocument.

Rückgabe		
	`Status`	Enthält den Ausführungsstatus der Operation.
	`SIG:VerificationResult`	 Das Element `Sig:VerificationResult` enthält das Ergebnis der Prüfung als Ampel, den Typ des zugehörigen angenommenen Signaturzeitpunkts und der angenommene Signaturzeitpunkt selbst.
	`SIG:HighLevelResult`	Das Ergebnis der Prüfung (Ampelschaltung) mit folgenden Werten: • VALID: alle Signaturen sind gültig • INVALID: mindestens eine der Signaturen ist ungültig • INCONCLUSIVE: in allen anderen Fällen
	`SIG:TimestampType`	Der Typ des angenommenen Signaturzeitpunkts mit folgenden Werten: • SIGNATURE_EMBEDDED_TIMESTAMP: in der Signatur eingebetter Zeitpunkt `Ermittelter_Signaturzeitpunkt_Eingebettet` • QUALIFIED_TIMESTAMP: qualifizierter Zeitstempel über die Signatur `Ermittelter_Signaturzeitpunkt_Qualifiziert` • SYSTEM_TIMESTAMP: Systemzeit des Konnektors bei Signaturprüfung `Ermittelter_Signaturzeitpunkt_System` • USER_DEFINED_TIMESTAMP: benutzerdefinierter Zeitpunkt `Benutzerdefinierter_Zeitpunkt`
	`SIG:OptionalOutputs`	Enthält (angelehnt an dss:OptionalOutputs, wie in Abschnitt 2.7 von [OASIS-DSS] beschrieben) optionale Ausgangselemente:

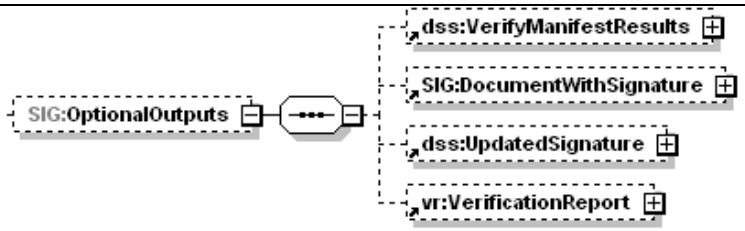
		
	`dss:VerifyManifestResults`	Dieses in Abschnitt 4.5.1 von [OASIS-DSS] definierte Element enthält Informationen zur Prüfung eines ggf. vorhandenen Signaturmanifests und wird zurückgeliefert, sofern beim Aufruf das `dss:VerifyManifest`-Element, aber nicht das `RequestVerificationReport` als optionales Eingabeelement übergeben wurde.
	`SIG:DocumentWithSignature`	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. Das XML-Attribut `ShortText` aus der Anfrage wird mit dem signierten `ShortText` überschrieben.
	`dss:UpdatedSignature`	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:VerificationReport`	Dieses in [OASIS-VR] spezifizierte Element wird zurückgeliefert, falls das `ReturnVerificationReport`-Element als Eingabeparameter verwendet wurde.
Vorbedingungen	Keine	
Nachbedingungen	Keine	

Tabelle 199: TAB_KON_760 Ablauf Operation VerifyDocument (nonQES und QES)

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“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; doNotNeedCardSession} Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.
3.	prüfe, ob QES oder 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

Nr.	Aufruf Technischer Use Case oder Interne Operation	Beschreibung
		liegt eine nonQES-Signatur vor.
Für QES-Signaturen wird Schritt 4 ausgeführt. Für nonQES-Signaturen wird Schritt 5 ausgeführt.		
4a.	Prüfe SAK-Modul	Prüfe, ob MGM_LU_SAK=Enabled. Ist dies nicht der Fall, so bricht die Operation mit Fehler 4125 ab.
4b.	TUC_KON_151 „QES Dokumentensignatur prüfen“	Die QES wird geprüft. Tritt hierbei ein Fehler auf, bricht die Operation ab.
5.	TUC_KON_161 „nonQES Dokumentensignatur prüfen“	Die nonQES wird geprüft. Tritt hierbei ein Fehler auf, bricht die Operation ab.

Tabelle 200: TAB_KON_761 Übersicht Fehler Operation VerifyDocument (nonQES und QES)

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen TUCs (siehe Tabelle 199: TAB_KON_760 Ablauf Operation VerifyDocument) können folgende weiteren Fehlercodes auftreten:			
4000	Technical	Error	Syntaxfehler
4125	Technical	Error	LU_SAK nicht aktiviert

⌫

4.1.8.5.3 ExternalAuthenticate

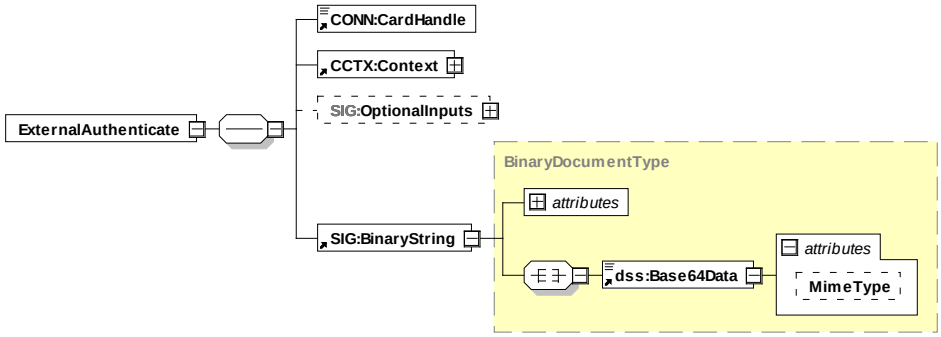
⌦ TIP1-A_5439 Operation ExternalAuthenticate

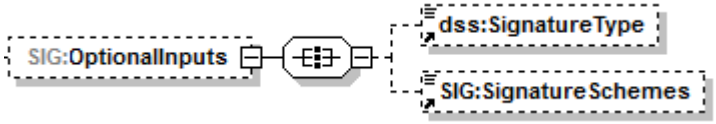
Der Signaturdienst des Konnektors MUSS an der Clientschnittstelle eine Operation ExternalAuthenticate anbieten**.**

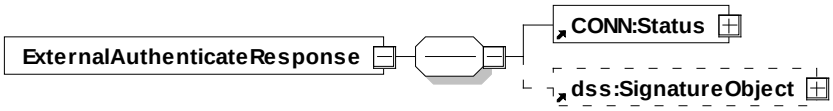
Tabelle 201: TAB_KON_781 Operation ExternalAuthenticate

Name	`ExternalAuthenticate`
Beschreibung	Diese Operation versieht einen Binärstring der maximalen Länge 512 Bit mit einer nicht-qualifizierten elektronischen Signatur (nonQES). Dazu wird das Signaturverfahren PKCS#1 verwendet. Das AUT-Zertifikat der SM-B und das AUT-Zertifikat des HBAx werden unterstützt.

Aufrufparameter



Name	Beschreibung
`CONN:CardHandle`	Identifiziert die zu verwendende Signaturkarte. Die Operation unterstützt HBAx und SM-B.
`CCTX:Context`	Aufrufkontext für HBAx: MandantId, ClientSystemId, WorkplaceId, UserId verpflichtend Aufrufkontext für SM-B: MandantId, ClientSystemId, WorkplaceId verpflichtend; UserId nicht ausgewertet
`SIG:OptionalInputs`	Enthält optionale Eingangsparameter: 
`SIG:BinaryString`	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.
`dss:SignatureType`	Durch dieses in [OASIS-DSS] (Abschnitt 3.5.1) beschriebene Element wird der Typ der zu erzeugenden Signatur bestimmt. Als Signaturtyp wird unterstützt : • **PKCS#1-Signatur** Durch Übergabe der URI urn:ietf:rfc:3447 wird eine PKCS#1 (Version 2.1) Signatur gemäß [RFC3447] erzeugt, die als `dss:Base64Signature` mit der oben genannten URI zurückgeliefert wird. Andere `SignatureType`-Angaben führen zu einer Fehlermeldung 4111 (Ungültiger Signaturtyp oder Signaturvariante). Fehlt dieses Element, so wird ebenfalls der Signaturtyp PKCS#1-Signatur verwendet.
`SIG:SignatureSchemes`	Durch dieses Element wird für PKCS#1-Signaturen zwischen den folgenden SignatureScheme-Optionen unterschieden: • RSASSA-PSS

		• RSASSA-PKCS1-v1_5 Fehlt dieses Element, so wird als Default-SignatureScheme RSASSA-PSS gewählt.
Rückgabe		
	`CONN:Status`	Enthält den Status der ausgeführten Operation.

	`dss:SignatureObject`	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. Das XML-Attribut `dss:SignatureObject/dss:Base64Signature/@Type` kennzeichnet durch den Wert urn:ietf:rfc:3447 den Signatur-Typ.
		Die XML-Elemente
		`dss:SignatureObject/ds:Signature`
		`dss:SignatureObject/dss:Timestamp`
		`dss:SignatureObject/dss:SignaturePtr`
		`dss:SignatureObject/dss:Other`
		werden nicht verwendet.

Vorbedingungen	Keine	
Nachbedingungen	Keine	

Der Ablauf der Operation ExternalAuthenticate ist in Tabelle TAB_KON_782 beschrieben:

Tabelle 202: TAB_KON_782 Ablauf Operation ExternalAuthenticate

Nr.	Aufruf Technischer Use Case oder Interne 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 „Prüfe Zugriffs-berechtigung“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; $context.userId; $cardHandle} Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.
3.	TUC_KON_026	Ermittle CardSession über TUC_KON_026 { MandantId, CsId,

Nr.	Aufruf Technischer Use Case oder Interne Operation	Beschreibung
	„Liefere CardSession“	CardHandle, UserId }
4.	TUC_KON_160	Dokumente nonQES signieren; im Aufruf von TUC_KON_160 den Parameter tvMode=NONE setzen.

Tabelle 203: TAB_KON_783 Übersicht Fehler Operation ExternalAuthenticate

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen TUCs können folgende weiteren Fehlercodes auftreten:			
4000	Technical	Error	Syntaxfehler
4058	Security	Error	Aufruf nicht zulässig

⌫

Die folgende Tabelle führt die zulässigen privaten Schlüssel für die Operation ExternalAuthenticate auf:

Tabelle 204: TAB_KON_784 Privater Schlüssel je Karte für ExternalAuthenticate

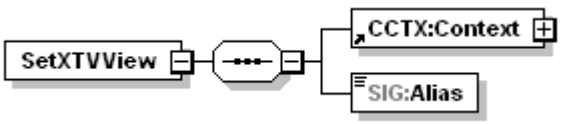
Karte	Schlüssel
SM-B	PrK.HCI.AUT in DF.ESIGN
HBAx	PrK.HP.AUT in DF.ESIGN

4.1.8.5.4 SetXTVView

⌦ TIP1-A_4677 Operation SetXTVView

Der Signaturdienst des Konnektors MUSS an der Clientsystemschnittstelle eine Operation SetXTVView anbieten.

Tabelle 205: TAB_KON_264 Operation SetXTVView

Name	`SetXTVView`	
Beschreibung	Diese Operation ermöglicht einem Clientsystem in vorgegebenen Grenzen die Festlegung der Darstellungsweise (Position und Größe) eines Extended Trusted Viewers (xTV) für alle folgenden Aufrufe, in denen dieser involviert ist. Damit kann ein Clientsystemanbieter steuern, wie sich ein xTV an einem Arbeitsplatz darstellt und dies mit den eigenen Masken im Rahmen eines Workflows harmonisieren.	
Aufrufparameter		
	Name	**Beschreibung**
	`CCTX:Context`	MandantId, ClientSystemId, WorkplaceId verpflichtend; UserId wird nicht ausgewertet

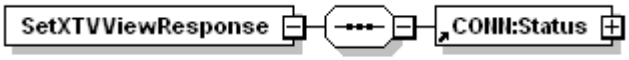
	`Alias`	Bezeichner einer xTV-Darstellungsweise für kommende Aufrufe (siehe Konfigurationsparameter SAK_ALIAS_SET in Kapitel 4.1.8.6)
Rückgabe		
	`Status`	Enthält den Ausführungsstatus der Operation.
Vorbedingungen	Es ist erforderlich, dass gemäß Infomodell die richtige workplaceId zur Auswahl der xTV-Instanz angegeben wird. Ein pauschales Setzen aller xTV wird nicht unterstützt.	
Nachbedingungen	Sobald im Rahmen eines fachlichen Ablaufs der betreffende xTV angezeigt wird, muss er an der gewählten Alias-Position angezeigt werden.	
Hinweise	Es ist angeraten, sprechende Bezeichner für vordefinierte Aliase zu wählen, um den Clientsystemen die Auswahl zu erleichtern.	

Tabelle 206: TAB_KON_762 Ablauf Operation SetXTVView

Nr.	Aufruf Technischer Use Case oder Interne Operation	Beschreibung
1.	TUC_KON_000 „Prüfe Zugriffs-berechtigung“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; $context.userId; $cardHandle} Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.
2.	checkArguments	Die übergebenen Werte werden auf Konsistenz und Gültigkeit überprüft. Treten hierbei Fehler auf, so bricht die Operation mit Fehler 4000 ab.
3.	Prüfe SAK-Modul	Prüfe, ob MGM_LU_SAK=Enabled. Ist dies nicht der Fall, so bricht die Operation mit Fehler 4125 ab.
4.	setXTVView	Der Alias-Konfigurationsparameter wird gesetzt.

Tabelle 207: TAB_KON_763 Übersicht Fehler Operation SetXTVView

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen TUCs Tabelle 156: TAB_KON_746 Ablauf EncryptDocument können folgende weiteren Fehlercodes auftreten:			
4000	Technical	Error	Syntaxfehler
4119	Technical	Error	Darstellungs-Alias unbekannt
4125	Technical	Error	LU_SAK nicht aktiviert

⌫

⌦ TIP1-A_4678 Pflicht-Alias-Bezeichner für Anzeige von 10 Kurztexten

Der Konnektor MUSS über die SetVTVView/GetXTVView-Schnittstelle einen Alias-Bezeichner anbieten, welcher im Ansichtsmodus des xTV die vollständige Anzeige

von mindestens 10 Kurztexten aus der Liste der zu signierenden Dokumente ermöglicht.⌫

4.1.8.5.5 GetXTVView

⌦ TIP1-A_4679 Operation GetXTVView

Der Signaturdienst des Konnektors MUSS an der Clientsystemschnittstelle eine Operation GetXTVView anbieten.

Tabelle 208: TAB_KON_265 Operation GetXTVView

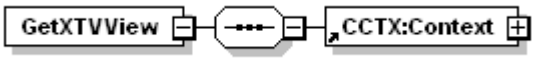
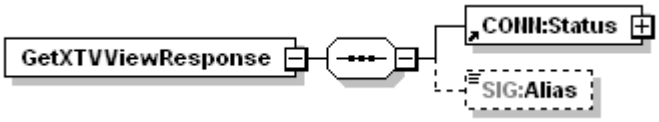
Name	`GetXTVView`	
Beschreibung	Diese Operation ermöglicht einem Clientsystem die aktuell eingestellte Darstellungsweise eines xTV auf einem bestimmten Arbeitsplatz zu erfragen. Auf Basis dieser Information kann das Clientsystem entweder die Änderung der Darstellungsweise durch Aufruf von SetXTVView veranlassen, oder die eigene Darstellung entsprechend passend wählen. Die Art der Reaktion obliegt dabei der Entscheidungshoheit des Clientsystems bzw., falls konfigurierbar, dem Administrator des Clientsystems.	
Aufrufparameter		
	Name	**Beschreibung**
	`CCTX:Context`	MandantId, ClientSystemId, WorkplaceId verpflichtend; UserId wird nicht ausgewertet
Rückgabe		
	`Alias`	Bezeichner einer xTV-Darstellungsweise.
	`Status`	Enthält den Ausführungsstatus der Operation.
Vorbedingungen	Es ist erforderlich, dass gemäß Infomodell die richtige workplaceId zur Auswahl der xTV-Instanz angegeben wird. Ein pauschales Erfragen aller xTV wird nicht unterstützt.	
Nachbedingungen	keine	
Hinweise	Welche Einstellungen letztlich mit dem gelieferten Alias verbunden sind, kann nicht über die Schnittstelle ermittelt werden. Daher ist organisatorisch sicherzustellen, dass die Interpretation des gelieferten Alias-Wertes korrekt ist. Aufgrund der Vielfalt an möglichen Parametern zum Steuern von Ausgaben/Fenstern auf unterschiedlichsten Plattformen wurde eine generalisierte und beherrschbare Form der Steuerung gewählt.	

Tabelle 209: TAB_KON_764 Ablauf Operation GetXTVView

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“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; doNotNeedCardSession } Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.
3.	Prüfe SAK-Modul	Prüfe, ob MGM_LU_SAK=Enabled. Ist dies nicht der Fall, so bricht die Operation mit Fehler 4125 ab.
4.	getXTVView	Der Alias-Konfigurationsparameter wird zurückgeliefert.

Tabelle 210: TAB_KON_765 Übersicht Fehler Operation GetXTVView

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen TUCs können folgende weiteren Fehlercodes auftreten:			
4000	Technical	Error	Syntaxfehler
4125	Technical	Error	LU_SAK nicht aktiviert

⌫

4.1.8.6 Betriebsaspekte

⌦ TIP1-A_4680 Konfigurationswerte des Signaturdienstes

Die Managementschnittstelle MUSS es einem Administrator ermöglichen Konfigurationsänderungen gemäß Tabelle TAB_KON_596 vorzunehmen:

Tabelle 211 TAB_KON_596 Konfigurationswerte des Signaturdienstes (Administrator)

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
SAK_ALIAS_SET	`freie Belegung, z.B. Screen1Top, Screen2Left, ScreenRight, DeskTopMiddle`	Alias-Bezeichner zur Auswahl von Einstellungen der Position des Extended Trusted Viewers. Der Hersteller kann entweder ein festes Set an Aliasen definieren (z. B. Screen1Top, Screen2Left, ScreenRight, DeskTopMiddle usw.) oder die Vergabe eigener Aliase durch den Administrator (via Konfig-Datei oder Web-Interface) unterstützen. Die konkrete Belegung von Parametern (Screen-Nr; X-,Y-Pos; Breite, Höhe usw.) wird durch die SAK verwaltet.
SAK_SIMPLE_SIGNATURE_ MODE	SE#1 SE#2	Aktivierung/Deaktivierung des „Einfachsignaturmodus“ für alle HBAx für die

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
		Durchführung von Einfachsignaturen im SecurityEnvironment #1 (SE#1) für Dokumentenstapel der Größe 1 anstelle der Verwendung des SE#2. Default-Wert = SE#1
SAK_TIMEOUT_xTV	`Sekunden` `Default: 60`	Wartezeit (Timeout), nach der der Konnektor die Signaturerzeugung abbrechen soll, wenn im xTV keine Bestätigung für die Signaturerzeugung eingegeben wurde. Der Konnektor muss dem Administrator die Möglichkeit bieten, die Wartezeit des xTV zu konfigurieren.

⌫

⌦ TIP1-A_4681 Instanzen des xTV konfigurieren

Der Konnektor MUSS es dem Administrator ermöglichen, Instanzen des xTV in die Konfiguration aufzunehmen und wieder entfernen zu können.⌫

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.

Innerhalb des Zertifikatsdienstes werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): „CERT“
- Konfigurationsparameter: „CERT_“

4.1.9.1 Funktionsmerkmalweite Aspekte

⌦ 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_4683 Sicheres Einbringen des QES-Vertrauensankers

Den QES-Vertrauensanker der BNetzA MUSS zum Auslieferungszeitpunkt des Konnektors integritätsgeschützt im Konnektor hinterlegt sein. Zur Sicherstellung der Integrität MUSS die Dateiablage EF.C.BNetzA.RCA der Anwendung DF.Sicherheitsanker der gSMC-K [gemSpec_gSMC-K_ObjSys#5.7.1] 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 Konnektors 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.

⌦ **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.⌫

⌦ **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.⌫

⌦ **TIP1-A_4692 Missbrauchserkennung, zu kontrollierende Operationen**

Der Konnektor MUSS zur Unterstützung von Missbrauchserkennungen die in Tabelle TAB_KON_597 gelisteten TUCs als Einträge in EVT_MONITOR_OPERATIONS berücksichtigen.

Tabelle 212: TAB_KON_597 Operationen in EVT_MONITOR_OPERATIONS

Operationsname	OK_Val	NOK_Val	Alarmwert (Default-Grenzwert 10 Minuten- Σ)
`TUC_KON_037 „Zertifikat prüfen“`	1	5	401

⌫

4.1.9.2 Durch Ereignisse ausgelöste Reaktionen

Keine.

4.1.9.3 Interne TUCs, nicht durch Fachmodule nutzbar

4.1.9.3.1 TUC_KON_032 „TSL aktualisieren“

⌦ **TIP1-A_4693 TUC_KON_032 “TSL aktualisieren”**

Der Konnektor MUSS den technischen Use Case TUC_KON_032 „TSL aktualisieren“ umsetzen.

Tabelle 213: 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 TrustStore 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 neuen BNetzA-Root-Zertifikats und/oder CVC-Root-CA-Zertifikats werden die zugehörigen BNetzA-Cross-Zertifikate bzw. das CVC-Root-CA-Zertifikat und die Cross-CV-Zertifikate aus der TSL in den Truststore

Element	Beschreibung
	eingestellt.
Auslöser	• Aufruf durch andere TUCs
Vorbedingungen	• Ein gültiger TI-Vertrauensanker ist vorhanden • Das XML-Schema der TSL-Datei liegt vor
Eingangsdaten	• TSL aus manuellem Import (Optional) • Referenzzeitpunkt (Default: aktuelles Datum) • Flag „MGM_LU_ONLINE“ für Offline/Online-Modus
Komponenten	Konnektor
Ausgangsdaten	• Status der Prüfung
Nachbedingungen	• Aktuelle TSL-Informationen inkl. sämtlicher BNetzA-Cross-Zertifikate und CVC-Root-CA- und Cross-CV-Zertifikate liegen im Truststore vor. • Ein ggf. gelieferter neuer Vertrauensanker der TI ist in einem sicheren Speicherort gespeichert
Standardablauf	1. Der Konnektor prüft und aktualisiert ggf. die TSL durch Aufruf von TUC_PKI_001. 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 (`StatusStartingTime` des neuen TSL-Signer-CA-Zertifikats) aktiviert ist. 2. Ggf. vorhandene Cross-Zertifikate der BNetzA, CVC-Root-CA-Zertifikat und Cross-CV-Zertifikate werden genauso wie und zusammen mit den anderen CA-Zertifikaten aus der TSL extrahiert. 3. Alle Informationen aus der TSL werden im TrustStore gespeichert 4. Der Konnektor löst TUC_KON_256 {"CERT/TSL/UPDATED"; Op; Info; „"; doLog; noDisp} aus. 5. CERT_CRL_DOWNLOAD_ADDRESS wird mit den CRL-Download-Adressen aus der TSL überschrieben.
Varianten/Alternativen	(→1) Wird eine zu importierende TSL 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. (→ 1) Falls MGM_LU_ONLINE=Disabled, kann der Sperrstatus des TSL-Signer-Zertifikats nicht überprüft werden. In diesem Fall wird die Aktivierung der TSL auch ohne Prüfung des Sperrstatus durchgeführt. (→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 TSL gefunden, so wird dieser, wie dort beschrieben, extrahiert und in einem sicheren Speicherort gespeichert Vor Erreichen des Aktivierungsdatums wird die TSL ausschließlich mit dem alten TSL-Signer-Zertifikat signiert. Ab dem Aktivierungsdatum werden die publizierten TSL mit einem TSL-Signer-Zertifikat signiert, das von der neuen TSL-Signer-CA ausgestellt wurde.

Element	Beschreibung
Fehlerfälle	(→1) Tritt beim manuellen Import der Datei ein Fehler auf, wird TUC_KON_256 {„CERT/TSL/IMPORT“; Op; Error; „$Fehlerbeschreibung“; doLog; noDisp} ausgelöst. Fehlercode 4128. (→1) Schlägt der Aufruf des TUC_PKI_001 oder eines durch ihn aufgerufenen TUCs fehl, geht der Konnektor in den Betriebszustand EC_TSL_Update_Not_Successful. Die vorhandene TSL und Vertrauensanker werden weiter verwendet. Fehlercode 4127.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 214: TAB_KON_598 Übersicht Fehlercodes für „TSL aktualisieren“

Fehlercode	ErrorTyp e	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4127	Security	Error	TSL nicht gültig
4128	Technical	Error	Der manuelle Import der TSL-Datei schlägt fehl

⌫

Für die Umsetzung des TI-Vertrauensankerwechsels kann die Datei EF.C.TSL_CA_2 auf der gSMC-K als sicherer Speicherort des neuen Vertrauensankers verwendet werden.

4.1.9.3.2 TUC_KON_040 „CRL aktualisieren“

⌦ **TIP1-A_4694 TUC_KON_040 „CRL aktualisieren“**

Der Konnektor MUSS den technischen Use Case TUC_KON_040 „CRL aktualisieren“ umsetzen.

Tabelle 215: TAB_KON_767 TUC_KON_040 „CRL aktualisieren“

Element	Beschreibung
Name	TUC_KON_040 „CRL aktualisieren“
Beschreibung	Dieser TUC aktualisiert die CRL
Auslöser	• Aufruf durch andere TUCs
Vorbedingungen	• Ein gültiger Vertrauensraum
Eingangsdaten	• Manuell importierte CRL (Optional)
Komponenten	Konnektor
Ausgangsdaten	Keine
Nachbedingungen	• Eine aktuelle, gültige CRL liegt vor

Element	Beschreibung
Standardablauf	1. Der Konnektor lädt die aktuelle CRL von `CERT_CRL_DOWNLOAD_ADDRESS` herunter. 2. Die Prüfung der CRL-Signatur mit dem CRL-Signer-Zertifikat setzt sich aus folgenden Teilschritten zusammen a) Prüfung auf zeitliche Gültigkeit des CRL-Signer-Zertifikats mittels TUC_PKI_002 "Gültigkeitsprüfung des Zertifikats" mit Referenzzeitpunkt = aktuelle Systemzeit b) Auswahl des öffentlichen Schlüssels des CRL-Signer-Zertifikats (CRL-Signer-Zertifikat im Truststore) c) Die Signatur und der verwendete Algorithmus werden aus der CRL ausgelesen 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 {"CERT/CRL/INVALID"; Op; Error; ""; doLog; noDisp} aus. 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. 3. Nach einer erfolgreichen Prüfung speichert der Konnektor die neue CRL und löst das Ereignis TUC_KON_256{„CERT/CRL/UPDATED“; Op; Info; „“; doLog; noDisp} aus.
Varianten/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 {„CERT/CRL/IMPORT“; Op; Error; „${Fehlerbeschreibung}“; noDisp} ausgelöst. (→2) Signaturprüfung der CRL fehlgeschlagen: Fehlercode 4130
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 216: TAB_KON_599 Übersicht Fehlercodes für „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

⌫

4.1.9.3.3 TUC_KON_033 „Zertifikatsablauf prüfen“

TIP1-A_4695 TUC_KON_033 „Zertifikatsablauf prüfen“

Der Konnektor MUSS den technischen Use Case TUC_KON_033 “Zertifikatsablauf prüfen” umsetzen.

Tabelle 217: 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 Zertifikats einer Karte.
Auslöser	• Aufruf durch andere TUCs des Konnektors oder • über die Managementschnittstelle
Vorbedingungen	Keine
Eingangsdaten	• doInformClients • CardSession oder checkSMCK
Komponenten	Konnektor
Ausgangsdaten	• Ablaufdatum
Standardablauf	1. TUC_KON_216 „LeseZertifikat“ für: • Bei checkSMCK das Zertifikat der gSMC-K (C.NK.VPN) • bei CardSession die Zertifikate der identifizierten Karte. i. Für die eGK: C.CH.AUT ii. Für den HBAx: C.HP.AUT iii. Für SM-B: C.HCI.AUT 2. Das Ablaufdatum wird aus dem Feld validity ausgelesen. 3. Falls das Zertifikat abgelaufen ist, Systemereignis absetzen: • gSMC-K: TUC_KON_256 {„CERT/CARD/EXPIRATION”; Op; Warning; „CARD_TYPE=gSMC-K; ICCSN=$ICCSN; Konnektor=$MGM_KONN_HOSTNAME, ExpirationDate=$validity “; doLog; $doInformClients} • Sonstige Karten (mit CARD(CardSession)): TUC_KON_256 {„CERT/CARD/EXPIRATION”; Op; Warning; „CARD_TYPE=$Type; ICCSN=$ICCSN; CARD_HANDLE=$CardHandle; CardHolderName=$CardHolderName; ExpirationDate=$validity“; noLog; $doInformClients} 4. Alternativ bei Ablauf des Zertifikats innerhalb von CERT_EXPIRATION_WARN_DAYS Systemereignis absetzen: • gSMC-K: TUC_KON_256 {„CERT/CARD/EXPIRATION”; Op; Info; „CARD_TYPE=gSMC-K; ICCSN=$ICCSN; Konnektor=$MGM_KONN_HOSTNAME;

Element	Beschreibung
	ExpirationDate=$validity; DAYS_LEFT=$validity-$Today“; noLog; $doInformClients} • Sonstige mit CARD(CardSession)): TUC_KON_256 {„CERT/CARD/EXPIRATION”; Op; Info; „CARD_TYPE=$Type; ICCSN=$ICCSN; CARD_HANDLE=$CardHandle; CardHolderName=$CardHolderName; ExpirationDate=$validity; DAYS_LEFT=$validity-$Today“; noLog; $doInformClients} 5. Das Ablaufdatum wird zurückgegeben.
Varianten/Alternativen	Keine
Fehlerfälle	(→1) Zur angegebenen CardSession keine Karte gefunden: Fehlercode 4131. (→4) Extraktion des Ablaufsdatums fehlgeschlagen: Fehlercode 4132.
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 218: TAB_KON_600 Übersicht Fehlercodes für „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

⌫

4.1.9.4 Interne TUCs, auch durch Fachmodule nutzbar

4.1.9.4.1 TUC_KON_037 „Zertifikat prüfen"

⌦ **TIP1-A_4696 TUC_KON_037 „Zertifikat prüfen“**

Der Konnektor MUSS den technischen Use Case „Zertifikat prüfen“ gemäß TUC_KON_037 „Zertifikat prüfen“ umsetzen.⌫

Tabelle 219: TAB_KON_769 TUC_KON_037 „Zertifikat prüfen“

Element	Beschreibung
Name	TUC_KON_037 „Zertifikat prüfen“
Beschreibung	Der TUC beschreibt • die Prüfung der Gültigkeit eines CV-Zertifikates, indem die Korrektheit der Signatur des CV- und des CVC-CA-Zertifikates

Element	Beschreibung
	geprüft wird. • die Prüfung eines X.509- Zertifikats gegen den Vertrauensraum
Auslöser	• Aufruf in einem Fachmodul oder • technischen Use Case
Vorbedingungen	• TSL-Informationen vorhanden
Eingangsdaten	• CV-Zertifikat und der öffentliche Schlüssel der zugehörigen ausstellenden CVC-CA oder X.509-Zertifikat • Bei X.509-Prüfung: o QUALIFIED={not_required|required| if_QC_present} o Referenzzeitpunkt: Zeitpunkt, für den das Zertifikat geprüft werden soll (optional; bei Nichtangabe Verwendung der Systemzeit des Konnektors) o OFFLINE_ALLOW_NOCHECK. (true/false; Default: false) o PolicyList: Zugelassene Zertifikatstyp-OID gemäß [gemSpec_OID#GS-A_4445] o Nur für nonQES-Zertifikate: o Vorgesehene KeyUsage (intendedKeyUsage) o Vorgesehene ExtendedKeyUsage (intendedExtendedKeyUsage) o Grace Period: maximal zulässiger Zeitraum, den letzte OCSP-Antwort aus dem Cache bezüglich des Referenzzeitpunkts zurückliegen darf (optional; Default-Wert CERT_OCSP_DEFAULT_GRACE_PERIOD_NONQES o Prüfmodus: ▪ OCSP: Es wird mittels OCSP geprüft. Dabei wird, falls die Grace Period noch nicht abgelaufen ist, die OCSP-Antworten aus dem Cache des Konnektors verwendet. ▪ CRL: Es wird gegen die aktuelle CRL auf dem Konnektor geprüft. ▪ NONE: Keine Prüfung von Statusinformationen o OCSP-Response (nonQES) / Liste von OCSP-Responses (QES) o getOCSPResponses [Boolean] - *optional; Default: false* (liefert die Information, ob eine Liste von OCSP-Responses (QES) oder einzelne OCSP-Antwort (nonQES) des geprüften Zertifikats an den Aufrufer zurückzugegeben ist) o Liste von Attributzertifikaten (optional, QES)
Komponenten	Konnektor
Ausgangsdaten	• Status und Liste von Warnungen/Fehlern bei der Zertifikatsprüfung • Ermittelte Rolle (X.509-Zertifikate) Werte ermittelt aus dem Zertifikat aus „Tab_PKI_406 OID-Festlegung technische Rolle in X.509-Zertifikaten“ oder „Tab_PKI_402 OID-Festlegung Rolle im X.509-Zertifikat für

Element	Beschreibung
	Berufsgruppen“ [gemSpec_OID] • QCStatements des Zertifikats • ocspResponsesRenewed – *optional / verpflichtend, wenn Eingabeparameter getOCSPResponses = true* (eine Liste von OCSP-Responses (QES) oder einzelne OCSP-Antwort (nonQES) des geprüften Zertifikats)
Standardablauf	1. Abhängig davon, ob ein CV-Zertifikat oder ein X.509-Zertifikat geprüft werden soll, werden folgende Schritte ausgeführt: A. CV-Zertifikat: CV-Zertifikate sind entsprechend [gemSpec_PKI#8.7 “Zertifikatsprüfung CVC“] und [gemSpec_PKI#8.8 “Zertifikatsprüfung CV-Zertifikate der 2. Generation“] zu prüfen. i. Die Signatur des CVC-CA-Zertifikats (von der zu prüfenden Karte) wird mit dem öffentlichen Root-Schlüssel der ausstellenden CVC-Root-CA geprüft. Der benötigte Root-Key für G2 befindet sich im Truststore des Konnektors und für G1+ auf der gSMC-K in der Datei EF.PuK.RCA.CS.R2048. ii. Die Signatur des CV-Zertifikats wird mit dem öffentlichen Schlüssel der ausstellenden CVC-CA (per Message Recovery in Schritt i aus dem CVC-CA-Zertifikat gewonnen) geprüft. B. X.509-Zertifikat: B.1 Wenn das X.509-Zertifikat von einem CA-Zertifikat in CERT_IMPORTED_CA_LIST ausgestellt wurde, erfolgt eine Zertifikatsprüfung analog zu den Festlegungen in TUC_PKI_018 “Zertifikatsprüfung”. Dabei sind zu prüfen: Zeitliche Gültigkeit, mathematische Prüfung der Zertifikatssignatur, die Prüfung der Zweckbindung gemäß der im Zertifikat hinterlegten keyUsage. TSL-bezogene Prüfungen im TUC_PKI_018 werden in diesem Fall nicht durchgeführt. Ebenso erfolgt keine OCSP-Prüfung. B.2 Wenn das zum X.509-Zertifikat gehörende CA-Zertifikat nicht in CERT_IMPORTED_CA_LIST enthalten ist, werden, abhängig vom Parameter QUALIFIED, folgende TUCs unter Weitergabe aller Eingangsparameter sowie der Negation des Werts von MGM_LU_ONLINE als Parameter „Offline-Modus“ aufgerufen: a) Für QUALIFIED=not_required: TUC_PKI_018 “Zertifikatsprüfung in der TI” Ist der Eingangsparameter ocspResponses mit einer OCSP-Antwort gefüllt, so wird dieser übergeben.

Element	Beschreibung
	Die aktuell aus der OCSP-Abfrage resultierte OCSP-Antwort, falls vorhanden, wird an den Aufrufer weitergegeben. b) Für QUALIFIED=required: TUC_PKI_030 „QES-Zertifikatsprüfung“ Dabei werden das Basiszertifikat und die Attributzertifikate übergeben. Ist Eingangsparamter ocspResponses mit einer Liste von OCSP-Responses gefüllt, so wird dieser übergeben. Die aktuell aus der OCSP-Abfrage resultierten OCSP-Responses, falls vorhanden, werden an den Aufrufer weitergegeben. c) Für QUALIFIED=if_QC_present: 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-Zertifikatsprüfung mittels TUC_PKI_030 „QES-Zertifikatsprüfung“, sonst um eine nonQES-Zertifikatsprüfung mittels TUC_PKI_018 „Zertifikatsprüfung“ Als Timeout wird CERT_OCSP_TIMEOUT_NONQES bzw. bei QUALIFIED CERT_OCSP_TIMEOUT_QES verwandt. Als TOLERATE_OCSP_FAILURE wird OFFLINE_ALLOW_NOCHECK verwandt. Für die QES-Zertifikatsprüfung wird das zu prüfende QES-Zertifikat zur Bildung der Zertifikatsliste (tbvCerts – die Liste der Zertifikate, die die Kette zur BNetzA-Root bilden) und zur Validierung der Zertifizierungspfades (tbvPath) an TUC_PKI_030 „QES-Zertifikatsprüfung“ übergeben. Das BNetzA-Root-Zertifikat befindet sich auf der gSMC-K (in der Datei EF.C.BNetzA.RCA). Sämtliche Cross-Zertifikate der BNetzA und QES-CA-Zertifikate sind im TrustStore vorhanden. Wird im Aufruf der Eingangsparameter getOCSPResponses=false mit übergeben, werden keine OCSP-Responses an den Aufrufer zurückgegeben. Wenn der Eingangsparameter „Prüfmodus“ den Wert NONE hat, werden die TUC_PKI_018 Eingangsparameter • „Offline-Modus“ unabhängig von MGM_LU_ONLINE auf „ja“ gesetzt und • „Prüfmodus“ auf „OCSP“. 2. Der Status der Prüfung wird zurückgegeben.
Varianten/Alternativen	
Fehlerfälle	(→1.A.) Fehler bei der CV-Zertifikatsprüfung: Fehlercode 4196.
Nichtfunktionale	Der Konnektor MUSS unter Einhaltung aller anderen Anforderungen

Element	Beschreibung
Anforderungen	an die Zertifikatsprüfung die Anzahl der OCSP-Abfragen minimieren. Dies MUSS durch Caching (unter Berücksichtigung der Grace Period) und DARF NICHT durch Bündelung von OCSP-Anfragen geschehen.
Zugehörige Diagramme	keine

Tabelle 220: TAB_KON_601 Übersicht Fehlercodes für das Prüfen eines Zertifikats

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

4.1.9.4.2 TUC_KON_034 „Zertifikatsinformationen extrahieren“

⌦ **TIP1-A_4697 TUC_KON_034 “Zertifikatsinformationen extrahieren”**

Der Konnektor MUSS den technischen Use Case TUC_KON_034 “Zertifikatsinformationen extrahieren” umsetzen.

Tabelle 221: 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	• Aufruf durch ein Fachmodul oder eine Basisanwendung des Konnektors
Vorbedingungen	Keine
Eingangsdaten	• Aufrufkontext (Mandant) • CardHandle oder checkSMCK • QES (true/false; Default: false) – Angabe, ob die QES-Identität oder die nonQES-Identität der Karte interessiert
Komponenten	Konnektor
Ausgangsdaten	• Zertifikatstyp • Zertifikatsinformationen (s. Standardablauf) • ggf. QCStatements
Nachbedingungen	Keine
Standardablauf	1. Je nach Kartentyp wird aus der Karte das passende Zertifikat über TUC_KON_216 "LeseZertifikat" {selektiertes Zertifikat} ausgelesen. a. Bei QES=false: i. Für die eGK: C.CH.AUT

Element	Beschreibung
	ii. Für den HBAx: C.HP.AUT iii. Für SM-B: C.HCI.AUT iv. Für gSMC-K: C.NK.VPN b. Bei QES=true: i. Für den HBAx: C.HP.QES 2. Die Zertifikatsbezeichnung aus Schritt 1 („C.XXX.YYY“) wird als Ausgangsdatum „Zertifikatstyp“ zurückgegeben. 3. Zusätzlich werden aus dem Zertifikat folgende Informationen extrahiert und zurückgegeben: c. X509SerialNumber d. Issuer (Distinguished Name) nach RFC 2253 e. Subject (CommonName) nach RFC 2253 f. Aus der Extension Admission: i. eine Liste von Rollen durch Aufruf von TUC_PKI_009 “Rollenermittlung” ii. registrationNumber (=Telematik-ID; falls vorhanden) g. id-etsi-qcs-QcCompliance (0.4.0.1862.1.1) in QCStatements, falls vorhanden h. Restriction, falls vorhanden i. validity
Varianten/Alternativen	Keine
Fehlerfälle	(→1) Wenn im Aufrufkontext (also erreichbar durch den Mandanten) zum angegebenen CardHandle keine Karte gefunden werden kann, bricht der TUC mit Fehlercode 4146 ab. (→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). (→1) Wenn aus anderen Gründen die Extraktion der Zertifikatsinformationen fehlschlägt, bricht der TUC mit Fehlercode 4148 ab.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 222: TAB_KON_602 Übersicht Fehlercodes für „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

⌫

4.1.9.5 Operationen an der Außenschnittstelle

⌦ **TIP1-A_4698 Basisanwendung Zertifikatsdienst**

Der Konnektor MUSS Clientsystemen eine Basisanwendung Zertifikatsdienst zur Verfügung stellen

Tabelle 223: TAB_KON_771 Basisanwendung Zertifikatsdienst

Name	`CertificateService`	
Version (KDV)	wird im Produktsteckbrief des Konnektors definiert	
Namensraum	Siehe Anhand D	
Namensraum-Kürzel	`CERT für Schema und CERTW für WSDL`	
Operationen	Name	Kurzbeschreibung
	`ReadCardCertificate`	Zertifikat von einer Karte lesen
	`CheckCertificateExpiration`	Ablaufdatum von Zertifikaten erfragen
	`VerifyCertificate`	Prüfung des Status eines Zertifikats
WSDL	`CertificateService.wsdl`	
Schema	`CertificateService.xsd`	

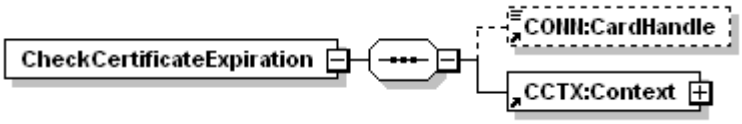
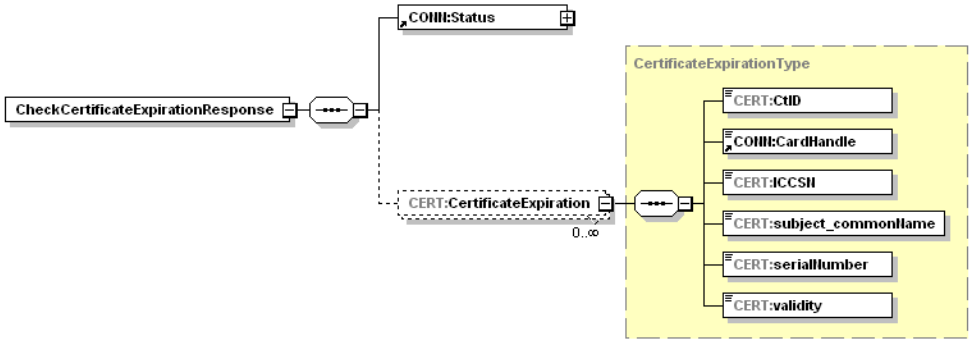
⌫

4.1.9.5.1 CheckCertificateExpiration

⌦ **TIP1-A_4699 Operation CheckCertificateExpiration**

Die Basisanwendung Zertifikatsdienst des Konnektors MUSS an der Clientschnittstelle eine Operation CheckCertificateExpiration anbieten.

Tabelle 224: TAB_KON_676 Operation CheckCertificateExpiration

Name	`CheckCertificateExpiration`	
Beschreibung	Gibt das Datum des Ablaufs eines bestimmten Zertifikats oder gesammelt des Zertifikats der gSMC-K sowie aller gesteckten HBAx und SM-B des Mandanten zurück.	
Aufrufparameter		
	Name	**Beschreibung**
	`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), die zum Mandanten passen, berücksichtigt.
	`Context`	MandantId, CsId, WorkplaceId verpflichtend; UserId optional
Rückgabe		
	`Status`	Enthält den Ausführungsstatus der Operation.
	`CertificateExpiration`	Eine Liste von Tupeln aus (CtID, CardHandle, ICCSN, subject.CommonName, serialNumber, validity) der Zertifikate der Karten.
Vorbedingungen	Keine	
Nachbedingungen	Keine	

Der Ablauf der Operation CheckCertificateExpiration ist in Tabelle 225 beschrieben:

Tabelle 225: 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

Nr.	Aufruf Technischer Use Case oder Interne Operation	Beschreibung
		ab.
2.	TUC_KON_000 „Prüfe Zugriffs-berechtigung“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; $context.userId; $cardHandle} 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. Wenn der Parameter CardHandle leer war, wird eine Liste der CardHandles aller für den Konnektor erreichbaren Karten (inkl. gSMC-K), die zum Mandanten passen, erstellt.
Für jedes CardHandle der in Schritt 3 erzeugten Liste werden folgende Schritte ausgeführt:		
4.	TUC_KON_026 „Liefere CardSession“	Ermittle CardSession über TUC_KON_026 { MandantId, CsId, CardHandle, UserId }
5.	TUC_KON_033 „Zertifikatsablauf prüfen“	Das Gültigkeitsdatum des Zertifikats wird geprüft.
6.	TUC_KON_034 “Zertifikatsinformationen extrahieren”	Beim Aufruf des TUC_KON_034 ist der Parameter QES = false zu setzen. Aus den jeweiligen Rückgabewerten entsteht eine Liste aus Tupeln (CtID, CardHandle, ICCSN, subject.CommonName, serialNumber, validity). Diese wird von der Operation zurückgegeben.

Tabelle 226 TAB_KON_603 Fehlerfälle 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

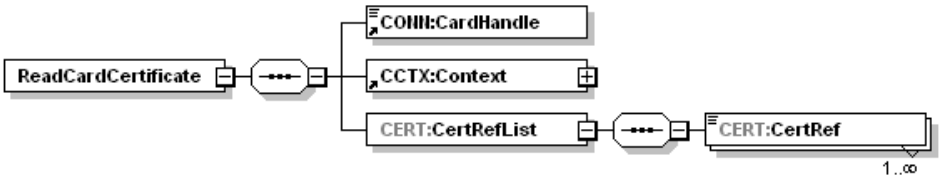
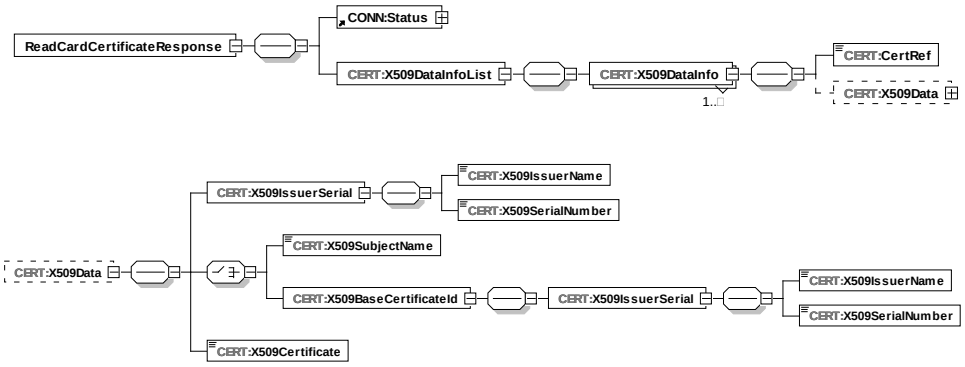
⌫

4.1.9.5.2 ReadCardCertificate

⌦ **TIP1-A_4700 Operation ReadCardCertificate**

Die Basisanwendung Zertifikatsdienst des Konnektors MUSS an der Client-schnittstelle eine Operation ReadCardCertificate wie in Tabelle 227: TAB_KON_678 Operation ReadCardCertificate beschrieben anbieten**.**

Tabelle 227: TAB_KON_678 Operation ReadCardCertificate

Name	`ReadCardCertificate`		
Beschreibung	Liest X.509-Zertifikate von einer Karte.		
Aufrufparameter			
	Name	**Beschreibung**	
	`CardHandle`	Gibt die Karte an, von der das Zertifikat gelesen werden soll. Es können Zertifikate von HBAx (HBA, HBA-VK), SM-B ausgelesen werden. Die Operation ReadCardCertificate DARF das Lesen von Zertifikaten der eGK NICHT unterstützen.	
	`Context`	Aufrufkontext (Mandant)	
	`CertRefList`	Gibt an, welche(s) Zertifikat(e) gelesen werden soll. Mögliche Werte für CertRef sind: `C.AUT, C.ENC, C.SIG, C.QES, C.QES-AC1, C.QES-AC2, C.QES-AC3`	
Rückgabe			
	`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.	
		`X509IssuerName`	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.)
		`X509SerialNumber`	Enthält die serialNumber des Zertifikats.
		`X509SubjectName`	Enthält das Feld subject.CommonName eines Public-Key-Zertifikats bzw. das Feld

			holder eines Attributzertifikats. Bezüglich des Encodings sind die in [XML DSig#4.4.4.4.1] angegebenen Regeln zu beachten (Escaping von Sonderzeichen etc.).
		`X509BaseCertificateId`	Enthält im Falle eines Attributzertifikats Issuer-Name und Seriennummer des Basiszertifikates, falls vorhanden.
		`X509Certificate`	Enthält das base64-codierte Zertifikat, dessen Binärstruktur wiederum ASN.1-codiert (gemäß [COMMON_PKI]) vorliegt.
Vorbedingungen	Keine		
Nachbedingungen	Keine		

Der Ablauf der Operation ReadCardCertificate ist in Tabelle 228: TAB_KON_679 Ablauf ReadCardCertificate beschrieben:

Tabelle 228: TAB_KON_679 Ablauf ReadCardCertificate

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. Wurde als Zielkarte eine eGK adressiert, wird Fehlercode 4090 zurückgeliefert.
2.	TUC_KON_000 „Prüfe Zugriffs-berechtigung“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; $context.userId; $cardHandle} 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 { MandantId, CsId, CardHandle, UserId }
4.	getEF	Für jedes Paar von CertRef und CardHandle wird gemäß folgender Tabelle das zu lesende File (EF) bestimmt: <u>Referenz</u> <u>EF</u> `C.AUT` HBA-VK: EF.C.HP.AUT HBA: EF.C.HP.AUT.R2048 SM-B: EF.C.HCI.AUT `C.ENC` HBA-VK: EF.C.HP.ENC HBA: EF.C.HP.ENC1.2048 SM-B: EF.C.HCI.ENC1.R2048

Nr.	Aufruf Technischer Use Case oder Interne Operation	Beschreibung
		`C.SIG` SM-B: EF.C.HCI.OSIG.R2048 `C.QES` HBA-VK: EF.C.HP.QES HBA: EF.C.HP.QES.R2048 `C.QES-AC1` HBAx: EF.C.HP.QES-AC1 `C.QES-AC2` HBAx: EF.C.HP.QES-AC2 `C.QES-AC3` HBAx: EF.C.HP.QES-AC3 Ist die übergebene Zertifikatsreferenz ungültig, wird Fehlercode 4149 zurückgegeben. Das Lesen von Zertifikaten der eGK ist aus Sicherheitsgründen für Clientsysteme nicht zulässig.
5.	TUC_KON_216 „LeseZertifikat“	Für jedes Paar von CardHandle und EF wird nun durch Aufruf von TUC_KON_216 „LeseZertifikat“ das Zertifikat ausgelesen.
6.	Zertifikatsattribute extrahieren	Aus jedem Zertifikat werden die zu liefernden Attribute extrahiert. Die Ergebnisstruktur wird mit den erhaltenen Rückgabewerten gefüllt.

Tabelle 229 TAB_KON_604 Fehlerfälle 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

⌫

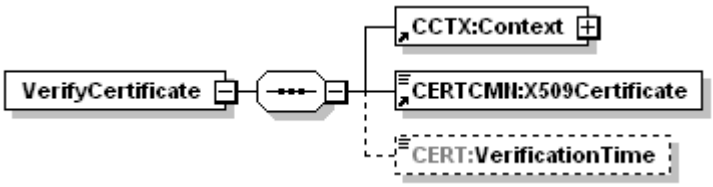
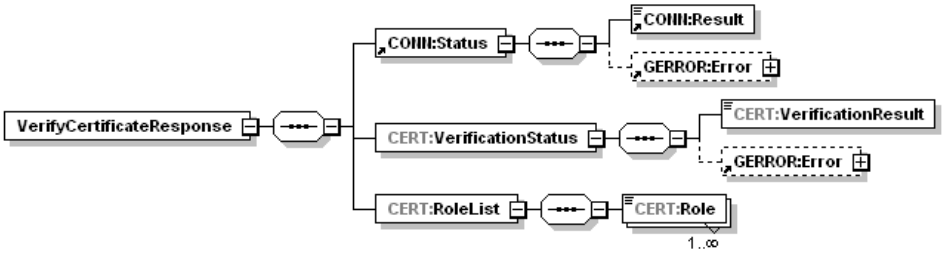
4.1.9.5.3 VerifyCertificate

⌦ **TIP1-A_5449 Operation VerifyCertificate**

Die Basisanwendung Zertifikatsdienst des Konnektors MUSS an der Client-schnittstelle eine Operation VerifyCertificate wie in Tabelle 230: TAB_KON_795 Operation VerifyCertificate beschrieben anbieten**.**

Tabelle 230: TAB_KON_795 Operation VerifyCertificate

Name	`VerifyCertificate`
Beschreibung	Prüft den Status eines Zertifikats.

Aufrufparameter		
	Name	**Beschreibung**
	`CCTX:Context`	Aufrufkontext (Mandant)
	`CERTCMN: X509Certificate`	Zu prüfendes Zertifikat (base64 kodiert), wie in Response zur Operation ReadCardCertificate enthalten.
	`CERT: VerificationTime`	Der für die Prüfung zu verwendende Referenzzeitpunkt. Falls der Parameter nicht angegeben ist, wird als Referenzzeitpunkt die Systemzeit verwendet.
Rückgabe		
	`CONN:Status`	Enthält den Ausführungsstatus der Operation.
	`CERT: VerificationStatus`	Enthält eines der drei möglichen Prüfungsergebnisse in CERT:VerificationResult • VALID • INCONCLUSIVE • INVALID sowie weiter Details zu den Zuständen „INCONCLUSIVE“ und „INVALID“ in GERROR:Error.
	`CERT:RoleList`	OIDs der im Zertifikat gespeicherten Rollen.
Vorbedingungen	Keine	
Nachbedingungen	Keine	

Der Ablauf der Operation VerifyCertificate ist in Tabelle 231: TAB_KON_797 Ablauf VerifyCertificate beschrieben:

Tabelle 231: TAB_KON_797 Ablauf VerifyCertificate

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

Nr.	Aufruf Technischer Use Case oder Interne Operation	Beschreibung
		ab.
2.	TUC_KON_000 „Prüfe Zugriffs-berechtigung“	Die Prüfung erfolgt durch den Aufruf TUC_KON_000 {$context.mandantId; $context.clientsystemId; $context.workplaceId; ; } Tritt bei der Prüfung ein Fehler auf, bricht die Operation mit Fehlercode aus TUC_KON_000 ab.

3.	TUC_KON_037 „Zertifikat prüfen“	Die Zertifikatsprüfung erfolgt durch Aufruf von TUC_KON_037. Als Parameter des TUC-Aufrufs gilt für Zertifikate aus CERT_IMPORTED_CA_LIST: {
		X.509-Zertifikat = `CERTCMN:X509Certificate`
		QUALIFIED= not_required;
		Referenzzeitpunkt= `CERT:VerificationTime`;
		OFFLINE_ALLOW_NOCHECK=true;
		PolicyList= keine Einschränkung;
		intendedKeyUsage=empty;
		intendedExtendedKeyUsage=empty;
		Grace Period = empty;
		Prüfmodus = NONE;
		OCSP-Response / Liste von OCSP-Responses = empty}
		
		für alle anderen Zertifikate gilt: {
		X.509-Zertifikat: `CERTCMN:X509Certificate`
		QUALIFIED=if_QC_present;
		Referenzzeitpunkt= `CERT:VerificationTime`;
		OFFLINE_ALLOW_NOCHECK=true;
		PolicyList= alle zugelassenen Zertifikatstyp-OIDs;
		intendedKeyUsage=empty;
		intendedExtendedKeyUsage=empty;
		Grace Period = empty;
		Prüfmodus = OCSP;
		OCSP-Response / Liste von OCSP-Responses = empty}.

4.		Wenn der Prüfprozess fehlerhaft war und nicht zu einem Ergebnis im Sinne eines VerificationResult führt, wird eine FaultMessage erzeugt. War der Prüfprozess erfolgreich, wird eine VerifyCertificateResponse mit einem VerificationStatus und den ermittelten Rollen-OIDs erzeugt. Fehlermeldungen im Rahmen des Prüfprozesses werden dabei in CONN:Status/GERROR:Error abgelegt. Ein Prüfergebnis „INCONCLUSIVE“ bzw. „INVALID“ wird in CERT:VerificationStatus/GERROR:Error mit den zugehörigen Fehlermeldungen detailliert.

Tabelle 232 TAB_KON_800 Fehlerfälle VerifyCertificate

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			

Fehlercode	ErrorType	Severity	Fehlertext
4000	Technical	Error	Syntaxfehler
4058	Security	Error	Aufruf nicht zulässig

⌫

4.1.9.6 Betriebsaspekte

4.1.9.6.1 TUC_KON_035 „Zertifikatsdienst initialisieren“

⌦ **TIP1-A_4701 TUC_KON_035 „Zertifikatsdienst initialisieren“**

In der Bootup-Phase MUSS der Konnektor den Zertifikatsdienst durch Aufruf des TUC_KON_035 „Zertifikatsdienst initialisieren“ initialisieren.

Tabelle 233: 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	• Bootup des Konnektors
Vorbedingungen	keine
Eingangsdaten	keine
Komponenten	Konnektor
Ausgangsdaten	• Status der Initialisierung des TrustStore
Nachbedingungen	Keine
Standardablauf	Für den übergebenen Status der Initialisierung des TrustStore werden folgende Schritte durchgeführt: 1. Durch eine DNS-Anfrage an den DNS-Forwarder zur Auflösung der SRV-RR mit dem Bezeichner "_ocsp._tcp.<DOMAIN_SRVZONE_TI>“ erhält der Konnektor Adressen des http-Forwarders des VPN-Zugangsdienststandortes. Durch DNS-Anfragen an den DNS-Forwarder zur Auflösung der SRV-RR und TXT-RR mit dem Bezeichner "_tsl._tcp.tsl.<TOP_LEVEL_DOMAIN_TI>“ erhält der Konnektor die URLs des TSL-Downloadpunktes der TI-Plattform. 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“. 3. Der Konnektor prüft die Gültigkeitsdauer der Zertifikate aller gesteckten Karten (inkl. gSMC-K) mittels Aufruf von TUC_KON_033{doInformClients=Ja}.

Element	Beschreibung
	4. 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/Alternativen	Keine
Fehlerfälle	Keine
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 234: TAB_KON_605 Übersicht Fehlercodes für „Zertifikatsdienst initialisieren“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können keine weiteren Fehlercodes auftreten.			

⌫

⌦ TIP1-A_4702 Konfigurierbarkeit des Zertifikatsdienstes

Der Administrator MUSS die in TAB_KON_606 aufgelisteten Parameter über die Managementschnittstelle konfigurieren und die in TAB_KON_733 aufgelisteten Parameter ausschließlich einsehen können.

Tabelle 235: TAB_KON_606 Konfiguration des Zertifikatsdienstes

ReferenzID	Belegung	Bedeutung
`CERT_CRL_DOWNLOAD_ADDRESS`	`2 URIs`	Download-Adressen für die CRL
`CERT_TSL_DEFAULT_GRACE_PERIOD_DAYS`	`X Tage`	Default Grace Period TSL in Tagen Gibt an, wie viele Tage der Konnektor mit einer zeitlich abgelaufenen TSL weiter betrieben werden kann. Der Wert MUSS zwischen 1 und 30 Tagen liegen. Default-Wert = 30 Tage *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“).*
`CERT_OCSP_DEFAULT_GRACE_PERIOD_NONQES`	`X Minuten`	Default Grace Period OCSP für nonQES in Minuten. Der Wert MUSS zwischen 0 und 20 Minuten liegen. Default-Wert = 10 Minuten
`CERT_OCSP_TIMEOUT_NONQES`	`X Sekunden`	Timeout für OCSP-Abfragen bei der Prüfung von nonQES-Zertifikaten.

ReferenzID	Belegung	Bedeutung
		Der Wert MUSS zwischen 1 und 120 Sekunden liegen. Default-Wert = 30 Sekunden
CERT_OCSP_TIMEOUT_QES	X Sekunden	Timeout für OCSP-Abfragen bei der Prüfung von QES-Zertifikaten. Der Wert muss zwischen 1 und 120 Sekunden liegen. Default-Wert = 30 Sekunden
CERT_EXPIRATION_WARN_DAYS	X Tag(e)	Warnung X Tage vor Ablauf von Zertifikaten im Managementinterface und per Ereignis. Der Wert muss zwischen 0 und 180 Tagen (0=keine Warnung) liegen. Default-Wert = 90 Tage
CERT_EXPIRATION_CARD_CHECK_DAYS	X Tag(e)	Alle X Tage wird der Ablauf aller gesteckten Karten überprüft. Der Wert muss zwischen 0 und 365 liegen (0=kein Check). Default-Wert = 1 Tag
CERT_IMPORTED_CA_LIST	Liste von manuell importierten CA-Zertifikaten	Der Administrator MUSS CA-Zertifikate importieren, anzeigen und löschen können. Der Konnektor DARF CA-Zertifikate zur Ableitung von QES-Zertifikaten NICHT importieren. Default-Wert = leere Liste

Tabelle 236 TAB_KON_733 Einsehbare Konfigurationsparameter des Zertifikatsdienstes

ReferenzID	Belegung	Bedeutung
CERT_OCSP_FORWARDER_ADDRESS	2 FQDNs	Adressen der OCSP-Forwarder (HTTPS-Proxy) beim Zugangsdienstprovider 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

⌫

⌦ **TIP1-A_4703 Vertrauensraumstatus anzeigen**

Das Managementinterface des Konnektors MUSS einem authentisierten Administrator die Anzeige des Status des Vertrauensraums in Form folgender Daten anbieten: Sequenznummer, StatusStartingTime, NextUpdate, Gültigkeit der TSL sowie optional für den Administrator einsehbar der Fingerprint des TSL-Signer-Zertifikats.⌫

⌦ **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) mittels TUC_KON_033{doInformClients=Nein} an zentraler Stelle in der Managementschnittstelle auslösen können und das Ergebnis angezeigt bekommen.⌫

⌦ 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.⌫

⌦ TIP1-A_4706 CRL manuell importieren

Der Konnektor SOLL es dem Administrator ermöglichen, eine CRL manuell von einer lokalen Datenquelle einzuspielen. In dem Fall MUSS der Konnektor TUC_KON_040{CRL-Datei} mit der manuell importierten CRL aufrufen.⌫

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 Fachmodulen werden in das Fachmodul-Performanceprotokoll geschrieben. Alle anderen Ereignisse werden in das Systemprotokoll oder die Fachmodulprotokolle geschrieben (grundsätzlich trifft die Entscheidung über den zu verwendenden Protokollspeicher der Aufrufer des Protokolldienstes).

Die Protokolle werden persistiert.

Hinweis:
Ereignisse im Protokollierungsdienst adressieren nicht nur zu protokollierende Events im Sinne des Systeminformationsdienstes sondern alles, was zu einem Protokolleintrag führen soll (z.B. Fehler, Informationen zu Ablauf, Debug, Performance).

Innerhalb des Protokollierungsdienstes werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): „LOG“
- Konfigurationsparameter: „LOG_“

4.1.10.1 Funktionsmerkmalweite Aspekte

⌦ TIP1-A_4708 Protokollierungsfunktion

Der Konnektor MUSS einen Protokollierungsdienst anbieten. Dabei MUSS der Konnektor zwischen System- und Sicherheitsprotokoll, sowie Fachmodulprotokollen unterscheiden. Je Fachmodul MUSS ein getrenntes Protokoll vorhanden sein.

Die Protokolleinträge MÜSSEN durch den Konnektor lokal persistiert werden.⌫

⌦ **TIP1-A_5654 Sicherheits-Protokollierung**

Der Konnektor MUSS herstellerspezifische Fehler, die Auswirkungen auf Sicherheitsmerkmale des Konnektors haben, in das Sicherheitsprotokoll schreiben.⌫

⌦ **TIP1-A_4709 Integrität des Sicherheitsprotokolls**

Der Konnektor MUSS sicherstellen, dass Einträge in das Sicherheitsprotokoll nicht von außen und nicht durch den Administrator verändert und gelöscht werden können. ⌫

⌦ **TIP1-A_4710 Protokollierung personenbezogener und medizinischer Daten**

Der Konnektor DARF medizinische Daten NICHT in die Protokolldateien schreiben.

Personenbezogene Daten DÜRFEN NICHT in Protokolleinträgen gespeichert werden, es sei denn, sie sind zur Analyse von (Sicherheits-)Vorfällen erforderlich. ⌫

⌦ **TIP1-A_4711 Kapazität der Protokolldateien**

Der Konnektor MUSS über eine Speichergröße für Protokolldateien verfügen, so dass Einträge (protokollierte Ereignisse ab der Schwere „Warning“) über einen Zeitraum von bis zu einem Jahr darin vorgehalten werden können. ⌫

Da sich die Menge an Einträgen nach der Größe der LE-Umgebung richtet, ist die Speichergröße nach den in [gemSpec_Perf#3.1.1] beschriebenen LE-Umgebungen ausreichend zu wählen.

⌦ **TIP1-A_4712 Protokollierung erfolgreicher Kryptooperationen**

Wenn `LOG_SUCCESSFUL_CRYPTO_OPS` = Enabled MUSS der Konnektor alle erfolgreich durchlaufenen Außenoperationen protokollieren, in denen mit Schlüsselmaterial der gesteckten Karten kryptographische Operationen durchgeführt wurden. Dazu MUSS er TUC_KON_256{"LOG/CRYPTO_OP"; Sec; Info; „Operation=$Operationsname;<für alle betroffenen Schlüssel:>Karte=$ICCSN;Keyref=<Referenz auf den Schlüssel>; CARD_HANDLE=$CardHandle; CardHolderName=$CardHolderName“; noDisp} 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.⌫

4.1.10.2 Durch Ereignisse ausgelöste Reaktionen

Keine.

4.1.10.3 Interne TUCs, nicht durch Fachmodule nutzbar

Keine.

4.1.10.4 Interne TUCs, auch durch Fachmodule nutzbar

4.1.10.4.1 TUC_KON_271 „Schreibe Protokolleintrag“

TIP1-A_4715 TUC_KON_271 “Schreibe Protokolleintrag”

Der Konnektor MUSS den technischen Use Case TUC_KON_271 “Schreibe Protokolleintrag” umsetzen.

Tabelle 237: 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 „Systemereignis absetzen“
Vorbedingungen	Keine
Eingangsanforderung	Keine
Eingangsdaten	• Zu protokollierendes Ereignis o fmName – *optional / verpflichtend für Aufruf durch Fachmodule* (Name des aufrufenden Fachmoduls; Default: „“: das Ereignis wird in das entsprechende Konnektor-Protokoll geschrieben) o Typ (Sec, Op, Perf) definiert den Protokolltyp, in welchen das Ereignis geschrieben wird; Sec = Security: Ereignis wird in das Securityprotokoll geschrieben Op = Operation: Wenn fmName =““ wird das Ereignis in das Systemprotokoll geschrieben. Wenn fmName gesetzt ist, wird das Ereignis in das durch fmName definierte Fachmodul-Protokoll geschrieben. Perf = Performance: Wenn fmName =““ wird das Ereignis in das Konnektor-Performanceprotokoll geschrieben. Wenn fmName gesetzt ist, wird das Ereignis in das durch fmName definierte Fachmodul-Performanceprotokoll geschrieben. o Schwere (Debug = Debug Information, Info = Information, Warn = Warning, Err = Error, Fatal) o Parameter beinhaltet die Daten des Ereignisses, die im Protokolleintrag geschrieben werden

Element	Beschreibung
	Im Fall eines zu protokollierenden Ereignisses entsprechen die Bezeichnungen Op, Sec, Info, Warning, Err, Fatal den Werten Operation, Security, Info, Warning, Error, Fatal in den jeweiligen Ereignisnachrichten gemäß Schema EventService.xsd.
Komponenten	Konnektor
Ausgangsdaten	Keine
Nachbedingungen	
Standardablauf	1. Ist das Ereignis vom Typ Sec, so wird das Ereignis in das Sicherheitsprotokoll geschrieben. Falls fmName angegeben ist, wird er dem Eintrag hinzugefügt. 2. fmName ist angegeben (durch ein Fachmodul aufgerufen) und Typ= „Op“, so wird das Ereignis in das zugehörige Fachmodulprotokoll geschrieben. a. Gemäß den Festlegungen in den jeweiligen Fachmodulspezifikationen, werden nur Ereignisse in das Fachmodulprotokoll geschrieben, deren Schwere mindestens dem jeweils dort festgelegten Wert entsprechen. 3. fmName ist nicht angegeben (Aufruf durch ein Fachmodul) und Typ = „Op“, dann wird das Ereignis in das Systemprotokoll geschrieben. 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. 4. Wurde der TUC durch ein Fachmodul aufgerufen (fmName ist angegeben) und ist vom Typ Perf, so wird das Ereignis in das zugehörige Fachmodul-Performanceprotokoll geschrieben. 5. Wurde der TUC nicht durch ein Fachmodul aufgerufen (fmName ist nicht angegeben) und ist vom Typ Perf, so wird das Ereignis in das Konnektor-Performanceprotokoll geschrieben. Die geschriebenen Protokolleinträge MÜSSEN mindestens folgende Attribute beinhalten: • Datum und Uhrzeit • Übergebenes Ereignis Die Speicherung erfolgt rollierend. Dabei werden Einträge, die älter als LOG_DAYS bzw. FM_<fmName>_LOG_DAYS sind, überschrieben.
Varianten/Alternativen	• Ist der Speicherplatz eines Protokolls erschöpft, so werden auch Einträge gelöscht, die nicht älter als LOG_DAYS bzw. FM_<fmName>_LOG_DAYS sind. Neue Logeinträge dürfen dabei nur alte Logeinträge gleicher oder niedrigerer Severity überschreiben. Beim Löschen von Logeinträgen, die nicht älter als LOG_DAYS bzw. FM_<fmName>_LOG_DAYS sind, wird der Fehlerzustand EC_LOG_OVERFLOW ausgelöst.
Fehlerfälle	Fehler in den folgenden Schritten des Standardablaufs führen zu: a) Aufruf von TUC_KON_256 mit folgenden Parametern {"LOG/ERROR"; $ErrorType; $Severity; „Error=$Fehlercode; Bedeutung=$Fehlertext“; noLog} b) Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes

Element	Beschreibung
	(→1) In das Sicherheitsprotokoll kann nicht geschrieben werden: Fehlercode: 4152 (→2) In das Fachmodulprotokoll kann nicht geschrieben werden: Fehlercode: 4151 (→3) In das Systemprotokoll kann nicht geschrieben werden: Fehlercode: 4150 (→4) In das Fachmodul-Performanceprotokoll kann nicht geschrieben werden: Fehlercode: 4217 (→5) In das Konnektor-Performanceprotokoll kann nicht geschrieben werden: Fehlercode: 4216
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 238: TAB_KON_608 Übersicht Fehler 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

⌫

4.1.10.5 Operationen an der Außenschnittstelle

Keine

4.1.10.6 Betriebsaspekte

⌦ TIP1-A_4716 Einsichtnahme und Veränderung der Protokolle

Der Administrator MUSS die durch den Protokollierungsdienst geschriebenen Protokolle über die Managementschnittstelle einsehen können.

Eine Veränderung des Sicherheitsprotokolls DARF für den Administrator NICHT möglich sein.

Das Löschen folgender Protokolle MUSS fuïr den Administrator möglich sein:

- Systemprotokoll
  das jeweils durch <fmName> spezifizierte Fachmodulprotokoll

- Konnektor-Performanceprotokoll
  das jeweils durch <fmName> spezifizierte Fachmodul-Performanceprotokoll

Der Konnektor MUSS den Export von Protokolleinträgen oder ganzen Protokolldateien unterstützen.

Der Konnektor SOLL das Sortieren und Filtern der Protokolleinträge sowie das Suchen in den Protokolleinträgen unterstützen.⌫

⌦ **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 239 TAB_KON_609 Konfigurationswerte des Protokollierungsdienstes (Administrator)

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
`LOG_LEVEL_SYSLOG`	`Info, Warning, Error, Fatal`	Der Administrator MUSS den Detaillierungsgrad des Systemprotokolls durch Festlegung der Mindest-Schwere zu protokollierender Einträge festlegen können. Default-Wert: Warning
`LOG_DAYS`	`X Tage`	Der Administrator MUSS die Anzahl der gespeicherten Tage für Protokolle festlegen können: Es gibt einen Konfigurationsparameter LOG_DAYS für das Sicherheitsprotokoll. Es gibt einen gemeinsamen Konfigurationsparameter LOG_DAYS für das Systemprotokoll und das Konnektor-Performanceprotokoll. Dabei DARF der eingestellte Wert NICHT unter der Mindestgröße von 10 Tagen oder über der Maximalgröße von einem Jahr (365 Tage) liegen. Default-Wert: 180
`FM_<fmName>_LOG_DAYS`	`X Tage`	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. Dabei DARF der eingestellte Wert NICHT unter der Mindestgröße von 10 Tagen oder über der Maximalgröße von einem Jahr (365 Tage) liegen. Default-Wert: 180 Die Definition des fachmodulspezifischen Konfigurationswertes ist Bestandteil der

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
		entsprechenden Fachmodulspezifikation. Ist kein fachmodulspezifischer Konfigurationsparameter spezifiziert, dann gilt LOG_DAYS.
`LOG_SUCCESSFUL_CRYPTO_OPS`	`Enabled / Disabled`	Der Administrator MUSS festlegen können, ob auch erfolgreich ausgeführte Kryptooperationen im Sicherheitslog protokolliert werden sollen. Default-Wert: Disabled

⌫

4.1.10.6.1 TUC_KON_272 "Initialisierung Protokollierungsdienst

⌦ **TIP1-A_4718 TUC_KON_272 "Initialisierung Protokollierungsdienst"**

Der Konnektor MUSS den technischen Use Case TUC_KON_272 "Initialisierung Protokollierungsdienst" umsetzen.

Tabelle 240 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.
Eingangsanforderung	Keine
Auslöser und Vorbedingungen	Bootup
Eingangsdaten	Keine
Komponenten	Konnektor
Ausgangsdaten	Keine
Standardablauf	1. Prüfen, ob Schreib-/Lesezugriff auf Sicherheitsprotokoll möglich ist 2. Prüfen, ob Schreib-/Lesezugriff auf Systemprotokoll möglich ist 3. Prüfen, ob Schreib-/Lesezugriff auf Fachmodulprotokolle möglich ist 4. Prüfen, ob Schreib-/Lesezugriff auf Konnektor-Performanceprotokoll möglich ist 5. Prüfen, ob Schreib-/Lesezugriff auf Fachmodul-Performanceprotokolle möglich ist
Varianten/Alternativen	Keine

Element	Beschreibung
Fehlerfälle	Fehler in den folgenden Schritten des Standardablaufs führen zu: a) Aufruf von TUC_KON_256 mit folgenden Parametern {"LOG/ERROR"; $ErrorType; $Severity; „Error=$Fehlercode; Bedeutung=$Fehlertext"; noLog } b) Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes (→1) Zugriff nicht möglich: Fehlercode: 4153 (→2) Zugriff nicht möglich: Fehlercode: 4154 (→3) Zugriff nicht möglich: Fehlercode: 4155 (→4) Zugriff nicht möglich: Fehlercode: 4218 (→5) Zugriff nicht möglich: Fehlercode: 4219
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 241: TAB_KON_611 Übersicht Fehler TUC_KON_272 „Initialisiere Protokollierungs-dienst"

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4153	Technical	Error	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

⌫

4.1.11 TLS-Dienst

Fachmodule müssen gesicherte Verbindungen zu Fachdiensten in der TI aufbauen können. Dabei sollen sie sich mit einer Organisationsidentität (einer SM-B) authentisieren können. Der TLS-Dienst stellt hierfür TUCs für einen TLS-Verbindungsaufbau und -Verbindungsabbau zur Verfügung. Die gesicherte Kommunikation selbst erfolgt dann durch das Fachmodul unter Nutzung der etablierten Verbindung.

Die Funktionalität steht nur zur Verfügung, wenn MGM_LU_ONLINE aktiv ist (siehe Kapitel 4.3.6)

4.1.11.1 Funktionsmerkmalweite Aspekte

4.1.11.2 Durch Ereignisse ausgelöste Reaktionen

⌦ **TIP1-A_4719 TLS-Dienst reagiert auf Veränderung LU_ONLINE**

Tritt das Ereignis „MGM/LU_CHANGED/LU_ONLINE“ ein, so MUSS

- wenn „Active=Ja“ der Dienst bereitgestellt werden
- wenn „Active=Nein“ der Dienst gestoppt werden.
  Sind TLS-Verbindungen aktiv, so MUSS für jede TUC_KON_111 "Kartenbasierte TLS-Verbindung abbauen" gerufen werden.⌫

4.1.11.3 Interne TUCs, nicht durch Fachmodule nutzbar

Keine.

4.1.11.4 Interne TUCs, auch durch Fachmodule nutzbar

4.1.11.4.1 TUC_KON_110 "Kartenbasierte TLS-Verbindung aufbauen"

⌦ **TIP1-A_4720 TUC_KON_110 "Kartenbasierte TLS-Verbindung aufbauen"**

Der Konnektor MUSS den technischen Use Case "Kartenbasierte TLS-Verbindung aufbauen" gemäß TUC_KON_110 umsetzen.

Tabelle 242: 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	• Role – *optional / wenn Rollenprüfung durchgeführt werden soll* • CardSession (SM-B) (optional) • URI des Verbindungsziels
Komponenten	Konnektor, eHealth-Kartenterminal, Karte, Server des Fachdienstes
Ausgangsdaten	• TLSConnectionIdentifier

Standardablauf	Der Konnektor MUSS folgende Schritte durchlaufen:
	1. Auflösen des FQDN im URI per 'TUC_KON_361 "DNS Namen auflösen"
	2. TLS-Verbindung mit ermittelter Adresse aufbauen:
	a) Prüfe Server-Zertifikat mittels TUC_KON_037 “Zertifikat prüfen“
	{ certificate = C.FD.TSL-S;
	qualifiedCheck = not_required;
	offlineAllowNoCheck = true;

Element	Beschreibung

	policyList = oid_fd_tls_s;
	intendedKeyUsage = digitalSignature&keyEncipherment;
	intendedExtendedKeyUsage = id-kp-serverAuth;
	validationMode = OCSP}
	Das Server-Zertifikat MUSS C.FD.TLS-S sein
	b) Prüfe in a) zurückgegebene Rolle ("ermittelte Rolle") = Role
	c) Wenn CardSession übergeben: Clientauthentisierung mittels ID.HCI.AUT
	3. Der TLSConnectionIdentifier der erzeugten Verbindung wird zurückgegeben

Varianten/Alternativen	• Keine
Fehlerfälle	Fehler in den folgenden Schritten des Ablaufs führen zu einem Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes (→1) Der Name der Gegenstelle kann nicht aufgelöst werden (→2b) Rollenprüfung fehlgeschlagen: Fehlercode 4220 (→2) Server konnte nicht authentisiert werden: Fehlercode 4156 (→2) Clientauthentisierung fehlgeschlagen: Fehlercode 4157
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 243: TAB_KON_612 Übersicht Fehler 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

⌫

4.1.11.4.2 TUC_KON_111 "Kartenbasierte TLS-Verbindung abbauen"

⌦ **TIP1-A_4721 TUC_KON_111 "Kartenbasierte TLS-Verbindung abbauen"**

Der Konnektor MUSS den technischen Use Case "Kartenbasierte TLS-Verbindung abbauen" gemäß TUC_KON_111 umsetzen.

Tabelle 244: 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	• TLSConnectionIdentifier
Komponenten	Konnektor, Server des Fachdienstes
Ausgangsdaten	Keine
Standardablauf	Der Konnektor MUSS folgende Schritte durchlaufen: 1. Trennen der über TLSConnectionIdentifier adressierten TLS-Verbindung
Varianten/Alternativen	keine
Fehlerfälle	Fehler in den folgenden Schritten des Ablaufs führen zu einem Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes: (→1) Keine Verbindung mit angegebenem TLSConnectionIdentifier vorhanden: Fehlercode 4158
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 245: TAB_KON_613 Übersicht Fehler TUC_KON_111 "Kartenbasierte TLS-Verbindung 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

⌫

4.1.11.5 Operationen an der Außenschnittstelle

Keine.

4.1.11.6 Betriebsaspekte

⌦ **TIP1-A_4722 TLS-Dienst initialisieren**

Wenn MGM_LU_ONLINE = “Enabled“, MUSS der Basisdienst TLS-Dienst nach dem Bootup zur Nutzung zur Verfügung stehen.

Wenn MGM_LU_ONLINE = “Disabled“, DARF der Basisdienst TLS-Dienst nach dem Bootup NICHT zur Nutzung zur Verfügung stehen.⌫

4.1.12 LDAP-Proxy

Der Konnektor ermöglicht es Clientsystemen und Fachmodulen durch Nutzung des LDAP-Proxies Daten aus dem Verzeichnisdienst der TI-Plattform (VZD) abzufragen. Die Kommunikation erfolgt über das LDAPv3 Protokoll.

Die Funktionalität steht nur zur Verfügung, wenn MGM_LU_ONLINE=Enabled ist (siehe Kapitel 4.3.6)

4.1.12.1 Funktionsmerkmalweite Aspekte

4.1.12.2 Durch Ereignisse ausgelöste Reaktionen

⌦ **TIP1-A_5516 LDAP-Proxy reagiert auf Veränderung LU_ONLINE**

Tritt das Ereignis „MGM/LU_CHANGED/LU_ONLINE“ ein, so MUSS

- wenn „Active=Ja“ der Dienst bereitgestellt werden
- wenn „Active=Nein“ der Dienst gestoppt werden.
  Ist eine Verbindung zum VZD aktiv, so MUSS diese abgebaut werden.⌫

4.1.12.3 Interne TUCs, nicht durch Fachmodule nutzbar

Keine.

4.1.12.4 Interne TUCs, auch durch Fachmodule nutzbar

4.1.12.4.1 TUC_KON_290 „LDAP-Verbindung aufbauen“

⌦ **TIP1-A_5517 Konnektor, TUC_KON_290 „LDAP-Verbindung aufbauen“**

Der Konnektor MUSS den technischen Use Case TUC_KON_290 „LDAP-Verbindung aufbauen“ gemäß TAB_KON_805 umsetzen.

Tabelle 246: 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	Aufruf durch einen LDAPv3 Bind oder StartTLS Request gemäß [RFC4511] und [RFC4513] von einem Fachmodul-TUC oder einem Clientsystem
Vorbedingungen	• MGM_LU_ONLINE=Enabled • MGM_LOGICAL_SEPARATION=Disabled
Eingangsdaten	keine

Element	Beschreibung
Komponenten	Konnektor, VZD
Ausgangsdaten	keine
Standardablauf	1. Wenn Auslöser = StartTLS Request o Der Konnektor sendet eine StartTLS Response an den LDAP-Client. o Die TLS-Verbindung zwischen LDAP-Client und Konnektor wird aufgebaut. Der Konnektor authentisiert sich beim LDAP-Client mit der Identität ID.AK.AUT. o Der Konnektor empfängt vom LDAP Client einen LDAPv3 Bind Request. 2. Der Konnektor ermittelt den FQDN und Port des VZD durch eine DNS-SD Namensauflösung gemäß [RFC6763] mit dem Bezeichner "_ldap._tcp.vzd.<DNS_TOP_LEVEL_DOMAIN_TI>.". 3. Der Konnektor baut mittels LDAPv3 StartTLS Request gemäß [RFC4511]#4.14.1 eine Verbindung zum VZD auf. 4. Der Konnektor empfängt vom VZD eine StartTLS Response gemäß [RFC4511]#4.14.2. 5. Der Konnektor baut die TLS-Verbindung zum VZD auf. Der Konnektor prüft das vom VZD empfangene Zertifikat C.ZD.TLS-S des VZD gemäß TUC_PKI_018 "Zertifikatsprüfung in der TI" prüfen. Die ermittelte Rolle MUSS oid_vzd_ti sein. Bei negativem Ergebnis MUSS der Konnektor die Verbindung abbauen. 6. Der Konnektor antwortet mit einer Bind Response.
Varianten/Alternativen	keine
Fehlerfälle	(→3) Wenn die Verbindung zum VZD nicht aufgebaut werden konnte, muss der Konnektor eine Unsolicited Notification/Notice of Disconnection gemäß [RFC4511] an den LDAP-Client senden. Weitere auftretende Fehler werden gemäß [RFC4511]# Appendix A behandelt.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

⌫

4.1.12.4.2 TUC_KON_291 „Verzeichnis abfragen“

⌦ **TIP1-A_5518 Konnektor, TUC_KON_291 „Verzeichnis abfragen“**

Der Konnektor MUSS den technischen Use Case TUC_KON_291 „Verzeichnis abfragen“ gemäß TAB_KON_815 umsetzen.

Tabelle 247: 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 einen Fachmodul-TUC oder einem Clientsystem
Vorbedingungen	• MGM_LU_ONLINE=Enabled • MGM_LOGICAL_SEPARATION=Disabled • Eine LDAPv3-Verbindung LDAP-Client sowie eine LDAPv3 Verbindung vom Konnektor zum VZD wurden aufgebaut (TUC_KON_290 „LDAP-Verbindung aufbauen“)
Eingangsdaten	• LDAPv3 Search Request gemäß [RFC4511]#4.5.1
Komponenten	Konnektor, VZD
Ausgangsdaten	• LDAPv3 Search Response gemäß [RFC4511]#4.5.2
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

⌫

4.1.12.4.3 TUC_KON_292 „LDAP-Verbindung trennen"

⌦ **TIP1-A_5519 Konnektor, TUC_KON_292 „LDAP-Verbindung trennen"**

Der Konnektor MUSS den technischen Use Case „LDAP-Verbindung trennen" gemäß TAB_KON_816 umsetzen.

Tabelle 248: 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 einen Fachmodul-TUC oder einem Clientsystem
Vorbedingungen	• MGM_LU_ONLINE=Enabled

Element	Beschreibung
	• MGM_LOGICAL_SEPARATION=Disabled • Eine LDAPv3-Verbindung LDAP-Client sowie eine LDAPv3 Verbindung vom Konnektor zum VZD wurden aufgebaut (TUC_KON_290 „Verbindungsaufbau zum VZD“)
Eingangsdaten	keine
Komponenten	Konnektor, VZD
Ausgangsdaten	keine
Standardablauf	1. Der Konnektor empfängt vom LDAP-Client einen Unbind Request gemäß [RFC4511]#4.3. 2. Der Konnektor sendet zum VZD einen Unbind Request. 3. Der Konnektor beendet die Verbindung zum VZD und zum LDAP Client gemäß [RFC4511]#5.3.
Varianten/Alternativen	keine
Fehlerfälle	Auftretende Fehler werden gemäß [RFC4511]# Appendix A behandelt.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

⌫

4.1.12.4.4 TUC_KON_293 „Verzeichnisabfrage abbrechen"

⌦ **TIP1-A_5520 Konnektor, TUC_KON_293 „Verzeichnisabfrage abbrechen"**

Der Konnektor MUSS den technischen Use Case TUC_KON_293 „Verzeichnisabfrage abbrechen" gemäß TAB_KON_817 umsetzen.

Tabelle 249: 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 einen Fachmodul-TUC oder einem Clientsystem
Vorbedingungen	• MGM_LU_ONLINE=Enabled • MGM_LOGICAL_SEPARATION=Disabled • 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.
Eingangsdaten	keine
Komponenten	Konnektor, VZD

Element	Beschreibung
Ausgangsdaten	keine
Standardablauf	1. Der Konnektor empfängt vom LDAP-Client einen Abandon Request gemäß [RFC4511]#4.11. 2. Der Konnektor sendet zum VZD einen Abandon Request gemäß [RFC4511]#4.11
Varianten/Alternativen	keine
Fehlerfälle	Auftretende Fehler werden gemäß [RFC4511]# Appendix A behandelt.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

⌫

4.1.12.5 Operationen an der Außenschnittstelle

4.1.12.5.1 Unterstützte LDAPv3 Operationen

⌦ **TIP1-A_5521 Konnektor, LDAPv3 Operationen**

Der Konnektor MUSS an der Client-Schnittstelle die folgenden LDAPv3 Operationen gemäß [RFC4511] anbieten.

- Bind Operation
- Unbind Operation
- Search Operation
- Abandon Operation
- StartTLS Operation

Andere LDAPv3 Operationen werden mit dem LDAP-Fehler unwillingToPerform (53) beantwort.

Der TCP Port 389 muss verwendet werden.

Wenn ANCL_TLS_MANDATORY=Enabled, muss der Konnektor sicherstellen, dass nur über eine bestehende TLS-Verbindung (aufgebaut über eine StartTLS Operation gemäß [RFC4511]#4.14 und [RFC4513]) Daten abgefragt werden können.

Fehler müssen gemäß [RFC4511]#Appendix A behandelt werden.⌫

4.1.12.6 Betriebsaspekte

keine

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 des Leistungserbringers, 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, NET_TI_GESICHERTE_FD, NET_TI_ZENTRAL, NET_SIS 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 LEKTR-Umgebung MUSS die IP-Adresse VPN_TUNNEL_TI_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 beim LE verwenden. ⌫

⌦ **TIP1-A_4726 Internet Anbindung nur bei MGM_LU_ONLINE**

Der Hersteller des Konnektor MUSS sicher stellen, 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_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 250: TAB_KON_680 Mapping der Netzwerksegmente

ReferenzID im Konnektor	Adressbereich für die TI-Produktivumgebung	Adressbereich für die TI-Testumgebung	Adressbereich für die TI-Referenz-umgebung
NET_SIS	TI_Dezentral_SIS - Konnektoren	TI_Test_Dezentral_SIS - Konnektoren	Ist durch den Testbetriebs-verantwortlichen zu definieren.
NET_TI_DEZENTRAL	TI_Dezentral - Konnektoren	TI_Test_Dezentral - - Konnektoren	Ist durch den Testbetriebs-verantwortlichen zu definieren.
NET_TI_ZENTRAL	TI_Zentral - Zentrale Dienste	TI_Test_Zentral - Zentrale Dienste	Ist durch den Testbetriebs-verantwortlichen zu definieren.
NET_TI_OFFENE_FD	TI_Fachdienste - Offene Fachdienste	TI_Test_Fachdienste - Offene Fachdienste	Ist durch den Testbetriebs-verantwortlichen zu definieren.
NET_TI_GESICHERTE_FD	TI_Fachdienste - Gesicherte Fachdienste	TI_Test_Fachdienste - Gesicherte Fachdienste	Ist durch den Testbetriebs-verantwortlichen zu definieren.
NET_LEKTR	Liste der Netzwerke die in der LE-Umgebung über den Konnektor erreichbar sind. Ein Eintrag der Liste enthält die Netzwerkadresse und den Netzwerkprefix.		
ANLW_BESTANDSNETZE	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_BESTANDSNETZE	Liste der an die TI angeschlossenen und vom Administrator freigeschalteten Bestandsnetze		

Tabelle 251: 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 Address-Segment NET_SIS

Tabelle 252: TAB_KON_682 Definition der Konnektor IP-Adressen

ReferenzID	Bedeutung / Belegung
ANLW_LAN_IP_ADDRESS	Dies ist die IP-Adresse des LAN-Adapters. Aus dem Netz des Leistungserbringers/Kostenträgers (ANLW_LAN_NETWORK_SEGMENT) die vom Konnektor verwendete IP-Adresse. Unter dieser Adresse werden die Dienste des Konnektor im lokalen Netzwerk bereitgestellt. Diese Adresse entspricht dem in Tabelle 259: 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.

4.2.1.1.2 Routing und Firewall

Darstellung der Kommunikationsregeln des Konnektors

Diese Abbildung dient der Veranschaulichung der im Konnektor verwendeten Kommunikationsregeln welche in den nachfolgenden Afo definiert werden.

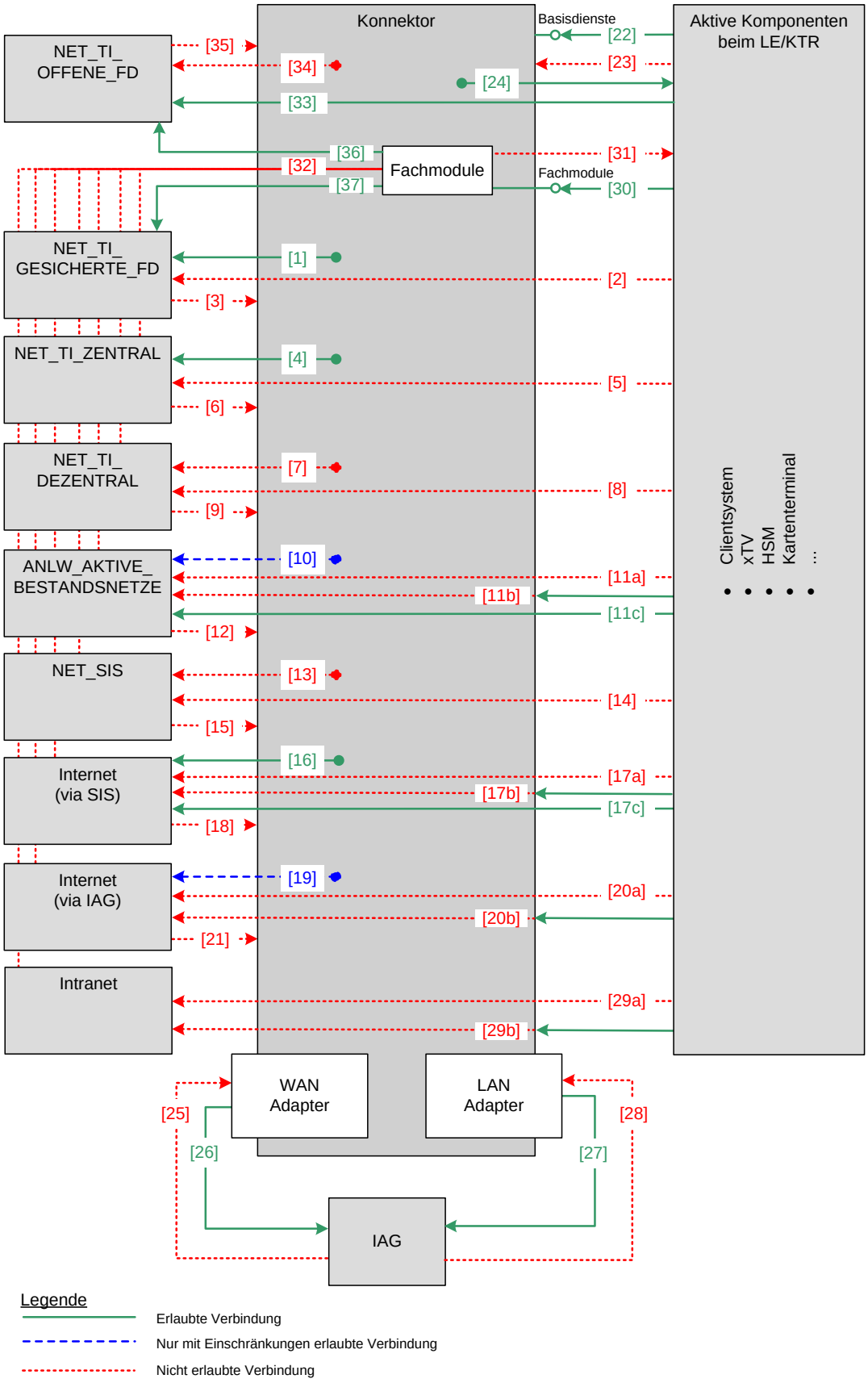


Abbildung 19: PIC_KON_115 Kommunikationsregeln Konnektor

⌦ **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 ausschließlich in den VPN-Tunnel der TI (VPN_TI) geleitet werden. ⌫

⌦ **TIP1-A_4732 Kommunikation mit NET_TI_DEZENTRAL**

Der Konnektor MUSS sicherstellen, dass die Adressen aus dem Adressbereich NET_TI_DEZENTRAL nur für die Kommunikation mit der TI/den Bestandsnetzen in Form der inner IP (VPN_TUNNEL_TI_INNER_IP) des VPN-Tunnel der TI (VPN_TI) verwendet wird.

Der Konnektor MUSS die Kommunikation mit Systemen des Netzwerksegments NET_TI_DEZENTRAL für folgende Fälle unterstützen:

- keine

Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit Systemen des Netzwerksegments NET_TI_DEZENTRAL für folgende Fälle blockieren:

- [7] vom Konnektor kommend (zur Verhinderung des Zugriffs auf fremde Konnektoren)
- [8] von „Aktive Komponenten“
- [9] in Richtung Konnektor gehend ⌫

⌦ **TIP1-A_4733 Kommunikation mit ANLW_AKTIVE_BESTANDSNETZE**

Der Konnektor MUSS sicherstellen, dass IP-Pakete mit einer Absenderadresse aus dem Adressbereich ANLW_AKTIVE_BESTANDSNETZE verworfen werden, wenn sie nicht aus dem VPN-Tunnel der TI (VPN_TI) stammen.

Der Konnektor MUSS die Kommunikation mit Systemen des Netzwerksegments ANLW_AKTIVE_BESTANDSNETZE für folgende Fälle unterstützen:

- [10] wenn (MGM_LU_ONLINE=Enabled und MGM_LOGICAL_SEPARATION=Disabled) vom Konnektor kommend nur für die DNS-Namensauflösung mittels DNS_SERVERS_BESTANDSNETZE
- [11c] wenn (MGM_LU_ONLINE=Enabled und MGM_LOGICAL_SEPARATION=Disabled) von „Aktive Komponenten“ kommend

Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit Systemen des Netzwerksegments ANLW_AKTIVE_BESTANDSNETZE für folgende Fälle blockieren:

- [11a] wenn (MGM_LOGICAL_SEPARATION=Enabled) für freigegebene Bestandsnetze (ANLW_AKTIVE_BESTANSNETZE) von „Aktive Komponenten“ kommend
- [11b] für nicht freigegebene Bestandsnetze (ANLW_BESTANSNETZE abzüglich ANLW_AKTIVE_BESTANSNETZE) von „Aktive Komponenten“ kommend;
- [12] in Richtung Konnektor gehend (und den dahinterliegenden „Aktive Komponenten“)

Der Konnektor MUSS sicherstellen, dass die aus einer unterstützten Kommunikation mit Systemen aus dem Netzwerksegment ANLW_AKTIVE_BESTANDSNETZE bestimmten IP-Pakete ausschließlich in den VPN-Tunnel der TI (VPN_TI) geleitet werden. ⌫

⌦ **TIP1-A_4734 Kommunikation mit NET_SIS**

Der Konnektor MUSS sicherstellen, dass eine Adresse aus dem Adressbereich NET_SIS nur für die Kommunikation mit dem Internet (via SIS) in Form der inner IP (VPN_TUNNEL_SIS_INNER_IP) des VPN-Tunnel der SIS (VPN_SIS) verwendet wird.

Der Konnektor MUSS insbesondere die Kommunikation mit Systemen des Netzwerksegments NET_SIS für folgende Fälle unterstützen:

- keine

Der Konnektor MUSS die Kommunikation an seinen Außenschnittstellen mit NET_SIS für folgende Fälle blockieren:

- [13] vom Konnektor kommend
- [14] von „Aktive Komponenten“ kommend
- [15] in Richtung Konnektor gehend (und den dahinterliegenden „Aktiven Komponenten“)⌫

⌦ **TIP1-A_4735 Kommunikation mit dem Internet (via SIS)**

Der Konnektor MUSS sicherstellen, dass IP-Pakete mit einer Absenderadresse aus dem Adressbereich NET_TI_ZENTRAL, NET_TI_GESICHERTE_FD; NET_TI_OFFENE_FD, NET_TI_DEZENTRAL, ANLW_AKTIVE_BESTANDSNETZE, ANLW_LAN_ADDRESS_SEGMENT, aus einem der Netzwerksegmente in ANLW_LEKTR_INTRANET_ROUTES oder ANLW_WAN_NETWORK_SEGMENT verworfen werden, wenn sie aus dem VPN-Tunnel der SIS (VPN_SIS) stammen.

Der Konnektor MUSS die Kommunikation mit Systemen des Netzwerksegments Internet (via SIS) für folgende Fälle unterstützen:

- [16] wenn (MGM_LU_ONLINE=Enabled und MGM_LOGICAL_SEPARATION=Disabled und ANLW_INTERNET_MODUS=SIS) vom Konnektor kommend
- [17c] wenn (MGM_LU_ONLINE=Enabled und MGM_LOGICAL_SEPARATION=Disabled und ANLW_INTERNET_MODUS=SIS) von „Aktive Komponenten“ kommend

Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit Internet (via SIS) für folgende Fälle blockieren oder umleiten:

- [17a] blockieren, wenn (MGM_LU_ONLINE=Enabled und MGM_LOGICAL_SEPARATION=Disabled und ANLW_INTERNET_MODUS=KEINER) von „Aktive Komponenten“ kommend
- [17b] umleiten, wenn (MGM_LU_ONLINE=Enabled und MGM_LOGICAL_SEPARATION=Disabled und ANLW_INTERNET_MODUS=SIS) von „Aktive Komponenten“ kommend;
  ➔ Der Konnektor MUSS an Hosts im Internet gerichtete IP-Pakete gemäß [RFC792] umleiten (ICMP Redirect).
- [18] blockieren, wenn von SIS kommend in Richtung Konnektor (und die dahinterliegenden „Aktive Komponenten“)

Der Konnektor MUSS sicherstellen, dass die für die Kommunikation mit dem Internet (via SIS) bestimmten IP-Pakete ausschließlich in den VPN-Tunnel des SIS (VPN_SIS) geleitet werden. ⌫

⌦ **TIP1-A_4736 Kommunikation mit dem Internet (via IAG)**

Der Konnektor MUSS sicherstellen, dass eingehende IP-Pakete von der Kommunikation mit dem Internet mit der Empfängeradresse ungleich (ANLW_LAN_IP_ADDRESS oder aus einem der Netzwerksegmente in ANLW_LEKTR_INTRANET_ROUTES wenn ANLW_WAN_ADAPTER_MODUS=DISABLED) oder (ANLW_WAN_IP_ADDRESS wenn ANLW_WAN_ADAPTER_MODUS=ENABLED) verworfen werden.

Der Konnektor MUSS sicherstellen, dass ausgehende IP-Pakete für die Kommunikation mit dem Internet mit der Absenderadresse ungleich (ANLW_LAN_IP_ADDRESS oder aus einem der Netzwerksegmente in ANLW_LEKTR_INTRANET_ROUTES wenn ANLW_WAN_ADAPTER_MODUS=DISABLED) oder (ANLW_WAN_IP_ADDRESS wenn ANLW_WAN_ADAPTER_MODUS=ENABLED) verworfen werden.

Der Konnektor MUSS die Kommunikation mit Systemen des Netzwerksegments Internet (via IAG) für folgende Fälle unterstützen:

- [19] vom Konnektor kommend zu den folgenden Systemen für das Protokoll IPsec
  - VPN_KONZENTRATOR_TI_IP_ADDRESS
  - VPN_KONZENTRATOR_SIS_IP_ADDRESS
- [19] vom Konnektor kommend zu den folgenden Systemen für HTTP und HTTPS

- o CERT_CRL_DOWNLOAD_ADDRESS
- o DNS_ROOT_ANCHOR_URL
- o hash&URL-Server
- o Registrierungsserver
- o Remote-Managementserver

- [19] vom Konnektor kommend zu den folgenden Systemen für das Protokoll DNS
  - o beliebige Hosts

Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit Internet (via IAG) für folgende Fälle blockieren oder umleiten:

- [20a] blockieren, wenn (ANLW_INTERNET_MODUS=KEINER oder MGM_LU_ONLINE=Disabled oder MGM_LOGICAL_SEPARATION=Enabled) von „Aktive Komponenten“ kommend
- [20b] mittels ICMP Redirect gemäß [RFC792] zum Default Gateway umleiten, wenn die Zieladresse des IP-Pakets nicht innerhalb der Adressbereiche (NET_TI_ZENTRAL, NET_TI_OFFENE_FD, NET_TI_GESICHERTE_FD und ANLW_AKTIVE_BESTANDSNETZE) ist und ANLW_INTERNET_MODUS≠SIS und von „Aktive Komponenten“ kommend.
- [21] blockieren, wenn von IAG kommend in Richtung Konnektor (und die dahinterliegenden „Aktive Komponenten“)⌫

⌦ **TIP1-A_4737 Kommunikation mit „Aktive Komponenten“**

Der Konnektor MUSS sicherstellen, dass ausgehende IP-Pakete für die Kommunikation mit „Aktive Komponenten“ mit einer Absenderadresse ungleich ANLW_LAN_IP_ADDRESS, einer Adresse aus einem Netzwerksegment in ANLW_LEKTR_INTRANET_ROUTES oder 0.0.0.0 verworfen werden.

Der Konnektor MUSS die Kommunikation mit „Aktive Komponenten“ für folgende Fälle unterstützen:

- [22] auf den Konnektor (mittels der Schnittstelle Basisdienste)
- [24] vom Konnektor kommend

Der Konnektor MUSS insbesondere die Kommunikation an seinen Außenschnittstellen mit „Aktive Komponenten“ für folgende Fälle blockieren:

- [23] zum Konnektor eingehend (direkt – ohne eine der Schnittstellen Fachmodule oder Basisdienste zu nutzen)⌫

⌦ **TIP1-A_4738 Route zum IAG**

Der Konnektor MUSS die Kommunikation mit dem IAG (beim LE) 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 (beim LE) 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:

- keine

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“
- [32] zu den Netzwerksegmenten, NET_TI_ZENTRAL, NET_TI_DEZENTRAL, ANLW_AKTIVE_BESTANDSNETZE, Internet (via SIS), Internet (via IAG) und Intranet⌫

TIP1-A_4744 Firewall - Drop statt Reject

Die Firewall des Konnektor MUSS alle abgelehnten IP-Pakete verwerfen (DROP) ohne ein ICMP-Destination-Unreachable (Type 3) zu schicken. ⌫

TIP1-A_4746 Firewall – Abwehr von IP-Spoofing, DoS/DDoS-Angriffe und Martian Packets

Der Konnektor MUSS geeignete technische Funktionen zur Abwehr von IP-Spoofing und DoS/DDoS-Angriffen implementieren.

Der Konnektor MUSS Martian Packets (Absender– oder Empfängeradressen aus den von der IETF als Special-Purpose definierten Netzbereichen), mindestens jedoch aus folgenden Netzbereichen 0.0.0.0/8, 127.0.0.0/8, 169.254.0.0/16, 192.0.0.0/24, 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, 240.0.0.0/4 verwerfen. Die in [RFC1918] und [RFC 6598] definierten Netzbereiche sind hiervon ausgenommen. ⌫

TIP1-A_4745 Eingeschränkte Nutzung von „Ping“

Die Firewall des Konnektor MUSS TCP-Port-7(Echo)-Pakete verwerfen.

Die Firewall des Konnektor MUSS ICMP-Echo-Request (Typ 8) und ICMP-Echo-Response (Typ 0) ausschließlich für die folgenden Kommunikationen zulassen:

- vom Konnektor zu den VPN-Konzentratoren für SIS und TI über das Transportnetz (via IAG)
- vom Konnektor zu dem CRL-Webservern (im Transportnetz) über das Internet (via SIS) und das Transportnetz (via IAG)
- vom Konnektor zu dem IAG (beim Leistungserbringer)
- vom Konnektor zu NET_TI_ZENTRAL
- vom Konnektor zu NET_TI_GESICHERTE_FD
- vom Konnektor zu NET_TI_OFFENE_FD
- vom NET_TI_DEZENTRAL zum Konnektor
- vom NET_SIS zum Konnektor
- vom lokalen Netzwerk (Adressen aus ANLW_LAN_NETWORK_SEGMENT (jedoch ohne die ANLW_LAN_IP_ADDRESS) oder Adressen aus einem der Netzwerksegmente in ANLW_LEKTR_INTRANET_ROUTES) zum Konnektor
- vom lokalen Netzwerk in ANLW_AKTIVE_BESTANDSNETZE (die freigegebenen Bestandsnetze)
- vom lokalen Netzwerk in das Internet (via SIS)

Die Firewall des Konnektors MUSS für alle anderen Kommunikationen ein ICMP-Echo-Request (Typ 8) verworfen werden. ⌫

TIP1-A_4747 Firewall – Einschränkungen der IP-Protokolle

Der Konnektor MUSS alle IP-Protokolle außer 1 (ICMP), 17 (UDP), 6 (TCP) und 50 (ESP) für alle ein- oder ausgehenden Pakete an allen seinen Adaptern verwerfen.

TIP1-A_4748 Firewall – Routing-Regeln

Der Konnektor DARF seine Routing-Regeln NICHT durch IP-Kommunikation beeinflussen lassen, weder mittels eines Routing-Protokolls (wie BGP oder RIP) noch mittels ICMP-Kommandos (wie Redirect (5), Router Advertisement (9/10) oder auch Mobile Host Redirect (32)) sondern MUSS diese ausschließlich durch TUC_KON_304 „Netzwerk-Routen einrichten“ setzen.

Die Firewall des Konnektor MUSS alle aus einem der Tunnel (VPN_TI oder VPN_SIS) kommenden DHCP-Pakete verwerfen.

Die Firewall des Konnektors MUSS an den Konnektor gerichtete IPsec-Pakete (IKE, ESP und IPsec NAT-T) verwerfen, sofern sie nicht einer vom Konnektor initiierten IPsec-Verbindung (VPN_TI und VPN_SIS) zugeordnet werden können.

TIP1-A_4749 Firewall Restart

Der Konnektor MUSS gewährleisten, dass unmittelbar nach einer Änderung der Parameter eines Adapters (LAN-Adapter, WAN-Adapter, virtueller Adapter VPN_TI oder virtueller Adapter VPN_SIS) die Firewall des Konnektor neu erstellt und geladen wird.

Wenn der WAN-Adapter verwendet wird (ANLW_WAN_ADAPTER_MODUS=ENABLED) DARF die Firewall des Konnektor bei einer Änderung der ANLW_WAN_IP_ADDRESS NICHT die Verbindungen über den LAN-Adapter durch einen Restart der Firewall beeinflussen.

Wenn der WAN-Adapter verwendet wird (ANLW_WAN_ADAPTER_MODUS=ENABLED), DARF die Firewall des Konnektor bei einer Änderung der ANLW_LAN_IP_ADDRESS NICHT die Verbindungen über die Adapter WAN, VPN_TI oder VPN_SIS durch einen Restart der Firewall beeinflussen.

Umsetzungshinweis für den Hersteller: Es können zwei getrennten Firewall-Regelsets für den LAN- bzw. für den WAN-Adapter verwendet werden.

TIP1-A_4750 Firewall-Protokollierung

Der Konnektor MUSS bei Start und Stopp der Firewall einen Protokolleintrag mit der Schwere „Warning“ und dem Typ „Operations“ sowie mindestens folgenden Informationen generieren:

• Zeitstempel, Aktion (Start/Stop), Ergebnis (Erfolg/Fehler), Auslöser (Prozess/User)

Der Konnektor MUSS bei Konfigurationsänderungen der Firewall einen Protokolleintrag mit der Schwere „Warning“ und dem Typ „Operations“ sowie mindestens folgenden Informationen generieren:

• Zeitstempel, Aktion (Add/Delete/Change), Details (Beschreibung der Änderung), Auslöser (Prozess/User)

Der Konnektor MUSS für alle vom Konnektor ausgehenden, nicht zugelassenen Kommunikationsversuche einen Protokolleintrag mit der Schwere „Warning“ und dem Typ „Security“ sowie mindestens folgenden Informationen generieren:

• Zeitstempel, Aktion (Drop, Reject), Absender-IP-Adresse, Empfänger-IP-Adresse, Protokoll, Absender-Port und Empfänger-Port, Interface über das das Paket empfangen wurde

Der Konnektor MUSS für alle verworfenen IP-Spoofing- und Martian-Packets einen Protokolleintrag mit der Schwere „Warning“ und dem Typ „Security“ sowie mindestens folgenden Informationen generieren:

• Zeitstempel, Aktion (Drop, Reject), Absender-IP-Adresse, Empfänger-IP-Adresse, Protokoll, Absender-Port und Empfänger-Port, Interface über das das Paket empfangen wurde

Der Konnektor MUSS für alle von der Firewall verworfenen IP-Pakete einen Protokolleintrag mit der Schwere „Info“ und dem Typ „Security“ sowie mindestens folgenden Informationen generieren:

• Zeitstempel, Aktion (Drop, Reject), Absender-IP-Adresse, Empfänger-IP-Adresse, Protokoll, Absender-Port und Empfänger-Port, Interface über das das Paket empfangen wurde

Der Konnektor MUSS für die Firewall-Protokollierung den TUC_KON_271 „Schreibe Protokolleintrag“ nutzen. ⌫

4.2.1.2 Durch Ereignisse ausgelöste Reaktionen

⌦ **TIP1-A_4751 Reagiere auf LAN_IP_Changed**

Beim Auftreten eines der nachfolgenden Events MUSS der Konnektor den TUC_KON_305 „LAN-Adapter initialisieren“ starten.

- Event ANLW/LAN/IP_CHANGED
- Event DHCP/LAN_CLIENT/RENEW⌫

⌦ **TIP1-A_4752 Reagiere auf WAN_IP_Changed**

Beim Auftreten eines der nachfolgenden Events MUSS der Konnektor TUC_KON_306 „WAN-Adapter initialisieren“ starten.

- Event ANLW/WAN/IP_CHANGED
- Event DHCP/WAN_CLIENT/RENEW⌫

⌦ **TIP1-A_4753 Ereignisbasiert Netzwerkrouten einrichten**

Beim Auftreten eines der nachfolgenden Events MUSS der Konnektor den TUC_KON_304 „Netzwerk-Routen einrichten“ aufrufen.

- Event NETWORK/VPN_TI/UP
- Event NETWORK/VPN_TI/DOWN
- Event NETWORK/VPN_SIS/UP

- Event NETWORK/VPN_SIS/DOWN
- Event MGM/LU_CHANGED/LU_ONLINE⌫

4.2.1.3 Interne TUCs, nicht durch Fachmodule nutzbar

4.2.1.3.1 TUC_KON_305 „LAN-Adapter initialisieren“

⌦ **TIP1-A_4754 TUC_KON_305 „LAN-Adapter initialisieren“**

Der Konnektor MUSS den technischen Use Case TUC_KON_305 „LAN-Adapter initialisieren“ umsetzen.

Tabelle 253: TAB_KON_614 - TUC_KON_305 „LAN-Adapter initialisieren“

Element	Beschreibung
Name	TUC_KON_305 LAN-Adapter initialisieren
Beschreibung	Initialisieren der LAN-Netzwerkschnittstelle
Auslöser	• Event ANLW/LAN/IP_CHANGED • Event DHCP/LAN_CLIENT/RENEW
Vorbedingungen	• Wenn die IP-Konfiguration des LAN-Adapters statisch (DHCP_CLIENT_LAN_STATE=Disabled) gesetzt wird, MUSS der Konnektor gewährleisten, dass alle Konfigurationsparameter gemäß „Tabelle 259: TAB_KON_683 LAN-Adapter IP-Konfiguration“ vorab über die Managementschnittstelle gesetzt wurden. • 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.
Eingangsdaten	Keine
Komponenten	Konnektor
Ausgangsdaten	Keine
Standardablauf	1) Die in „Tabelle 259: TAB_KON_683 LAN-Adapter IP-Konfiguration“ und „Tabelle 260: TAB_KON_684 LAN-Adapter Erweiterte Parameter“ gesetzten Werte sind zur Konfiguration des LAN-Adapter zu verwenden. 2) Rufe TUC_KON_304 „Netzwerk-Routen einrichten“ 3) Wenn (ANLW_WAN_ADAPTER_MODUS = DISABLED) und MGM_LU_ONLINE = ENABLED: • Rufe TUC_KON_321 „Verbindung zu dem VPN-Konzentrator der TI aufbauen“. • Rufe TUC_KON_322 „Verbindung zu dem VPN-

Element	Beschreibung
	Konzentrator des SIS aufbauen“ 4) Firewall-Regeln aktualisieren und aktivieren. Tritt der Fehler 4164 auf, geht der Konnektor in den Betriebszustand EC_Firewall_Not_Reliable über.
Varianten / Alternativen	Keine
Fehlerfälle	(→ 1) Fehlerhafte LAN IP-Konfiguration; 4162 (→ 4) Beim Aktualisieren oder Aktivieren der Firewall-Regeln ist es zu einem Fehler gekommen; 4164
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 254: TAB_KON_615 Übersicht Fehler 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.

⌫

4.2.1.3.2 TUC_KON_306 „WAN-Adapter initialisieren“

⌦ **TIP1-A_4755 TUC_KON_306 „WAN-Adapter initialisieren“**

Der Konnektor MUSS den technischen Use Case TUC_KON_306 „WAN-Adapter initialisieren“ umsetzen.

Tabelle 255: TAB_KON_616 - TUC_KON_306 "WAN-Adapter initialisieren"

Element	Beschreibung
Name	TUC_KON_306 WAN-Adapter initialisieren
Beschreibung	Initialisieren der WAN-Netzwerkschnittstelle
Auslöser	• Event ANLW/WAN/IP_CHANGED • Event DHCP/WAN_CLIENT/RENEW
Vorbedingungen	
Eingangsdaten	Keine
Komponenten	Konnektor
Ausgangsdaten	Keine
Standardablauf	1) Wenn ANLW_WAN_ADAPTER_MODUS = DISABLED

Element	Beschreibung
	oder MGM_LU_ONLINE = Disabled: a) Aktive VPN-Tunnel TI oder SIS (VPN_TI oder VPN_SIS) müssen gestoppt werden, 2) Wenn ANLW_WAN_ADAPTER_MODUS = ENABLED und MGM_LU_ONLINE = ENABLED: a) Der WAN-Adapter wird abhängig von DHCP_CLIENT_WAN_STATE statisch oder dynamisch über DHCP konfiguriert. Die in „Tabelle 261: TAB_KON_685 WAN-Adapter IP-Konfiguration“ und „Tabelle 262: TAB_KON_686 WAN-Adapter Erweiterte Parameter“ gesetzten Werte sind zur Konfiguration des WAN-Adapter zu verwenden. b) Rufe TUC_KON_304 „Netzwerk-Routen einrichten“ c) Rufe TUC_KON_321 „Verbindung zu dem VPN-Konzentrator der TI aufbauen“. d) Rufe TUC_KON_322 „Verbindung zu dem VPN-Konzentrator des SIS aufbauen“ e) Firewall-Regeln aktualisieren und aktivieren. Tritt der Fehler 4164 auf, geht der Konnektor in den Betriebszustand EC_Firewall_Not_Reliable über.
Varianten / Alternativen	Keine
Fehlerfälle	(→ 1b1) Fehlerhafte WAN IP-Konfiguration; 4163 (→ 2) Beim Aktualisieren oder Aktivieren der Firewall-Regeln ist es zu einem Fehler gekommen; 4164
Nichtfunktionale Anforderungen	Keine

Tabelle 256: TAB_KON_617 Übersicht Fehler 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.

⌫

4.2.1.3.3 TUC_KON_304 „Netzwerk-Routen einrichten“

⌦ **TIP1-A_4758 TUC_KON_304 „Netzwerk-Routen einrichten“**

Der Konnektor MUSS den technischen Use Case TUC_KON_304 „Netzwerk-Routen einrichten“ umsetzen.

Tabelle 257: TAB_KON_622 - TUC_KON_304 „Netzwerk-Routen einrichten“

Element	Beschreibung
Name	TUC_KON_304 Netzwerk-Routen einrichten
Beschreibung	Anpassen der Routing-Tabelle
Auslöser	• TUC_KON_305 „LAN-Adapter initialisieren“ • TUC_KON_306 „WAN-Adapter initialisieren“ • Event NETWORK/VPN_TI/UP • Event NETWORK/VPN_TI/DOWN • Event NETWORK/VPN_SIS/UP • Event NETWORK/VPN_SIS/DOWN • Event MGM/LU_CHANGED/LU_ONLINE
Vorbedingungen	keine
Eingangsdaten	• IP-Konfiguration des LAN-Interface (gemäß Tabelle 259: TAB_KON_683 LAN-Adapter IP-Konfiguration) • IP-Konfiguration des WAN-Interface (gemäß Tabelle 261: TAB_KON_685 WAN-Adapter IP-Konfiguration) • ANLW_IAG_ADDRESS (IP-Adresse des IAG beim LE) • DNS_SERVERS_INT
Komponenten	Konnektor
Ausgangsdaten	Keine
Nachbedingungen	• Die Routing-Einträge im Konnektor wurden gesetzt.
Standardablauf	Alle bestehenden Routen MÜSSEN vollständig durch die in diesem TUC ermittelten Routen ersetzt werden. **1) Wenn (MGM_LU_ONLINE=Enabled** Der Konnektor MUSS die nachfolgenden Routen bereitstellen. a) i. Ziel: Lokale Netze der LE/KTR gemäß ANLW_LEKTR_INTRANET_ROUTES Next Hop: gemäß ANLW_LEKTR_INTRANET_ROUTES b) Wenn die VPN-Tunnel zur TI und zum SIS nicht aufgebaut sind: i. Ziel: Default Route Next Hop: ANLW_IAG_ADDRESS c) Wenn der VPN-Tunnel zur TI aufgebaut und

Element	Beschreibung
	der VPN-Tunnel zum SIS nicht aufgebaut sind: i. Ziel: Default Route Next Hop: ANLW_IAG_ADDRESS ii. Ziel: TI (NET_TI_OFFENE_FD, NET_TI_GESICHERTE_FD und NET_TI_ZENTRAL) Next Hop: Innere Tunnel IP-Adresse des VPN-Konzentrators TI iii. Ziel: ANLW_AKTIVE_BESTANDSNETZE Next Hop: Innere Tunnel IP-Adresse des VPN-Konzentrators TI iv. Ziel: VPN-Konzentrator TI Next Hop: ANLW_IAG_ADDRESS d) Wenn die VPN-Tunnel zur TI und zum SIS aufgebaut sind: i. Ziel: Default Route Next Hop: Innere Tunnel IP-Adresse des VPN-Konzentrators SIS ii. Ziel: TI (NET_TI_OFFENE_FD, NET_TI_GESICHERTE_FD und NET_TI_ZENTRAL) Next Hop: Innere Tunnel IP-Adresse des VPN-Konzentrators TI iii. Ziel: ANLW_AKTIVE_BESTANDSNETZE Next Hop: Innere Tunnel IP-Adresse des VPN-Konzentrators TI iv. Ziel: VPN-Konzentrator TI Next Hop: ANLW_IAG_ADDRESS v. Ziel: VPN-Konzentrator SIS Next Hop: ANLW_IAG_ADDRESS Hinweis: 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. **2) Wenn (MGM_LU_ONLINE=Disabled)** 1. Der Konnektor MUSS die nachfolgenden Routen bereitstellen. i. Ziel: Lokale Netze der LE/KTR gemäß ANLW_LEKTR_INTRANET_ROUTES Next Hop: gemäß ANLW_LEKTR_INTRANET_ROUTES

Element	Beschreibung
	3) Firewall aktualisieren: 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 (→ 3) Beim Aktualisieren oder Aktivieren der Firewall-Regeln ist es zu einem Fehler gekommen; 4164
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 258: TAB_KON_623 Übersicht Fehler 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: Ein oder mehrere Adressen sind ungültig.
4164	Technical	Fatal	Beim Aktualisieren oder Aktivieren der Firewall-Regeln ist es zu einem Fehler gekommen.

⌫

4.2.1.4 Interne TUCs, auch durch Fachmodule nutzbar

Keine.

4.2.1.5 Operationen an der Außenschnittstelle

Keine

4.2.1.6 Betriebsaspekte

⌦ **TIP1-A_5414 Initialisierung „Anbindung LAN/WAN“**

Der Konnektor MUSS in der Bootup-Phase zur Initialisierung des Funktionsmerkmals „Anbindung LAN/WAN“:

- den LAN Adapter initialisieren (TUC_KON_305)
- den WAN Adapter initialisieren (TUC_KON_306)
- die Infrastrukturdaten vom KSR einlesen (TUC_KON_283)⌫

⌦ TIP1-A_4759 Konfiguration LAN-Interface

Der Konnektor MUSS gewährleisten, dass die Konfiguration nur dann gespeichert wird, 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 {"ANLW/LAN/IP_CHANGED"; Op; Info;IP=$dieNeueIP; noDisp}

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 259: 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 LEKTR an das der LAN-Adapter des Konnektors angeschlossen ist. Der Konnektor MUSS gewährleisten, das das Netzwerksegment NICHT mit einem der folgenden Netzwerksegmente überlappt: 1. NET_TI_DEZENTRAL 2. NET_TI_ZENTRAL 3. NET_TI_OFFENE_FD

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
		4. NET_TI_GESICHERTE_FD 5. NET_SIS 6. ANLW_BESTANDSNETZE 7. ANLW_AKTIVE_BESTANDSNETZE 8. ANLW_LEKTR_INTRANET_ROUTES

Der Administrator des Konnektor MUSS die Werte der folgenden Tabelle über die Managementschnittstelle setzen können.

Tabelle 260: 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, das der konfigurierte Wert in den Grenzen von 576 bis 9000 liegt. Default-Wert: 1500
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.

⌫

⌦ **TIP1-A_4760 Konfiguration WAN-Interface**

Der Konnektor MUSS gewährleisten, dass die Konfiguration nur dann gespeichert wird, wenn alle Parameter der nachfolgenden Tabellen den dazugehörigen Bedingungen entsprechen.

Wenn die Konfiguration per Managementschnittstelle geändert wurde, MUSS das folgende Systemereignis ausgelöst werden: TUC_KON_256 {"ANLW/WAN/IP_CHANGED"; Op; Info;IP=$dieNeueIP; noDisp}

Wenn (DHCP_CLIENT_WAN_STATE=Disabled) gesetzt ist, MUSS der Administrator des Konnektors die Werte der folgenden Tabelle über die Managementschnittstelle setzen können.

Wenn (DHCP_CLIENT_WAN_STATE=Enabled) gesetzt ist, MUSS der Administrator des Konnektors die Werte der folgenden Tabelle angezeigt bekommen, kann diese jedoch nicht ändern.

Tabelle 261: 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. Nur wenn DHCP_CLIENT_WAN_STATE=Disabled und

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
		ANLW_WAN_ADAPTER_MODUS=ENABLED MUSS der Administrator die WAN-seitige IP-Adresse des Konnektors setzen können. 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. Der Administrator MUSS die Subnetzmaske setzen können. 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. Der Wert bezeichnet ein lokales Netzwerk in der Umgebung des LEKTR an das der WAN-Adapter des Konnektors angeschlossen ist. Der Konnektor MUSS gewährleisten, dass das Netzwerksegment nicht mit einem der folgenden Netzwerksegmente überlappt: 1. NET_TI_DEZENTRAL 2. NET_TI_ZENTRAL 3. NET_TI_OFFENE_FD 4. NET_TI_GESICHERTE_FD 5. NET_SIS 6. ANLW_ BESTANDSNETZE 7. ANLW_AKTIVE_BESTANDSNETZE 8. ANLW_LAN_NETWORK_SEGMENT 9. ANLW_LEKTR_INTRANET_ROUTES

Der Administrator des Konnektor MUSS die Werte der folgenden Tabelle über die Managementschnittstelle setzen können.

Tabelle 262: 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. Der Konnektor MUSS sicherstellen, das der konfigurierte Wert in den Grenzen von 576 bis 9000 liegt. Default-Wert: 1500
ANLW_WAN_PARAMETER	Liste von IP, UDP	Der Administrator SOLL weitere

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
	und/oder TCP Parametern	Konfigurationsparameter gemäß [gemSpec_Net#2.2.2.1,2.5] konfigurieren können.

⌫

⌦ TIP1-A_4761 Konfiguration Anbindung LAN/WAN

Die Managementschnittstelle MUSS es einem Administrator ermöglichen Konfigurationsänderungen gemäß Tabelle 263: 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 263: 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 beim LE / KTR geschaltet. Wenn ANLW_WAN_ADAPTER_MODUS= ENABLED befindet sich der Konnektor in diesem Anbindungsmodus. Der Administrator MUSS diesen Wert einsehen und DARF ihn NICHT ändern können.
	Parallel	Der Konnektor ist parallel (zu allen bestehenden Systemen) ins Netzwerk des LE / KTR angebunden. Wenn ANLW_WAN_ADAPTER_MODUS= DISABLED befindet sich der Konnektor in diesem Anbindungsmodus. 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.

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
		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_INTRANET_ROUTES_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_ADAPTER_MODUS	ENABLED	Dieser Parameter ändert den Interface-Status des WAN-Adapters. Der Administrator MUSS diesen Wert einsehen können. Der Administrator MUSS diesen Wert ändern können.
	DISABLED	Dieser Parameter ändert den Interface-Status des WAN-Adapters. Der Administrator MUSS diesen Wert einsehen können. Der Administrator MUSS diesen Wert ändern können.
ANLW_LEKTR_INTRANET_ROUTES	Tupel aus Netzwerksegment und dazugehörigem Next-Hop	Der Administrator MUSS in diese Liste Einträge hinzufügen, editieren und löschen können. Liste von Routen zur Erreichung der Clientsysteme und Kartenterminals vom Konnektor; jeweils mit IP-Netzwerk dazugehörigem Next Hop. Die Netzwerksegmente DÜRFEN NICHT mit den Netzbereichen - NET_SIS - NET_TI_DEZENTRAL - NET_TI_ZENTRAL - NET_TI_OFFENE_FD - NET_TI_GESICHERTE_FD

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
		- ANLW_BESTANDSNETZE kollidieren.
ANLW_SERVICE_TIMEOUT	X Sekunden	Der Administrator MUSS die maximale Zeit konfigurieren können, in der ein Service antworten muss, bevor das System einen Timout-Fehler meldet. Default-Wert: 60 Sekunden
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 Bestandsnetzen (gemäß TUC_KON_283 „Infrastruktur Konfiguration aktualisieren“) einzelne freischalten können. Nur die freigegeben Bestandsnetze werden in dieser Variablen erfasst. Nur die freigeschalteten Bestandsnetze sind aus den LE/KTR-Netzwerken erreichbar. Wird eine Änderung an der Liste der aktiven Bestandsnetze vorgenommen, so MUSS der Konnektor für jedes aktive Bestandsnetz 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 „nicht freigeschaltete“ oder zwischenzeitlich gelöschte Bestandsnetze DARF der Konnektor NICHT Referer-Einträge in DNS_SERVERS_BESTANDSNETZE enthalten. Die Einträge in

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
		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.

⌫

⌦ **TIP1-A_5537 Anzeige IP-Routinginformationen**

Der Konnektor MUSS über die Managmentschnittstelle 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. ⌫

⌦ **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 264: 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.
ANLW_FW_SECURITY_LOGGING	Enabled / Disabled	Der Administrator des Konnektor MUSS das erweiterte Logging der Firewall zuschalten können. Default-Wert: Disabled

⌫

4.2.2 DHCP-Server

Innerhalb des Kapitels DHCP-Servers werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): „DHCP“
- Konfigurationsparameter: „DHCP_SERVER_“

4.2.2.1 Funktionsmerkmalweite Aspekte

⌦ TIP1-A_4763 DHCP-Server des Konnektors

Der Konnektor MUSS an seiner LAN-Schnittstelle einen DHCP-Server gemäß [RFC2131] und [RFC2132] anbieten. ⌫

4.2.2.2 Durch Ereignisse ausgelöste Reaktionen

Keine.

4.2.2.3 Interne TUCs, nicht durch Fachmodule nutzbar

Keine.

4.2.2.4 Interne TUCs, auch durch Fachmodule nutzbar

Keine.

4.2.2.5 Operationen an der Außenschnittstelle

4.2.2.5.1 Liefere Netzwerkinformationen über DHCP

⌦ TIP1-A_4765 Liefere Netzwerkinformationen über DHCP

Der DHCP-Server des Konnektors MUSS an der Client-Schnittstelle eine Operation zur Lieferung von Netzwerkinformationen über DHCP anbieten**.**

Tabelle 265 TAB_KON_626 "Liefere Netzwerkinformationen über DHCP"

Name	Liefere Netzwerkinformationen über DHCP
Beschreibung	Der Konnektor MUSS anfragenden Clients per DHCP die konfigurierten Netwerkinformationen liefern (siehe Tabelle 267 und Tabelle 268).
Aufrufparameter	gemäß [RFC2131], [RFC2132]
Rückgabe	gemäß [RFC2131], [RFC2132]
Standardablauf	Die an den aufrufenden Client zu übergebenden Parameter ergeben sich aus Tabelle 267 und Tabelle 268: Falls DHCP_SERVER_STATE = Enabled: • Anhand der MAC-Adresse des anfragenden Client wird die Clientgruppe aus DHCP_SERVER_CLIENTGROUPS bzw. DHCP_SERVER_DEFAULT_CLIENTGROUP ausgewählt. • DHCP_OWNDNS_ENABLED o Enabled: DNS-Server = <konnektoreigene Adresse>

Name	Liefere Netzwerkinformationen über DHCP
	o Disabled: DNS-Server = DHCP_DNS_ADDR • DHCP_NTP o Enabled: NTP-Server = <konnektoreigene Adresse> o Disabled: Keine Wertübermittlung • DHCP_OWNDGW_ENABLED o Enabled: DGW = <konnektoreigene Adresse> o Disabled: DGW = DHCP_DGW_ADDR • Falls Client-MAC-Adresse in DHCP_STATIC_LEASE o IP_Address = die in der Static Lease konfigurierte Adresse. o Sonst: IP_Address = IP_Address aus DHCP_IP_RANGE • Netzmaske = DHCP_IP_NETMASK • Domainname = DHCP_DOMAINNAME • Hostname = DHCP_HOSTNAME • Lease Dauer = DHCP_LEASE_TTL • Routen bestehend aus o DHCP_AKTIVE_BESTANDSNETZE_ROUTES o DHCP_INTRANET_ROUTES o DHCP_ROUTES • Weitere DHCP-Optionen = DHCP_OPTIONS • MTU = ANLW_LAN_MTU
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

⌫

4.2.2.6 Betriebsaspekte

⌦ **TIP1-A_4766 Deaktivierbarkeit des DHCP-Servers**

Der DHCP- Server des Konnektors MUSS durch den Administrator über die Managementschnittstelle aktivierbar und deaktivierbar sein (gemäß TAB_KON_627). Der DHCP-Server MUSS bei der Auslieferung deaktiviert sein.

Bei der Aktivierung MUSS der Konnektor den TUC_KON_343 "Initialisierung DHCP-Server" durchlaufen.

Sobald DHCP_SERVER_STATE geändert wurde, muss TUC_KON_256{"DHCP/SERVER/STATECHANGED"; Op; Info; "STATE=$DHCP_SERVER_STATE "} aufgerufen werden.

Tabelle 266 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.

⌫

⌦ **TIP1-A_4767 Konfiguration des DHCP-Servers**

Der Konnektor MUSS die Möglichkeit bieten die in Tabelle 267 und Tabelle 268 beschriebenen Parameter des DHCP-Servers über die Managementschnittstelle zu konfigurieren.

Tabelle 267 TAB_KON_628 "Basiskonfiguration des DHCP-Servers"

Referenz ID	Belegung	Bedeutung
DHCP_SERVER_NETWORK	IP-Adresse	IP-Netzwerk des LE.
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; 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.

Tabelle 268 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. 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 des LE erreichbaren DNS-Server als Parameter übergeben werden. Der Administrator MUSS diese Adressen konfigurieren können.
DHCP_NTP	Enabled /	Der Administrator MUSS konfigurieren

ReferenzID	Belegung	Bedeutung
	Disabled	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_LEASE_TTL	X Minuten	Der Administrator MUSS Lease-Dauer der dynamischen Adressen konfigurieren können.
DHCP_AKTIVE_BESTANDSNETZE_ROUTES	Liste von Tupel: `Netzwerksegment je INTRANET und Adresse für Next Hop je` freigegebenem Bestandsnetz	Der Administrator MUSS je freigegebenem Bestandsnetz (aus ANLW_AKTIVE_BESTANDSNETZE) den an den Client zu übermittelnden Routen-Eintrag aktivieren oder deaktivieren können.
DHCP_INTRANET_ROUTES	Liste von Tupel: Netzwerksegment je INTRANET und Adresse für Next Hop in die definierten 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_ROUTES	Tupel Netzwerksegment und Adresse für 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: - dem Netzwerksegment

ReferenzID	Belegung	Bedeutung
		ANLW_LAN_NETWORK_SEGMENT - dem Netzwerksegment ANLW_WAN_NETWORK_SEGMENT - jedes Netzsegmente in ANLW_BESTANDSNETZE ANLW_AKTIVE_BESTANDSNETZE ANLW_LEKTR_INTRANET_ROUTES Die Routen SOLLEN über DHCP Option 121 (Windows Vista oder höher) bzw. DHCP Option 249 (Windows XP und darunter) verteilt werden.
DHCP_OPTIONS	Liste an weiteren DHCP-Optionen.	Vom Administrator konfigurierbare Liste an weiteren DHCP-Options gemäß [RFC2132]

⌫

4.2.2.6.1 TUC_KON_343 „Initialisierung DHCP-Server“

⌦ **TIP1-A_4768 TUC_KON_343 „Initialisierung DHCP-Server“**

Der Konnektor MUSS in der Bootup-Phase TUC_KON_343 "Initialisierung DHCP-Server" durchlaufen.

Tabelle 269 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

Element	Beschreibung
Standardablauf	Falls DHCP_SERVER_STATE = enabled - den DHCP-Server starten Falls DHCP_SERVER_STATE = disabled - den DHCP-Server stoppen
Varianten/Alternativen	Keine
Fehlerfälle	4168: DHCP-Server konnte nicht gestartet werden
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 270: TAB_KON_631 Übersicht Fehler TUC_KON_343 "Initialisierung DHCP-Server"

Fehlercode	ErrorType	Severity	Fehlertext
4168	Technical	Error	Der DHCP-Server des Konnektors konnte nicht gestartet werden.

⌫

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 muss dem Interface zugewiesen und in der Variable ANLW_LAN_IP_ADDRESS bzw. ANLW_WAN_IP_ADDRESS gespeichert 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 271 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 des LE konfigurieren können.
Anwendungsumfeld	Netzwerkkonfiguration für den Betrieb des Konnektors
Eingangsanforderung	Der Konnektor muss zur Netzwerk-Interface-Konfiguration DHCP nutzen sofern keine statischen Informationen vorhanden sind.
Auslöser und Vorbedingungen	Bootup, Ablauf einer DHCP-Lease, manuell angestoßenes DHCP-Renew, Aktivierung der DHCP-Client-Funktionalität.
Eingangsdaten	DHCP-Informationen aus dem Netz des LE
Komponenten	Konnektor
Ausgangsdaten	Keine
Standardablauf	• Falls DHCP_CLIENT_LAN_STATE = Enabled o Vgl. [RFC2131], [RFC2132] o Nur die Werte gemäß Tabelle 259: TAB_KON_683 LAN-Adapter IP-Konfiguration dürfen belegt werden o Aufruf TUC_KON_256{"DHCP/LAN_CLIENT/RENEW"; Op; Info; "IP_ADDRESS=$Belegung"} • Falls DHCP_CLIENT_WAN_STATE = Enabled o Vgl. [RFC2131], [RFC2132] o Nur die Werte gemäß Tabelle 261: TAB_KON_685 WAN-Adapter IP-Konfiguration dürfen belegt werden o Aufruf TUC_KON_256{"DHCP/WAN_CLIENT/RENEW"; Op; Info; "IP_ADDRESS=$Belegung"}.
Varianten/Alternativen	Keine

Element	Beschreibung
Fehlerfälle	4169: Konnektor erhält keine DHCP-Informationen 4170: Konnektor besitzt identische IP-Adressen am WAN- und LAN-Interface
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 272: TAB_KON_633 Übersicht Fehler 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

⌫

4.2.3.4 Interne TUCs, auch durch Fachmodule nutzbar

Keine.

4.2.3.5 Operationen an der Außenschnittstelle

Keine.

4.2.3.6 Betriebsaspekte

⌦ TIP1-A_4773 Konfiguration des DHCP-Clients

Die DHCP-Client Funktionalität MUSS für LAN- und WAN-Interface vom Administrator getrennt aktivierbar und deaktivierbar sein (gemäß TAB_KON_634). Falls der DHCP-Client nicht verwendet wird MUSS sichergestellt werden, dass eine statische Konfiguration, für den LAN-Adapter gemäß Tabelle 259: TAB_KON_683 LAN-Adapter IP-Konfiguration bzw. für den WAN-Adapter gemäß Tabelle 261: TAB_KON_685 WAN-Adapter IP-Konfiguration, existiert bevor die Netzwerkeinstellungen übernommen werden.

Sobald Parameter geändert wurden, MUSS TUC_KON_256 „Systemereignis absetzen“ je nachdem auf welchem Interface der Client aktiviert oder deaktiviert wurde mit folgenden Parameter aufgerufen werden:

TUC_KON_256{"DHCP/LAN_CLIENT/STATECHANGED"; Op; Info;
"STATE=$DHCP_CLIENT_LAN_STATE"}

oder

TUC_KON_256{"DHCP/WAN_CLIENT/STATECHANGED "; Op; Info;
"STATE=$DHCP_CLIENT_WAN_STATE "}

Tabelle 273 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.

⌫

⌦ **TIP1-A_4774 Manuelles anstoßen eines DHCP-Lease-Renew**

Der Administrator MUSS die Möglichkeit haben die DHCP-Lease des Konnektors für jedes Interface getrennt zu erneuern. ⌫

⌦ **TIP1-A_4775 Aktive DHCP-Clients bei Auslieferung des Konnektors**

Der DHCP-Client des Konnektors auf WAN- und LAN-Schnittstelle MUSS bei Auslieferung aktiviert sein.⌫

⌦ **TIP1-A_4776 Setzen der IP-Adresse nach Timeout**

Falls der DHCP-Client auf der LAN-Seite nach einem Timeout von 30s keine IP-Adresse bezogen hat, MUSS gemäß [RFC3927] eine Default-Adresse aus 169.254/16 vergeben werden.⌫

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 Konzentrator 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-Konzentrator 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
- Event MGM/LU_CHANGED/LU_ONLINE⌫

⌦ 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)⌫

⌦ TIP1-A_4782 SIS VPN-Client Stop Events

Beim Auftreten einer der nachfolgenden Events MUSS der Konnektor den VPN-Tunnel zum SIS beenden:

- MGM/LOGICAL_SEP_CHANGED mit (Active=Enabled)

- MGM/LU_CHANGED/LU_ONLINE mit (Active=Disabled)⌫

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 274: 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 TUC_KON_305 „LAN-Adapter initialisieren“ TUC_KON_306 „WAN-Adapter initialisieren“ Event MGM/LU_CHANGED/LU_ONLINE Event NETWORK/VPN/CONFIG_CHANGED 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. • Innere Tunnel IP-Adresse des VPN-Konzentrators TI • DNS_SERVERS_TI • NTP_SERVER_ADDR • CERT_OCSP_FORWARDER_ADDRESS • CERT_OCSP_FORWARDER_PORT • MGM_KSR_KONFIG_URL • MGM_KSR_FIRMWARE_URL • VPN_KONZENTRATOR_TI_IP_ADDRESS

Element	Beschreibung
Standardablauf	1) Wenn der Auslöser = Event NETWORK/VPN/CONFIG_CHANGED ist, muss der VPN-Tunnel TI abgebaut werden. 2) Wenn der VPN-Tunnel TI noch aktiv ist, ist der Ablauf abgeschlossen. Anderenfalls ist mit Ablaufschritt 3) fortzufahren. 3) Prüfen, MGM_LU_ONLINE = Enabled, falls nicht ist der TUC ohne Ausgabe einer Fehlermeldung zu beenden. 4) Prüfen, ob die im Konnektor hinterlegte CRL noch gültig ist. falls nicht, muss der TUC_KON_040 „CRL aktualisieren“ aufgerufen werden, Anschließend erneut prüfen, ob die im Konnektor hinterlegte CRL nun gültig ist. Falls die CRL noch nicht gültig ist, ist der TUC mit Fehler zu beenden. 5) Aufrufen von TUC_VPN-ZD_0001 “IPsec Tunnel TI aufbauen” Die folgenden Rückgabewerte des TUC_VPN-ZD_0001 “IPsec Tunnel TI aufbauen” sind in die laufende Konfiguration des Konnektors zu übernehmen: • VPN_TUNNEL_TI_INNER_IP • DNS_SERVERS_TI • NTP_SERVER_ADDR • CERT_OCSP_FORWARDER_ADDRESS • CERT_OCSP_FORWARDER_PORT • MGM_KSR_KONFIG_URL • MGM_KSR_FIRMWARE_URL • VPN_KONZENTRATOR_TI_IP_ADDRESS 6) Aufrufen von TUC_KON_304 „Netzwerk-Routen einrichten“ Sobald der Tunnel erfolgreich aufgebaut wurde, ist der folgende Event zu generieren: TUC_KON_256 {"NETWORK/VPN_TI/UP"; Op; Info;IP= $VPN_TUNNEL_TI_INNER_IP}
Varianten / Alternativen	Keine
Fehlerfälle	(→3) Keine Online-Verbindung zulässig; 4172 (→4) CRL ist Abgelaufen (outdated); 4173 (→5) VPN-Tunnel konnte nicht aufgebaut werden; Fehlercode: 4174
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 275: TAB_KON_636 Übersicht Fehler TUC_KON_321

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
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: Verbindung konnte nicht aufgebaut werden

⌫

4.2.4.3.2 TUC_KON_322 „Verbindung zu dem VPN-Konzentrator des SIS aufbauen“

⌦ **TIP1-A_4784 TUC_KON_322 „Verbindung zu dem VPN-Konzentrator des SIS aufbauen“**

Der Konnektor MUSS den technischen Use Case TUC_KON_322 „Verbindung zu dem VPN-Konzentrator der SIS aufbauen“ umsetzen.

Tabelle 276: TAB_KON_637 - TUC_KON_322 „Verbindung zu dem VPN-Konzentrator der 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 TUC_KON_305 „LAN-Adapter initialisieren“ TUC_KON_306 „WAN-Adapter initialisieren“ Event MGM/LU_CHANGED/LU_ONLINE Event NETWORK/VPN/CONFIG_CHANGED Manueller Aufruf über Managementschnittstelle
Vorbedingungen	ANLW_INTERNET_MODUS = SIS Die Verbindung VPN-Konzentrator TI ist aufgebaut 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.

Element	Beschreibung
	• Innere Tunnel-IP-Adresse des VPN-Konzentrators SIS • VPN_KONZENTRATOR_SIS_IP_ADDRESS • DNS_SERVER_SIS
Standardablauf	1) Wenn der Auslöser Event NETWORK/VPN/CONFIG_CHANGED ist, muss der VPN-Tunnel SIS abgebaut werden. 2) Wenn der VPN-Tunnel SIS noch aktiv ist, ist der Ablauf abgeschlossen. Anderenfalls ist mit Ablaufschritt 3) fortzufahren. 3) Prüfen, ob (MGM_LU_ONLINE=Enabled). falls nicht ist der TUC ohne Ausgabe einer Fehlermeldung zu beenden. 4) Prüfen, ob (MGM_LOGICAL_SEPARATION=Disabled), falls nicht ist der TUC mit einer Fehlermeldung zu beenden. 5) Prüfen, ob (ANLW_INTERNET_MODUS=SIS) gesetzt ist, falls nicht ist der TUC zu beenden. 6) Prüfen, ob die im Konnektor hinterlegte CRL noch gültig ist. falls nicht, MUSS der TUC_KON_040 „CRL aktualisieren“ aufgerufen werden, Anschließend erneut prüfen, ob die im Konnektor hinterlegte CRL nun gültig ist. Falls die CRL noch nicht gültig ist, ist der TUC mit Fehler zu beenden. 7) Aufrufen von TUC_VPN-ZD_0002 “IPsec Tunnel SIS aufbauen” 8) Aufrufen von TUC_KON_304 „Netzwerk-Routen einrichten“ Sobald der Tunnel erfolgreich aufgebaut wurde, ist der folgende Event zu generieren: TUC_KON_256 {"NETWORK/VPN_SIS/UP"; Op; Info;IP= $VPN_TUNNEL_SIS_INNER_IP}
Varianten / Alternativen	Keine
Fehlerfälle	(→ 3) Keine Online-Verbindung zulässig; 4172 (→ 4) Keine Online-Verbindung zulässig; 4172 (→ 5) Keine Online-Verbindung zulässig; 4172 (→ 6) CRL ist Abgelaufen (outdated);4173 (→ 7) VPN Tunnel konnte nicht aufgebaut werden; Fehlercode: 4176
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 277: TAB_KON_638 Übersicht Fehler TUC_KON_322

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
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: Verbindung konnte nicht aufgebaut werden

⌫

4.2.4.4 Interne TUCs, auch durch Fachmodule nutzbar

Keine

4.2.4.5 Operationen an der Außenschnittstelle

Keine

4.2.4.6 Betriebsaspekte

⌦ **TIP1-A_5415 Initialisierung „VPN-Client“**

Der Konnektor MUSS in der Bootup-Phase zur Initialisierung des Funktionsmerkmals „VPN-Client“:

- die Verbindung zum VPN-Konzentrator TI aufbauen (TUC_KON_321)
- die Verbindung zum VPN-Konzentrator SIS aufbauen (TUC_KON_322)⌫

⌦ **TIP1-A_4785 Konfigurationsparameter VPN-Client**

Die Managementschnittstelle MUSS es einem Administrator ermöglichen Konfigurationsänderungen am VPN-Client gemäß Tabelle TAB_KON_639 vorzunehmen.

Der Konnektor MUSS bei einer Änderung der Konfigurationswerte

den folgenden Event auslösen:

Rufe TUC_KON_256 {"NETWORK/VPN/CONFIG_CHANGED"; Op; Info;; noDisp}

Tabelle 278: TAB_KON_639 - Konfigurationsparameter VPN-Client

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
IKE_KEEPALIVE_MODUS	Enabled/Disabled	Der Administrator MUSS einstellen können, ob IKE Keep-Alive-Pakete gesendet werden. Ein Hinweis MUSS ausgegeben werden, dass dies bei Nutzung von Dial-Up-Verbindungen nicht zu empfehlen ist. Dies dient der Vermeidung von Kosten für LE bei Nutzung

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
		eines Internetzugangs ohne Flatrate. Default-Wert: Enabled
IKE_KEEPALIVE_INTERVAL	X Sekunden	Der Administrator MUSS die Zeit in Sekunden angeben können, nach der ein neues IKE Keep-Alive-Paket gesendet wird. Default-Wert: 30
IKE_KEEPALIVE_RETRY	X	Der Administrator MUSS angeben können, nach wie vielen IKE Keep-Alive-Paketen ohne Acknowledge Message die Verbindung beendet wird. Default-Wert: 3
VPN_IDLE_TIMEOUT_MODUS	Enabled/Disabled	Der Administrator MUSS einstellen können, ob nach Inaktivität die VPN-Verbindung automatisch abgebaut werden soll. Ein Hinweis MUSS ausgegeben werden, dass dies insbesondere bei Nutzung von Dial-Up-Verbindungen Enabled werden sollte. 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. Default-Wert: 600
NAT_KEEPALIVE_MODUS	Enabled/Disabled	Der Administrator MUSS einstellen können, ob NAT Keep-Alive-Pakete gesendet werden. Ein Hinweis MUSS ausgegeben werden, dass dies insbesondere bei Nutzung von Dial-Up-Verbindungen nicht zu empfehlen ist. Default-Wert: Enabled
NAT_KEEPALIVE_INTERVAL	X Sekunden	Der Administrator MUSS die Zeit in Sekunden angeben können, nach der ein neues NAT Keep-Alive-Paket gesendet wird. 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_ADDRESS	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. Default-Wert: 1418
VPN_SIS_MTU	Paketgröße in Byte	Der Administrator MUSS die MTU für ESP Pakete zum SIS (excl. ESP-Header-Size) in

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
		den Grenzen von 576 bis 8076 konfigurieren können. Default-Wert: 1418
HASH_AND_URL	Enabled/Disabled	Der Administrator MUSS die Nutzung des hash&URL-Verfahrens zum Zertifikatsaustausch konfigurieren können. 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_INT>“ ermittelt. Default-Wert: Disabled

⌫

4.2.5 Zeitdienst

Der Zeitdienst schafft die Grundlage einer gleichen Systemzeit für alle in der TI einzusetzenden Produkttypen. Grundsätzlich ist ein NTP-Server der Stratum-3-Ebene innerhalb des Konnektors erforderlich, welcher die Zeitangaben eines NTP-Servers Stratum-2-Ebene abfragt (GS-A_3942). Die in [gemSpec_Net#5.1] „NTP-Topologie“ getroffenen Anforderungen werden durch dieses Kapitel erweitert.

Innerhalb des Zeitdienstes werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): „NTP“
- Konfigurationsparameter: „NTP_“

4.2.5.1 Funktionsmerkmalweite Aspekte

⌦ **TIP1-A_4786 Maximale Zeitabweichung**

Falls der Leistungsumfang Online nicht aktiviert ist (MGM_LU_ONLINE=Disabled), MUSS sichergestellt werden, dass der maximale zulässige Fehler von +/- 20ppm (part per million) gegenüber einer Referenzuhr nicht überschritten wird. Dies entspricht einer maximalen Abweichung im Freilauf von +/- 34,56 Sekunden über 20 Tage.⌫

⌦ **TIP1-A_4787 Konfigurationsabhängige Funktionsweise**

Der NTP-Server des Konnektors MUSS deaktiviert sein, falls der Konnektor Leistungsumfang Online nicht aktiviert ist (MGM_LU_ONLINE=Disabled.⌫

Falls die Systemzeit des Konnektors zu stark von der Zeit der zentralen TI-Plattform abweicht, deutet dies auf ein schwerwiegendes Problem im Konnektor oder der Umgebung hin, da dies im ordnungsgemäßen Betrieb nicht auftreten sollte.

⌦ **TIP1-A_4788 Verhalten bei Abweichung zwischen lokaler Zeit und erhaltenen Zeit**

Der Konnektor DARF die im Konnektor vorgehaltene Systemzeit im Rahmen einer automatisierten Synchronisation NICHT aktualisieren, wenn die lokale Zeit von der

im Rahmen der Synchronisation erhaltenen Zeit um mehr als NTP_MAX_TIME-DIFFERENCE abweicht. Dies betrifft NICHT Änderungen in der Darstellung der Systemzeit, die zeitzonenbedingt sind (MEZ -> MESZ -> MEZ), da die Zeitsynchronisation grundsätzlich UTC berücksichtigt. Bei einer erstmaligen Synchronisierung nach dem Boot-Vorgang oder bei einer erstmaligen Synchronisierung bei der Inbetriebnahme des Konnektors darf eine Synchronisation trotz einer Zeitabweichung größer einer Stunde durchgeführt werden. Daher MUSS der Konnektor bei einer Abweichung von mehr als einer Stunde in den kritischen Betriebszustand EC_TIME_DIFFERENCE_INTOLERABLE übergehen, ein weiterer fachlicher Betrieb des Konnektors DARF NICHT mehr erfolgen.⌫

Der kritische Betriebszustand kann anschließend über einen manuellen Eingriff (z. B. Reboot) behoben werden (siehe 3.3 Betriebszustand).

⌦ **TIP1-A_4789 Zustandsvariablen des Konnektor Zeitdiensts**

TAB_KON_640 listet die zu verwendenden Zustandsvariablen des Konnektor NTP-Servers. Diese Werte DÜRFEN NICHT durch den Administrator geändert werden.

Tabelle 279 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. Dieser Parameter wirkt nur bei MGM_LU_ONLINE = Enabled.
NTP_MAX_TIMEDIFFERENCE	3600	Maximale Zeitabweichung in Sekunden zwischen Systemzeit und Zeit des Stratum-2-Zeitservers zum Zeitpunkt der Zeitsynchronisierung. Dieser Parameter wirkt nur bei MGM_LU_ONLINE = Enabled.

⌫

4.2.5.2 Durch Ereignisse ausgelöste Reaktionen

Keine.

4.2.5.3 Interne TUCs, nicht durch Fachmodule nutzbar

Keine.

4.2.5.4 Interne TUCs, auch durch Fachmodule nutzbar

4.2.5.4.1 TUC_KON_351 Liefere Systemzeit

⌦ **TIP1-A_4790 TUC_KON_351 „Liefere Systemzeit“**

Der Konnektor MUSS den technischen Use Case TUC_KON_351 „Liefere Systemzeit“ umsetzen.

Tabelle 280 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

Tabelle 281: TAB_KON_641 Übersicht Fehler TUC_KON_351 "Liefere Systemzeit"

Fehlercode	ErrorType	Severity	Fehlertext
4178	Technical	Error	Das Fachmodul konnte die aktuelle Systemzeit des Konnektors nicht abrufen

⌫

4.2.5.5 Operationen an der Außenschnittstelle

4.2.5.5.1 Sync_Time

TIP1-A_4791 Operation sync_Time

Der NTP-Server des Konnektors MUSS an der Client-Schnittstelle eine Operation sync_Time anbieten.

Tabelle 282 TAB_KON_642 Operation sync_Time

Name	I_NTP_Time_Information:sync_Time
Beschreibung	Der Konnektor MUSS anfragenden Clients (z.B. Ärztearbeitsplatz) per NTP-Version 4 die Systemzeit liefern
Aufrufparameter	Vgl. [NTPv4]
Rückgabe	Vgl. [NTPv4]
Vorbedingungen	MGM_LU_ONLINE=Enabled und MGM_LOGICAL_SEPARATION=Disabled
Nachbedingungen	Der anfragende Client hat die korrekte Zeit geliefert bekommen.
Hinweise	Keine
Fehler	Der Aufruf schlägt fehl (bleibt unbeantwortet), wenn MGM_LU_ONLINE=Disabled oder MGM_LOGICAL_SEPARATION=Enabled

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.

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 283 TAB_KON_643 Konfiguration des Konnektor NTP-Servers

ReferenzID	Belegung	Bedeutung
NTP_TIMEZONE	Zeitzone	Der Administrator MUSS die Zeitzone des Konnektors einstellen können. Default-Wert: Central European Time/Mitteleuropäische Zeit (CET/MEZ)
NTP_TIME	Zeit	Der Administrator MUSS die Zeit des Konnektors

ReferenzID	Belegung	Bedeutung
		(NTP_TIME) über die Managementschnittstelle manuell einstellen können.

Tabelle 284 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.

⌫

⌦ **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.⌫

4.2.5.6.1 TUC_KON_352 Initialisierung Zeitdienst

⌦ **TIP1-A_4795 TUC_KON_352 „Initialisierung Zeitdienst“**

Der Konnektor MUSS in der Bootup-Phase TUC_KON_352 "Initialisierung Zeitdienst" durchlaufen.

Tabelle 285 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 und Vorbedingungen	Bootup Verbindung zum VPN-Konzentrator TI muss aufgebaut sein
Eingangsdaten	NTP-Server der zentralen TI-Plattform
Komponenten	Konnektor
Ausgangsdaten	Keine

Element	Beschreibung
Standardablauf	• Falls MGM_LU_ONLINE=Enabled: o Durch eine DNS-Anfrage an den DNS-Forwarder zur Auflösung des SRV-RR mit dem Bezeichner "_ntp._udp.<DOMAIN_SRVZONE_TI>“ erhält der Konnektor Adressen der NTP-Server der zentralen TI-Plattform. o gemäß [NTPv4] o Falls nach ANLW_SERVICE_TIMEOUT keine Antwort erfolgt ist oder falls der Zeitserver nicht erreichbar ist, wird Fehler 4177 ausgelöst
Varianten/Alternativen	Keine
Fehlerfälle	4177: Der NTP-Server des Konnektors empfängt keine Systemzeit
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 286: TAB_KON_645 Übersicht Fehler TUC_KON_352 "Initialisierung Zeitdienst "

Fehlercode	ErrorType	Severity	Fehlertext
4177	Technical	Warning	Der NTP-Server des Konnektors konnte nicht synchronisiert werden.

⌫

4.2.6 Namensdienst und Dienstlokalisierung

Innerhalb des Namensdienstes werden folgende Präfixe für Bezeichner verwendet:

- Events (Topic Ebene 1): *keine Events vorhanden*
- Konfigurationsparameter: „DNS_“

4.2.6.1 Funktionsmerkmalseite Aspekte

⌦ **TIP1-A_4796 Grundlagen des Namensdienstes**

Der Konnektor MUSS einen Recursive Caching Nameserver zur Auflösung von DNS-Anfragen bereitstellen.

Der Nameserver des Konnektors MUSS für Clientsysteme aus dem lokalen Netzwerk (ANLW_LAN_NETWORK_SEGMENT oder ANLW_LEKTR_INTRANET_ROUTES) erreichbar sein.

Der Nameserver des Konnektor MUSS DNS-Zonetransfers unterbinden.

Der Nameserver des Konnektors MUSS Anfragen aus dem Internet, der TI, den Bestandsnetzen oder dem SIS unterbinden.

Der Nameserver des Konnektors MUSS einen Timeout von max. ANLW_SERVICE_TIMEOUT Sekunden beachten. Konnte in dieser Zeit eine DNS-Abfrage nicht durchgeführt werden, MUSS die Bearbeitung abgebrochen werden.

Der Nameserver des Konnektors MUSS DNS-Abfragen immer mit gesetztem CD-Bit senden.

Der Nameserver des Konnektors MUSS als Validating Resolver konfiguriert sein und DNS-Anfragen wenn möglich (RRSIG Resource Record und ZSK sind für angefragte Daten verfügbar) gemäß DNSSEC Protokoll validieren.⌫

⌦ **TIP1-A_4797 DNS-Forwards des DNS-Servers**

Der DNS-Server des Konnektor MUSS die folgenden DNS-Forwards durchführen:

Tabelle 287: TAB_KON_687 DNS-Forwards des DNS-Servers

Domain	Liste der DNS-Server/Liste der DNS-RR-Einträge	Bemerkungen
Namensraum TI (*.DNS_TOP_LEVEL_D) OMAIN_TI	DNS_SERVERS_TI	Auflösung der DNS-Namen der TI, nur erlaubt, wenn MGM_LU_ONLINE=Enabled Wenn (MGM_LOGICAL_SEPARATION=Enabl ed) gesetzt ist, DARF der Konnektor diesen Namensraum NICHT zur Namensauflösung von Anfragen aus dem lokalen Netzwerk bereitstellen. Wenn (MGM_LOGICAL_SEPARATION=Enabl ed) gesetzt ist MUSS der Konnektor diesen Namensraum ausschließlich für interne Dienste und die internen Fachanwendungen auflösen.
Namensraum Bestandsnetze (Domainnamen von Bestandsnetzen gemäß Bestandsnetze.xml)	DNS_SERVERS_BESTAN DSNETZE (Je Domainnamen eines Bestandsnetzes alle zugehörigen DNS-Server IP-Adressen gemäß Bestandsnetze.xml)	Ausschließlich wenn (MGM_LU_ONLINE=Enabled und MGM_LOGICAL_SEPARATION=Disabl ed) MUSS der Konnektor diese Namenräume für Anfragen aus dem lokalen Netzwerk zur Namensauflösung bereitstellen.
Namensraum LEKTR (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

⌫

⌦ **TIP1-A_4798 DNS Stub-Resolver**

Der Stub-Resolver im Konnektor MUSS von allen internen Diensten zur Namensauflösung genutzt werden.

Der Stub-Resolver im Konnektor MUSS immer den DNS-Server im Konnektor anfragen. ⌫

⌦ **TIP1-A_4799 Aktualität der DNS-Vertrauensanker sicherstellen**

Der Hersteller des Konnektors MUSS gewährleisten, dass mit jedem Firmware-Update auch die jeweils aktuellen Vertrauensanker mitgeliefert werden.

Der Konnektor SOLL den DNSSEC-Vertrauensanker gemäß den DNSSEC-Standards bei der Namensermittlung aktualisieren.

Der Konnektor MUSS den DNSSEC-Vertrauensanker der TI aus dem Zertifikatspeicher in den DNS-Server übernehmen, wenn ein Fehler bei der Validierung der Namensauflösung der TI aufgetreten ist. ⌫

Für DNSSEC muss der Konnektor die Vorgaben aus [gemSpec_Krypt#3.3.3] befolgen.

4.2.6.2 Durch Ereignisse ausgelöste Reaktionen

Keine.

4.2.6.3 Interne TUCs, nicht durch Fachmodule nutzbar

Keine.

4.2.6.4 Interne TUCs, auch durch Fachmodule nutzbar

4.2.6.4.1 TUC_KON_361 „DNS-Namen auflösen“

⌦ **TIP1-A_4801 TUC_KON_361 „DNS-Namen auflösen“**

Der Konnektor MUSS den technischen Use Case TUC_KON_361 „DNS-Namen auflösen“ umsetzen.

Tabelle 288: 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

Element	Beschreibung
	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. Für alle ermittelten IPv4-Adressen und IPv6-Adressen werden als LIST_OF_IP_ADDRESSES zurückgeliefert. Da IPv6 nicht produktiv eingesetzt wird muss die aufrufende Instanz die IPv6-Adressen ignorieren. Falls keine IP-Adressen ermittelt werden konnten, wird eine leere Liste zurückgeliefert.
Varianten / Alternativen	Keine
Fehlerfälle	(→ 1) Timeout der Anfrage; Fehlercode 4179 (→ 1) DNS-Fehler; Fehlercode 4180
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 289: TAB_KON_647 Übersicht Fehler 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 abgebrochen, da der Timeout von ANLW_SERVICE_TIMEOUT Sekunden überschritten wurde.”
4180	Technical	Fatal	„DNS: Es ist ein Fehler bei der Namensauflösung aufgetreten“ Die Fehlerdetails sind gemäß DNS-Protokoll zu ergänzen.

⌫

4.2.6.4.2 TUC_KON_362 „Liste der Dienste abrufen“

⌦ **TIP1-A_4802 TUC_KON_362 „Liste der Dienste abrufen“**

Der Konnektor MUSS den technischen Use Case TUC_KON_362 „Liste der Dienste abrufen“ umsetzen.

Tabelle 290: 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-

Element	Beschreibung
	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 DNS-Server des Konnektor gestellt.
Varianten / Alternativen	Keine
Fehlerfälle	(→ 1) Timeout der Anfrage; Fehlercode 4179 (→ 1) DNS-Fehler; Fehlercode 4180
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 291: TAB_KON_649 Übersicht Fehler 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 abgebrochen, da der Timeout von ANLW_SERVICE_TIMEOUT Sekunden überschritten wurde.”
4180	Technical	Fatal	„DNS: Es ist ein Fehler bei der Namensauflösung aufgetreten“ Die Fehlerdetails sind gemäß [gemSpec_Net] zu ergänzen.

⌫

4.2.6.4.3 TUC_KON_363 „Dienstdetails abrufen“

⌦ **TIP1-A_4803 TUC_KON_363 „Dienstdetails abrufen“**

Der Konnektor MUSS den technischen Use Case TUC_KON_363 „Dienstdetails abrufen“ umsetzen.

Tabelle 292: 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

Element	Beschreibung
	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 LIST_OF_SRV_DETAILS
Standardablauf	1) Mit dem FQDN wird eine Typ-„SRV“-Anfrage an den DNS-Server des Konnektor 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. 2) Mit dem FQDN wird zusätzlich eine Typ-„TXT“-Anfrage an den DNS-Server des Konnektor gestellt. Wenn ein oder mehrere entsprechende Einträge gefunden werden konnte, 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.
Varianten / Alternativen	Keine
Fehlerfälle	(→ 1-2) Timeout der Anfrage; Fehlercode 4179 (→ 1-2) DNS Fehler; Fehlercode 4180
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 293: TAB_KON_651 Übersicht Fehler TUC_KON_363 „Dienstdetails abrufen“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			

Fehlercode	ErrorType	Severity	Fehlertext
4179	Technical	Error	„DNS: Anfrage wurde abgebrochen, da der Timeout von ANLW_SERVICE_TIMEOUT Sekunden überschritten wurde.”
4180	Technical	Fatal	„DNS: Es ist ein Fehler bei der Namensauflösung aufgetreten“ Die Fehlerdetails sind gemäß [gemSpec_Net] zu ergänzen.

⌫

4.2.6.5 Operationen an der Außenschnittstelle

⌦ **TIP1-A_4804 Basisanwendung Namensdienst**

Der Konnektor MUSS für Clients eine Basisanwendung Namensdienst anbieten.

Tabelle 294 TAB_KON_652 Basisanwendung Namensdienst

Name	Namendienst	
Version	wird im Produktsteckbrief des Konnektors definiert	
Namensraum	Keiner	
Namensraum-Kürzel	Keiner	
Operationen	Name	Kurzbeschreibung
	GetIPAddress	Diese Operation ermöglicht die Auflösung von IP-Adressen in FQDN im Namensraum der TI.
WSDL	Keines	
Schema	Keines	

⌫

4.2.6.5.1 GetIPAddress

⌦ **TIP1-A_5035 Operation GetIPAddress**

Der Namensdienst des Konnektors MUSS an der Client-Schnittstelle eine Operation GetIPAddress anbieten**.**

Tabelle 295: TAB_KON_653 Operation GetIPAddress

Name	GetIPAddress
Beschreibung	Diese Operation ermöglicht die Auflösung von FQDN in IP-Adressen. (DNS-Forwarder Abfrage ohne Cache)
Aufrufparameter	Address (FQDN) DNSSECValidation (Boolean)

Rückgabe	IPAddr (IPAddress) DNSSECValidatidated (Boolean)
Vorbedingungen	Der DNS-Server im Konnektor muss aktiv sein. Sie Forward Nameserver (DNS_SERVERS_TI, DNS_SERVERS_SIS, DNS_SERVERS_BESTANDSNETZE) müssen konfiguriert sein.
Nachbedingungen	Keine
Standardablauf	Für Details zu DNS Namensauflösung wird auf [gemSpec_Net] verweisen.

⌫

4.2.6.6 Betriebsaspekte

⌦ **TIP1-A_5416 Initialisierung „Namensdienst und Dienstlokalisierung“**

Der Konnektor MUSS in der Bootup-Phase zur Initialisierung des Funktionsmerkmals „Namensdienst und Dienstlokalisierung“:

- den DNS-Server starten.⌫

⌦ **TIP1-A_4805 Konfigurationsparameter Namensdienst und Dienstlokalisierung**

Der Administrator MUSS die in TAB_KON_654 aufgelisteten Parameter über die Managementschnittstelle konfigurieren und die in TAB_KON_731 aufgelisteten Parameter ausschließlich einsehen können.

Nach jeder Änderung MUSS sichergestellt werden, dass die Änderungen sofort am DNS-Server zur Verfügung stehen.

Tabelle 296: 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 Transportnetz (nur für den Konnektor, zum Auflösen der FQDN aller zum VPN-Tunnel-Aufbau erforderlichen Adressen) Alle genannten IP-Adressen MÜSSEN aus einem öffentlich zugänglichen Adressbereich stammen.
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 LE/KTR-Umgebung verwendet werden. Der Administrator MUSS die Liste von DNS-Servern, die die DNS_DOMAIN_LEKTR auflösen, bearbeiten können. Die IP-Adressen der DNS-Server MÜSSEN aus dem Adressbereich LAN_IP_ADDRESS

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
		stammen.
DNS_DOMAIN_LEKTR	DNS Domainname	DNS Domainname, der von einem DNS-Server beim Leistungserbringer aufgelöst wird. Der Name DARF NICHT mit einem „.“ beginnen und nicht mit einem „.“ enden.
DNS_ROOT_ANCHOR_URL	URL	Der Administrator MUSS die URL für den Download der IANA DNSSEC Root Keys (im Format http://$hostname:$port/$path oder https://$hostname:$port/$path) eintragen können. Default: http://data.iana.org/root-anchors/root-anchors.xml

Tabelle 297 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 Bestandsnetz	Liste von DNS-Servern, die zur Namensauflösung der Namensräume von Bestandsnetzen verwendet werden.
DNS_TOP_LEVEL_DOMAIN_TI	DNS Domainname	Top Level Domain des Namensraumes TI

⌫

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.

Die Verbindung zur Managementschnittstelle MUSS zur Sicherung der Vertraulichkeit, Integrität und Authentizität durch Nutzung eines kryptographischen Verfahrens gemäß [gemSpec_Krypt] abgesichert werden, falls die Sicherheit der übertragenen Daten nicht auf andere Weise erreicht wird. Die Absicherung der Daten kann z. B. durch Nutzung von TLS unter Berücksichtigung der in [gemSpec_Krypt] angegebenen Algorithmen und Schlüssellängen geschehen.

Die Managementschnittstelle MUSS in thematisch gegliederte Konfigurationsbereiche unterteilt sein. Die konkrete Gliederung selbst ist herstellerspezifisch.

Die Managementschnittstelle KANN einen Managementbereich aufweisen, der nur für autorisierte Techniker des Herstellers zugänglich ist. Ein Zugriff auf diesen Bereich MUSS durch eine eigene Authentisierungsfunktion geschützt werden (z. B. durch Passwortschutz).⌫

Die über die Managementschnittstelle zu erreichenden und zu verändernden Inhalte werden erhoben in:

- diesem Kapitel
- in allen Betriebsaspektekapiteln der Funktionsmerkmale, sowie der Übergreifenden Festlegungen
- den Fachmodulspezifikationen der Fachanwendungen (siehe Kapitel 4.3.4).
- Den übergreifenden Spezifikationen [gemSpec_Net] und [gemSpec_PKI]

Eine Ergänzung um weitere, herstellerspezifische Konfigurationsinhalte ist möglich. Eine Übersicht der Konfigurationsparameter findet sich in Anhang E.

⌦ **TIP1-A_4807 Mandantenübergreifende Managementschnittstelle**

Das Management des Konnektors MUSS über die Managementschnittstelle mandantenübergreifend erfolgen. Dies bedeutet insbesondere, dass ein Administrator (gemäß seiner Zugriffsberechtigungen) in einer Management-Session alle Einstellungen einsehen und verändern können MUSS, egal welchem Mandanten diese Werte zugeordnet sind.⌫

⌦ **TIP1-A_5658 Konnektor, rollenspezifische Endpunkte der Managementschnittstelle**

Der Konnektor MUSS die Managementschnittstelle mit zwei getrennten Endpunkten implementieren. Der Konnektor MUSS sicherstellen, dass auf den einen Endpunkt nur Nutzer mit der Rolle Lokaler-Administrator oder Super-Administrator zugreifen können, und auf den anderen Endpunkt nur Nutzer mit der Rolle Remote-Administrator.⌫

⌦ **TIP1-A_5005 Protokollierung in der Managementschnittstelle**

Jede Änderung, die ein Benutzer (Administrator) vornimmt, MUSS protokolliert werden durch TUC_KON_271 „Schreibe Protokolleintrag“ {„MGM/ADMINCHANGES“; Op; Info; „User=$AdminUsername; RefID=$ReferenzID; NewVal=$NeuEingestellterWert“}.⌫

4.3.1 Zugang und Benutzerverwaltung des Konnektormanagements

Der Konnektor verfügt über keine Verwaltung der fachlichen Nutzer, wohl aber über eine Verwaltung der Nutzer, die in der Rolle eines Administrators den Konnektor konfigurieren und die Protokolle einsehen dürfen. Dabei werden drei Administrator-Rollen unterschieden:

1. Lokaler-Administrator: zur Konfiguration des Konnektors über die lokale Managementschnittstelle
2. Remote-Administrator: zur Konfiguration des Konnektors über die remote Managementschnittstelle.
3. Super-Administrator: zur Verwaltung von Benutzerkonten und zur Konfiguration des Konnektors über die lokale Managementschnittstelle

⌦ **TIP1-A_4808 Zugangsschutz der Managementschnittstelle**

Der Konnektor MUSS sicherstellen, dass die Managementschnittstelle vor unberechtigtem Zugang geschützt ist. Die Managementschnittstelle MUSS durch eine Kombination aus Benutzername und Passwort oder einen mindestens gleich starken Mechanismus vor unberechtigtem Zugang geschützt sein.

Für die Erstellung und Verarbeitung von Passwörtern der Managementschnittstelle MÜSSEN die Empfehlungen der Grundschutz-Kataloge des BSI beachtet werden (siehe Maßnahme „M 2.11 Regelung des Passwortgebrauchs“ in [BSI_GK]).

Für die Passworterstellung MUSS der Konnektor mindestens folgende Aspekte berücksichtigen:

- dem Benutzer muss es möglich sein, die Zeichen eines Passworts aus den Zeichenklassen Großbuchstaben, Kleinbuchstaben, Sonderzeichen und Ziffern zu wählen. Ein Passwort muss Zeichen aus mindestens drei dieser Zeichenklassen enthalten.
- ein Passwort muss mindestens 8 Zeichen lang sein
- ein Passwort darf nicht die zugehörige Benutzerkennung enthalten (weder vorwärts noch rückwärts, bei Vergleich unter Ignorierung der Groß- und Kleinschreibung)

- die Wiederholung alter Passwörter beim Passwortwechsel durch den Benutzer selbst muss vom Konnektor verhindert werden (Passworthistorie). Dazu muss der Konnektor mindestens die letzten drei Passwörter eines Benutzers bei der Passwortneuvergabe erkennen und als neues Passwort ablehnen.

Für die Passwortverarbeitung MUSS der Konnektor mindestens folgende Aspekte berücksichtigen

- für die Erstanmeldung neuer Benutzer müssen Einmalpasswörter vergeben werden, also Passwörter, die nach einmaligem Gebrauch gewechselt werden müssen. Gleiches gilt, wenn ein Passwort eines Benutzers vom Super-Admin zurückgesetzt wird.
- jeder Benutzer muss sein eigenes Passwort jederzeit ändern können
- bei der Eingabe darf das Passwort nicht im Klartext auf dem Bildschirm angezeigt werden
- die Passwörter müssen im Konnektor zugriffssicher gespeichert werden
- der Konnektor muss nach einem durch den Super-Admin konfigurierbaren Zeitraum (Voreinstellung: 120 Tage) einen Passwortwechsel beim nächsten Login initiieren
- erfolglose Anmeldeversuche müssen mit einer kurzen Fehlermeldung ohne Angabe von näheren Einzelheiten abgelehnt werden. Insbesondere darf bei erfolglosen Anmeldeversuchen nicht erkennbar sein, ob der eingegebene Benutzername oder das eingegebene Passwort (oder beides) falsch ist.
- Nach einer Fehleingabe des Passworts muss eine Verzögerung bis zur nächsten Eingabemöglichkeit des Passworts für dieselbe Benutzerkennung erfolgen. Die Verzögerung soll 3 Sekunden betragen.

⌫

Näheres hierzu regeln die Schutzprofile des Konnektors.

⌦ **TIP1-A_4810 Benutzerverwaltung der Managementschnittstelle**

Der Konnektor MUSS eine Benutzerverwaltung für die Managementschnittstelle enthalten, in der anmeldeberechtigte Administratoren-Benutzer definiert werden können.

Die Benutzerverwaltung MUSS die Administrator-Rollen Lokaler-Administrator, Remote-Administrator und Super-Administrator unterstützen.

Den Administrator-Rollen MÜSSEN folgende Rechte zugewiesen sein:

- Lokaler-Administrator:
  - ausschließlicher Zugriff über lokalen Endpunkt der Managementschnittstelle
  - Verwaltung aller Konfigurationsdaten mit Ausnahme von Benutzerverwaltung
- Remote-Administrator:

- ausschließlicher Zugriff über remote-Endpunkt der Managementschnittstelle
- Verwaltung aller Konfigurationsdaten mit Ausnahme von Benutzerverwaltung und Werksreset

- Super-Administrator:
  - ausschließlicher Zugriff über lokalen Endpunkt der Managementschnittstelle
  - Benutzerverwaltung gemäß Tabelle TAB_KON_655
  - Verwaltung aller Konfigurationsdaten

Tabelle 298 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. Benutzerkonten MÜSSEN angelegt, geändert und gelöscht werden können. Das Passwort eines Benutzerkonten MUSS neu gesetzt werden können.
MGM_ADMIN_RIGHTS	Liste von Zugriffsrechten eines Benutzers	i. Eindeutige Zuordnung eines Benutzerkontos zu einer Rolle. Die Benutzerverwaltung MUSS sicherstellen, dass zu jeder Zeit mindestens ein Benutzerkonto mit der Rolle Super-Administrator vorhanden ist. Gewähren / Entziehen von Rechten für Benutzerkonten: ii. Zugriffsrechte bezüglich der Konfigurationsbereiche. iii. Recht zum Aufbau einer Remote-Management-Session (USER_INIT_REMOTESESSION). v. Recht für einen Werksreset (USER_RESET_PERMISSION)

Die Benutzerverwaltung MUSS es jedem Benutzer ermöglichen Konfigurationsänderungen gemäß Tabelle TAB_KON_656 vorzunehmen:

Tabelle 299 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 Konnektorname 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 Konnektorname MUSS dauerhaft an der Managementschnittstelle angezeigt werden.

Die Managementschnittstelle MUSS es einem Administrator ermöglichen Konfigurationsänderungen gemäß Tabelle TAB_KON_657 vorzunehmen:

Tabelle 300 TAB_KON_657 Konfigurationsparameter des Konnektornamens

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
MGM_KONN_HOSTNAME	12 Zeichen	Der Konnektorname MUSS folgende Anforderungen erfüllen (in Anlehnung an die Definition eines „Labels“ in [RFC1034]): Verwendung der Buchstaben „A bis Z“ und „a bis z“, Verwendung der Ziffern „0 bis 9“, als Sonderzeichen „–“ (Minus), sowie eine maximale Länge von 12 Zeichen, 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.⌫

4.3.3 Konfigurationsdaten: Persistieren sowie Export- Import

⌦ TIP1-A_4813 Persistieren der Konfigurationsdaten

Der Konnektor MUSS die Konfigurationsdaten nach Änderung persistieren. Dabei MÜSSEN Integrität, Authentizität und Vertraulichkeit der Konfigurationsdaten gewährt sein. Der Mechanismus hierfür ist herstellerspezifisch.

Der Konnektor MUSS sicherstellen, dass immer ein integerer Konfigurationssatz persistiert ist.

Bei der Konnektorinitialisierung MÜSSEN die persistierten Konfigurationsdaten eingelesen werden.

Die Verpflichtung zur Persistierung gilt für alle innerhalb der Konnektor- und Fachmodul-Spezifikationen erhobenen Konfigurationsdaten.⌫

⌦ **TIP1-A_4814 Export- Import von Konfigurationsdaten**

Der Administrator MUSS die gesamten Konfigurationsdaten des Konnektors ex- und importieren können. Dabei KÖNNEN die Konfigurationsdaten des Anwendungs- und Netzkonnektors gemeinsam oder getrennt exportiert bzw. importiert werden. Das Format der Konfigurationsdaten ist herstellerspezifisch.

Auf hardwareseitig baugleichen Geräten:

- MUSS der Import von Konfigurationsdateien möglich sein, die unter der gleichen oder einer früheren Firmwareversion exportiert wurden
- SOLL der Import von Konfigurationsdateien möglich sein, die unter einer neueren Firmwareversion exportiert wurden

Der Import von Konfigurationsdateien, die von einem Konnektor mit anderer Hardwareversion exportiert wurden, KANN ermöglicht werden.

(für Fachmodule siehe Kapitel 4.3.4)

Der Konnektor MUSS sicherstellen, dass der Exportvorgang nur von einem am Konnektor angemeldeten User mit mindestens der Rolle Administrator ausgelöst werden kann.

Der Konnektor MUSS sicherstellen, dass der Importvorgang nur von einem am Konnektor angemeldeten User mit der Rolle Super-Administrator ausgelöst werden kann.

Sowohl Ex- als auch Import MÜSSEN protokolliert werden durch TUC_KON_271 „Schreibe Protokolleintrag“ {„MGM/CONFIG_EXIMPORT“; Op; Info; „User=$AdminUsername; Mode=[Export / Import]“}.⌫

Nähere Vorgaben zum Ablauf des Imports der Kartenterminalinformationen finden sich im Kapitel 4.1.4.6.3.

⌦ **TIP1-A_4815 Export: Schutz der Integrität, Authentizität und Nichtabstreitbarkeit**

Die **Integrität, Authentizität und Nichtabstreitbarkeit** der exportierten Daten MUSS sichergestellt werden. Dies MUSS durch eine Signatur mit der OSIG-Identität der SM-B oder mit einem herstellerspezifischen Schlüsselpaar realisiert werden. In die zu signierenden Daten MUSS eine Zeitangabe zum Signaturzeitpunkt integriert werden. Beim Import MUSS die Signatur vor der Übernahme der Daten erfolgreich verifiziert worden sein. Im Laufe des Importvorgangs MUSS dem Administrator das zur Signatur zugehörige Zertifikat (oder der herstellerspezifische öffentliche Schlüssel) sowie die Zeitangabe zum Signaturzeitpunkt der exportierten Konfiguration angezeigt werden, und der Administrator MUSS explizit bestätigen, dass er die zu dem angezeigten Zeitpunkt gehörige Konfiguration importieren will.

Wird die SM-B zur Signatur eingesetzt, so MUSS die Prüfung des genutzten Signaturzertifikats anhand von TUC_KON_037 erfolgen. Das Zertifikat der OSIG-Identität, mit dem die Daten signiert wurden, MUSS zusammen mit den exportierten Daten gespeichert werden, um eine Verifikation der Signatur auf neuen Konnektoren auch ohne Zugriff auf die entsprechende SM-B zu ermöglichen.

Da Konfigurationsdaten mit einem Schutzbedarf von mindestens „Hoch“ für Authentizität und Nichtabstreitbarkeit exportiert werden (z. B. Pairing-Geheimnisse (ShS.KT.AUT) der Kartenterminals), MUSS durch geeignete Maßnahmen sichergestellt werden, dass der Zugriff auf die Daten auf eine natürliche Person rückführbar ist. Dies kann organisatorisch (durch Einträge des Administrators in ein Betriebsführungs-Handbuch beim Leistungserbringer) technisch (durch eine personenbezogene Administratorenverwaltung) oder äquivalent herstellerspezifisch erreicht werden. ⌫

⌦ **TIP1-A_4816 Export: Schutz der Vertraulichkeit**

Zum Schutz der **Vertraulichkeit** der exportierten Daten MÜSSEN die Daten vor dem Export verschlüsselt werden. Dies kann durch asymmetrische oder symmetrische Verschlüsselungsverfahren nach [gemSpec_Krypt] realisiert werden.

Wird ein rein symmetrisches Verfahren eingesetzt, so MUSS als Mindestanforderung eine Passphrase einer Mindestlänge von 16 Zeichen (Groß- und Kleinbuchstaben, Ziffern und Sonderzeichen) zur Verschlüsselung der Daten eingesetzt werden. Diese Passphrase MUSS dabei vom Konnektor zufällig generiert werden und aus einer Kombination von Buchstaben und Ziffern bestehen. Diese Passphrase MUSS dem Administrator anschließend angezeigt werden.⌫

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.⌫

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 durch TUC_KON_271 „Schreibe Protokolleintrag“ {„MGM/FACTORYSETTINGS“; Op; Info; „User=$AdminUsername“} protokolliert werden. Dieser Protokolleintrag DARF durch einen erfolgreichen Werksreset NICHT verloren gehen.

Der Hersteller MUSS ferner einen alternativen, herstellerspezifischen Weg zum Auslösen des Werksresets vorsehen, welcher die Arbeitsabläufe beim Leistungserbringer nur minimal unterbricht. Auch für diesen zusätzlichen Weg MUSS zuvor eine Authentisierung durch eine Kombination aus Benutzername 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 erfolgreichen anmelden kann. ⌫

4.3.6 Leistungsumfänge und Standalone-Szenarios

Obgleich der Konnektor in seinem Auslieferungszustand alle Leistungsmerkmale aufweisen muss, die gemäß Produkttypssteckbrief 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 Aktivieren/Deaktivieren von Leistungsumfängen**

Die Managementschnittstelle MUSS es einem Administrator ermöglichen Konfigurationsänderungen gemäß Tabelle TAB_KON_658 vorzunehmen:

Tabelle 301 TAB_KON_658 Aktivieren/Deaktivieren von Leistungsumfängen

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
`MGM_LU_ONLINE`	`Enabled / Disabled`	Der Administrator MUSS den „Leistungsumfang Online“ aktivieren und deaktivieren können. Default-Wert: Enabled Bei Veränderung MUSS TUC_KON_256 gerufen

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
		werden {"MGM/LU_CHANGED/LU_ONLINE"; Op; Info; „Active=$MGM_LU_ONLINE"}
MGM_LU_SAK	Enabled / Disabled	Der Administrator MUSS den „Leistungsumfang Signaturanwendungskomponente" aktivieren und deaktivieren können. Default-Wert: Enabled Bei Veränderung MUSS TUC_KON_256 gerufen werden {"MGM/LU_CHANGED/LU_SAK"; Op; Info; „Active=$MGM_LU_SAK"}

⌫

Der Konfigurationsparameter MGM_LU_SAK wirkt hauptsächlich in dem Funktionsmerkmal „Signaturdienst und SAK" (siehe Kapitel 4.1.8).

Ist MGM_LU_ONLINE Disabled, so baut der Konnektor grundsätzlich keine Online-Verbindungen auf (weder zur TI, noch zum SIS). Der Parameter wirkt hauptsächlich in den Funktionsmerkmalen:

- „Zertifikatsdienst" (Kapitel 4.1.9)
- „TLS-Dienst" (Kapitel 4.1.11)
- „Anbindung LAN/WAN" (Kapitel 4.2.1)
- „VPN-Client" (Kapitel 4.2.4)
- „Zeitdienst" (Kapitel 4.2.5)
- „Software-Aktualisierungsdienst (KSR-Client)" (Kapitel 4.3.9)
- LDAP-Proxy (Kapitel 4.1.12)

Ob es sich bei einem Konnektor um den losgelösten (stand alone) vom Leistungserbringernetz betriebenen handelt, also ein Konnektor, auf welchen kein Clientsystem zugreift, muss diesem mitgeteilt werden:

⌦ **TIP1-A_4822 Konnektor Standalone einsetzen**

Die Managementschnittstelle MUSS es einem Administrator ermöglichen Konfigurationsänderungen gemäß Tabelle TAB_KON_659 vorzunehmen:

Tabelle 302 TAB_KON_659 Konnektor Standalone einsetzen

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
MGM_STANDALONE_KON	Enabled / Disabled	Der Administrator MUSS den Konnektor als alleinstehend konfigurieren können. Default-Wert: Disabled Bei Veränderung MUSS TUC_KON_256 gerufen werden {"MGM/STANDALONE_CHANGED"; Op; Info; „Active=$MGM_STANDALONE_KON"}

⌫

Das Setzen von MGM_STANDALONE_KON auf Enabled dient dem Konnektor als Anzeige, dass dieser ohne angeschlossenes Clientsystem (Primärsystem) betrieben wird. Diese Information kann seitens der Fachmodule verwendet werden, damit diese sich im Standalone-Fall anders als im Normalfall verhalten.

⌦ **TIP1-A_4823 Konnektor mit logischer Trennung**

Die Managementschnittstelle MUSS es einem Administrator ermöglichen Konfigurationsänderungen gemäß Tabelle TAB_KON_660 vorzunehmen:

Die Aktivierung der logischen Trennung DARF sich NICHT auf die Anzahl benötigter Kartenterminals auswirken. Alle mit deaktivierter logischer Trennung mit einem Kartenterminal verfügbaren Funktionen des Konnektors müssen auch mit aktivierter logischer Trennung weiterhin mit nur einem Kartenterminal zur Verfügung stehen.

Tabelle 303 TAB_KON_660 Konnektor mit logischer Trennung

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
`MGM_LOGICAL_SEPARATION`	`Enabled / Disabled`	Der Administrator MUSS die logische Separation zwischen TI und lokalem LE-Netz aktivieren / deaktivieren können. Default-Wert: Disabled Bei Veränderung MUSS TUC_KON_256 gerufen werden {"MGM/LOGICAL_SEP_CHANGED"; Op; Info; „Active=$MGM_LOGICAL_SEPARATION"}

⌫

MGM_LOGICAL_SEPARATION wirkt hauptsächlich in den Funktionsmerkmalen:

- „Zertifikatsdienst" (Kapitel 4.1.9)
- „Anbindung LAN/WAN" (Kapitel 4.2.1)
- „VPN-Client" (Kapitel 4.2.4)
- „Zeitdienst" (Kapitel 4.2.5)

4.3.7 Online-Anbindung verwalten

Um Zugang zur TI erlangen zu können, muss der Betriebsstättenverantwortliche einen Vertrag mit einem Zugangsdienstprovider (ZGDP) abgeschlossen haben. Von diesem erhält er eine ContractID. Der Administrator muss den Konnektor (genauer das NK-Zertifikat C.NK.VPN) mit dieser Information unter Nutzung einer SM-B über den Registrierungsdienst des ZGDP bei diesem freischalten.

Die Berechtigung zur Einwahl in die TI ist von der Gültigkeit der **beiden** bei der Freischaltung übermittelten Zertifikate abhängig (C.NK.VPN und C.HCI.OSIG). Die Berechtigung zur Einwahl in die TI wird verweigert, bzw. eine bestehende Verbindung zur TI wird beendet, wenn ein Zertifikat abgelaufen oder gesperrt ist. Aus diesem Grund muss der Administrator vor Ablauf eines der beiden Zertifikate eine wiederholte Registrierung mit neuem Netzkonnektorzertifikat bzw. neuer SM-B beim ZGDP durchführen. (Hinweis:

neue NK-Zertifikate werden erst mit Etablierung der Nachladefunktionalität für gSMC-K verfügbar sein.)

Soll ein Konnektor außer Betrieb genommen werden oder wird der Vertrag mit einem ZGDP gekündigt, muss der Administrator den Konnektor über den Registrierungsdienst des ZGDP abmelden.

⌦ TIP1-A_4824 Freischaltdaten des Konnektors bearbeiten

Der Administrator MUSS die in TAB_KON_661 aufgelisteten Parameter über die Managementschnittstelle konfigurieren und die in TAB_KON_732 aufgelisteten Parameter ausschließlich einsehen können.

Tabelle 304 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 305 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 306 TAB_KON_662 Zustandswerte der Konnektorfreischaltung.

Im Auslieferungszustand MUSS MGM_TI_ACCESS_GRANTED=Disabled belegt sein.

Tabelle 306 TAB_KON_662 Zustandswerte der Konnektorfreischaltung

ReferenzID	Belegung	Zustandswerte
MGM_TI_ACCESS_GRANTED	Enabled / Disabled	Status der Freischaltung des Konnektors: - Enabled: Konnektor wurde erfolgreich beim Zugangsdienstprovider freigeschaltet - 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“

   b. Persistiere diese Zustandsinformation zusammen mit dem Freischaltezeitpunkt

   c. Verteile das folgende interne Ereignis über TUC_KON_256: {"MGM/TI_ACCESS_GRANTED“; Op; Info; „Active=$MGM_TI_ACCESS_GRANTED“; noDisp}

Tritt während der Verarbeitungskette ein Fehler auf, so bricht die weitere Verarbeitung ab und der Administrator MUSS darüber geeignet informiert werden (u.a. Klartextanzeige des vom Registrierungsdienst gemeldeten Fehlers).⌫

⌦ **TIP1-A_4826 Status Konnektorfreischaltung einsehen**

Der Administrator MUSS über die Managementschnittstelle den aktuellen Freischaltstatus einsehen können (MGM_TI_ACCESS_GRANTED). Ist der Konnektor aktuell freigeschaltet, so MUSS ihm angezeigt werden:

- wann die Freischaltung erfolgte
- welche SM-B für die Freischaltung verwendet wurde⌫

Möchte ein Konnektoreigentümer das Gerät weiterveräußern oder vollständig außer Betrieb nehmen, so sollte er eine vorhandene Freischaltung zuvor rückgängig machen.

⌦ TIP1-A_4827 Konnektorfreischaltung zurücknehmen

Ist MGM_TI_ACCESS_GRANTED=Enabled, dann MUSS der Administrator über die Managementschnittstelle des Konnektors die Freischaltung über den folgenden Mechanismus zurücknehmen können:

1. Der Administrator MUSS eine Sicherheitsabfrage zur Zurücknahme der Freischaltung bestätigen
2. Der Konnektor MUSS eine deRegisterKonnektorRequest-Struktur gemäß [gemSpec_VPN_ZugD] erstellen und mit den entsprechenden Parametern befüllen (aktuelles Datum/Uhrzeit, C.NK.VPN, MGM_ ZGDP_CONTRACTID)
3. Der Konnektor MUSS die Request-Nachricht mittels einer verfügbaren SM-B (ID.HCI.OSIG) im Element deRegisterKonnektorRequest/Signature signieren. (MGM_ ZGDP_SMCB ist zu bevorzugen, es kann aber auch jede andere SM-B verwendet werden)
   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.
4. 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>“
5. Der Konnektor ruft unter Verwendung der erzeugten Request-Nachricht die in [gemSpec_VPN_ZugD#Tab_ZD_deregisterKonnektor] definierte Operation I_Registration_Service::deRegisterKonnektor mit der Zieladresse MGM_ZGDP_REGSERVER auf.
6. Der Konnektor zeigt dem Administrator den Inhalt von deregisterKonnektorResponse/AdditionalInformation /ContractStatus und /RegistrationStatus an
7. Der Response der Operation wird verarbeitet:
   a. Setze MGM_TI_ACCESS_GRANTED auf
      - Enabled, wenn /RegistrationStatus = „Registriert“
      - Disabled, wenn /RegistrationStatus = „Nicht registriert“
   b. Persistiere diese Zustandsinformation zusammen mit dem Zeitpunkt
   c. Verteile das folgende interne Ereignis über TUC_KON_256: {"MGM/TI_ACCESS_GRANTED“; Op; Info; „Active=$MGM_TI_ACCESS_GRANTED“; noDisp}

Tritt während der Verarbeitungskette ein Fehler auf, so bricht die weitere Verarbeitung ab und der Administrator MUSS darüber geeignet informiert werden (u.a. Klartextanzeige des vom Registrierungsdienst gemeldeten Fehlers).⌫

⌦ TIP1-A_5655 Deregistrierung bei Außerbetriebnahme

Der Hersteller des Konnektors MUSS im Handbuch den Administrator darüber informieren, dass der Konnektor bei dauerhafter Außerbetriebnahme (z. B. Verkauf, Schenkung, Entsorgung) beim Zugangsdienstprovider deregistriert werden muss. ⌫

4.3.8 Remote Management

Im Betreibermodell der TI wird unter Remote Management ein delegierter Betrieb dezentraler Produkte durch einen durch den Anwender beauftragten Servicepartner verstanden. Der Servicepartner stellt als Vertragsbestandteil bevollmächtigte Personen zur Verfügung, die sich ständig um die Betriebs- und Datensicherheit der dezentralen Produkte im Rahmen eines Remote Managements kümmern.

Voraussetzung für die Etablierung dieses Bestandteils des Betreibermodells der TI ist, dass ein dezentrales Produkt ein Remote Management technisch unterstützt. Die nachfolgend aufgeführten Anforderungen bilden die Grundlage für die Nutzung von Remote Management am Konnektor. Der Hersteller des Konnektors kann unter Berücksichtigung dieser Anforderungen weitere herstellerspezifische Funktionen zum Remote Management implementieren.

⌦ **TIP1-A_5647 Remote Management Konnektor: Personenbezogene Daten**

Der Konnektor DARF über die Remote-Managementschnittstelle KEINE personenbezogenen Daten übertragen oder darstellen.⌫

⌦ **TIP1-A_5648 Remote Management Konnektor: Offene Schnittstelle**

Der Hersteller des Konnektors MUSS die zur Nutzung der Remote-Managemenschnittstelle notwendigen Informationen offenlegen. Der Hersteller des Konnektors MUSS die Remote-Managementschnittstelle so spezifizieren und implementieren, dass diese auch für Dritte (z.B. einen durch den Anwender beauftragten Servicepartner) nutzbar ist. ⌫

⌦ **TIP1-A_5649 Remote Management Konnektor: Standardbasierte Protokolle**

Der Hersteller des Konnektors SOLL für die Implementierung der Remote-Managementschnittstelle standardbasierte Verfahren und Protokolle einsetzen.⌫

⌦ **TIP1-A_5650 Remote Management Konnektor: Aufbau der Verbindung**

Der Konnektor MUSS sicherstellen, dass die Initiierung einer Remote-Managementverbindung im Sinne des Verbindungsaufbaus immer vom Konnektor ausgeht und nur durch einen angemeldeten Benutzer mit der Rolle Lokaler Administrator und dem Recht USER_INIT_REMOTESESSION ausgelöst werden darf.⌫

Nach Aufbau der Verbindung muss sich der Remote-Administrator gemäß TIP1-A_4808 zuerst authentisieren.

⌦ **TIP1-A_5651 Remote Management Konnektor: Absicherung der Verbindung**

Der Konnektor MUSS die Remote-Management-Verbindung durch Nutzung eines kryptographischen Verfahrens gemäß [gemSpec_Krypt] hinsichtlich Vertraulichkeit, Integrität und Authentizität absichern und eine gegenseitige Authentifizierung durchführen. ⌫

⌦ **TIP1-A_5652 Remote Management Konnektor: Konfiguration Remote Management**

Der Konnektor MUSS sicherstellen, dass es ausschließlich einem Super-Administrator möglich ist, Konfigurationsänderungen gemäß TAB_KON_663 vorzunehmen.

Tabelle 307 TAB_KON_663 Konfigurationen des Remote Managements

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
MGM_REMOTE_ALLOWED	Enabled / Disabled	Der Administrator MUSS einstellen können, ob der Konnektor eine Remote-Management-Verbindung aufbauen kann. Enabled: Der Konnektor kann eine Remote-Management-Verbindung aufbauen Disabled: Der Konnektor kann keine Remote-Management-Verbindung aufbauen Default-Wert: Disabled

⌫

⌦ **TIP1-A_5653 Remote Management Konnektor: Protokollierung Remote Management**

Der Konnektor MUSS im Rahmen des Remote-Managements folgende Aktionen protokollieren:

- Versuch Verbindungsaufbau Remote-Session durch TUC_KON_271 „Schreibe Protokolleintrag“ {„MGM/REMOTE_SESSION“; Op; Info; „InitUser=$AdminUsername; RemoteID=<Kennung der Gegenstelle>; Mode=[InitSuccess / InitFail]“}
- Verbindungsabbau Remote-Session durch TUC_KON_271 „Schreibe Protokolleintrag“ {„MGM/REMOTE_SESSION“; Op; Info; „InitUser=$AdminUsername; RemoteID=<Kennung der Gegenstelle>; Mode=Exit“} ⌫

Ein Softwareupdate gemäß TIP1-A_5657 kann auch über das Remote Management initiiert und aktiviert werden. Auch in diesem Fall muss jedes Softwareupdate vor Aktivierung einzeln vom Leistungserbringer bzw. einem von ihm beauftragten lokalen Administrator freigeschaltet werden, um die Anforderungen des SigG/SigV an eine SAK zu erfüllen.

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>=“zugewiesen“ und

CT.VALID_VERSION=True und CT.IS_PHYSICAL = Nein) 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 Freischaltung von Softwareupdates

Der Konnektor MUSS sicherstellen, dass Softwareupdates durch den Leistungserbringer bzw. einen von ihm beauftragten lokalen Administrator aus der Leistungserbringerumgebung heraus einzeln freigeschaltet werden. Der Konnektor DARF ein Softwareupdate NICHT ohne vorher erfolgte Freischaltung aktivieren.

TIP1-A_5659 Bewusste Entscheidung bei Freischaltung von Softwareupdates

Der Hersteller des Konnektors MUSS in seinem Handbuch den Anwender (Leistungserbringer bzw. von ihm beauftragter lokaler Administrator) darauf hinweisen, dass der Anwender ein Softwareupdate nur dann aktivieren soll, wenn er ausreichend Informationen über den Inhalt des Softwareupdates erhalten hat, die ihm eine bewusste Entscheidung bei der Freischaltung ermöglichen.

4.3.9.2 Durch Ereignisse ausgelöste Reaktionen

TIP1-A_4831 KT-Update nach Wiedererreichbarkeit erneut anstoßen

Wenn aus (TIP1-A_4840 Auslösen der durchzuführenden Updates) heraus für ein Kartenterminal noch ein ausstehendes Updates vorhanden ist, dessen Ausführungszeitpunkt nicht gesetzt oder überschritten ist, und für dieses Kartenterminal das Ereignis „CT/CONNECTED“ eintritt, so MUSS TUC_KON_281 „Kartenterminal-aktualisierung anstoßen“ für dieses KT gerufen werden.

4.3.9.3 Interne TUCs, nicht durch Fachmodule nutzbar

4.3.9.3.1 TUC_KON_280 „Konnektoraktualisierung durchführen“

TIP1-A_4832 TUC_KON_280 „Konnektoraktualisierung durchführen“

Der Konnektor MUSS den technischen Use Case TUC_KON_280 „Konnektoraktualisierung durchführen“ umsetzen.

Tabelle 308: 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	Der Administrator hat UpdateInformation zur Anwendung ausgewählt und bestätigt bzw. ein lokales Updatepaket bezogen und zur Anwendung übergeben.
Vorbedingungen	Der Administrator hat bewusst das übergebene Paket für eine Installation ausgewählt
Eingangsdaten	• UpdateInformation (gemäß [gemSpec_KSR#5.2]) oder • Updatepaket (herstellerspezifisch, von lokaler Datenquelle, mit enthaltenen FirmwareFiles)
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: 1. Integrität und Authentizität der UpdateInformation prüfen (Mechanismus ist herstellerspezifisch) 2. Download aller in UpdateInformation.FirmwareFiles gelisteten Dateien. Dabei wird die Komprimierung des File Transfers vom Konfigurationsdienst über http „Content Coding“ [RFC2616] „gzip“ genutzt. 3. Integrität und Authentizität jeder der via UpdateInformation/FirmwareFiles heruntergeladenen Dateien prüfen (Mechanismus ist herstellerspezifisch) 4. Prüfen auf Zulässigkeit des Updates basierend auf der Firmware-Gruppe (siehe [gemSpec_OM#2.5] 5. Anwenden der zur Verfügung stehenden FirmwareFiles a) TUC_KON_256{"KSR/UPDATE/START"; Sec; Info; „Target=Konnektor; Name=$MGM_KONN_HOSTNAME“} (betroffene Fachmodule und Basisdienste reagieren und stoppen sich) b) Herstellerspezifischer Mechanismus zur Aktualisierung der internen Konnektorsoftware durch die FirmwareFiles inklusive anschließender Prüfung auf Erfolg. c) Bestehende Konfigurationsdaten des Konnektors MÜSSEN erhalten bleiben und sofern erforderlich und möglich automatisch auf die Definitionen der neuen Firmware angepasst werden. d) Ist ein händischer Anpassungs- oder Ergänzungsbedarf der Konfigurationsdaten erforderlich, so MUSS der Administrator hierüber geeignet informiert werden

Element	Beschreibung
	e) Absetzen TUC_KON_256{"KSR/UPDATE/SUCCESS"; Sec; Info; <Params>"} mit <Params>= „Target=Konnektor; Name= $MGM_KONN_HOSTNAME; NewFirmwareversion=$UpdateInformation.FirmwareVersion; ConfigurationChanged=<Ja/Nein>; ManualInputNeeded=<Ja/Nein>" Der TUC endet in jedem Fall mit: • TUC_KON_256{"KSR/UPDATE/END"; Sec; Info; „Target=Konnektor; Name=$MGM_KONN_HOSTNAME"} (betroffene Fachmodule und Basisdienste reagieren und starten sich)
Varianten/Alternativen	Sofern direkt ein Updatepaket (mit enthaltenen FirmwareFiles) übergeben wurde beginnt der Ablauf ab Nummer 4 mit der Integritätsprüfung des Updatepakets
Fehlerfälle	Fehler in den folgenden Schritten des Ablaufs führen zu: a) Aufruf von TUC_KON_256 mit folgenden Parametern {"KSR/ERROR"; $ErrorType; $Severity; „Target=Konnektor; Name= $MGM_KONN_HOSTNAME; Error=$Fehlercode; Bedeutung=$Fehlertext"} b) Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes (→1) Integritätsprüfung UpdateInformation fehlgeschlagen, Fehlercode: 4181 (→2) Fehler bei der Downloaddurchführung, Fehlercode: 4182 (→3) Integritätsprüfung eines FirmwareFiles fehlgeschlagen, Fehlercode: 4183 (→ 4) Firmwaregruppenprüfung fehlgeschlagen, Fehlercode: 4185 (→5b) Interne Aktualisierung fehlgeschlagen, dann: 1. Rollback auf vorherige Version 2. Abbruch mit Fehlercode: 4184
Nichtfunktionale Anforderungen	Der laufende Updatevorgang MUSS in der Managementschnittstelle ausgewiesen und der Fortschritt mindestens für die Schritte 1-5b dargestellt werden.
Zugehörige Diagramme	Abbildung 20: PIC_KON_105 Aktivitätsdiagramm Konnektoraktualisierung durchführen

Tabelle 309: TAB_KON_665 Übersicht Fehlercodes für „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>).

Fehlercode	ErrorType	Severity	Fehlertext
4185	Security	Error	Firmware-Version liegt außerhalb der gültigen Firmware-Gruppe

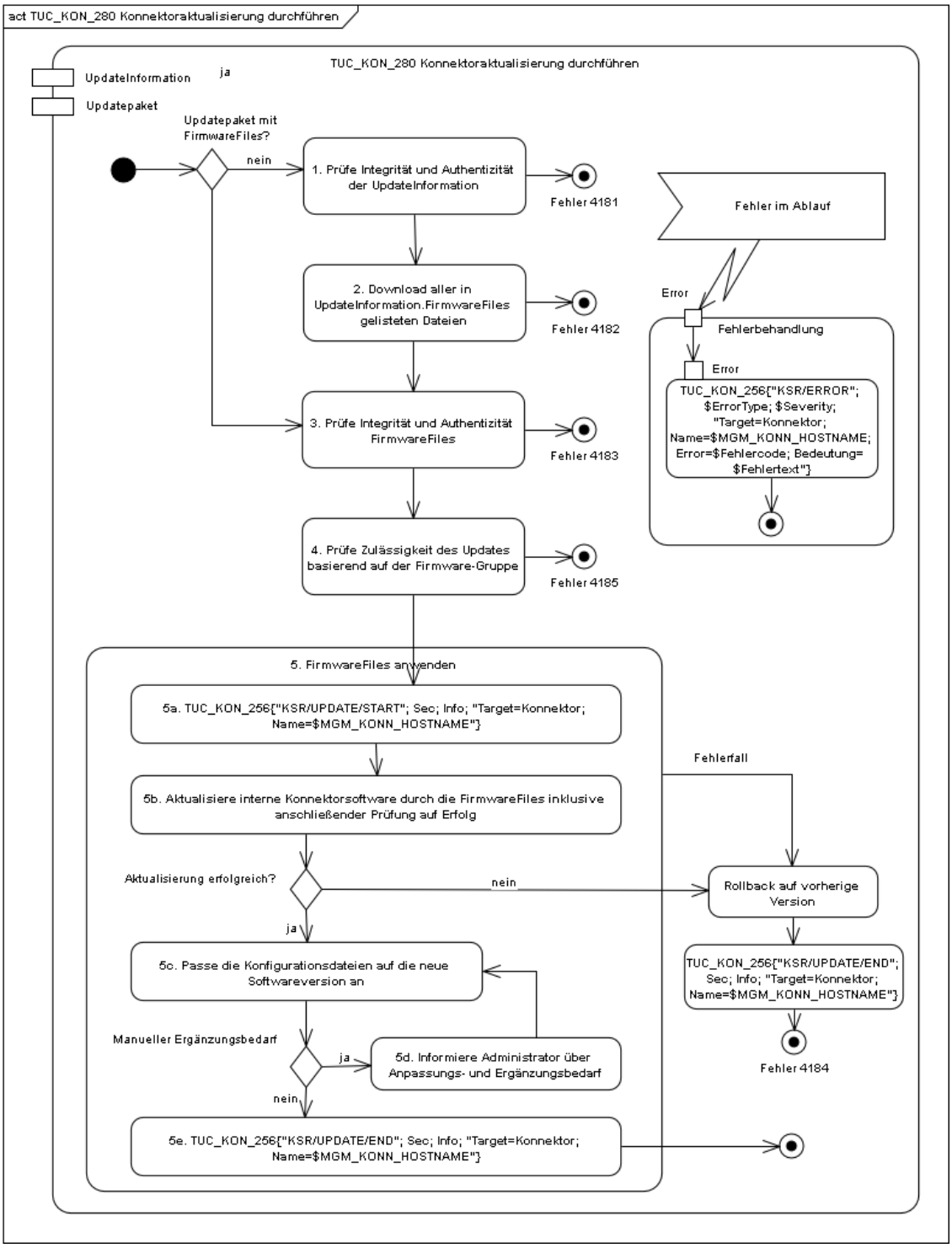


Abbildung 20: PIC_KON_105 Aktivitätsdiagramm Konnektoraktualisierung durchführen

⌫

4.3.9.3.2 TUC_KON_281 „Kartenterminalaktualisierung anstoßen“

Im Vergleich zur Durchführung des Konnektor-Update (TUC_KON_280), werden die Updates der Kartenterminals nur durch den Konnektor initiiert. Der Konnektor liefert dem Kartenterminal das Updatefile, der eigentliche Updatevorgang (inklusive der Prüfung des Updatepakets auf Integrität und Authentizität) erfolgt ausschließlich und eigenverantwortlich auf Seiten des Kartenterminals.

⌦ TIP1-A_4833 TUC_KON_281 „Kartenterminalaktualisierung anstoßen“

Der Konnektor MUSS den technischen Use Case TUC_KON_281 „Kartenterminalaktualisierung anstoßen“ umsetzen.

Tabelle 310: 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	Der Administrator hat UpdateInformation für ein Kartenterminal zur Anwendung ausgewählt und bestätigt bzw. ein lokales Updatepaket für ein Kartenterminal bezogen und zur Anwendung übergeben.
Vorbedingungen	• Der Administrator hat bewusst das übergebene Paket für eine Installation ausgewählt • CT(ctID).IS_PHYSICAL=Ja
Eingangsdaten	• ctID (ID des Ziel-KTs) • UpdateInformation (gemäß [gemSpec_KSR]) oder • Updatepaket (herstellerspezifisch, von lokaler Datenquelle, mit enthaltenen FirmwareFiles)
Komponenten	Konnektor, Kartenterminal
Ausgangsdaten	Keine
Nachbedingungen	Das Kartenterminal arbeitet basierend auf der übergebenen, im Updatepaket enthaltenen neuen Version.
Standardablauf	Der Konnektor MUSS die zur Anwendung übergebene UpdateInformation wie folgt anwenden: 1. Download der in UpdateInformation/FirmwareFiles gelisteten Datei (für KT-Updates darf nur genau ein FirmwareFile angegeben werden) 2. TUC_KON_256{"KSR/UPDATE/START"; Sec; Info; „Target=KT; CTID=$ctID“} 3. Durchführen des KT-Updates durch: a) Wechsel in eine Admin-Session durch TUC_KON_050 „Beginne Kartenterminalsitzung“{Admin; ctID} b) Senden der SICCT Kommandos: SICCT CT Download INIT, SICCT CT Download DATA (Übermittlung des UpdateFiles) und SICCT CT Download FINISH an ctID

Element	Beschreibung
	c) TUC_KON_256{"KSR/UPDATE/SUCCESS"; Sec; Info; „Target=KT; Name= $CT.HOSTNAME;CTID=$ctID; NewFirmwareversion=<UpdateInformation.FirmwareVersion>“} Der TUC endet in jedem Fall mit: • TUC_KON_256{"KSR/UPDATE/END"; Sec; Info; „Target=KT;CTID=$ctID“}
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	Fehler in den folgenden Schritten des Ablaufs führen zu: a) Aufruf von TUC_KON_256 mit folgenden Parametern {"KSR/ERROR"; $ErrorType; $Severity; „Target=KT; Name=$CT.HOSTNAME;CTID=$ctID; Error=$Fehlercode; Bedeutung=$Fehlertext“} b) Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes (→1) Download fehlgeschlagen, Fehlercode: 4186 (→3b) SICCT-Download fehlgeschlagen, Fehlercode: 4187
Nichtfunktionale Anforderungen	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 Management-schnittstelle ausgewiesen und der Fortschritt dargestellt werden. Der Konnektor MUSS mindestens 5 Kartenterminal-Updates parallel durchführen können.
Zugehörige Diagramme	keine

Tabelle 311: TAB_KON_667 Übersicht Fehlercodes für „Kartenterminalaktualisierung 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>)

⌫

4.3.9.3.3 TUC_KON_282 „UpdateInformationen beziehen“

⌦ **TIP1-A_4834 TUC_KON_282 „UpdateInformationen beziehen“**

Der Konnektor MUSS den technischen Use Case TUC_KON_282 „UpdateInformationen beziehen“ umsetzen.

Tabelle 312: 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	• Manuell durch den Administrator • Automatisch wenn `MGM_KSR_AUTOCHECK=Enabled`
Vorbedingungen	Keine
Eingangsdaten	Keine
Komponenten	Konnektor, Konfigurationsdienst
Ausgangsdaten	• Keine
Nachbedingungen	Der Konnektor verfügt über alle aktuellen UpdateInformationen
Standardablauf	Der Konnektor MUSS folgende Schritte durchlaufen: 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" {C.ZD.TSL-S; not_required; ; true ; oid_zd_tls_s;digitalSignature&keyEncipherment; id-kp-serverAuth; ; OCSP} auf Gültigkeit prüfen. Das Server-Zertifikat C.ZD.TLS-S MUSS für den Konfigurationsdienst ausgestellt sein. 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>=“zugewiesen“ folgende Schritte durchlaufen: a) Belegen von listUpdatesRequest mit den korrekten Werten für ManufacturerID, ProductShortName, HardwareVersion und FirmwareVersion b) Aufruf von I_KSRS_Download::list_Updates 3. Beenden der TLS-Verbindung
Varianten/Alternativen	Keine
Fehlerfälle	Fehler in den folgenden Schritten des Ablaufs führen zu: a) Aufruf von TUC_KON_256 mit folgenden Parametern {"KSR/ERROR"; $ErrorType; $Severity; „Error=$Fehlercode; Bedeutung=$Fehlertext“} b) Abbruch der Verarbeitung mit den ausgewiesenen Fehlercodes (→1) Konfigurationsdienst nicht erreichbar, Fehlercode: 4188 (→1) Serverzertifikat ist nicht C.ZD.TLS_S, Fehlercode: 4189 (→2b) Fehler beim Beziehen der Updatelisten, Fehlercode: 4190
Nichtfunktionale Anforderungen	Der Konnektor muss die Vorgaben aus [gemSpec_Krypt#3.3.2] für TLS-Verbindungen befolgen.

Element	Beschreibung
Zugehörige Diagramme	keine

Tabelle 313: TAB_KON_669 Übersicht Fehlercodes für „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

⌫

4.3.9.3.4 TUC_KON_283 „Infrastruktur Konfiguration aktualisieren“

⌦ **TIP1-A_5153 TUC_Kon_283 „Infrastruktur Konfiguration aktualisieren**“

Der Konnektor MUSS den technischen Use Case TUC_Kon_283 „Infrastruktur Konfiguration aktualisieren“ umsetzen.

Tabelle 314: 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
Vorbedingungen	Der TUC_KON_304 „Netzwerk-Routen einrichten“ MUSS fehlerfrei durchgelaufen sein. Der TUC_KON_321 „Verbindung zu dem VPN-Konzentrator der TI aufbauen“ MUSS fehlerfrei durchgelaufen sein.
Eingangsdaten	Keine
Komponenten	Konnektor, Konfigurationsdienst
Ausgangsdaten	Keine
Standardablauf	Der Konnektor MUSS folgende Schritte durchlaufen: 1. „Einlesen des Konfigurations-XML“: 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" {C.ZD.TSL-S; not_required; ; true; oid_zd_tls_s; digitalSignature&keyEncipherment; id-

Element	Beschreibung
	kp-serverAuth; ; OCSP} auf Gültigkeit prüfen. Das Server-Zertifikat C.ZD.TLS-S MUSS für den Konfigurationsdienst ausgestellt sein. b) Herunterladen der Konfigurationsdaten mittels I_KSRS_Download::get_Ext_Net_Config (MGM_KSR_KONFIG_URL, „Bestandsnetze.xml“) c) Beenden der TLS-Verbindung 2. „Prüfen der Versionskennung auf Änderungen“: 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 werden. 3. Aktualisieren der Gesamtnetzliste Alle in der Datei enthaltenen Netzsegmente sind nach ANLW_BESTANDSNETZE zu übernehmen 4. „Aktualisieren von Konfigurationsinformationen“ Haben sich Konfigurationsdaten zu einem in ANLW_AKTIVE_BESTANDSNETZE gelisteten Netz verändert, so a) sind die Änderungen entsprechend zu übernehmen und zu aktivieren (Anpassung ANLW_AKTIVE_BESTANDSNETZE, DHCP_AKTIVE_BESTANDSNETZE_ROUTES, DNS_SERVERS_BESTANDSNETZE). b) ist anschließend TUC_KON_304 „Netzwerk-Routen einrichten“ aufzurufen War ein Bestandsnetz bereits durch den Administrator freigegeben, so muss diese Freigabe erhalten bleiben. 5. „Entfernen von nicht mehr gültigen Bestandsnetzen“ Ist ein Netz in der neuen Datei gegenüber der alten Datei nicht mehr vorhanden, so: 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). b) ist anschließend TUC_KON_304 „Netzwerk-Routen einrichten“ aufzurufen.
Varianten / Alternativen	Keine
Fehlerfälle	(→ 1-5) Es ist ein unerwarteter Fehler aufgetreten; Fehlercode: 4198
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 315: Tab_Kon_726 Übersicht Fehler 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 Bestandsnetze ist ein Fehler aufgetreten.

⌫

4.3.9.4 Interne TUCs, auch durch Fachmodule nutzbar

Keine.

4.3.9.5 Operationen an der Außenschnittstelle

Keine.

4.3.9.6 Betriebsaspekte

4.3.9.6.1 TUC_KON_284 KSR-Client initialisieren

⌦ **TIP1-A_5938 TUC_KON_284 „KSR-Client initialisieren“**

Der Konnektor MUSS in der Bootup-Phase TUC_KON_284 "KSR-Client initialisieren" durchlaufen.

Tabelle 316 TAB_KON_644 - 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 Verbindung zum VPN-Konzentrator TI muss aufgebaut sein
Eingangsdaten	Keine
Komponenten	Konnektor
Ausgangsdaten	Keine

Element	Beschreibung
Standardablauf	• Falls MGM_LU_ONLINE=Enabled: o Durch DNS-Anfragen an den DNS-Forwarder zur Auflösung der SRV-RR und TXT-RR mit den Bezeichnern "_ksrkonfig._tcp.ksr.<TOP_LEVEL_DOMAIN_TI>“ und "_ksrfirmware._tcp.ksr.<TOP_LEVEL_DOMAIN_TI>“ erhält der Konnektor URLs der Downloadpunkte des KSR für Konfigurationsdaten (MGM_KSR_KONFIG_URL) und für Firmware (MGM_KSR_FIRMWARE_URL).
Varianten/Alternativen	Keine
Fehlerfälle	Keine
Nichtfunktionale Anforderungen	Keine
Zugehörige Diagramme	Keine

Tabelle 317: TAB_KON_822 Übersicht Fehler TUC_KON_284 "Initialisierung Konfigurationsdienst"

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können keine weiteren Fehlercodes auftreten.			

⌫

⌦ TIP1-A_4835 Konfigurationswerte des KSR-Client

Der Administrator MUSS die in TAB_KON_670 aufgelisteten Parameter über die Managementschnittstelle konfigurieren und die in TAB_KON_820 aufgelisteten Parameter ausschließlich einsehen können.

Tabelle 318 TAB_KON_670 Konfigurationsparameter der Software-Aktualisierung

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
`MGM_KSR_AUTOCHECK`	Enabled / Disabled	Der Administrator MUSS die automatische Prüfung auf verfügbare Update-Pakete an- und abschalten können. Default-Wert: Enabled
`MGM_KSR_AUTODOWNLOAD`	Enabled / Disabled	Sofern `MGM_KSR_AUTODOWNLOAD` auf Enabled gesetzt ist, MUSS der Administrator den automatische Download verfügbarer Update-Pakete an- und abschalten können. Default-Wert: Disabled

Tabelle 319 TAB_KON_820 Einsehbare Konfigurationsparameter der Software-Aktualisierung

ReferenzID	Belegung	Bedeutung und Administrator-Interaktion
MGM_KSR_KONFIG_URL	URL	SOAP-Endpunkt des Konfigurationsdienstes zum Download von Konfigurationsdaten, dynamisch ermittelt im Rahmen des VPN-Verbindungsaufbaus zur TI
MGM_KSR_FIRMWARE_URL	URL	SOAP-Endpunkt des Konfigurationsdienstes zum Download der Firmware, dynamisch ermittelt im Rahmen des VPN-Verbindungsaufbaus zur TI

⌫

Hinweis: Die Adressen des Konfigurationsdienstes werden im Rahmen des VPN-Verbindungsaufbaus ermittelt (siehe [gemSpec_VPN_ZugD#5.1.1.2 TUC_VPN-ZD_0001])

⌦ **TIP1-A_4836 Automatische Prüfung und Download von Update-Paketen**

Wenn MGM_KSR_AUTOCHECK auf Enabled gesetzt ist, MUSS der Konnektor täglich 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“ {"KSR/UPDATES_AVAILABLE"; Op; Info; <Param>; noLog} informieren. Je gefundenem Update MUSS <Param> mit folgender Werten belegt sein:
   „ManufacturerID=$UpdateInformation/ManufacturerID;
   ProductShortName=$UpdateInformation/ProductShortName;
   ProductName=$UpdateInformation/ProductName;
   FirmwareVersion=$UpdateInformation/FirmwareVersion“
4. 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.
5. 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
6. 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

Der Konnektor MUSS immer nur die neusten Update-Pakete für eine Nutzung vorhalten. Eventuell vorhandene ältere, nicht genutzte Updatepakte KÖNNEN überschrieben werden.⌫

⌦ **TIP1-A_4837 Übersichtsseite des KSR-Client**

Die Administrationsoberfläche des KSR-Clients MUSS dem Administrator eine Übersichtseite anbieten, die einen Geräteeintrag für den Konnektor selbst, sowie eine Liste von Geräteeinträgen für jedes Kartenterminal (CT) aus CTM_CT_LIST mit CT.IS_PHYSICAL=Ja und CT.CORRELATION>="zugewiesen" enthält.

Der Administrator MUSS die Liste der Kartenterminals nach Kartenterminalmodellen gruppieren können (gleiche Werte für ManufacturerID, ProduktShortName, HardwareVersion und FirmwareVersion).

Je Geräteeintrag MÜSSEN die über „Automatische Prüfung und Download von Update-Paketen" ermittelten listUpdatesResponse bereitstehen.

Der Administrator MUSS eine Aktualisierung aller listUpdatesResponse über TUC_KON_282 „UpdateInformationen beziehen" auslösen können.

Geräteeinträge, die über listUpdatesResponse mit neuerer Firmwareversion als das zugehörige Gerät verfügen, MÜSSEN hervorgehoben werden.⌫

⌦ **TIP1-A_4838 Einsichtnahme in Update-Informationen**

Für alle Geräteeinträge MUSS der Administrator zu den listUpdatesResponse sowohl die FirmwareGroupReleaseNotes als auch jedes enthaltene UpdateInformation-Element einsehen können. Dazu MUSS der Konnektor

- alle Felder der Struktur verständlich umsetzen und strukturiert anzeigen (inkl. der Notes für jedes Firmwarefiles- und Documentationsfiles-Element)
- jedes über das Documentationfiles-Element erreichbare Dokument auf Anforderung des Administrator herunterladen und anzeigen. Es MÜSSEN dabei mindestens die folgenden Dokumentenformate zur Anzeige gebracht werden können: Text, PDF, JPEG, TIFF⌫

⌦ **TIP1-A_4839 Festlegung der durchzuführenden Updates**

Der Administrator MUSS in der Übersichtsliste einzelne Geräteeinträge bzw. Gruppen mit der jeweils anzuwendende UpdateInformation für die Durchführung eines Updates markieren können.

Alternativ MUSS der Administrator neben der Markierung je Geräteeintrag bzw. Gruppe Update-Pakete lokal einspielen können (etwa durch ein Upload- bzw. Download-Interface in der Administrationsoberfläche).

Je Geräteeintrag MUSS der Administrator einen individuellen Ausführungszeitpunkt für die Durchführung des Updates einstellen können.

Der Administrator MUSS für den Geräteeintrag Konnektor festlegen können, ob dieses Update erst gestartet werden darf, wenn zuvor alle festgelegten KT-Updates erfolgreich durchlaufen wurden.

Der Administrator MUSS zu jeder Zeit die gerätebezogene Festlegung für ein Update ändern bzw. löschen können, sofern dieses konkrete Update noch nicht begonnen wurde.⌫

⌦ **TIP1-A_4840 Auslösen der durchzuführenden Updates**

Der Administrator MUSS für die Liste der markierten Geräteeinträge ein gesammeltes Update auslösen können. Dieses MUSS nach folgendem Muster ablaufen:

1. Alle Kartenterminaleinträge abarbeiten. Pro markiertem Kartenterminal:
   - Wenn Ausführungszeitpunkt nicht gesetzt:
     Anwenden des definierten Updates mittels TUC_KON_281 „Kartenterminalaktualisierung anstoßen“
   - Wenn Ausführungszeitpunkt gesetzt:
     Anwenden des definierten Updates mittels TUC_KON_281 sobald der Ausführungszeitpunkt erreicht ist oder, sofern der Konnektor zum Ausführungszeitpunkt nicht in Betrieb war, überschritten wurde. Konnte das Kartenterminal nicht erreicht werden, so MUSS das gesetzte Update im KSR-Client für eine spätere Anwendung erhalten bleiben (wird ereignisgesteuert neu ausgelöst).
2. Sofern die KonnektorUpdate-Abhängigkeit von KT-Updates nicht gesetzt wurde oder wenn alle vorgesehenen Kartenterminal-Updates durchgeführt wurden, MUSS das Konnektor-Updates mittels TUC_KON_280 „Konnektoraktualisierung durchführen“ wie folgt begonnen werden:
   - wenn Ausführungszeitpunkt nicht gesetzt: TUC-Aufruf direkt
   - wenn Ausführungszeitpunkt gesetzt: TUC-Aufruf direkt sobald der Ausführungszeitpunkt erreicht ist oder, sofern der Konnektor zum Ausführungszeitpunkt nicht in Betrieb war, überschritten wurde

Der Konnektor DARF NICHT automatisch ein Update für sich oder einem seiner verwalteten Kartenterminals ausführen, wenn der Administrator diesem Update zuvor nicht explizit zugestimmt hat.⌫

4.3.10 Konnektorstatus

⌦ **TIP1-A_5542 Konnektor, Funktion zur Prüfung der Erreichbarkeit von Systemen**

Der Konnektor MUSS an der Managementschnittstelle eine Funktion anbieten, die es ermöglicht die Erreichbarkeit von Systemen durch Eingabe der IP-Adresse oder des FQDN zu prüfen. Das Ergebnis des Tests MUSS angezeigt werden.⌫

4.4 Hardware-Merkmale des Konnektors

⌦ **TIP1-A_4841 Hardware für Dauerbetrieb**

Der Konnektor MUSS sowohl in seiner Stromversorgung als auch in seinen restlichen Hardwarekomponenten auf einen 24x7-Dauerbetrieb ausgelegt sein.

Der Hersteller DARF NICHT davon ausgehen oder gar in seiner Guidance darauf verweisen, dass der Konnektor mehrere Stunden am Tag nicht betrieben wird.⌫

Diese Anforderung verlangt keinen Schutz gegen Stromausfall in den Betriebsräumen.

⌦ **TIP1-A_4842 Gehäuseversiegelung**

Jeder Konnektor, der als Appliance (dezidierte, geschlossene Kombination aus spezifischer Hard- und Software) ausgeprägt ist, MUSS über eine fälschungssichere Gehäuseversiegelung verfügen. Die Versiegelung MUSS so angebracht werden, dass eine Öffnung des Gehäuses nicht ohne Beschädigung des Siegels erfolgen kann.

Der Konnektor MUSS die Umsetzung entsprechend der Festlegungen für das Kartenterminal nach der TR-03120 [BSI TR-03120], Kapitel bzgl. Gehäuseversiegelung 9 vornehmen.

Die optische Gestaltung der Siegel ist herstellerspezifisch.⌫

Die Prüfung auf Einhaltung der Versigelungsvorgaben erfolgt nicht im Rahmen der CC-Evaluierung, sondern im Zuge der Prüfung auf funktionale Eignung.

⌦ **TIP1-A_4843 Zustandsanzeige**

Im Betrieb MUSS der Zustand des Konnektors erkennbar sein. Zur Anzeige des Betriebszustandes des Konnektors SOLL es eine Signaleinrichtung (z. B. über Status-LEDs) am Konnektor geben**.** Falls keine Signalvorrichtung am Konnektorgehäuse verwendet wird MUSS es eine softwareseitige Lösung über das Managementinterface geben. Bei verbauter Hardware-Signalgebung KANN eine softwareseitige Lösung zusätzlich angeboten werden.

Es MÜSSEN mindestens folgende angezeigt werden:

- Power ON,
- Link Status pro physischer Netzwerkschnittstelle
- Fehler/Kritischer Betriebszustand gemäß Kapitel 3.3

Es SOLLEN folgende Zustände angezeigt werden:

- Status pro IPsec-Verbindung⌫

⌦ **TIP1-A_4844 Ethernet-Schnittstellen**

Der Konnektor MUSS mindestens zwei Ethernetinterfaces nach [IEEE802.3] als physikalischen Schnittstelle zu Verfügung stellen. Jedes Interface MUSS als Fast-Ethernet Schnittstelle in der Ausführung 100Base-TX nach [IEEE802.3 Clause 25] ausgeführt sein. Optional KANN der Konnektor die Schnittstelle in der Ausführung 1000Base-T nach [IEEE802.3 Clause 40] anbieten. ⌫

⌦ **TIP1-A_4845 Verwendungsumgebung - Klima**

Als normaler Einsatzort wird für den Konnektor ein Büroraum angenommen. Der Konnektor MUSS die in Tabelle 320 aufgeführten Anforderungen erfüllen, welche unter der Annahme des normalen Einsatzortes erhoben werden.

Tabelle 320 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.

⌫

⌦ TIP1-A_4846 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 321 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 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 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 m/s^2 (10 g) Amplitude und einer Dauer von 16 ms belastet.

⌫

Anhang A - Verzeichnisse

A1 – Abkürzungen

Kürzel	Erläuterung

A2 – 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.

Das Glossar wird als eigenständiges Dokument, vgl. [gemGlossar] zur Verfügung gestellt.

A3 – Abbildungsverzeichnis

Abbildung 1 PIC_KON_116 Schnittstellen des Konnektors von und zu anderen Produkttypen 20

Abbildung 2 PIC_KON_117 Logische Zerlegung des Konnektors in Anwendungs- und Netzkonnektor 22

Abbildung 3 PIC_KON_107 XML-Struktur des Status-Elements einer SOAP-Antwort 48

Abbildung 4: PIC_Kon_100 Informationsmodell des Konnektors 54

Abbildung 5: PIC_KON_101 Aufrufkontext der Operation 62

Abbildung 6: PIC_KON_118 Aktivitätsdiagramm zu „TUC_KON_000 Prüfe Zugriffsberechtigung“.....66

Abbildung 7 PIC_KON_071 Korrelationszustände eines eHealth-KT 84

Abbildung 8: PIC_KON_110 Aktivitätsdiagramm zu BeginneKartenterminalsitzung.....92

Abbildung 9: PIC_KON_057 Aktivitätsdiagramm zu PaireKartenterminal.....98

Abbildung 10: PIC_KON_111 Aktivitätsdiagramm zu „PIN verifizieren“.....129

Abbildung 11: PIC_KON_022 Grundsätzlicher Aufbau der Ereignisnachricht 175

Abbildung 12: PIC_KON_112 Aktivitätsdiagramm zu „Systemereignis absetzen“ 180

Abbildung 13: PIC_KON_058 Aktivitätsdiagramm „Daten hybrid verschlüsseln“.....209

Abbildung 14: PIC_KON_059 Aktivitätsdiagramm „Daten hybrid entschlüsseln“.....213

Abbildung 15: PIC_KON_103 Use Case Diagramm Signaturdienst (nonQES) 235
Abbildung 16: PIC_KON_104 Use Case Diagramm Signaturdienst (QES) 236
Abbildung 17: PIC_KON_113 Aktivitätsdiagramm zu „QES Signatur erstellen“.....247
Abbildung 18: PIC_KON_114 Aktivitätsdiagramm zu „Dokument QES signieren“.....260
Abbildung 19: PIC_KON_115 Kommunikationsregeln Konnektor 340
Abbildung 20: PIC_KON_105 Aktivitätsdiagramm Konnektoraktualisierung durchführen 411
Abbildung 21: PIC_KON_120 Abbildung von CardSessions auf logische Kanäle 523
Abbildung 22: PIC_KON_007 Übersicht Zeichensatz ISO646DE / DIN66003.....525
Abbildung 23 - Szenario einer einfachen Installation.....527
Abbildung 24 - Szenario einer Installation mit mehreren Behandlungsräumen.....529
Abbildung 25 - Szenario einer Integration der TI Produkte in eine bestehende Infrastruktur 530
Abbildung 26 - Szenario einer Integration der TI Produkte in eine bestehende Infrastruktur mit existierendem Router 532
Abbildung 27 – Szenario mit zentral gesteckten HBA und SMC-B 533
Abbildung 28 - Szenario mit zentralem Primärsystem als Clientsystem 535
Abbildung 29 - Szenario für den Zugriff.....537
Abbildung 30 Standalone-Szenario mit logischer Trennung im Konnektor 539
Abbildung 31 Standalone-Szenario mit physischer Trennung im Konnektor 541

A4 – Tabellenverzeichnis

Tabelle 1: TAB_KON_500 Wertetabelle Kartentypen.....29
Tabelle 2 TAB_KON_502 Fehlercodes „Betriebszustand“ 34
Tabelle 3 TAB_KON_503 Betriebszustand_Fehlerzustandsliste.....35
Tabelle 4 TAB_KON_504 Ausführungserlaubnis für Dienste in kritischen Fehlerzuständen 40
Tabelle 5 TAB_KON_505 Konfigurationswerte Missbrauchserkennung.....43
Tabelle 6 TAB_KON_506 Konfigurationsparameter der Clientsystem-Authentisierung....46
Tabelle 7: TAB_KON_812 Umgebungsabhängige Konfigurationsparameter 50
Tabelle 8 TAB_KON_507 Informationsmodell Entitäten.....54
Tabelle 9 TAB_KON_508 Informationsmodell Attribute.....57
Tabelle 10 TAB_KON_509 Informationsmodell Entitätenbeziehungen.....58
Tabelle 11 TAB_KON_510 Informationsmodell Constraints.....60
Tabelle 12: TAB_KON_511 - TUC_KON_000 „Prüfe Zugriffsberechtigung“.....63
Tabelle 13: TAB_KON_512 Zugriffsregeln Beschreibung 65
Tabelle 14: TAB_KON_513 Zugriffsregeln Regelzuordnung 67

Tabelle 15: TAB_KON_514 Zugriffsregeln Definition 68
Tabelle 16 TAB_KON_515 Fehlercodes TUC_KON_000 „Prüfe Zugriffsberechtigung“....69
Tabelle 17: TAB_KON_143 - TUC_KON_080 „Dokument validieren“ 71
Tabelle 18: TAB_KON_144 Übersicht Fehlercodes für „Dokument validieren“.....74
Tabelle 19: TAB_KON_516 Basisanwendung Dienstverzeichnisdienst.....76
Tabelle 20 TAB_KON_517 Schemabeschreibung Produktinformation (ProductInformation.xsd).....77
Tabelle 21 TAB_KON_518 Schemabeschreibung Serviceinformation (Serviceinformation.xsd) 78
Tabelle 22 TAB_KON_519 - TUC_KON_041 "Einbringen der Endpunktinformationen während der Bootup-Phase" 79
Tabelle 23: TAB_KON_520 Übersicht Fehler TUC_KON_041 "Einbringen der Endpunktinformationen während der Bootup-Phase".....80
Tabelle 24: TAB_KON_521 Schnittstelle der Basisanwendung Dienstverzeichnisdienst..80
Tabelle 25 TAB_KON_522 Parameterübersicht des Kartenterminaldienstes 82
Tabelle 26: TAB_KON_727 Terminalanzeigen beim Anfordern und Auswerfen von Karten 86
Tabelle 27: TAB_KON_039 - TUC_KON_050 „Beginne Kartenterminalsitzung“ 88
Tabelle 28: TAB_KON_523 Übersicht Fehlercodes für „Beginne Kartenterminalsitzung“.93
Tabelle 29: TAB_KON_524 - TUC_KON_054 „Kartenterminal hinzufügen“ 93
Tabelle 30: TAB_KON_525 Übersicht Fehlercodes für „Kartenterminal hinzufügen“.....94
Tabelle 31: TAB_KON_041 - TUC_KON_053 „Paire Kartenterminal“ 95
Tabelle 32: TAB_KON_113 Übersicht Fehler TUC_KON_053 „PaireKartenterminal“.....97
Tabelle 33: TAB_KON_526 - TUC_KON_055 „Befülle CT-Object“.....99
Tabelle 34: TAB_KON_112 - TUC_KON_051 „Mit Anwender über Kartenterminal interagieren“ 100
Tabelle 35: TAB_KON_114 Übersicht Fehler TUC_KON_051 „Mit Anwender über Kartenterminal interagieren“.....102
Tabelle 36: TAB_KON_723 - TUC_KON_056 „Karte anfordern“.....102
Tabelle 37: TAB_KON_724 Übersicht Fehler TUC_KON_056 „Karte anfordern“ 103
Tabelle 38: TAB_KON_725 - TUC_KON_057 „Karte auswerfen“.....104
Tabelle 39: TAB_KON_796 Übersicht Fehler TUC_KON_057 „Karte auswerfen“ 105
Tabelle 40: TAB_KON_722 Basisdienst Kartenterminaldienst 106
Tabelle 41: TAB_KON_716 Operation RequestCard 106
Tabelle 42: TAB_KON_717 Ablauf RequestCard.....107
Tabelle 43: TAB_KON_718 Übersicht Fehler Operation „RequestCard“ 108
Tabelle 44: TAB_KON_719 Operation EjectCard.....108
Tabelle 45: TAB_KON_720 Ablauf EjectCard 109

Tabelle 46: TAB_KON_721 Übersicht Fehler Operation „EjectCard“ 109
Tabelle 47 TAB_KON_527 Konfigurationswerte eines Kartenterminalobjekts.....110
Tabelle 48 TAB_KON_528 Informationsparamter des Kartenterminaldienstes 111
Tabelle 49 TAB_KON_529 Anzeigewerte zu einem Kartenterminalobjekt.....112
Tabelle 50 TAB_KON_530 Konfigurationswerte eines Kartenterminalobjekts.....113
Tabelle 51 TAB_KON_531 Parameterübersicht des Kartendienstes.....116
Tabelle 52: TAB_KON_090 Terminalanzeigen beim Eingeben der PIN am Kartenterminal 119
Tabelle 53: TAB_KON_734 - TUC_KON_001 „Karte öffnen“ 123
Tabelle 54: TAB_KON_735 - TUC_KON_026.....125
Tabelle 55: TAB_KON_087 - TUC_KON_012 „PIN verifizieren“ 126
Tabelle 56: TAB_KON_089 Übersicht Fehler TUC_KON_012 „PIN verifizieren“.....129
Tabelle 57: TAB_KON_736 - TUC_KON_019 „PIN ändern“.....130
Tabelle 58: TAB_KON_093 Übersicht Fehler TUC_KON_019 „PIN ändern“.....132
Tabelle 59: TAB_KON_236 – TUC_KON_021 „PIN entsperren“.....133
Tabelle 60: TAB_KON_193 Übersicht Fehler TUC_KON_021 „PIN entsperren“.....135
Tabelle 61 TAB_KON_532 – TUC_KON_022 „Liefere PIN-Status“.....135
Tabelle 62: TAB_KON_091 Übersicht Fehler TUC_KON_022 „Liefere PIN-Status“.....136
Tabelle 63: TAB_KON_533 - TUC_KON_023 “Karte reservieren” 137
Tabelle 64: TAB_KON_534 Übersicht Fehler TUC_KON_023“Karte reservieren“.....138
Tabelle 65: TAB_KON_096 - TUC_KON_005 “Card-to-Card authentisieren”.....139
Tabelle 66: TAB_KON_673 AuthMode für C2C 141
Tabelle 67: TAB_KON_674 Erlaubte Parameterkombinationen und resultierende CV-Zertifikate für C2C.....141
Tabelle 68: TAB_KON_535 Übersicht Fehler TUC_KON_005 „Card-to-Card authentisieren“.....142
Tabelle 69: TAB_KON_218 – TUC_KON_202 „LeseDatei“.....142
Tabelle 70: TAB_KON_536 Übersicht Fehler TUC_KON_202 „Lese Datei“.....143
Tabelle 71: TAB_KON_219 – TUC_KON_203 „SchreibeDatei“ 143
Tabelle 72: TAB_KON_537 Übersicht Fehler TUC_KON_203 „Schreibe Datei“.....144
Tabelle 73: TAB_KON_538 – TUC_KON_209 „LeseRecord“.....145
Tabelle 74: TAB_KON_539 Übersicht Fehler TUC_KON_209 „Lese Record“.....146
Tabelle 75: TAB_KON_224 – TUC_KON_210 „SchreibeRecord“ 146
Tabelle 76: TAB_KON_540 Übersicht Fehler TUC_KON_210 „Schreibe Record“ 147
Tabelle 77: TAB_KON_228 – TUC_KON_214 „FügeHinzuRecord“ 148
Tabelle 78: TAB_KON_541 Übersicht Fehler TUC_KON_214 „FügeHinzuRecord“ 149
Tabelle 79: TAB_KON_229 – TUC_KON_215 „SucheRecord“ 149

Tabelle 80: TAB_KON_542 Übersicht Fehler TUC_KON_215 „SucheRecord“ 150
Tabelle 81: TAB_KON_110 - TUC_KON_018 „eGK-Sperrung prüfen“.....150
Tabelle 82: TAB_KON_239 Übersicht Fehler TUC_KON_018 „eGK-Sperrung prüfen“ ..151
Tabelle 83: TAB_KON_108 - TUC_KON_006 „Datenzugriffsaudit eGK schreiben“.....153
Tabelle 84: TAB_KON_238 Übersicht Fehler TUC_KON_006 „Datenzugriffsaudit eGK schreiben“.....154
Tabelle 85: TAB_KON_231 – TUC_KON_218 „Signiere“.....154
Tabelle 86: TAB_KON_543 Übersicht Fehler TUC_KON_218 „Signiere“.....155
Tabelle 87: TAB_KON_232 – TUC_KON_219 „Entschlüssele“.....155
Tabelle 88: TAB_KON_210 Übersicht Fehler TUC_KON_219 „Entschlüssele“.....156
Tabelle 89: TAB_KON_215 TUC_KON_200 „SendeAPDU“.....157
Tabelle 90: TAB_KON_216 Übersicht Fehler TUC_KON_200 „SendeAPDU“.....157
Tabelle 91: TAB_KON_737 - TUC_KON_024 „Karte zurücksetzen“ 158
Tabelle 92: TAB_KON_544 Übersicht Fehler TUC_KON_024 „Karte zurücksetzen“.....159
Tabelle 93: TAB_KON_230 – TUC_KON_216 „LeseZertifikat“ 159
Tabelle 94: TAB_KON_209 Übersicht Fehler TUC_KON_216 „LeseZertifikat“.....160
Tabelle 95: TAB_KON_038 Basisanwendung Karten- und Kartenterminaldienst.....160
Tabelle 96: TAB_KON_047 Operation VerifyPin.....160
Tabelle 97: TAB_KON_738 Ablauf VerifyPin 162
Tabelle 98: TAB_KON_545 Übersicht Fehler Operation „VerifyPin“.....163
Tabelle 99: TAB_KON_049 Operation ChangePin.....163
Tabelle 100: TAB_KON_546 Ablauf ChangePin 165
Tabelle 101: TAB_KON_547 Übersicht Fehler Operation „ChangePin“ 165
Tabelle 102: TAB_KON_051 Operation GetPinStatus 165
Tabelle 103: TAB_KON_548 Ablauf GetPinStatus.....167
Tabelle 104: TAB_KON_549 Übersicht Fehler Operation „GetPinStatus“ 167
Tabelle 105: TAB_KON_053 Operation UnblockPin 168
Tabelle 106: TAB_KON_550 Ablauf UnblockPIN.....169
Tabelle 107: TAB_KON_551 Übersicht Fehler Operation „UnblockPin“.....170
Tabelle 108: TAB_KON_055 Operation AuthorizeSMC 170
Tabelle 109: TAB_KON_552 Ablauf AuthorizeSMC.....171
Tabelle 110: TAB_KON_553 Übersicht Fehler Operation „AuthorizeSMC“ 171
Tabelle 111: TAB_KON_554 Konfiguration des Kartendienstes.....172
Tabelle 112: TAB_KON_555 - TUC_KON_025 „Initialisierung Kartendienst“.....172
Tabelle 113: TAB_KON_030 Ereignisnachricht.....175
Tabelle 114: TAB_KON_556 - TUC_KON_256 „Systemereignis absetzen“.....176

Tabelle 115: TAB_KON_557 Übersicht Fehler TUC_KON_256 „Systemereignis absetzen“ 181
Tabelle 116: TAB_KON_558 - TUC_KON_252 „Liefere KT_Liste“....181
Tabelle 117: TAB_KON_559 - TUC_KON_253 „Liefere Karten_Liste“....182
Tabelle 118: TAB_KON_560 Übersicht Fehler TUC_KON_253 „Liefere Karten_Liste“ ..183
Tabelle 119: TAB_KON_561 - TUC_KON_254 „Liefere Ressourcendetails“....183
Tabelle 120: TAB_KON_562 Übersicht Fehler TUC_KON_254 „Liefere Ressourcendetails“....185
Tabelle 121 TAB_KON_029 Basisanwendung Systeminformationsdienst 185
Tabelle 122: TAB_KON_563 Operation GetCardTerminals 186
Tabelle 123: TAB_KON_564 Ablauf GetCardTerminals....188
Tabelle 124: Tab_KON_565 Operation GetCards....188
Tabelle 125: TAB_KON_566 Ablauf GetCards....191
Tabelle 126 TAB_KON_567 Fehlerfälle GetCards 191
Tabelle 127: TAB_KON_568 Operation GetResourceInformation....192
Tabelle 128: TAB_KON_569 Ablauf GetResourceInformation 194
Tabelle 129 TAB_KON_570 Fehlerfälle GetResourceInformation....195
Tabelle 130: TAB_KON_571 Operation Subscribe 195
Tabelle 131: TAB_KON_572 Ablauf Subscribe....196
Tabelle 132 TAB_KON_573 Fehlerfälle Subscribe 197
Tabelle 133: TAB_KON_574 Operation Unsubscribe....197
Tabelle 134: TAB_KON_575 Ablauf Unsubscribe 198
Tabelle 135 TAB_KON_576 Fehlerfälle Unsubscribe....199
Tabelle 136: TAB_KON_792 Operation RenewSubscriptions 199
Tabelle 137: TAB_KON_793 Ablauf RenewSubscriptions 200
Tabelle 138 TAB_KON_794 Fehlerfälle RenewSubscriptions 200
Tabelle 139: TAB_KON_577 Operation GetSubscription 201
Tabelle 140: TAB_KON_578 Ablauf GetSubscription....202
Tabelle 141 TAB_KON_579 Fehlerfälle GetSubscription 202
Tabelle 142 TAB_KON_580 Konfigurationswerte des Systeminformationsdienstes (Administrator) 203
Tabelle 143 TAB_KON_581 Verschlüsselungsdienst-Operationen für EVT_MONITOR_OPERATIONS....204
Tabelle 144: TAB_KON_739 - TUC_KON_070 „Daten hybrid verschlüsseln“ 204
Tabelle 145: TAB_KON_073 Vorgaben zum Format verschlüsselter XML-Dokumente..210
Tabelle 146: TAB_KON_740 Übersicht Fehlercodes für „Daten hybrid verschlüsseln“...210
Tabelle 147: TAB_KON_140 - TUC_KON_071 „Daten hybrid entschlüsseln“ 210

Tabelle 148: TAB_KON_142 Übersicht Fehlercodes für „Daten hybrid entschlüsseln“...214
Tabelle 149: TAB_KON_741 - TUC_KON_072 „Daten symmetrisch verschlüsseln“.....214
Tabelle 150: TAB_KON_742 Übersicht Fehlercodes für „Daten symmetrisch verschlüsseln“.....215
Tabelle 151: TAB_KON_743 - TUC_KON_073 „Daten symmetrisch entschlüsseln“.....215
Tabelle 152: TAB_KON_744 Übersicht Fehlercodes für „Daten symmetrisch entschlüsseln“.....216
Tabelle 153: TAB_KON_745 Basisdienst Verschlüsselungsdienst.....216
Tabelle 154: TAB_KON_071 Operation EncryptDocument.....217
Tabelle 155: TAB_KON_141 Übersicht Fehlercodes für „EncryptDocument“.....220
Tabelle 156: TAB_KON_746 Ablauf EncryptDocument.....221
Tabelle 157: TAB_KON_747 KeyReference für Encrypt-/DecryptDocument.....221
Tabelle 158: TAB_KON_075 Operation DecryptDocument.....222
Tabelle 159: TAB_KON_145 Übersicht Fehlercodes für „DecryptDocument“.....223
Tabelle 160: TAB_KON_076 Ablauf DecryptDocument.....223
Tabelle 161: TAB_KON_582 – Signaturverfahren Dokumentensignatur.....225
Tabelle 162: TAB_KON_585 – Zusätzliche Signaturverfahren für Dokumentensignaturprüfung.....225
Tabelle 163: TAB_KON_778 - Einsatzbereich der Signaturvarianten für XAdES, CAdES und PAdES.....226
Tabelle 164: TAB_KON_583 – Default-Signaturverfahren.....228
Tabelle 165: TAB_KON_584 nonQES-Operationen für EVT_MONITOR_OPERATIONS.....228
Tabelle 166: TAB_KON_780 – Signaturverfahren Externe Authentisierung.....234
Tabelle 167: TAB_KON_748 - TUC_KON_155 „Dokumente zur Signatur vorbereiten“..236
Tabelle 168: TAB_KON_586 Übersicht Fehlercodes für „Dokumente zur Signatur vorbereiten“.....238
Tabelle 169: TAB_KON_749 - TUC_KON_165 „Signaturvoraussetzungen für nonQES prüfen“.....239
Tabelle 170: TAB_KON_587 Übersicht Fehlercodes für „Signaturvoraussetzungen für nonQES prüfen“.....239
Tabelle 171: TAB_KON_750 - TUC_KON_166 „nonQES Signaturen erstellen“.....240
Tabelle 172: TAB_KON_120 Übersicht Fehlercodes für „nonQES Signaturen erstellen“.....241
Tabelle 173: TAB_KON_751 - TUC_KON_152 „Signaturvoraussetzungen für QES prüfen“.....241
Tabelle 174: TAB_KON_588 Übersicht Fehlercodes für „Signaturvoraussetzungen für QES prüfen“.....242
Tabelle 175: TAB_KON_131 - TUC_KON_153 „Dokumentenliste im xTV anzeigen“.....242

Tabelle 176: TAB_KON_589 Übersicht Fehlercodes für „Dokumentenliste im xTV anzeigen“ 244
Tabelle 177: TAB_KON_752 - TUC_KON_154 „QES Signaturen erstellen“ 245
Tabelle 178: TAB_KON_126 Übersicht Fehlercodes für „QES Signaturen erstellen“ 248
Tabelle 179: TAB_KON_293 - TUC_KON_168 „Einzelsignatur QES erstellen“ 248
Tabelle 180: TAB_KON_590 Übersicht Fehlercodes für „Einzelsignatur QES erstellen“ 249
Tabelle 181: TAB_KON_753 - TUC_KON_160 „Dokumente nonQES signieren“ 249
Tabelle 182: TAB_KON_127 Übersicht Fehlercodes für „Dokumente nonQES signieren“ 251
Tabelle 183: TAB_KON_121 - TUC_KON_161 „nonQES Dokumentsignatur prüfen“ 252
Tabelle 184: TAB_KON_124 Übersicht Fehlercodes für „nonQES Dokumentensignatur prüfen“ 255
Tabelle 185: TAB_KON_754 Übersicht Status für Prüfung einer Dokumentensignatur ..255
Tabelle 186: TAB_KON_755 - TUC_KON_150 „Dokumente QES signieren“ 257
Tabelle 187: TAB_KON_128 Übersicht Fehlercodes für „Dokument QES signieren“ 261
Tabelle 188: TAB_KON_192 Verhalten des Konnektors beim Abbruch einer Stapelsignatur 265
Tabelle 189: TAB_KON_591 - TUC_KON_151 „QES-Dokumentensignatur prüfen“ 267
Tabelle 190: TAB_KON_592 Übersicht Fehlercodes für „QES Dokumentensignatur prüfen“ 270
Tabelle 191: TAB_KON_593 Übersicht Status für Prüfung einer Dokumentensignatur ..271
Tabelle 192: TAB_KON_197 Basisdienst Signaturdienst (nonQES und QES) 274
Tabelle 193: TAB_KON_065 Operation SignDocument (nonQES und QES) 274
Tabelle 194: TAB_KON_756 Ablauf Operation SignDocument (nonQES und QES) 283
Tabelle 195: TAB_KON_757 Übersicht Fehler Operation SignDocument (nonQES und QES) 283
Tabelle 196: TAB_KON_758 Zertifikat und privater Schlüssel je Karte für Sign/VerifyDocument (nonQES) 284
Tabelle 197: TAB_KON_759 Zertifikat und privater Schlüssel je Karte für Sign/VerifyDocument (QES) 284
Tabelle 198: TAB_KON_066 Operation VerifyDocument (nonQES und QES) 284
Tabelle 199: TAB_KON_760 Ablauf Operation VerifyDocument (nonQES und QES) 288
Tabelle 200: TAB_KON_761 Übersicht Fehler Operation VerifyDocument (nonQES und QES) 289
Tabelle 201: TAB_KON_781 Operation ExternalAuthenticate 289
Tabelle 202: TAB_KON_782 Ablauf Operation ExternalAuthenticate 291
Tabelle 203: TAB_KON_783 Übersicht Fehler Operation ExternalAuthenticate 292
Tabelle 204: TAB_KON_784 Privater Schlüssel je Karte für ExternalAuthenticate 292
Tabelle 205: TAB_KON_264 Operation SetXTVView 292

Tabelle 206: TAB_KON_762 Ablauf Operation SetXTVView 293
Tabelle 207: TAB_KON_763 Übersicht Fehler Operation SetXTVView 293
Tabelle 208: TAB_KON_265 Operation GetXTVView 294
Tabelle 209: TAB_KON_764 Ablauf Operation GetXTVView 295
Tabelle 210: TAB_KON_765 Übersicht Fehler Operation GetXTVView 295
Tabelle 211 TAB_KON_596 Konfigurationswerte des Signaturdienstes (Administrator) 295
Tabelle 212: TAB_KON_597 Operationen in EVT_MONITOR_OPERATIONS 298
Tabelle 213: TAB_KON_766 TUC_KON_032 „TSL aktualisieren“ 298
Tabelle 214: TAB_KON_598 Übersicht Fehlercodes für „TSL aktualisieren“.....300
Tabelle 215: TAB_KON_767 TUC_KON_040 „CRL aktualisieren“.....300
Tabelle 216: TAB_KON_599 Übersicht Fehlercodes für „CRL aktualisieren“ 301
Tabelle 217: TAB_KON_768 TUC_KON_033 „Zertifikatsablauf prüfen“.....302
Tabelle 218: TAB_KON_600 Übersicht Fehlercodes für „Zertifikatsablauf prüfen“ 303
Tabelle 219: TAB_KON_769 TUC_KON_037 „Zertifikat prüfen“ 303
Tabelle 220: TAB_KON_601 Übersicht Fehlercodes für das Prüfen eines Zertifikats.....307
Tabelle 221: TAB_KON_770 TUC_KON_034 „Zertifikatsinformationen extrahieren“ 307
Tabelle 222: TAB_KON_602 Übersicht Fehlercodes für „Zertifikatsinformationen extrahieren“ 309
Tabelle 223: TAB_KON_771 Basisanwendung Zertifikatsdienst 309
Tabelle 224: TAB_KON_676 Operation CheckCertificateExpiration.....310
Tabelle 225: TAB_KON_677 Ablauf CheckCertificateExpiration 310
Tabelle 226 TAB_KON_603 Fehlerfälle CheckCertificateExpiration.....311
Tabelle 227: TAB_KON_678 Operation ReadCardCertificate 312
Tabelle 228: TAB_KON_679 Ablauf ReadCardCertificate.....313
Tabelle 229 TAB_KON_604 Fehlerfälle ReadCardCertificate 314
Tabelle 230: TAB_KON_795 Operation VerifyCertificate 314
Tabelle 231: TAB_KON_797 Ablauf VerifyCertificate.....315
Tabelle 232 TAB_KON_800 Fehlerfälle VerifyCertificate 316
Tabelle 233: TAB_KON_772 TUC_KON_035 „Zertifikatsdienst initialisieren“ 317
Tabelle 234: TAB_KON_605 Übersicht Fehlercodes für „Zertifikatsdienst initialisieren“.318
Tabelle 235: TAB_KON_606 Konfiguration des Zertifikatsdienstes.....318
Tabelle 236 TAB_KON_733 Einsehbare Konfigurationsparameter des Zertifikatsdienstes 319
Tabelle 237: TAB_KON_607 - TUC_KON_271 „Schreibe Protokolleintrag“ 322
Tabelle 238: TAB_KON_608 Übersicht Fehler TUC_KON_271 „Schreibe Protokolleintrag“ 324

Tabelle 239 TAB_KON_609 Konfigurationswerte des Protokollierungsdienstes (Administrator) 325
Tabelle 240 TAB_KON_610 - TUC_KON_272 " Initialisierung Protokollierungsdienst" ..326
Tabelle 241: TAB_KON_611 Übersicht Fehler TUC_KON_272 „Initialisiere Protokollierungsdienst“ 327
Tabelle 242: TAB_KON_773 - TUC_KON_110 "Kartenbasierte TLS-Verbindung aufbauen" 328
Tabelle 243: TAB_KON_612 Übersicht Fehler TUC_KON_110 "Kartenbasierte TLS-Verbindung aufbauen" 329
Tabelle 244: TAB_KON_774 - TUC_KON_111 "Kartenbasierte TLS-Verbindung abbauen" 330
Tabelle 245: TAB_KON_613 Übersicht Fehler TUC_KON_111 "Kartenbasierte TLS-Verbindung abbauen" 330
Tabelle 246: TAB_KON_805 - TUC_KON_290 „LDAP-Verbindung aufbauen“ 331
Tabelle 247: TAB_KON_815 - TUC_KON_291 „Verzeichnis abfragen“ 333
Tabelle 248: TAB_KON_816 - TUC_KON_292 „LDAP-Verbindung trennen" 333
Tabelle 249: TAB_KON_817 - TUC_KON_293 „Verzeichnisabfrage abbrechen" 334
Tabelle 250: TAB_KON_680 Mapping der Netzwerksegmente 338
Tabelle 251: TAB_KON_681 Definition der vom Konnektor verwendeten VPN-Tunnel ..338
Tabelle 252: TAB_KON_682 Definition der Konnektor IP-Adressen 339
Tabelle 253: TAB_KON_614 - TUC_KON_305 „LAN-Adapter initialisieren“ 350
Tabelle 254: TAB_KON_615 Übersicht Fehler TUC_KON_305 „LAN-Adapter initialisieren“ 351
Tabelle 255: TAB_KON_616 - TUC_KON_306 "WAN-Adapter initialisieren" 351
Tabelle 256: TAB_KON_617 Übersicht Fehler TUC_KON_306 "WAN-Adapter initialisieren" 352
Tabelle 257: TAB_KON_622 - TUC_KON_304 „Netzwerk-Routen einrichten“ 353
Tabelle 258: TAB_KON_623 Übersicht Fehler TUC_KON_304 „Netzwerk-Routen einrichten“ 355
Tabelle 259: TAB_KON_683 LAN-Adapter IP-Konfiguration 356
Tabelle 260: TAB_KON_684 LAN-Adapter Erweiterte Parameter 357
Tabelle 261: TAB_KON_685 WAN-Adapter IP-Konfiguration 357
Tabelle 262: TAB_KON_686 WAN-Adapter Erweiterte Parameter 358
Tabelle 263: TAB_KON_624 - Konfigurationsparameter der Anbindung LAN/WAN" 359
Tabelle 264: TAB_KON_625 - Konfigurationsparameter Firewall-Schnittstelle 362
Tabelle 265 TAB_KON_626 "Liefere Netzwerkinformationen über DHCP" 363
Tabelle 266 TAB_KON_627 „Aktivierung des DHCP-Servers“ 365
Tabelle 267 TAB_KON_628 "Basiskonfiguration des DHCP-Servers" 365

Tabelle 268 TAB_KON_629 "Client-Gruppenspezifische Konfigurationsoptionen des Konnektor-DHCP-Servers" 365
Tabelle 269 TAB_KON_630 - TUC_KON_343 "Initialisierung DHCP-Server" 367
Tabelle 270: TAB_KON_631 Übersicht Fehler TUC_KON_343 "Initialisierung DHCP-Server" 368
Tabelle 271 TAB_KON_632 - TUC_KON_341 "DHCP Informationen beziehen" 369
Tabelle 272: TAB_KON_633 Übersicht Fehler TUC_KON_341 „DHCP-Informationen beziehen“ 370
Tabelle 273 TAB_KON_634 "Konfiguration des DHCP-Clients" 371
Tabelle 274: TAB_KON_635 - TUC_KON_321 „Verbindung zu dem VPN-Konzentrator der TI aufbauen“ 373
Tabelle 275: TAB_KON_636 Übersicht Fehler TUC_KON_321 375
Tabelle 276: TAB_KON_637 - TUC_KON_322 „Verbindung zu dem VPN-Konzentrator der SIS aufbauen“ 375
Tabelle 277: TAB_KON_638 Übersicht Fehler TUC_KON_322 377
Tabelle 278: TAB_KON_639 - Konfigurationsparameter VPN-Client 377
Tabelle 279 TAB_KON_640 Zustandswerte für Konnektor NTP-Server 380
Tabelle 280 TAB_KON_776 TUC_KON_351 "Liefere Systemzeit" 381
Tabelle 281: TAB_KON_641 Übersicht Fehler TUC_KON_351 "Liefere Systemzeit" 381
Tabelle 282 TAB_KON_642 Operation sync_Time 382
Tabelle 283 TAB_KON_643 Konfiguration des Konnektor NTP-Servers 382
Tabelle 284 TAB_KON_730 Einsehbare Konfigurationsparameter des Konnektor NTP-Servers 383
Tabelle 285 TAB_KON_644 - TUC_KON_352 "Initialisierung Zeitdienst " 383
Tabelle 286: TAB_KON_645 Übersicht Fehler TUC_KON_352 "Initialisierung Zeitdienst " 384
Tabelle 287: TAB_KON_687 DNS-Forwards des DNS-Servers 385
Tabelle 288: TAB_KON_646 - TUC_KON_361 „DNS-Namen auflösen“ 386
Tabelle 289: TAB_KON_647 Übersicht Fehler TUC_KON_361 „DNS Namen auflösen“ 387
Tabelle 290: TAB_KON_648 - TUC_KON_362 „Liste der Dienste abrufen“ 387
Tabelle 291: TAB_KON_649 Übersicht Fehler TUC_KON_362 „Liste der Dienste abrufen“ 388
Tabelle 292: TAB_KON_650 - TUC_KON_363 „Dienstdetails abrufen“ 388
Tabelle 293: TAB_KON_651 Übersicht Fehler TUC_KON_363 „Dienstdetails abrufen“ .389
Tabelle 294 TAB_KON_652 Basisanwendung Namensdienst 390
Tabelle 295: TAB_KON_653 Operation GetIPAddress 390
Tabelle 296: TAB_KON_654 - Konfigurationsparameter Namensdienst 391
Tabelle 297 TAB_KON_731 Einsehbare Konfigurationsparameter Namensdienst 392

Tabelle 298 TAB_KON_655 Konfigurationen der Benutzerverwaltung (Super-Administrator) 396
Tabelle 299 TAB_KON_656 Konfigurationen der Benutzerverwaltung....396
Tabelle 300 TAB_KON_657 Konfigurationsparameter des Konnektornamens....397
Tabelle 301 TAB_KON_658 Aktivieren/Deaktivieren von Leistungsumfängen 400
Tabelle 302 TAB_KON_659 Konnektor Standalone einsetzen....401
Tabelle 303 TAB_KON_660 Konnektor mit logischer Trennung....402
Tabelle 304 TAB_KON_661 Konfigurationsparameter der Konnektorfreischaltung....403
Tabelle 305 TAB_KON_732 Einsehbare Konfigurationsparameter der Konnektorfreischaltung 403
Tabelle 306 TAB_KON_662 Zustandswerte der Konnektorfreischaltung 403
Tabelle 307 TAB_KON_663 Konfigurationen des Remote Managements....407
Tabelle 308: TAB_KON_664 - TUC_KON_280 „Konnektoraktualisierung durchführen“.409
Tabelle 309: TAB_KON_665 Übersicht Fehlercodes für „Konnektoraktualisierung durchführen“ 410
Tabelle 310: TAB_KON_666 - TUC_KON_281 „Kartenterminalaktualisierung anstoßen“ 412
Tabelle 311: TAB_KON_667 Übersicht Fehlercodes für „Kartenterminalaktualisierung anstoßen“ 413
Tabelle 312: TAB_KON_668 - TUC_KON_282 „UpdateInformationen beziehen“ 414
Tabelle 313: TAB_KON_669 Übersicht Fehlercodes für „UpdateInformationen beziehen“ 415
Tabelle 314: TAB_KON_799 - TUC_KON_283 „Infrastruktur Konfiguration aktualisieren“ 415
Tabelle 315: Tab_Kon_726 Übersicht Fehler TUC_KON_283 „Infrastruktur Konfiguration aktualisieren“ 417
Tabelle 316 TAB_KON_644 - TUC_KON_284 "KSR-Client initialisieren" 417
Tabelle 317: TAB_KON_822 Übersicht Fehler TUC_KON_284 "Initialisierung Konfigurationsdienst" 418
Tabelle 318 TAB_KON_670 Konfigurationsparameter der Software-Aktualisierung....418
Tabelle 319 TAB_KON_820 Einsehbare Konfigurationsparameter der Software-Aktualisierung 419
Tabelle 320 TAB_KON_671 Anforderungen Klima 423
Tabelle 321 TAB_KON_672 Anforderungen Vibration 423
Tabelle 322: TAB_KON_779 „Profilierung der Signaturformate“ 444
Tabelle 323: TAB_KON_801 „Zu unterstützende XSL-FO-Objects und -Properties“....445
Tabelle 324: TAB_KON_802 „Bildungsvorschrift ShortTextSecure für PDF/A, TIFF und Text“ 450
Tabelle 325: SR_DF_SV_VHITG_ARZTBRIEF_01 - Elemente zur Steuerung der Signaturrichtline 451

Tabelle 326: SR_DF_SV_VHITG_ARZTBRIEF_02 - XML-Schemata 452
Tabelle 327: SR_DF_SV_VHITG_ARZTBRIEF_03 - erlaubte Root-Elemente 452
Tabelle 328: SR_DF_SV_VHITG_ARZTBRIEF_04 - XSL PDF.....453
Tabelle 329: SR_DF_SV_VHITG_ARZTBRIEF _05 - XSL Kurztext.....453
Tabelle 330: SR_DF_SV_VHITG_ARZTBRIEF_06 – Constraints SignDocument.....453
Tabelle 331: SR_DF_SV_VHITG_ARZTBRIEF_07 – Constraints VerifyDocument.....454
Tabelle 332: SR_DF_SV_EGES_01 - Elemente zur Steuerung der Signaturrichtline.....455
Tabelle 333: SR_DF_SV_EGES_02 - XML-Schemata.....455
Tabelle 334: SR_DF_SV_EGES_03 - erlaubte Root-Elemente.....456
Tabelle 335: SR_DF_EGES_04 - XSL PDF.....456
Tabelle 336: SR_DF_EGES_05 - XSL Kurztext.....456
Tabelle 337: SR_DF_SV_EGES_06 – Constraints SignDocument 457
Tabelle 338: SR_DF_SV_EGES_07 – Constraints VerifyDocument 458
Tabelle 339: TAB_KON_688 Version der Schemas aus dem Namensraum des Konnektors 459
Tabelle 340: TAB_KON_798 Schnittstellenversionen 460
Tabelle 341 – TAB_KON_689 Konfigurationsparameter und Zustandswerte des Konnektors 462
Tabelle 342 – TAB_KON_777 Events Interne Mechanismen 488
Tabelle 343 - TAB_KON_691 Allgemeine Fehlercodes.....501
Tabelle 344 - TAB_KON_692 Fehlercodes Zugriffsberechtigungsdienst.....501
Tabelle 345 - TAB_KON_693 Fehlercodes Dokumentenvalidierungsdienst.....502
Tabelle 346 - TAB_KON_694 Fehlercodes Dienstverzeichnisdienst 502
Tabelle 347 - TAB_KON_695 Fehlercodes Kartenterminaldienst.....502
Tabelle 348 - TAB_KON_696 Fehlercodes Kartendienst 503
Tabelle 349 - TAB_KON_697 Fehlercodes Systeminformationsdienst.....505
Tabelle 350 - TAB_KON_698 Fehlercodes Verschlüsselungsdienst 505
Tabelle 351 - TAB_KON_699 Fehlercodes Signaturdienst und SAK.....506
Tabelle 352 - TAB_KON_700 Fehlercodes Zertifikatsdienst 507
Tabelle 353 - TAB_KON_701 Fehlercodes Protokollierungsdienst 507
Tabelle 354 - TAB_KON_702 Fehlercodes TLS-Dienst 508
Tabelle 355 - TAB_KON_703 Fehlercodes Anbindung LAN/WAN 508
Tabelle 356 - TAB_KON_704 Fehlercodes DHCP-Server 508
Tabelle 357 - TAB_KON_705 Fehlercodes DHCP-Client.....508
Tabelle 358 - TAB_KON_706 Fehlercodes VPN-Client.....509
Tabelle 359 - TAB_KON_708 Fehlercodes Zeitdienst.....509
Tabelle 360 - TAB_KON_709 Fehlercodes Namensdienst und Dienstlokalisierung 509

Tabelle 361 - TAB_KON_710 Fehlercodes Software-Aktualisierungsdienst (KSR-Client) 509
Tabelle 362 - TAB_KON_711 Architektur der TI-Plattform, Berechtigt Fachmodule.....511
Tabelle 363 - TAB_KON_712 Architektur der TI-Plattform, Berechtigt Clientsysteme 514
Tabelle 364 - TAB_KON_713 Architektur der TI-Plattform, Berechtigt eHealth-KT 516
Tabelle 365 - TAB_KON_714 Architektur der TI-Plattform, Berechtigt Administrator.....516
Tabelle 366 - TAB_KON_715 Architektur der TI-Plattform, Berechtigt LE.....517

A5 - Referenzierte Dokumente

A5.1 – Dokumente der gematik

Die nachfolgende Tabelle enthält die Bezeichnung der in dem vorliegenden Dokument referenzierten Dokumente der gematik zur Telematikinfrastruktur. Der mit der vorliegenden Version korrelierende Entwicklungsstand dieser Konzepte und Spezifikationen wird pro Release in einer Dokumentenlandkarte definiert, Version und Stand der referenzierten Dokumente sind daher in der nachfolgenden Tabelle nicht aufgeführt. Deren zu diesem Dokument passende jeweils gültige Versionsnummer sind in der aktuellsten, von der gematik veröffentlichten Dokumentenlandkarte enthalten, in der 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_ORS1]	gematik: Testkonzept Online-Rollout (Stufe 1)
[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_eGK_ObjSys]	gematik: Spezifikation der elektronischen Gesundheitskarte eGK-Objektsystem - für Karten der Generation 2
[gemSpec_eGK_P1]	gematik: Die Spezifikation der elektronische Gesundheitskarte; Teil 1 – Spezifikation der elektrischen Schnittstelle - für Karten der Generation 1 und 1+
[gemSpec_eGK_P2]	gematik: Die Spezifikation der elektronische Gesundheitskarte; Teil 2 – Grundlegende Applikationen - für Karten der Generation 1 und 1+
[gemSpec_gSMC-K_ObjSys]	gematik: Spezifikation der gSMC-K Objektsystem
[gemSpec_gSMC-KT_ObjSys]	gematik: Spezifikation gSMC-KT Objektsystem
[gemSpec_HBA_ObjS	gematik: Spezifikation HBA Objektsystem

[Quelle]	Herausgeber: Titel
ys]	
[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

A5.2 – 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, veröffentlicht am 27.03.2013 im Bundesanzeiger www.bundesanzeiger.de (Suchebegriff "BAnz AT 27.03.2013 B4"). http://www.bundesnetzagentur.de/cln_1931/DE/Service-Funktionen/QualifizierteelektronischeSignatur/WelcheAufgabenhatdieBundesnetzagentur/GeeigneteAlgorithmenfestlegen/geeignetealgorithmenfestlegen-node.html
[BasicProfile1.2]	Basic Profile Version 1.2 http://www.ws-i.org/Profiles/BasicProfile-1.2-2010-11-09.html
[BSI_GK]	BSI (2005): IT-Grundschutz-Kataloge (11. Ergänzungslieferung 12/2008) https://www.bsi.bund.de/DE/Themen/weitereThemen/ITGrundschutzKataloge/itgrundschutzkataloge_node.html
[BSI-TR03114]	BSI (22.10.2007): Technische Richtlinie – Stapelsignatur mit dem Heilberufsausweis; Version 2.0 https://www.bsi.bund.de/cae/servlet/contentblob/477234/publicationFile/30605/BSI-TR-03114_pdf.pdf
[BSI TR-03120]	BSI (23.10.2007): BSI - Technische Richtlinie – Sichere Kartenterminalidentität (Betriebskonzept); Version 1.0 https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03120/BSI-TR-03120_pdf.pdf
[CAdES]	ETSI: *Electronic Signature Formats*, Electronic Signatures and

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
	Infrastructures (ESI) – Technical Specification, ETSI TS 101 733 V1.7.4, 2008-07, via http://www.etsi.org
[CanonXML1.1]	Canonical XML Version 1.1 http://www.w3.org/TR/2008/REC-xml-c14n11-20080502/
[CDA]	ISO/HL7 27932:2009 Data Exchange Standards -- HL7 Clinical Document Architecture, Release 2 http://www.hl7.org/documentcenter/private/standards/cda/r2/cda_r2_normativewebedition2010.zip
[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 Version 2.0, 20 January 2009 http://www.t7ev.org/themen/entwickler/common-pki-v20-spezifikation.html ISIS-MTT Core Specification, 2004, Version 1.1 http://www.common-pki.org/uploads/media/ISIS-MTT_Core_Specification_v1.1_03.pdf
[CMS]	Cryptographic Message Syntax (CMS), September 2009 http://tools.ietf.org/html/rfc5652
[DIN66003]	DIN 66003:1999 Informationsverarbeitung; 7-Bit-Code
[eAB]	http://www.d2d.de/uploads/media/PDF_Stylesheets_02.zip
[eGES]	Elektronische Gesamtaufstellung, Version 1.5 vom 08.02.2012 http://www.kvno.de/downloads/it_praxis/elektronische_gesamtaufstellung.zip
[HPC-P1]	Spezifikation des elektronischen Heilberufsausweises Version 2.3.2, 05.08.2009, Teil I: Kommandos, Algorithmen und Funktionen der COS Plattform http://www.bundesaerztekammer.de > Ärzte > e-Arztausweis/Telematik > Downloads > Technische Spezifikationen > HPC-Spezifikation 2.3.2 - COS (Teil 1) pdf
[HPC-P2]	Spezifikation des elektronischen Heilberufsausweises Version 2.3.2, 05.08.2009, Teil II: HPC - Anwendungen und Funktionen http://www.bundesaerztekammer.de > Ärzte > e-Arztausweis/Telematik > Downloads > Technische Spezifikationen > HPC-Spezifikation 2.3.2 - HPC (Teil 2) pdf
[HPC-P3]	Spezifikation des elektronischen Heilberufsausweises Version 2.3.2, 05.08.2009, Teil III: SMC - Anwendungen und Funktionen http://www.bundesaerztekammer.de > Ärzte > e-Arztausweis/Telematik > Downloads > Technische Spezifikationen > HPC-Spezifikation 2.3.2 - SMC (Teil 3) pdf
[HüKo06]	BSI (2006): Hühnlein, Detlef/Korte, Ulrike:Grundlagen der elektronischen Signatur
[IEEE802.3]	Technical Committee Computer Communications of the IEEE Computer Society, USA (1985): IEEE standards for local area networks: carrier sense multiple access with

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
	collision detection (CSMA/CD) access method and physical layer specifications ISBN: 0-7381-4253-0
[ISO8601]	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): Technische Spezifikation der Versichertenkarte Version: 2.08
[MIME]	RFC 2045, RFC 2046, RFC 2047, RFC 2048, RFC 2049
[NTPv4]	Internet Engineering Task Force (IETF) (06/2010): Network Time Protocol Version 4: Protocol and Algorithms Specification http://www.ietf.org/rfc/rfc5905.txt
[OASIS-AdES]	OASIS: Advanced Electronic Signature Profiles of the OASIS Digital Signature Service Version 1.0, OASIS Standard, http://docs.oasis-open.org/dss/v1.0/oasis-dss-profiles-AdES-spec-v1.0-os.pdf
[OASIS-DSS]	OASIS: Digital Signature Service Core Protocols, Elements, and Bindings, Version 1.0, OASIS Standard, via http://docs.oasis-open.org/dss/v1.0/oasis-dss-core-spec-v1.0-os.pdf
[OASIS-SP]	OASIS: Signature Policy Profile of the OASIS Digital Signature Services Version 1.0, Committee Draft 01, 18 May 2009, http://docs.oasis-open.org/dss-x/profiles/sigpolicy/oasis-dssx-1.0-profiles-sigpolicy-cd01.pdf
[OASIS-VR]	OASIS: Profile for comprehensive multi-signature verification reports for OASIS Digital Signature Services Version 1.0, Committee Specification 01, 12 November 2010, http://docs.oasis-open.org/dss-x/profiles/verificationreport/oasis-dssx-1.0-profiles-vr-cs01.pdf
[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 KON_ORS1]	Common Criteria Schutzprofil (Protection Profile) für einen Konnektor im elektronischen Gesundheitswesen Schutzprofil 2: Anforderungen an den Konnektor Online-Rollout (Stufe 1) BSI-CC-PP-0046

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
[PP NK]	Common Criteria Schutzprofil (Protection Profile) für einen Konnektor im elektronischen Gesundheitswesen Schutzprofil 1: Anforderungen an den Netzkonnektor (NK-PP) BSI-CC-PP-0047
[RFC792]	IETF (September 1981) INTERNET CONTROL MESSAGE PROTOCOL http://tools.ietf.org/html/rfc792
[RFC1034]	RFC 1034 (November 1987): Domain Names – Concepts and Facilities http://tools.ietf.org/html/rfc1034
[RFC1122]	RFC 1122 (Oktober 1989): Requirements for Internet Hosts -- Communication Layers http://tools.ietf.org/html/rfc1122
[RFC1812]	F. Baker (ed.): Requirements for IP Version 4 Routers, IETF RFC 1812, http://www.ietf.org/rfc/rfc1812.txt
[RFC1918]	RFC1918 (Februar 1996): Address Allocation for Private Internets http://tools.ietf.org/html/rfc1918
[RFC2119]	RFC 2119 (März 1997): Key words for use in RFCs to Indicate Requirement Levels S. Bradner, http://tools.ietf.org/html/rfc2119
[RFC2131]	Network Working Group (03/1997): Dynamic Host Configuration Protocol http://www.ietf.org/rfc/rfc2131.txt
[RFC2132]	Network Working Group (03/1997): DHCP Options and BOOTP Vendor Extensions http://www.ietf.org/rfc/rfc2132.txt
[RFC2617]	Network Working Group (06/1999): HTTP Authentication: Basic and Digest Access Authentication http://www.ietf.org/rfc/rfc2617.txt
[RFC2818]	Network Working Group (05/2000): HTTP Over TLS http://www.ietf.org/rfc/rfc2818.txt
[RFC3447]	B. Kaliski:_PKCS #1: RSA Encryption, Version 2.1, RFC3447,
[RFC2616]	Network Working Group (06/1999): Hypertext Transfer Protocol -- HTTP/1.1 http://www.ietf.org/rfc/rfc2616.txt
[RFC2644]	D. Senie: *Changing the Default for Directed Broadcasts in Routers*, IETF RFC 2644, http://www.ietf.org/rfc/rfc2644.txt
[RFC2663]	P. Srisuresh, M. Holdrege: *IP Network Address Translator (NAT) Terminology and Considerations*, IETF RFC 2663, http://www.ietf.org/rfc/rfc2663.txt
[RFC3022]	RFC 3022 (Januar 2001): Traditional IP Network Address Translator (Traditional NAT) http://tools.ietf.org/html/rfc3022
[RFC3275]	D. Eastlage, J. Reagle, D. Solo: *(Extensible Markup Language) XMLSignature Syntax and Processing*, IETF RFC 3275, via http://www.ietf.org/rfc/rfc3275.txt
[RFC3279]	W, Polk, R. Hously, L. Bassham: *Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List*

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
	(CRL) Profile, IETF RFC 3279, http://www.ietf.org/rfc/rfc3279.txt
[RFC3629]	Network Working Group (11/2003): UTF-8, a transformation format of ISO 10646 http://www.ietf.org/rfc/rfc3629.txt
[RFC3927]	Network Working Group (05/2005): Dynamic Configuration of IPv4 Link-Local Addresses http://www.ietf.org/rfc/rfc3927.txt
[RFC4122]	RFC 4122 (July 2005): A Universelly Unique Identifier UUID URN Namspace http://tools.ietf.org/html/rfc4122
[RFC4632]	Network Working Group (08/2006): Classless Inter-domain Routing (CIDR): The Internet Address Assignment and Aggregation Plan http://tools.ietf.org/html/rfc4632
[RFC5246]	RFC 5246 (August 2008): The Transport Layer Security (TLS) Protocol Version 1.2; http://tools.ietf.org/html/rfc5246
[RFC5652]	R. Housley: Cryptographic Message Syntax (CMS), RFC 5652 (September 2009) http://tools.ietf.org/html/rfc5652
[RFC 6598]	RFC 6598 (April 2012): IANA-Reserved IPv4 Prefix for Shared Address Space http://tools.ietf.org/html/rfc6598
[RFC6931]	RFC 6931 (April 2013): Additional XML Security Uniform Resource Identifiers (URIs) http://tools.ietf.org/html/rfc6931
[S/MIME]	RFC 5751 (Januar 2010): Secure/Multipurpose Internet Mail Extensions (S/MIME), Version 3.2, Message Specification http://www.ietf.org/rfc/rfc5751.txt
[SICCT]	TeleTrusT (17.12.2010): SICCT Secure Interoperable ChipCard Terminal, Version 1.21 http://www.teletrust.de/uploads/media/SICCT-Spezifikation-1.21.pdf
[SigG01]	Bundesgesetzblatt I (2001), S.876: Signaturgesetz vom 16. Mai 2001 (BGBl. I S. 876), zuletzt geändert durch Artikel 4 des Gesetzes vom 26. Februar 2007 (BGBl. I S. 179)
[SigV01]	Bundesgesetzblatt I (2001), S. 3074: Signaturverordnung vom 16. November 2001 (BGBl. I S. 3074), zuletzt geändert durch Artikel 9 Abs. 18 des Gesetzes vom 23. November 2007 (BGBl. I S. 2631)
[TIFF6]	TIFF Revision 6.0 (Final, June 3, 1992) http://partners.adobe.com/public/developer/en/tiff/TIFF6.pdf
[WSDL1.1]	W3C Note (15.03.2001): Web Services Description Language (WSDL) 1.1 http://www.w3.org/TR/wsdl
[XAdES]	European Telecommunications Standards Institute (ETSI): Technical Specication XML Advanced Electronic Signatures (XAdES). ETSI Technical Specication TS 101 903, Version 1.4.2, 2010

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
[XMLDSig]	W3C Recommendation (06.2008): XML-Signature Syntax and Processing http://www.w3.org/TR/2008/REC-xmldsig-core-20080610/
[XMLEnc]	XML Encryption Syntax and Processing W3C Candidate Recommendation 3 March 2012 http://www.w3.org/TR/xmlenc-core1/
[XPATH]	W3C Recommendation (14 December 2010) XML Path Language (XPath) 2.0 (Second Edition) http://www.w3.org/TR/2010/REC-xpath20-20101214/
[XSLT]	W3C Recommendation (23 January 2007) XSL Transformations (XSLT) Version 2.0 http://www.w3.org/TR/2007/REC-xslt20-20070123/
[XAdES Baseline 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 Baseline Profile]	European Telecommunications Standards Institute (ETSI): Electronic Signatures and Infrastructure (ESI); CAdES Baseline Profile; ETSI Technical Specification TS 103 173, Version 2.1.1, 2012-03
[PAdES Baseline Profile]	European Telecommunications Standards Institute (ETSI): Electronic Signatures and Infrastructure (ESI); PAdES Baseline Profile; ETSI Technical Specification TS 103 172, Version 2.1.1, 2012-03
[XSL]	W3C Recommendation (05.12.2006): Extensible Stylesheet language (XSL) Version1.1 http://www.w3.org/TR/2006/REC-xsl11-20061205/

Anhang B - Profilierung der Signatur- und Verschlüsselungsformate (normativ)

B1 – Profilierung der Verschlüsselungsformate

B2 – Profilierung der Signaturformate

Tabelle 322: TAB_KON_779 „Profilierung der Signaturformate“

Aspekt (QES/nonQES)	**Festlegung** (XML-Signatur/CMS-Signatur/PDF-Signatur)
Zertifikatsreferenz (QES und nonQES)	XML-Signatur Bei der Signaturerstellung ist das XML Element `SigningCertificate` 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. CMS-Signatur Bei der Signaturerstellung ist das Attribut `signing certificate reference` gemäß CAdES Kapitel 5.7.3 „Signing Certificate Reference Attributes“ [CAdES] anzulegen. Bei der Signaturprüfung ist es gemäß CAdES Kapitel 5.6.3 „Message signature verification process“ [CAdES] zu prüfen. Grundsätzlich sind auch Signaturen zu prüfen, die keine Zertifikatsreferenz enthalten. Das Prüfergebnis muss dann widerspiegeln, dass diese Sicherheitsfunktion nicht enthalten war. PDF-Signatur Bei der Signaturerstellung ist das Attribut `signing certificate reference` gemäß den Vorgaben aus PAdES Kapitel 4.4.3 „Signing Certificate Reference Attribute“ [PAdES] anzulegen. Bei der Signaturprüfung ist es gemäß PAdES Kapitel 4.6.1 „Signing Certificate Reference Validation“ [PAdES] zu prüfen. Grundsätzlich sind auch Signaturen zu prüfen, die keine Zertifikatsreferenz enthalten. Das Prüfergebnis muss dann widerspiegeln, dass diese Sicherheitsfunktion nicht enthalten war.
Parallelsignatur (QES und nonQES)	XML-Signatur Parallele Signaturen werden durch je ein `ds:signature`-Element pro Signatur abgebildet. Für die Signaturvariante „eveloping“ werden parallele Signaturen nicht angeboten. CMS-Signatur: Parallele Signaturen werden durch je einen SignerInfo-Container pro Signatur realisiert. PDF-Signatur:

Aspekt (QES/nonQES)	**Festlegung** (XML-Signatur/CMS-Signatur/PDF-Signatur)
	Parallele Signaturen werden nicht angeboten.
Dokumentexkludierende Gegensignatur (QES und nonQES)	XML-Signatur Die Implementierung erfolgt mittels Countersignature gemäß [XAdES], Kapitel 7.2.4. Jede vorhandene Parallel-Signatur wird gegensigniert. CMS-Signatur: Die Implementierung erfolgt mittels der Countersignature gemäß CMS-Spezifikation [RFC5652]. Jede vorhandene Parallel-Signatur wird gegensigniert. PDF-Signatur: Dokumentexkludierende Gegensignaturen werden nicht angeboten.
Dokumentinkludierende Gegensignatur (QES und nonQES)	XML-Signatur Wird als Enveloping XML-Signatur auf dem Gesamtdokument ausgeführt. CMS-Signatur: Dokumentinkludierende Gegensignaturen ist durch Signatur des gesamten SignedData Container zu realisieren. PDF-Signatur: Dokumentinkludierende Gegensignaturen sind gemäß [PAdES-1], Kapitel 4.4 PDF serial signatures, zu realisieren.

B3 – Profilierung der Transformation von XML-Dokumenten für die Anzeige

Funktional zu unterstützende XSL-FO-Objects und -Properties aus dem [XSL]-Standard.

Tabelle 323: TAB_KON_801 „Zu unterstützende XSL-FO-Objects und -Properties“

Kapitel	Formatting Object bzw. Property
§6.4.2	root
§6.4.3	declarations
§6.4.4	color-profile
§6.4.5	page-sequence
§6.4.7	layout-master-set
§6.4.8	page-sequence-master
§6.4.9	single-page-master-reference
§6.4.10	repeatable-page-master-reference
§6.4.11	repeatable-page-master-alternatives
§6.4.12	conditional-page-master-reference
§6.4.14	region-body
§6.4.15	region-before
§6.4.16	region-after
§6.4.17	region-start
§6.4.18	region-end

Kapitel	Formatting Object bzw. Property
§6.4.19	flow
§6.4.20	static-content
§6.4.21	title
§6.5.2	block
§6.6.2	bidi-override
§6.6.3	character
§6.6.5	external-graphic
§6.6.6	instream-foreign-object
§6.6.7	inline
§6.6.9	leader
§6.6.10	page-number
§6.7.4	table-column
§6.7.6	table-header
§6.7.7	table-footer
§6.7.8	table-body
§6.7.9	table-row
§6.7.10	table-cell
§6.8.2	list-block
§6.8.3	list-item
§6.8.4	list-item-body
§6.8.5	list-item-label
§6.9.2	basic-link
§6.11.1	bookmark-tree
§6.12.4	footnote-body
§6.13.4	wrapper
§6.13.5	marker
§6.13.6	retrieve-marker
§7.6.1	absolute-position
§7.6.2	top
§7.6.3	right
§7.6.4	bottom
§7.6.5	left
§7.8.2	background-color
§7.8.3	background-image
§7.8.4	background-repeat
§7.8.7	border-before-color
§7.8.8	border-before-style
§7.8.9	border-before-width
§7.8.10	border-after-color
§7.8.11	border-after-style
§7.8.12	border-after-width
§7.8.13	border-start-color
§7.8.14	border-start-style
§7.8.15	border-start-width
§7.8.16	border-end-color
§7.8.17	border-end-style

Kapitel	Formatting Object bzw. Property
§7.8.18	border-end-width
§7.8.19	border-top-color
§7.8.20	border-top-style
§7.8.21	border-top-width
§7.8.22	border-bottom-color
§7.8.23	border-bottom-style
§7.8.24	border-bottom-width
§7.8.25	border-left-color
§7.8.26	border-left-style
§7.8.27	border-left-width
§7.8.28	border-right-color
§7.8.29	border-right-style
§7.8.30	border-right-width
§7.8.31	padding-before
§7.8.32	padding-after
§7.8.33	padding-start
§7.8.34	padding-end
§7.8.35	padding-top
§7.8.36	padding-bottom
§7.8.37	padding-left
§7.8.38	padding-right
§7.9.4	font-size
§7.9.7	font-style
§7.10.1	country
§7.10.2	language
§7.10.3	script
§7.10.4	hyphenate
§7.10.5	hyphenation-character
§7.10.6	hyphenation-push-character-count
§7.10.7	hyphenation-remain-character-count
§7.11.1	margin-top
§7.11.2	margin-bottom
§7.11.3	margin-left
§7.11.4	margin-right
§7.11.7	start-indent
§7.11.8	end-indent
§7.14.1	alignment-adjust
§7.14.2	alignment-baseline
§7.14.3	baseline-shift
§7.14.5	dominant-baseline
§7.15.3	block-progression-dimension
§7.15.4	content-height
§7.15.5	content-width
§7.15.6	height
§7.15.7	inline-progression-dimension
§7.15.8	max-height

Kapitel	Formatting Object bzw. Property
§7.15.9	max-width
§7.15.10	min-height
§7.15.11	min-width
§7.15.12	scaling
§7.15.14	width
§7.16.2	hyphenation-ladder-count
§7.16.3	last-line-end-indent
§7.16.4	line-height
§7.16.5	line-height-shift-adjustment
§7.16.7	linefeed-treatment
§7.16.11	text-indent
§7.16.12	white-space-collapse
§7.17.1	character
§7.17.2	letter-spacing
§7.17.4	text-decoration
§7.17.6	text-transform
§7.17.8	word-spacing
§7.18.1	color
§7.20.1	break-after
§7.20.2	break-before
§7.20.6	orphans
§7.20.7	widows
§7.21.2	overflow
§7.21.3	reference-orientation
§7.21.4	span
§7.22.2	leader-pattern
§7.22.3	leader-pattern-width
§7.22.4	leader-length
§7.22.5	rule-style
§7.22.6	rule-thickness
§7.23.6	external-destination
§7.23.8	internal-destination
§7.25.1	marker-class-name
§7.25.3	retrieve-class-name
§7.53.4	retrieve-position
§7.25.5	retrieve-boundary
§7.26.1	format
§7.26.2	grouping-separator
§7.26.3	grouping-size
§7.26.4	letter-value
§7.27.1	blank-or-not-blank
§7.27.2	column-count
§7.27.3	column-gap
§7.27.4	extent
§7.27.5	flow-name
§7.27.6	force-page-count

Kapitel	Formatting Object bzw. Property
§7.27.7	initial-page-number
§7.27.8	master-name
§7.27.9	master-reference
§7.27.10	maximum-repeats
§7.27.12	odd-or-even
§7.27.14	page-position
§7.27.16	precedence
§7.27.17	region-name
§7.28.5	border-separation
§7.28.8	column-number
§7.28.9	column-width
§7.28.11	ends-row
§7.28.12	number-columns-repeated
§7.28.13	number-columns-spanned
§7.28.14	number-rows-spanned
§7.28.15	starts-row
§7.28.17	table-omit-footer-at-break
§7.28.18	table-omit-header-at-break
§7.30.11	provisional-label-separation
§7.30.12	provisional-distance-between-starts
§7.30.13	ref-id
§7.30.16	src
§7.31.2	background-position
§7.31.3	border
§7.31.4	border-bottom
§7.31.5	border-color
§7.31.6	border-left
§7.31.7	border-right
§7.31.8	border-style
§7.31.9	border-spacing
§7.31.10	border-top
§7.31.11	border-width
§7.31.14	margin
§7.31.15	padding
§7.31.16	page-break-after
§7.31.17	page-break-before
§7.31.18	page-break-inside
§7.31.20	position
§7.31.23	white-space
§7.31.24	xml:lang
§5.10.1	floor
§5.10.1	ceiling
§5.10.1	round
§5.10.1	min
§5.10.1	max
§5.10.1	abs

Kapitel	Formatting Object bzw. Property
§5.10.2	rgb
§5.10.2	rgb-icc
§5.10.2	system-color
§5.10.4	inherited-property-value
§5.10.4	label-end
§5.10.4	body-start
§5.10.4	from-parent
§5.10.4	from-nearest-specified-value
§5.10.4	from-table-column
§5.10.4	proportional-column-width

B4 – Bildungsvorschrift „ShortTextSecure“ für PDF/A, TIFF und Text

Tabelle TAB_KON_802 definiert die Bildungsvorschriften für den Kurztext „ShortTextSecure“ für die Dokumentenformate PDF/A, TIFF und Text.

Tabelle 324: TAB_KON_802 „Bildungsvorschrift ShortTextSecure für PDF/A, TIFF und Text“

Dokumentenformat	Bildungvorschrift ShortTextSecure
PDF/A	Der Konnektor muss den Kurztext „ShortTextSecure“ für PDF-Dokumente als die ersten 50 Zeichen des Titels aus den Dokumenteigenschaften/Dokumentinformationen bestimmen. Sind weniger als 50 Zeichen vorhanden, werden die vorhandenen Zeichen verwendet. Ist kein Titel vorhanden, gibt es keinen ShortTextSecure.
TIFF	Der Konnektor muss den Kurztext „ShortTextSecure“ für TIFF-Dokumente als die ersten 50 Zeichen des Tags ImageDescription (Code 0x010E) bestimmen. Sind weniger als 50 Zeichen vorhanden, werden die vorhandenen Zeichen verwendet. Ist kein ImageDescription vorhanden, gibt es keinen ShortTextSecure.
Text	Für das Dokumentenformat Text ist kein ShortTextSecure definiert.

Anhang C - QES-Dokumentenformate und -Signaturrichtlinien (normativ)

C1 – Dokumentenformat DF_SV_PDFA

DF_SV_PDFA, das Dokumentenformat für die sichere Anzeige von PDF-Dokumenten im xTV, ist definiert als PDF/A-2b [PDF/A-2] mit den nachfolgenden Einschränkungen:

- das Dokument enthält keine transparenten Elemente
- das Dokument bettet keine weiteren PDF-Dokumente ein
- das Dokument enthält keine JPEG 2000 Elemente
- das Dokument enthält keine Ebenen

C2 – Dokumentenformat DF_SV_TIFF

DF_SV_TIFF, das Dokumentenformat für die sichere Anzeige von TIFF-Dokumenten im xTV, ist definiert als TIFF 6.0: Part 1 Baseline TIFF [TIFF6] mit den nachfolgenden Einschränkungen:

- das Dokument enthält keine eingebetteten Unterdateien
- das Dokument enthält keine optionalen Felder

C3 – Signaturrichtlinie SR_DF_SV_VHITG_ARZTBRIEF

Die Signaturrichtlinie SR_DF_SV_VHITG_ARZTBRIEF zum Dokumentenformat DF_SV_VHITG_ARZTBRIEF wird im Folgenden definiert.

Gültigkeitsbereich

Die Signaturrichtline SR_DF_SV_VHITG_ARZTBRIEF wird für Aufrufe der Operation SignDocument bei den in Tabelle SR_DF_SV_VHITG_ARZTBRIEF_01 angegebenen Parametern im Kontext einer QES-Signatur wirksam. Für VerifyDocument wird sie wirksam, wenn in der Signatur die in Tabelle SR_DF_SV_VHITG_ARZTBRIEF_01 angegebene URI eingebettet ist. Die Signaturrichtlinie erlaubt eine enveloped oder eine enveloping XAdES-Signatur.

Tabelle 325: SR_DF_SV_VHITG_ARZTBRIEF_01 - Elemente zur Steuerung der Signaturrichtline

XML-Element oder XML-Attribut	URI Signaturrichtlinie
für SignDocument:	urn:gematik:fa:sak:cda:r1:v1

XML-Element oder XML-Attribut	URI Signaturrichtlinie

`/SIG:SignDocument/SIG:SignRequest/SIG:OptionalInputs/sp:GenerateUnderSignaturePolicy/sp:SignaturePolicyIdentifier`
 	
für VerifyDocument:
`ds:Signature/ds:Object/xades:QualifyingProperties/xades:SignedProperties/xades:SignedSignatureProperties/xades:SignaturePolicyIdentifier/xades:SignaturePolicyId/xades:SigPolicyId/xades:Identifier`	

Typkonformität

Das Dokument muss XML-Schema-valide zu den Schemata in Tabelle SR_DF_SV_VHITG_ARZTBRIEF_02 sein.

Tabelle 326: SR_DF_SV_VHITG_ARZTBRIEF_02 - XML-Schemata

Datei (Pfad und Name)	targetNamespace
`/fa/sak/cda/schemas/CDA.xsd`	`urn:hl7-org:v3`
`/fa/sak/cda/schemas/POCD_MT000040.xsd`	`urn:hl7-org:v3`
`/fa/sak/cda/coreschemas/datatypes.xsd`	
`/fa/sak/cda/coreschemas/datatypes-base.xsd`	
`/fa/sak/cda/coreschemas/voc.xsd`	`urn:hl7-org:v3`
`/fa/sak/cda/coreschemas/NarrativeBlock.xsd`	`urn:hl7-org:v3`

Die erlaubten Root-Elemente des Dokuments sind definiert durch Tabelle SR_DF_SV_VHITG_ARZTBRIEF_03.

Tabelle 327: SR_DF_SV_VHITG_ARZTBRIEF_03 - erlaubte Root-Elemente

URI Signaturrichtlinie	Namespace	Element
`urn:gematik:fa:sak:cda:r1:v1`	`urn:hl7-org:v3`	`ClinicalDocument`

Sichere Anzeige

Für die sichere Anzeige muss der Konnektor aus der XML-Eingangsdatei und dem in SR_DF_SV_VHITG_ARZTBRIEF_04 angegebenen und im Konnektor hinterlegten Stylesheet mittels XSL-FO-Prozessing ein PDF erzeugen.

Tabelle 328: SR_DF_SV_VHITG_ARZTBRIEF_04 - XSL PDF

URI Signaturrichtlinie	Datei (Pfad und Name)
`urn:gematik:fa:sak:cda:r1:v1`	`/fa/sak/cda/xsl/cda_arztbrief_pdf.xsl`

Kurztext ShortTextSecure

Der Kurztext zur sicheren Anzeige wird durch eine XSL-Transformation mit einem Stylesheet gemäß Tabelle SR_DF_SV_VHITG_ARZTBRIEF_05 aus dem Dokument erzeugt.

Tabelle 329: SR_DF_SV_VHITG_ARZTBRIEF _05 - XSL Kurztext

URI Signaturrichtlinie	Dokument-Namensraum	XSL-Datei (Pfad und Name)
`urn:gematik:fa:sak:cda:r1:v1`	`urn:hl7-org:v3`	`/fa/sak/cda/xsl/cda_arztbrief_kurztext.xsl`

Beispiel für einen so erzeugten ShortTextSecure:

CDA-R2: Mustermann, Max, 20050101

Härtung der Schnittstelle SignDocument

Der Konnektor muss bei Aufrufen der Schnittstelle SignDocument eine Parameterprüfung durchführen, in der die Konformität der Parameter gemäß Tabelle SR_DF_SV_VHITG_ARZTBRIEF_06 geprüft wird. Bei der Prüfung wird auf abweichende Werte sowie auf fehlende Elemente mit einem Fehler reagiert. Im Schnittstellenaufruf vorhandene optionale Elemente, die nicht in der Tabelle aufgeführt sind, werden ignoriert, d. h. bei der Bearbeitung des Aufrufs so behandelt, als ob sie im Aufruf nicht gesetzt wären.

Tabelle 330: SR_DF_SV_VHITG_ARZTBRIEF_06 – Constraints SignDocument

XML-Element oder –Attribut (XPath)	Wert	Kardinalität
`/SIG:SignDocument/SIG:SignRequest/SIG:Document/@ID`	Platzhalter für Dokumentbezeichner `CDA_DOC_ID`	1
`/SIG:SignDocument/SIG:SignRequest/SIG:Document/@RefURI`	Der Wert muss, wenn der Parameter verwendet wird, eine korrekte Referenz auf das Root-Element darstellen.	0-1
`/SIG:SignDocument/SIG:SignRequest/SIG:Document/@ShortText`	keine Einschränkung	1
`/SIG:SignDocument/SIG:SignRequest/SIG:Document/CONN:Base64XML`	keine Einschränkung	1
`/SIG:SignDocument/SIG:SignRequest/SIG:OptionalInputs`		1

XML-Element oder –Attribut (XPath)	Wert	Kardinalität
`/SIG:SignDocument/SIG:SignRequest/SIG:OptionalInputs/SIG:KeyReference`	`"C.QES"`	1
`/SIG:SignDocument/SIG:SignRequest/SIG:OptionalInputs/SIG:SignatureType`	`"urn:ietf:rfc:3275"`	1
`/SIG:SignDocument/SIG:SignRequest/SIG:OptionalInputs/dss:SignaturePlacement`	Einschränkung: Signatur muss als child des Root-Elements des Dokuments eingefügt werden.	0-1
`/SIG:SignDocument/SIG:SignRequest/SIG:OptionalInputs/SIG:IncludeObject`	Einschränkung: Signiert wird das Root-Elements des Dokuments.	0-1
`/SIG:SignDocument/SIG:SignRequest/SIG:OptionalInputs/dss:ReturnUpdatedSignature`	keine Einschränkung	0-1
`/SIG:SignDocument/SIG:SignRequest/SIG:OptionalInputs/SIG:AttributeCertificates`	keine Einschränkung	0-1
`/SIG:SignDocument/SIG:SignRequest/SIG:OptionalInputs/sp:GenerateUnderSignaturePolicy/sp:SignaturePolicyIdentifier`	`"urn:gematik:fa:sak:cda:r1:v1"`	1
`/SIG:SignDocument/SIG:SignRequest/SIG:IncludeRevocationInfo`	`”true” oder ”false”`	1
`/SIG:SignDocument/SIG:TvMode`	`”UNCONFIRMED” oder ”CONFIRMED”`	1

Härtung der Schnittstelle VerifyDocument

Der Konnektor muss bei Aufrufen der Schnittstelle VerifyDocument eine Parameterprüfung durchführen, in der die Konformität der Parameter gemäß Tabelle SR_DF_SV_VHITG_ARZTBRIEF_07 geprüft wird. Bei der Prüfung wird auf abweichende Werte sowie auf fehlende Elemente mit einem Fehler reagiert. Im Schnittstellenaufruf vorhandene optionale Elemente, die nicht in der Tabelle aufgeführt sind, werden ignoriert d. h. bei der Bearbeitung des Aufrufs so behandelt, als ob sie im Aufruf nicht gesetzt wären.

Tabelle 331: SR_DF_SV_VHITG_ARZTBRIEF_07 – Constraints VerifyDocument

XML-Element oder –Attribut (XPath)	Wert	Kardinalität
`/SIG:VerifyDocument/SIG:Document`		1

/SIG:VerifyDocument/SIG:Document/CONN:Base64XML	keine Einschränkung	1
/SIG:VerifyDocument/SIG:Document/@ShortText	keine Einschränkung	1
/SIG:VerifyDocument/SIG:OptionalInputs		1
/SIG:VerifyDocument/SIG:OptionalInputs/vr:ReturnVerificationReport	keine Einschränkung	0-1
/SIG:VerifyDocument/SIG:OptionalInputs/SIG:UseVerificationTime	keine Einschränkung	0-1
/SIG:VerifyDocument/SIG:TvMode	"UNCONFIRMED"	1

C4 – Signaturrichtlinie SR_DF_SV_EGES

Die Signaturrichtlinie SR_DF_SV_EGES zum Dokumentenformat DF_SV_EGES wird im Folgenden definiert.

Gültigkeitsbereich

Die Signaturrichtline SR_DF_SV_EGES wird für Aufrufe der Operation SignDocument bei den in Tabelle SR_DF_SV_EGES_01 angegebenen Parametern im Kontext einer QES-Signatur wirksam. Für VerifyDocument wird sie wirksam, wenn in der Signatur die in Tabelle SR_DF_SV_EGES_01 angegebene URI eingebettet ist. Die Signaturrichtlinie erlaubt eine enveloped oder eine enveloping XAdES-Signatur.

Tabelle 332: SR_DF_SV_EGES_01 - Elemente zur Steuerung der Signaturrichtline

XML-Element oder XML-Attribut	URI Signaturrichtlinie
für SignDocument: /SIG:SignDocument/SIG:SignRequest/SIG:OptionalInputs/ sp:GenerateUnderSignaturePolicy/sp:SignaturePolicyIdentifier für VerifyDocument: ds:Signature/ds:Object/xades:QualifyingProperties/xades:SignedProperties/xades:SignedSignatureProperties/xades:SignaturePolicyIdentifier/xades:SignaturePolicyId/xades:SigPolicyId/xades:Identifier	urn:gematik:fa:sak:eges:r1:v1 oder urn:gematik:fa:sak:eges:r2:v1 oder urn:gematik:fa:sak:eges:r3:v1

Typkonformität

Das Dokument muss XML-Schema-valide zu den Schemata in Tabelle SR_DF_SV_EGES_02 sein.

Tabelle 333: SR_DF_SV_EGES_02 - XML-Schemata

URI Signaturrichtlinie	Datei (Pfad und Name)	targetNamespace
urn:gematik:fa:sak:eges:r1:v1	/fa/sak/eges/schemas/gesamtaufstellung_arzt.xsd	urn::kvno-org/kvabrechnung
urn:gematik:fa:sak:eges:r2:v1	/fa/sak/eges/schemas/gesamtaufstellung_ermaechtigt.xsd	urn::kvno-org/kvabrechnung
urn:gematik:fa:sak:eges:r3:v1	/fa/sak/eges/schemas/gesamtaufstellung_krankenhaus.xsd	urn::kvno-org/kvabrechnung

Die erlaubten Root-Elemente des Dokuments sind definiert durch Tabelle SR_DF_SV_EGES_03.

Tabelle 334: SR_DF_SV_EGES_03 - erlaubte Root-Elemente

URI Signaturrichtlinie	Namespace	Element
urn:gematik:fa:sak:eges:r1:v1	urn::kvno-org/kvabrechnung	anlage_hvv
urn:gematik:fa:sak:eges:r2:v1	urn::kvno-org/kvabrechnung	anlage_hvv
urn:gematik:fa:sak:eges:r3:v1	urn::kvno-org/kvabrechnung	anlage_hvv

Sichere Anzeige

Für die sichere Anzeige muss der Konnektor aus der XML-Eingangsdatei und dem in SR_DF_SV_EGES_04 angegebenen und im Konnektor hinterlegten Stylesheet mittels XSL-FO-Prozessing ein PDF erzeugen.

Tabelle 335: SR_DF_EGES_04 - XSL PDF

URI Signaturrichtlinie	XSL-Datei (Pfad und Name)
urn:gematik:fa:sak:eges:r1:v1	/fa/sak/eges/xsl/GesamtFO-PDFA.xsl
urn:gematik:fa:sak:eges:r2:v1	/fa/sak/eges/xsl/GesamtFO-PDFA.xsl
urn:gematik:fa:sak:eges:r3:v1	/fa/sak/eges/xsl/GesamtFO-PDFA.xsl

Kurztext ShortTextSecure

Der Kurztext zur sicheren Anzeige wird durch eine XSL-Transformation mit einem Stylesheet gemäß Tabelle SR_DF_SV_EGES_05 aus dem Dokument erzeugt.

Tabelle 336: SR_DF_EGES_05 - XSL Kurztext

URI Signaturrichtlinie	XSL-Datei (Pfad und Name)
urn:gematik:fa:sak:eges:r1:v1	/fa/sak/eges/xsl/eges_kurztext.xsl
urn:gematik:fa:sak:eges:r2:v1	/fa/sak/eges/xsl/eges_kurztext.xsl

urn:gematik:fa:sak:eges:r3:v1	/fa/sak/eges/xsl/eges_kurztext.xsl

Beispiel für einen so erzeugten ShortTextSecure:

EGES: Hans Musterarzt, 2011-4

Härtung der Schnittstelle SignDocument

Der Konnektor muss bei Aufrufen der Schnittstelle SignDocument eine Parameterprüfung durchführen, in der die Konformität der Parameter gemäß Tabelle SR_DF_SV_EGES_06 geprüft wird. Bei der Prüfung wird auf abweichende Werte sowie auf fehlende Elemente mit einem Fehler reagiert. Im Schnittstellenaufruf vorhandene optionale Elemente, die nicht in der Tabelle aufgeführt sind, werden ignoriert, d. h. bei der Bearbeitung des Aufrufs so behandelt, als ob sie im Aufruf nicht gesetzt wären.

Tabelle 337: SR_DF_SV_EGES_06 – Constraints SignDocument

XML-Element oder –Attribut (XPath)	Wert	Kardinalität
/SIG:SignDocument/SIG:SignRequest /SIG:Document/@ID	Platzhalter für Dokumentbezeichner EGES_DOC_ID	1
/SIG:SignDocument/SIG:SignRequest /SIG:Document/@RefURI	Der Wert muss, wenn der Parameter verwendet wird, eine korrekte Referenz auf das Root-Element darstellen.	0-1
/SIG:SignDocument/SIG:SignRequest /SIG:Document/@ShortText	keine Einschränkung	1
/SIG:SignDocument/SIG:SignRequest /SIG:Document/CONN:Base64XML	keine Einschränkung	1
/SIG:SignDocument/SIG:SignRequest/SIG:OptionalInputs		1
/SIG:SignDocument/SIG:SignRequest /SIG:OptionalInputs/SIG:KeyRefere nce	"C.QES"	1
/SIG:SignDocument/SIG:SignRequest /SIG:OptionalInputs/SIG:Signature Type	"urn:ietf:rfc:3275"	1
/SIG:SignDocument/SIG:SignRequest /SIG:OptionalInputs/dss:Signature Placement	Einschränkung: Signatur muss als child des Root-Elements des Dokuments eingefügt werden.	0-1
/SIG:SignDocument/SIG:SignRequest /SIG:OptionalInputs/SIG:IncludeOb ject	Einschränkung: Signiert wird das Root-Elements des Dokuments.	0-1
/SIG:SignDocument/SIG:SignRequest /SIG:OptionalInputs/dss:ReturnUpd ated Signature	keine Einschränkung	0-1
/SIG:SignDocument/SIG:SignRequest /SIG:OptionalInputs/SIG:Attribute	keine Einschränkung	0-1

XML-Element oder –Attribut (XPath)	Wert	Kardinalität
`Certificates`		

`/SIG:SignDocument/SIG:SignRequest /SIG:OptionalInputs/ sp:GenerateUnderSignaturePolicy/sp:SignaturePolicyIdentifier`	`"urn:gematik:fa:sak:eges:r1:v1"`
	oder
	`"urn:gematik:fa:sak:eges:r2:v1"`
	oder
	`"urn:gematik:fa:sak:eges:r3:v1"`
		1

`/SIG:SignDocument/SIG:SignRequest /SIG:IncludeRevocationInfo`	`"true" oder "false"`	1
`/SIG:SignDocument/SIG:TvMode`	`"UNCONFIRMED" oder "CONFIRMED"`	1

Härtung der Schnittstelle VerifyDocument

Der Konnektor muss bei Aufrufen der Schnittstelle VerifyDocument eine Parameterprüfung durchführen, in der die Konformität der Parameter gemäß Tabelle SR_DF_SV_EGES_07 geprüft wird. Bei der Prüfung wird auf abweichende Werte sowie auf fehlende Elemente mit einem Fehler reagiert. Im Schnittstellenaufruf vorhandene optionale Elemente, die nicht in der Tabelle aufgeführt sind, werden ignoriert, d. h. bei der Bearbeitung des Aufrufs so behandelt, als ob sie im Aufruf nicht gesetzt wären.

Tabelle 338: SR_DF_SV_EGES_07 – Constraints VerifyDocument

XML-Element oder –Attribut (XPath)	Wert	Kardinalität
`/SIG:VerifyDocument/SIG:Document`		1
`/SIG:VerifyDocument/SIG:Document/ CONN:Base64XML`	keine Einschränkung	1
`/SIG:VerifyDocument/SIG:Document/ @ShortText`	keine Einschränkung	1
`/SIG:VerifyDocument/SIG:OptionalInputs`		1
`/SIG:VerifyDocument/SIG:OptionalInputs/vr:ReturnVerificationReport`	keine Einschränkung	0-1
`/SIG:VerifyDocument/SIG:OptionalInputs/SIG:UseVerificationTime`	keine Einschränkung	0-1
`/SIG:VerifyDocument/SIG:TvMode`	`"UNCONFIRMED"`	1

Anhang D - Übersicht über die verwendeten Versionen

Tabelle 339: TAB_KON_688 Version der Schemas aus dem Namensraum des Konnektors

Schemas aus dem Namensraum des Konnektors „http://ws.gematik.de/conn“		
	XSD Name	CardEvents.xsd
	XSD Schemaversion	6.0.0
	TargetNamespace	http://ws.gematik.de/conn/CardEvents/v6.0
	XSD Name	CardService.xsd
	XSD Schemaversion	8.1.0
	TargetNamespace	http://ws.gematik.de/conn/CardService/v8.1
	XSD Name	CardServiceCommon.xsd
	XSD Schemaversion	2.0.0
	TargetNamespace	http://ws.gematik.de/conn/CardServiceCommon/v2.0
	XSD Name	CardTerminalInfo.xsd
	XSD Schemaversion	8.1.0
	TargetNamespace	http://ws.gematik.de/conn/CardTerminalInfo/v8.1
	XSD Name	CardTerminalService.xsd
	XSD Schemaversion	1.1.0
	TargetNamespace	http://ws.gematik.de/conn/CardTerminalService/v1.1
	XSD Name	CertificateService.xsd
	XSD Schemaversion	6.0.0
	TargetNamespace	http://ws.gematik.de/conn/CertificateService/v6.0
	XSD Name	CertificateServiceCommon.xsd
	XSD Schemaversion	2.0.0
	TargetNamespace	http://ws.gematik.de/conn/CertificateServiceCommon/2.0
	XSD Name	ConnectorCommon.xsd
	XSD Schemaversion	5.0.0
	TargetNamespace	http://ws.gematik.de/conn/ConnectorCommon/v5.0
	XSD Name	ConnectorContext.xsd
	XSD Schemaversion	2.0.0
	TargetNamespace	http://ws.gematik.de/conn/ConnectorContext/v2.0
	XSD Name	EncryptionService.xsd
	XSD Schemaversion	6.1.0
	TargetNamespace	http://ws.gematik.de/conn/EncryptionService/v6.1
	XSD Name	EventService.xsd
	XSD Schemaversion	7.2.0
	TargetNamespace	http://ws.gematik.de/conn/EventService/ v7.2
	XSD Name	ServiceDirectory.xsd
	XSD Schemaversion	3.1.0
	TargetNamespace	http://ws.gematik.de/conn/ServiceDirectory/v3.1

	XSD Name	SignatureService.xsd
	XSD Schemaversion	7.3.0
	TargetNamespace	http://ws.gematik.de/conn/SignatureService/ v7.3

Tabelle 340: 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	8.1.0
	TargetNamespace	http://ws.gematik.de/conn/CardService/WSDL/v8.1
	verwendete XSDs	CardService.xsd, CardServiceCommon.xsd, ConnectorCommon.xsd, ConnectorContext.xsd, ProductInformation.xsd, TelematikError.xsd
Kartenterminaldienst (CardTerminalService)		
	WSDL Name	CardTerminalService.wsdl
	WSDL-Version	1.1.0
	TargetNamespace	http://ws.gematik.de/conn/CardTerminalService/ WSDL/v1.1
	verwendete XSDs	CardTerminalService.xsd, CardService.xsd, CardTerminalService.xsd, CardServiceCommon.xsd, ConnectorCommon.xsd, ConnectorContext.xsd, TelematikError.xsd
Systeminformationsdienst (EventService)		
	WSDL Name	EventService.wsdl
	WSDL-Version	7.2.0
	TargetNamespace	http://ws.gematik.de/conn/EventService/ WSDL/v7.2
	verwendete XSDs	CardService.xsd, CardServiceCommon.xsd, CardTerminalInfo.xsd, ConnectorCommon.xsd, ConnectorContext.xsd, EventService.xsd, ProductInformation.xsd, TelematikError.xsd
Zertifikatsdienst (CertificateService)		
	WSDL Name	CertificateService.wsdl
	WSDL-Version	6.0.0
	TargetNamespace	http://ws.gematik.de/conn/CertificateService/ WSDL/v6.0
	verwendete XSDs	CertificateService.xsd, CertificateServiceCommon.xsd, ConnectorCommon.xsd, ConnectorContext.xsd
Verschlüsselungsdienst (EncryptionService)		

	WSDL Name	EncryptionService.wsdl
	WSDL-Version	6.1.0
	TargetNamespace	http://ws.gematik.de/conn/EncryptionService/ WSDL/v6.1
	verwendete XSDs	ConnectorCommon.xsd, ConnectorContext.xsd, EncryptionService.xsd
Signaturdienst (SignatureService)		
	WSDL Name	SignatureService.wsdl
	WSDL-Version	7.3.0
	TargetNamespace	http://ws.gematik.de/conn/SignatureService/ WSDL/v7.3
	verwendete XSDs	CertificateServiceCommon.xsd, ConnectorCommon.xsd, ConnectorContext.xsd, SignatureService.xsd

Anhang E - Übersicht Konfigurationsparameter und Zustandswerte

Tabelle 341 – TAB_KON_689 Konfigurationsparameter und Zustandswerte des Konnektors

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
Interne Mechanismen			
	ANCL_TLS_MANDATORY	Enabled / Disabled	Der Administrator MUSS die verpflichtende Verwendung eines TLS gesicherten Kanals an- oder abschalten können. 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 verbunden Risiken informiert und darlegt, dass in diesem Fall der Leistungserbringer die Verantwortung für die Sicherstellung der vertraulichen Übertragung übernimmt. Default-Wert: Enabled
	ANCL_CAUT_MANDATORY	Enabled / Disabled	Der Administrator MUSS die verpflichtende Authentifizierung der Clientsysteme an- oder abschalten können. Defaultwert: Enabled
	ANCL_CAUT_MODE	CERTIFICATE / PASSWORD	Der Administrator MUSS konfigurieren können, welche Client Authentifizierungsmodus genutzt werden kann. Defaultwert: CERTIFICATE

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	ANCL_CCERT_LIST	Liste von X.509-Zertifikaten zugeordnet zu ClientID	Whitelist an importierten oder vom Konnektor erzeugten X.509-Zertifikaten und dazugehörenden Clientsystem IDs. Der Administrator MUSS die Liste der Zertifikate und den zugehörenden Clientsystemen verwalten können.
	ANCL_CUP_LIST	Liste von Benutzer/Passwort Kombinationen, zugeordnet zu ClientID	Whitelist an UserCredentials und dazugehörenden Clientsystem IDs. Der Administrator MUSS eine Liste von Credentials und zugehörendem 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 immer (ENABLED) auch über einen ungesicherten http-Kanal erfolgen kann, unabhängig von ANCL_TLS_MANDATORY = ENABLED oder ANCL_CAUT_MANDATORY = ENABLED. Defaultwert: Enabled
Zugriffsberechtigungsdienst			
Dokumentvalidierungsdienst			
Dienstverzeichnisdienst			
Kartenterminaldienst			
	CTM_SERVICE_DISCOVERY_PORT	Portnummer	Der Administrator MUSS die Portnummer eingeben können, auf der die KTs im lokalen Netz auf Dienstanfragen hören. Defaultwert = 4742
	CTM_SERVICE_DISCOVERY_TIME OUT	X Sekunden	Der Administrator MUSS die Anzahl Sekunden eingeben können, die der Konnektor auf Antworten zu Service Discovery Anfragen wartet

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
			Defaultwert=3
	CTM_SERVICE_ANNOUNCEMENT_PORT	Portnummer	Der Administrator MUSS die Portnummer eingeben können, auf der der Konnektor auf Dienstbeschreibungspakete hört. Defaultwert = 4742
	CTM_SERVICE_DISCOVERY_CYCLE	X Minuten	Der Administrator MUSS die Anzahl Minuten einstellen können, in denen der Konnektor wiederholt Service Discovery Nachrichten absetzt. Defaultwert=10, 0=Deaktiviert
x	CTM_SUPPORTED_KT_VERSIONS	Liste von KT-Produkttypversionen	Der Administrator MUSS die Liste der vom Konnektor unterstützten modellunabhängigen KT-Produkttypversionen einsehen können.
	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). Wertebereich: 1-60 Default-Wert=10
x	CTM_CT_LIST	Liste von CT-Objekten	Eine Liste von Repräsentanzen (CT-Objects) der dem Konnektor bekannten Kartenterminals.
x	CT.CTID	Identifier	Eindeutige, statische Identifikation des Kartenterminals
x	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.

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
x	CT.MAC_ADRESS	MAC-Adresse	Die MAC-Adresse des Kartenterminals
	CT.HOSTNAME	String	Der Administrator MUSS den SICCT-Terminalnamen (Hostname) - auch als FriendlyName bezeichnet - des Kartenterminals eingeben können.
	CT.IP_ADRESS	IP-Adresse	Der Administrator MUSS für KTs mit CT.IS_PHYSICAL=Ja die IP-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.
x	CT.SLOTCOUNT	Nummer	Anzahl der Slots des Kartenterminals
x	CT.SLOTS_USED	Liste	Liste der aktuell mit Karten belegten Slots
x	CT.PRODUCTINFORMATION	Inhalt ProductInformation.XSD	Die Herstellerinformationen zum Kartenterminal gemäß gemSpec_O&M
x	CT.VALID_VERSION	Boolean	True, wenn die Version des Kartenterminals durch den Konnektor unterstütz wird
x	CT.SMKT_AUT	X.509-Cert	C.SMKT.AUT-Zertifikat des Kartenterminals, gespeichert im Rahmen des Pairings
x	CT.SHARED_SECRET		ShS.KT.AUT, gespeichert im Rahmen des Pairings

© gematik – öffentlich

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
x	CT.CORRELATION	bekannt zugewiesen gepairt aktiv aktualisierend	Der Korrelationsstatus zum Konnektor: • bekannt (über Service Announcement/Service Discovery gelernte Kartenterminals), • zugewiesen (durch den Administrator aus dem Bereich der bekannten Kartenterminals oder manuell konfigurierte Kartenterminals), • gepairt, (Pairing erfolgreich, nicht zum Verbindungsaufbau vorgesehen, da nicht aktiv) • aktiv (ja nach erfolgreichem Pairing durch Administrator gesetzt), • 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" für das jeweilige Kartenterminal).
x	CT.CONNECTED	Ja / Nein	Der Verfügbarkeitsstatus des Kartenterminals (Ja = nach Aufbau der TLS/SSL-Verbindung und erfolgter zweiter Authentifizierung)
x	CT.ACTIVEROLE	User / Admin	Benutzerrolle, die für die aktuelle Session verwendet wird
	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.
Kartendienst			
	CARD_TIMEOUT_CARD	Sekunden	Maximale Zeit, die ein Aufruf einer Kartenoperation dauern darf, bevor der Aufruf abgebrochen wird. Der Konnektor MUSS dem Administrator die Möglichkeit bieten, diesen Parameter auf einen Wert zwischen 1 und 30 Sekunden einzustellen. Default-Wert: 2 Sekunden

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	CM_CARD_LIST	Liste	Liste der vom Konnektor verwalteten Karten
Systeminformationsdienst			
	EVT_MAX_TRY	Nummer	Der Administrator MUSS die Anzahl der Fehlversuche bzgl. Anmeldeversuchen bzw. Ereigniszustellungen, die zur automatischen Abmeldung eines Clientsystems User-Accounts führen, vergeben können.
	EVT_MONITOR_OPERATIONS	Liste von: - Operationsname - OK_Val (Nummer) - NOK_Val (Nummer) - Alarmwert (Nummer)	Der Administrator MUSS in der Liste der zur Missbrauchserkennung überwachbaren Operationen alle Listeneinträge einsehen können. Er MUSS den Alarmwert editieren können (0-9999, 0=deaktiviert).
Verschlüsselungsdienst			
Signaturdienst und SAK			
	SAK_ALIAS_SET	freie Belegung, z.B. Screen1Top, Screen2Left, ScreenRight, DeskTopMiddle	Alias-Bezeichner zur Auswahl von Einstellungen der Position des Extended Trusted Viewers. Der Hersteller kann entweder ein festes Set an Aliasen definieren (z. B. Screen1Top, Screen2Left, ScreenRight, DeskTopMiddle usw.) oder die Vergabe eigener Aliase durch den Administrator (via Konfig-Datei oder Web-Interface) unterstützen. Die konkrete Belegung von Parametern (Screen-Nr; X-,Y-Pos; Breite, Höhe usw.) wird durch die SAK verwaltet.
	SAK_SIMPLE_SIGNATURE_ MODE	SE#1 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. Defaultwert = SE#1

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	SAK_TIMEOUT_xTV	Sekunden Default: 60	Wartezeit (Timeout), nach der der Konnektor die Signaturerzeugung abbrechen soll, wenn im xTV keine Bestätigung für die Signaturerzeugung eingegeben wurde. Der Konnektor muss dem Administrator die Möglichkeit bieten, die Wartezeit des xTV zu konfigurieren.
Zertifikatsdienst			
	CERT_CRL_DOWNLOAD_ADDRESS	2 URIs	Download-Adressen für die CRL
	CERT_TSL_DEFAULT_GRACE_PERIOD_DAYS	X Tage	Default Grace Period TSL in Tagen Gibt an, wie viele Tage der Konnektor mit einer nicht aktualisierten TSL weiter betrieben werden kann. Der Wert MUSS zwischen 1 und 30 Tagen liegen. Default-Wert = 30 Tage
	CERT_OCSP_FORWARDER_ADDRESS	2 FQDNs	Adressen der OCSP-Forwarder (HTTPS-Proxy) beim Zugangsdienstprovider 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_OCSP_DEFAULT_GRACE_PERIOD_NONQES	X Minuten	Default Grace Period OCSP für nonQES in Minuten. Der Wert MUSS zwischen 0 und 20 Minuten liegen. Default-Wert = 10 Minuten
	CERT_OCSP_TIMEOUT_NONQES	X Sekunden	Timeout für OCSP-Abfragen bei der Prüfung von nonQES-Zertifikaten. Der Wert MUSS zwischen 1 und 120 Sekunden liegen. Default-Wert = 30 Sekunden
	CERT_OCSP_TIMEOUT_QES	X Sekunden	Timeout für OCSP-Abfragen bei der Prüfung von QES-Zertifikaten. Der Wert muss zwischen 1 und 120 Sekunden liegen. Default-Wert = 30 Sekunden

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	CERT_EXPIRATION_WARN_DAYS	X Tag(e)	Warnung X Tage vor Ablauf von Zertifikaten im Managementinterface und per Ereignis. Der Wert muss zwischen 0 und 180 Tagen (0=keine Warnung) liegen. Default-Wert = 90 Tage
	CERT_EXPIRATION_CARD_CHECK_DAYS	X Tag(e)	Alle X Tage wird der Ablauf aller gesteckten Karten überprüft. Der Wert muss zwischen 0 und 365 liegen (0=kein Check). Default-Wert = 1 Tag
	CERT_IMPORTED_CA_LIST	Liste von manuell importierten CA-Zertifikaten	Der Administrator MUSS CA-Zertifikate importieren, anzeigen und löschen können. Der Konnektor DARF CA-Zertifikate zur Ableitung von QES-Zertifikaten NICHT importieren. Default-Wert = leere Liste
Protokollierungsdienst			
	LOG_LEVEL_SYSLOG	Info, Warning, Error, Fatal	Der Administrator MUSS den Detaillierungsgrad des Systemprotokolls durch Festlegung der Mindest-Schwere zu protokollierender Einträge festlegen können. Defaultwert: Warning
	LOG_DAYS	X Tage	Der Administrator MUSS die Anzahl der gespeicherten Tage für Protokolle festlegen können: Es gibt einen Konfigurationsparameter LOG_DAYS für das Sicherheitsprotokoll. Es gibt einen gemeinsamen Konfigurationsparameter LOG_DAYS für das Systemprotokoll und das Konnektor-Performanceprotokoll. Dabei DARF der eingestellte Wert NICHT unter der Mindestgröße von 10 Tagen oder über der Maximalgröße von einem Jahr (365 Tage) liegen. Default-Wert: 180
	FM_<fmName>_LOG_DAYS	X Tage	Der Administrator MUSS die Anzahl der gespeicherten Tage für die fachmodulspezifischen Protokolle festlegen können. Es kann je Fachmodul einen Konfigurationsparameter für

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
			LOG_DAYS geben, der gemeinsam für das Fachmodulprotokoll und das Fachmodul-Performanceprotokoll gilt. Dabei DARF der eingestellte Wert NICHT unter der Mindestgröße von 10 Tagen oder über der Maximalgröße von einem Jahr (365 Tage) liegen. Default-Wert: 180 Die Definition des fachmodulspezifischen Konfigurationswertes ist Bestandteil der entsprechenden Fachmodulspezifikation. Ist kein fachmodulspezifischer Konfigurationsparameter spezifiziert, dann gilt LOG_DAYS.
	LOG_SUCCESSFUL_CRYPTO_OPS	Enabled / Disabled	Der Administrator MUSS festlegen können, ob auch erfolgreich ausgeführte Kryptooperationen im Sicherheitslog protokolliert werden sollen. Defaultwert: Disabled
TLS-Dienst			
Anbindung LAN/WAN			
x	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 LEKTR an das der LAN-Adapter des Konnektors angeschlossen ist. Der Konnektor MUSS gewährleisten, das das Netzwerksegment NICHT mit einem der folgenden Netzwerksegmente überlappt:

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
			1. NET_TI_DEZENTRAL 2. NET_TI_ZENTRAL 3. NET_TI_FACHDIENSTE 4. NET_SIS 5. ANLW_BESTANDSNETZE 6. ANLW_AKTIVE_BESTANDSNETZE 7. ANLW_LEKTR_INTRANET_ROUTES
	ANLW_LAN_IP_ADDRESS	IP-Adresse	Dies ist die IP-Adresse des LAN-Adapters.
	ANLW_LAN_SUBNETMASK	Subnetzmaske	Dies ist die zu ANLW_LAN_IP_ADDRESS gehörende Subnetzmaske.
	ANLW_LAN_MTU	Nummer	Der Administrator MUSS die LAN-seitige IP-Paketlänge (die Maximum Transmission Unit) setzen können. Der Konnektor MUSS sicherstellen, dass der konfigurierte Wert in den Grenzen von 576 bis 9000 liegt. Defaultwert: 1500
	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.
x	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. Der Wert bezeichnet ein lokales Netzwerk in der Umgebung des LEKTR an das der WAN-Adapter des Konnektors angeschlossen ist. Der Konnektor MUSS gewährleisten, das das Netzwerksegment nicht mit einem der folgenden

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
			Netzwerksegmente überlappt: 1. NET_TI_DEZENTRAL 2. NET_TI_ZENTRAL 3. NET_TI_FACHDIENSTE 4. NET_SIS 5. ANLW_ BESTANDSNETZE 6. ANLW_AKTIVE_BESTANDSNETZE 7. ANLW_LAN_NETWORK_SEGMENT 8. ANLW_LEKTR_INTRANET_ROUTES
	ANLW_WAN_IP_ADDRESS	IP-Adresse	Dies ist die IP-Adresse des WAN-Adapters.
	ANLW_WAN_SUBNETMASK	Subnetzmaske	Dies ist die zu ANLW_WAN_IP_ADDRESS gehörende Subnetzmaske.
	ANLW_WAN_MTU	Nummer	Der Administrator MUSS die WAN-seitige IP-Paketlänge (die Maximum Transmission Unit) setzen können. Der Konnektor MUSS sicherstellen, dass der konfigurierte Wert in den Grenzen von 576 bis 9000 liegt. Defaultwert: 1500
	ANLW_WAN_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.
x	ANLW_ANBINDUNGS_MODUS	InReihe / Parallel	InReihe: Der Konnektor wird in Reihe zu dem IAG beim LE / KTR geschaltet Parallel: Der Konnektor wird parallel (zu allen bestehenden Systemen) ins Netzwerk des LE / KTR angebunden. Wenn ANLW_WAN_ADAPTER_MODUS= ENABLED befindet sich der Konnektor im Anbindungsmodus InReihe. Wenn ANLW_WAN_ADAPTER_MODUS= DISABLED befindet sich der Konnektor im Anbindungsmodus Parallel.

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
			Der Administrator MUSS diesen Wert einsehen und DARF ihn NICHT ändern können.
	ANLW_INTERNET_MODUS	SIS / IAG / KEINER	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 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_INTRANET_ROUTES_MODUS	REDIRECT / BLOCK	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. KEINER: Der Konnektor MUSS alle IP-Pakete für ein Intranet (gemäß ANLW_LEKTR_INTRANET_ROUTES) ablehnen.
	ANLW_WAN_ADAPTER_MODUS	ENABLED/ DISABLED	Dieser Parameter ändert den Interface-Status des WAN-Adapters. Der Administrator MUSS diesen Wert einsehen können. Der Administrator MUSS diesen Wert ändern können.
	ANLW_LEKTR_INTRANET_ROUTES	Tuple aus Netzwerksegment und dazugehörigem Next-Hop	Der Administrator MUSS in diese Liste Einträge hinzufügen, editieren und löschen können. Liste von Routen zur Erreichung der Clientsysteme und Kartenterminals vom Konnektor; jeweils mit IP-Netzwerk dazugehörigem Next Hop. Die Netzwerksegmente DÜRFEN NICHT mit den

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
			Netzbereichen - NET_SIS - NET_TI_DEZENTRAL - NET_TI_ZENTRAL - NET_TI_GESICHERTE_FD - NET_TI_OFFENE_FD - ANLW_BESTANDSNETZE kollidieren.
	`ANLW_AKTIVE_BESTANDSNETZE`	Liste von IP-Address-Segmenten	Der Administrator MUSS manuell aus der Liste der zur Verfügung stehenden Bestandsnetzen (gemäß Bestandsnetze.xml) einzelne freischalten können. Nur die freigegeben Bestandsnetze werden in dieser Variablen erfasst. Nur die freigeschalteten Bestandsnetze sind aus dem LE/KTR Netzwerken erreichbar.
	`ANLW_SERVICE_TIMEOUT`	X Sekunden	Der Administrator MUSS die maximale Zeit konfigurieren können, in der ein Service antworten muss, bevor das System einen Timout-Fehler meldet. Defaultwert: 60 Sekunden

© gematik – öffentlich

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	ANLW_IAG_ADDRESS	IP Adresse	Wenn (ANLW_WAN_ADAPTER_MODUS=ENABLED und DHCP_CLIENT_WAN_STATE=DISABLED) MUSS der Administrator die IP-Adresse des Default Gateway (zum IAG im lokalen Netzwerk) setzen können. Diese IP-Adresse MUSS innerhalb des ANLW_WAN_NETWORK_SEGMENT liegen. Wenn (WAN_ADAPTER_MODUS=ACTIVE und ACTIVE und DHCP_CLIENT_WAN_STATE=ENABLED) MUSS der DHCP Client das über das WAN Interface erhaltene DGW in der Variable ANLW_IAG_ADDRESS speichern. Der Administrator des Konnektor MUSS diesen Wert dann ausschließlich einsehen können. Wenn (ANLW_WAN_ADAPTER_MODUS=DISABLED und DHCP_CLIENT_LAN_STATE=DISABLED) MUSS der Administrator die IP-Adresse des Default Gateway (zum IAG im lokalen Netzwerk) setzen können. Diese IP-Adresse MUSS innerhalb des ANLW_LAN_NETWORK_SEGMENT liegen. Wenn (WAN_ADAPTER_MODUS=DISABLED und DHCP_CLIENT_LAN_STATE=ENABLED) MUSS der über das LAN Interface erhaltene DGW in der Variable ANLW_IAG_ADDRESS gespeichert werden. Der Administrator des Konnektor MUSS diesen Wert dann ausschließlich einsehen können.
	ANLW_FW_SIS_ADMIN_RULES	Firewall Regelset	Der Administrator MUSS Firewall Regeln (für den einschränkenden Zugriff auf die SIS) einfügen, editieren und löschen können.
	ANLW_FW_SECURITY_LOGGING	Enabled / Disabled	Der Administrator des Konnektor MUSS das erweiterte Logging der Firewall zuschalten können. Default-Wert: Disabled
DHCP Server			
	DHCP_SERVER_STATE	Enabled / Disabled	Der DHCP Server MUSS durch den Administrator aktivierbar und deaktivierbar sein.

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	DHCP_SERVER_NETWORK	IP-Adresse	IP-Netzwerk des LE.
	DHCP_SERVER_BROADCAST	IP-Adresse	Die Broadcast Adresse des Konnektors am LAN Interface
	DHCP_SERVER_STATIC_RANGE	von – bis IP Adresse	Adressbereich für Adressen die per Static Lease vergeben werden dürfen.
	DHCP_SERVER_DYNAMIC_RANGE	von – bis IP Adresse	Adressbereich für Adressen die dynamisch vergeben werden dürfen.
	DHCP_SERVER_CLIENTGROUPS	Name der Clientgruppe; Liste an MAC Adressen	Der Konnektor MUSS dem Administrator über das Managementinterface die Möglichkeit bieten mindestens zwei Client Gruppen zu verwalten.
	DHCP_SERVER_DEFAULT_CLIENTGROUP	Clientgruppe	Standardmäßig eingestellte Client Gruppe. Wird verwendet falls DHCP Anfrage keiner anderen Client Gruppe zugeordnet werden kann.
	Die nachfolgende Parameterliste ist für jede Clientgruppe getrennt konfigurierbar		
	DHCP_OWNDNS_ENABLED	Enabled / Disabled	Der Administrator MUSS konfigurieren können, ob der konnektoreigene DNS Server als Parameter übergeben wird. Defaultwert: 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 des LE 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. Defaultwert: Enabled

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	DHCP_OWNDGW_ENABLED	Enabled / Disabled	Der Administrator MUSS konfigurieren können, ob der Konnektor beim Client als Default Gateway gesetzt werden soll. Defaultwert: 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_RANGE	von – bis IP Adresse	Der Administrator MUSS das zu der Clientgruppe gehörende IP Range für Clients 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_LEASE_TTL	X Minuten	Der Administrator MUSS die Leasedauer der dynamischen Adressen konfigurieren können
	DHCP_AKTIVE_BESTANDSNETZE_ROUTES	Liste von Tupel: Netzwerksegment je INTRANET und Adresse für Next Hop je freigegebenem Bestandsnetz	Der Administrator MUSS je freigegebenem Bestandsnetz (aus ANLW_AKTIVE_BESTANDSNETZE) den an den Client zu übermittelnden Routen-Eintrag aktivieren oder deaktivieren können.

© gematik – öffentlich

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	DHCP_INTRANET_ROUTES	Liste von Tupel: Netzwerksegment je INTRANET und Adresse für Next Hop in die definierten Intranets	Der Administrator MUSS je Intranet-Tupel (aus LEKTR_INTRANET_ROUTES) den an den Client zu übermittelnden Routeneintrag aktivieren oder deaktivieren können.
	DHCP_ROUTES	Tupel Netzwerksegment und Adresse für 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: - dem Netzwerksegment ANLW_LAN_NETWORK_SEGMENT - dem Netzwerksegment ANLW_WAN_NETWORK_SEGMENT - jedes Netzsegmente in ANLW_BESTANDSNETZE ANLW_AKTIVE_BESTANDSNETZE ANLW_LEKTR_INTRANET_ROUTES Die Routen SOLLEN über DHCP Option 121 (Windows Vista oder höher) bzw. DHCP Option 249 (Windows XP und darunter) verteilt werden.
	DHCP_OPTIONS	Liste an weiteren DHCP Optionen	Vom Administrator konfigurierbare Liste an weiteren DHCP Options gemäß [RFC2132]
DHCP Client			
	DHCP_CLIENT_LAN_STATE	Enabled / Disabled	Der Administrator muss den DHCP Client an der LAN Schnittstelle aktivieren oder deaktivieren können.

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	DHCP_CLIENT_WAN_STATE	Enabled / Disabled	Der Administrator muss den DHCP Client an der WAN Schnittstelle aktivieren oder deaktivieren können.
VPN-Client			
	IKE_KEEPALIVE_MODUS	Enabled/Disabled	Der Administrator MUSS einstellen können, ob IKE Keep-Alive-Pakete gesendet werden. Hinweis MUSS ausgegeben werden, dass dies ist insbesondere bei Nutzung von Dial-Up-Verbindungen nicht zu empfehlen ist. Default-Wert: Enabled
	IKE_KEEPALIVE_INTERVAL	X Sekunden	Der Administrator MUSS die Zeit in Sekunden angeben können, nach der ein neues IKE Keep-Alive-Paket gesendet wird. Default-Wert: 30
	IKE_KEEPALIVE_RETRY	X	Der Administrator MUSS angeben können, nach wie vielen IKE Keep-Alive-Paketen ohne Acknowledge Message die Verbindung beendet wird. Default-Wert: 3
	VPN_IDLE_TIMEOUT_MODUS	Enabled/Disabled	Der Administrator MUSS einstellen können, ob nach Inaktivität die VPN-Verbindung automatisch abgebaut werden soll. Ein Hinweis MUSS ausgegeben werden, dass dies insbesondere bei Nutzung von Dial-Up-Verbindungen Enabled werden sollte. Default-Wert: Disabled

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	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. Default-Wert: 600
	NAT_KEEPALIVE_MODUS	Enabled/Disabled	Der Administrator MUSS einstellen können, ob NAT Keep-Alive-Pakete gesendet werden. Hinweis MUSS ausgegeben werden, dass dies ist insbesondere bei Nutzung von Dial-Up-Verbindungen nicht zu empfehlen ist. Default-Wert: Enabled
	NAT_KEEPALIVE_INTERVAL	X Sekunden	Der Administrator MUSS die Zeit in Sekunden angeben können, nach der ein neues NAT Keep-Alive-Paket gesendet wird. 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_ADDRESS	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 (exkl. ESP-Header-Size) in den Grenzen von 576 bis 8076 konfigurieren können. Defaultwert: 1418

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	VPN_SIS_MTU	Paketgröße in Byte	Der Administrator MUSS die MTU für ESP-Pakete zum SIS (exkl. ESP-Header-Size) in den Grenzen von 576 bis 8076 konfigurieren können. Defaultwert: 1418
	HASH_AND_URL	Enabled/Disabled	Der Administrator MUSS die Nutzung des hash&URL Verfahrens zum Zertifikatsaustausch konfigurieren können. Wenn HASH_AND_URL = Enabled gesetzt ist, MUSS es dem Administrator ermöglicht werden die URL für das hash&URL Verfahren einzutragen oder automatisch zu ermitteln. Die URL kann durch eine DNS-SRV-Anfrage „_hashandurl._tcp.<DNS_DOMAIN_VPN_ZUGD_INT>“ ermittelt werden. Default-Wert: Disabled
Zeitdienst			
	NTP_TIMEZONE	Zeitzone	Der Administrator MUSS die Zeitzone des Konnektors einstellen können. Defaultwert: Central European Time/Mitteleuropäische Zeit (CET/MEZ)
	NTP_SERVER_ADDR	IP-Adressen	Die Adressen des Haupt- und sekundären Stratum 2 Zeitserver der zentralen TI-Plattform für die Synchronisation mit dem NTP Server des Konnektors.
	NTP_TIME	Zeit	Der Administrator MUSS die Zeit des Konnektors (NTP_TIME) über das Managementinterface einstellen können falls MGM_LU_ONLINE nicht aktiv ist.
	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

© gematik – öffentlich

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	NTP_MAX_TIMEDIFFERENCE	3600	Maximale Zeitabweichung in Sekunden zwischen Systemzeit und Zeit des Stratum 2 Zeitservers zum Zeitpunkt der Zeitsynchronisierung
Namensdienst und Dienstlokalisierung			
	DNS_SERVERS_INT	Liste von IP-Adressen der DNS-Server	Liste von DNS-Servern für Transportnetz (nur für den Konnektor, zum Auflösen der FQDN aller zum VPN-Tunnel-Aufbau erforderlichen Adressen) Alle genannten IP-Adressen MÜSSEN aus einem öffentlich zugänglichen Adressbereich stammen.
	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-Server	Liste von DNS-Servern die zur Namensauflösung der Namensräume von Bestandsnetzen verwendet werden. Der Administrator MUSS je freigegebenem Bestandsnetz (gemäß Bestandsnetze.xml) die vom Konnektor weiterzuleitenden Domainen de-/aktivieren, sowie deren DNS-Servern bearbeiten können. Alle genannten IP-Adressen MÜSSEN aus dem Adressbereich des jeweiligen Bestandsnetz stammen.

© gematik – öffentlich

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	DNS_SERVERS_LEKTR	Liste von IP-Adressen der DNS-Server	Liste von DNS-Servern die zur Namensaulösung von Namensräumen in der LE/KTR-Umgebung verwendet werden Der Administrator MUSS die Liste von DNS-Servern die die DNS_DOMAIN_LEKTR können bearbeiten können. Die IP-Adressen der DNS-Server MÜSSEN aus dem Adressbereich LAN_IP_ADDRESS stammen.
	DNS_TOP_LEVEL_DOMAIN_TI	DNS Domainname	Top Level Domain des Namensraumes TI
	DNS_DOMAIN_VPN_ZUGD_INT	DNS Domainname	DNS-Domainname für die Service Discovery der VPN-Konzentratoren des VPN-Zugangsdienstes
	DNS_DOMAIN_LEKTR	DNS Domain Name	DNS Domainname, der von einem DNS-Server beim Leistungserbringer aufgelöst wird. Der NAME DARF nicht mit einem „.“ beginnen und nicht mit einem „.“ enden.
	DNS_ROOT_ANCHOR_URL	URL	Der Administrator MUSS die URL für den Download der IANA DNSSEC Root Keys (im Format http://$hostname:$port/$path oder https://$hostname:$port/$path) eintragen können. Default: http://data.iana.org/root-anchors/root-anchors.xml
Zugang und Benutzerverwaltung des Konnektormanagements			
	MGM_USER_LIST	Liste von Benutzernamen und deren Kontaktdaten	Liste von Benutzern und deren Kontaktdaten. Benutzerkonten MÜSSEN angelegt, geändert und gelöscht werden können. Das Passwort eines Benutzerkonten MUSS neu gesetzt werden können.

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	MGM_ADMIN_RIGHTS	Liste von Zugriffsrechten eines Benutzers	v. Eindeutige Zuordnung eines Benutzerkontos zu einer Rolle. Die Benutzerverwaltung MUSS sicherstellen, dass zu jeder Zeit mindestens ein Benutzerkonto mit der Rolle Super-Administrator vorhanden ist. Gewähren / Entziehen für Benutzerkonten: i. von Zugriffsrechten bezüglich der Konfigurationsbereiche. ii. des Rechts zum Aufbau einer Remote-Management-Session (USER_INIT_REMOTESESSION). iii. des Rechts für einen Werksreset (USER_RESET_PERMISSION)
Werksreset			
	MGM_RESET_WITHOUT_AUTHENTICATION	Enabled / Disabled	Der Super-Administrator MUSS die Möglichkeit zum Auslösen eines Werksresets ohne Authentisierung aktivieren und deaktivieren können. Defaultwert: Disabled
Leistungsumfänge und Standalone-Szenarios			
	MGM_LU_ONLINE	Enabled / Disabled	Der Administrator MUSS den „Leistungsumfang Online“ aktivieren und deaktivieren können. Defaultwert: Enabled Bei Veränderung MUSS TUC_KON_256 gerufen werden {"MGM/LU_CHANGED/LU_ONLINE“; Op; Info; „Active=$MGM_LU_ONLINE“}

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	MGM_LU_SAK	Enabled / Disabled	Der Administrator MUSS den „Leistungsumfang Signaturanwendungskomponente“ aktivieren und deaktivieren können. Defaultwert: Enabled Bei Veränderung MUSS TUC_KON_256 gerufen werden {"MGM/ LU_CHANGED/LU_SAK“; Op; Info; „Active=$MGM_LU_SAK“}
	MGM_STANDALONE _KON	Enabled / Disabled	Der Administrator MUSS den Konnektor als alleinstehend konfigurieren können. Defaultwert: Disabled Bei Veränderung MUSS TUC_KON_256 gerufen werden {"MGM/STANDALONE_CHANGED“; Op; Info; „Active=$MGM_STANDALONE_KON“}
	MGM_LOGICAL_SEPARATION	Enabled / Disabled	Der Administrator MUSS die logische Separation zwischen TI und lokalem LE-Netz aktivieren / deaktivieren können. Defaultwert: Disabled Bei Veränderung MUSS TUC_KON_256 gerufen werden {"MGM/LOGICAL_SEP_CHANGED“; Op; Info; „Active=$MGM_STANDALONE_KON“}
In- und Außerbetriebnahme			
	MGM_ZGDP_CONTRACTID	String	Der Administrator MUSS die vom Zugangsdienstprovider für die Freischaltung des Konnektors erhaltene ContractID eingeben können.
	MGM_ZGDP_REGSERVER	URI	URI des Registrierungsservers des Zugangsdienstprovider
	MGM_ZGDP_SMCB	ICCSN	Der Administrator MUSS die zur Freischaltung zu verwendende SM-B aus der Liste der verwalteten SM-Bs auswählen können.

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
x	MGM_TI_ACCESS_GRANTED	Enabled / Disabled	Status der Freischaltung des Konnektors: - Enabled: Konnektor wurde erfolgreich beim Zugangsdienstprovider freigeschaltet - Disabled: Freischaltung noch nicht erfolgt
Software- und Konfigurationsaktualisierungsdienst (KSR-Client)			
x	MGM_KSR_FIRMWARE_URL	URL	SOAP-Endpunkt des Konfigurationsdienstes zum Download der Firmware, dynamisch ermittelt im Rahmen des VPN-Verbindungsaufbaus zur TI
x	MGM_KSR_Konfig_URL	URL	SOAP-Endpunkt des Konfigurationsdienstes zum Download von Konfigurationsdaten, dynamisch ermittelt im Rahmen des VPN-Verbindungsaufbaus zur TI
	MGM_KSR_AUTOCHECK	Enabled / Disabled	Der Administrator MUSS die automatische Prüfung auf verfügbare Updatepakete an- und abschalten können. Defaultwert: Enabled
	MGM_KSR_AUTODOWNLOAD	Enabled / Disabled	Sofern MGM_KSR_AUTODOWNLOAD auf Enabled gesetzt ist, MUSS der Administrator den automatische Download verfügbarer Updatepakete an- und abschalten können. Defaultwert: Disabled
Konnektorname und Versionsinformationen			
	MGM_KONN_HOSTNAME	12 Zeichen	Der Konnektorname MUSS folgende Anforderungen erfüllen (in Anlehnung an die Definition eines „Labels“ in [RFC1034]): • Verwendung der Buchstaben „A bis Z“ und „a bis z“, • Verwendung der Ziffern „0 bis 9“, • als Sonderzeichen „–“ (Minus), sowie • eine maximale Länge von 12 Zeichen, Die Verwendung weiterer Sonderzeichen sowie des Leerzeichens DARF NICHT möglich sein.
Remote Management			

Zustandswert?	ReferenzID	Belegung	Bedeutung und Administrator-Interaktion bzw. Zustandswerte
	MGM_REMOTE_ALLOWED	Enabled / Disabled	Der Administrator MUSS einstellen können, ob der Konnektor eine Remote-Management-Verbindung aufbauen kann. Enabled: Der Konnektor kann eine Remote-Management-Verbindung aufbauen Disabled: Der Konnektor kann keine Remote-Management-Verbindung aufbauen Default-Wert: Disabled

Anhang F - Übersicht Events

Tabelle 342 – TAB_KON_777 Events Interne Mechanismen

Topic Ebene1	Topic Ebene2	Topic Ebene3	Typ	Schwere	Prot	An Clients	Parameter	Bedeutung	Auslöser (TUC/Op)
Interne Mechanismen									
BOOTUP	BOOTUP_COMPLETE		Op	Info	x	x		Änderung des Betriebszustandes	
OPERATIONAL_STATE	EC_CardTerminal_Software_Out_Of_Date($ctId)		Op	Info	x	x	Value=true/false; CtId=$ctId; Bedeutung=$EC.description	Änderung des Betriebszustandes durch Änderung im Fehlerzustand (Änderung im value).	
OPERATIONAL_STATE	EC_Connector_Software_Out_Of_Date		Op	Info	x	x	Value=true/false; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_Time_Sync_Not_Successful		Op	Info	x	x	Value=true/false; LastSyncAttempt=$lastSyncAttemptTimestamp; LastSyncSuccess=$lastSyncSuccessTimestamp; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_TSL_Update_Not_Successful		Op	Info	x	x	Value=true/false; Bedeutung=$EC.description; LastUpdateTSL=$lastUpdateTSLTimestamp	"	
OPERATIONAL_STATE	EC_TSL_Expiring		Sec	Info	x	x	Value=true/false; NextUpdateTSL=$NextUpdate-Element der TSL Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_TSL_Trust_Anchor_Expiring		Sec	Info	x	x	Value=true/false; ExpiringDateTrustAnchor=Ablaufdatum der Vertrauensankergültigkeit	"	

© gematik – öffentlich

Topic Ebene1	Topic Ebene2	Topic Ebene3	Typ	Schwere	Prot	An Clients	Parameter	Bedeutung	Auslöser (TUC/Op)
							Bedeutung=$EC.description		
OPERATIONAL_STATE	EC_LOG_OVERFLOW		Op	Warning	x	x	Value=true/false; Bedeutung=$EC.description	"	TUC_KON_271
OPERATIONAL_STATE	EC_CRL_Expiring		Sec	Warning	x	x	Value=true/false; NextUpdateTSL=$NextUpdate-Element der TSL Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_Time_Sync_Pending_Warning		Sec	Warning	x	x	Value=true/false; LastSyncSuccess=$lastSyncSuccessTimestamp; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_TSL_Out_Of_Date_Within_Grace_Period		Sec	Warning	x	x	Value=true/false; NextUpdateTSL=$NextUpdate-Element der TSL; GracePeriodTSL=CERT_TSL_DEFAULT_GRACE_PERIOD_DAYS; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_CardTerminal_Not_Available($CtId)		Op	Error	x	x	Value=true/false; CtId=$ctId; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_No_VPN_TI_Connection		Op	Error	x	x	Value=true/false; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_No_VPN_SIS_Connection		Op	Error	x	x	Value=true/false; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_No_Online_Connection		Op	Error	x	x	Value=true/false; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_FeatureOrTUC_Not_Available($Dienst/$Operation)		Op	Error	x	x	Value=true/false; Dienst=$service Operation=$operation Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_IP_Adresses_Not_Available		Sec	Error	x	x	Value=true/false; Bedeutung=$EC.description	"	

Topic Ebene1	Topic Ebene2	Topic Ebene3	Typ	Schwere	Prot	An Clients	Parameter	Bedeutung	Auslöser (TUC/Op)
OPERATIONAL_STATE	EC_CRL_Out_Of_Date		Sec	Fatal	x	x	Value=true/false; NextUpdateCRL=$NextUpdate der CRL; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_Firewall_Not_Reliable		Sec	Fatal	x	x	Value=true/false; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_Random_Generator_Not_Reliable		Sec	Fatal	x	x	Value=true/false; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_Secure_KeyStore_Not_Available		Sec	Fatal	x	x	Value=true/false; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_Security_Log_Not_Writable		Op	Fatal	x	x	Value=true/false; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_Software_Integrity_Check_Failed		Sec	Fatal	x	x	Value=true/false; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_Time_Difference_Intolerable		Sec	Fatal	x	x	Value=true/false; Bedeutung=$EC.description; NtpTimedifference=Zeitabweichung; NtpMaxAllowedTimedifference=NTP_MAX_TIMEDIFFERENCE; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_Time_Sync_Pending_Critical		Sec	Fatal	x	x	Value=true/false; LastSyncSuccess=$lastSyncSuccessTimestamp; NtpGracePeriod=NTP_GRACE_PERIOD Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_TSL_Trust_Anchor_Out_Of_Date		Sec	Fatal	x	x	Value=true/false; ExpiringDateTrustAnchor=Ablaufdatum der Vertrauensankergültigkeit; Bedeutung=$EC.description NextUpdateTSL=$NextUpdate-Element der TSL; GracePeriodTSL=CERT_TSL_DEFA	"	

Topic Ebene1	Topic Ebene2	Topic Ebene3	Typ	Schwere	Prot	An Clients	Parameter	Bedeutung	Auslöser (TUC/Op)
							ULT_GRACE_PERIOD_DAYS; Bedeutung=$EC.description		
OPERATIONAL_STATE	EC_TSL_Out_Of_Date_Beyond_Grace_Period		Sec	Fatal	x	x	Value=true/false; NextUpdateTSL=$NextUpdate-Element der TSL; GracePeriodTSL=CERT_TSL_DEFAULT_GRACE_PERIOD_DAYS; Bedeutung=$EC.description	"	
OPERATIONAL_STATE	EC_CRYPTOPERATION_ALARM		Sec	Warning	x	x	Value=true/false; Operation=$Operationsname; Count=$Summenwert; Arbeitsplatz=$<Liste operationsaufrufenden workplaceIDs>; Meldung='Auffällige Häufung von Operationsaufrufen in den letzten 10 Minuten'	"	
OPERATIONAL_STATE	EC_OTHER_ERROR_STATE($no)		$Type	$Severity	x	x	Value=true/false; Bedeutung=$EC.description	"	
Zugriffsberechtigungsdienst									
Dokumentvalidierungsdienst									
Dienstverzeichnisdienst									
Kartenterminaldienst									
CT	ERROR		$Error Type	$Severity	x	x	CTID=$CT.ID; Name=$CT.HOSTNAME; Error=$Fehlercode; Bedeutung=$Fehlertext	Bei der Kommunikation mit dem KT ist ein Fehler aufgetreten	TUC_KON_051 TUC_KON_053
CT	CONNECTED		Op	Info	x	x	CtID=$CT.CTID,	Die Verbindung zu einem	

Topic Ebene1	Topic Ebene2	Topic Ebene3	Typ	Schwere	Prot	An Clients	Parameter	Bedeutung	Auslöser (TUC/Op)
							Hostname=$CT.HOSTNAME	Kartenterminal wurde hergestellt	
CT	DISCONNECTED		Op	Info	x	x	CtID=$CT.CTID, Hostname=$CT.HOSTNAME	Die Verbindung zu einem Kartenterminal wurde unterbrochen	
CT	TLS_ESTABLISHMENT_FAILURE		$ErrorType	$Severity	x	x	CTID=$CT.ID; Name=$CT.HOSTNAME; Error=$Fehlercode; Bedeutung=$Fehlertext	Im Rahmen des Verbindungsaufbaus sind Fehler aufgetreten	TUC_KON_050
CT	CT_ADDING_ERROR		$ErrorType	$Severity	x	x	IP=$IP-Adresse; Error=$Fehlercode; Bedeutung=$Fehlertext	Bei dem Versuch ein KT der Verwaltung zuzufügen ist ein Fehler aufgetreten	TUC_KON_054
CT	SLOT_FREE		Op	Info	-	-	CtID=$CT.CTID; SlotNo=$CT.SLOTS_USED[X]	Internes Event von Kartenterminaldienst --> Kartendienst. Informiert, dass ein Slot frei wurde. Wird im Kartendienst ausgewertet und verursacht dort CARD/REMOVED	
CT	SLOT_IN_USE		Op	Info	-	-	CtID=$CT.CTID; SlotNo=<FU-Nummer aus Ereignisnachricht>	Internes Event von Kartenterminaldienst --> Kartendienst. Informiert, dass ein Slot belegt wurde. Wird im Kartendienst ausgewertet und verursacht dort CARD/INSERTED	
Kartendienst									
CARD	INSERTED		Op	Info	x	x	CardHandle=$CARD.CARDHANDLE; CardType=$CARD.TYP; CardVersion=$CARD.VER;	Eine Karte wurde gesteckt	TUC_KON_001 (als Reaktion auf CTM/SLOT_IN_USE)

Topic Ebene1	Topic Ebene2	Topic Ebene3	Typ	Schwere	Prot	An Clients	Parameter	Bedeutung	Auslöser (TUC/Op)
							ICCSN=$CARD.ICCSN CtID=$CARD.CTID SlotID=$CARD.SLOTID InsertTime=$CARD.INSERTTIME CardHolderName=$CARD.CARDHOLDERNAME KVNR=$CARD.KVNR		
CARD	REMOVED		Op	Info	x	x	CardHandle=$CARD.CARDHANDLE; CardType=$CARD.TYP; CardVersion=$CARD.VER; ICCSN=$CARD.ICCSN CtID=$CARD.CTID SlotID=$CARD.SLOTID InsertTime=$CARD.INSERTTIME CardHolderName=$CARD.CARDHOLDERNAME KVNR=$CARD.KVNR	Eine Karte wurde gezogen	Reaktion auf CTM/SLOT_FREE
CARD	PIN	VERIFY_STARTED	Op	Info	-	x			
CARD	PIN	VERIFY_FINISHED	Op	Info	-	x			
CARD	PIN	CHANGE_STARTED	Op	Info	-	x			
CARD	PIN	CHANGE_FINISHED	Op	Info	-	x			
Systeminformationsdienst									
Verschlüsselungsdienst									

Topic Ebene1	Topic Ebene2	Topic Ebene3	Typ	Schwere	Prot	An Clients	Parameter	Bedeutung	Auslöser (TUC/Op)
Signaturdienst und SAK									
Zertifikatsdienst									
CERT	TSL	IMPORT	Op	Error	x	-	$Fehlerbeschreibung	Manueller Import der TSL fehlgeschlagen	TUC_KON_032 "TSL aktualisieren"
CERT	TSL	UPDATED	Op	Info	x	-		Eine neue TSL wurde erfolgreich in den TrustStore eingespielt	TUC_KON_032 "TSL aktualisieren"
CERT	CRL	INVALID	Op	Error	x	-		Prüfung der Signatur der CRL fehlgeschlagen	TUC_KON_040 "CRL aktualisieren"
CERT	CRL	IMPORT	Op	Error	x	-	$Fehlerbeschreibung	Manueller Import der CRL fehlgeschlagen	TUC_KON_040 "CRL aktualisieren"
CERT	CRL	UPDATED	Op	Info	x	-		Die CRL wurde erfolgreich aktualisiert	TUC_KON_040 "CRL aktualisieren"
CERT	CARD	EXPIRATION	Op	Warning	x	x	CARD_TYPE=gSMC-K; ICCSN=$ICCSN; Konnektor=$MGM_KONN_HOSTNAME, ExpirationDate=$validity	gSMC-K abgelaufen	TUC_KON_033 "Zertifikatsablauf prüfen"
CERT	CARD	EXPIRATION	Op	Warning	-	x	CARD_TYPE=$Type; ICCSN=$ICCSN; CARD_HANDLE=$CardHandle; CardHolderName=$CardHolderName; ExpirationDate=$validity	Sonstige Karte abgelaufen	TUC_KON_033 "Zertifikatsablauf prüfen"
CERT	CARD	EXPIRATION	Op	Info	-	x	CARD_TYPE=gSMC-K; ICCSN=$ICCSN;	gSMC-K läuft innerhalb von	TUC_KON_033 "Zertifikatsablau

Topic Ebene1	Topic Ebene2	Topic Ebene3	Typ	Schwere	Prot	An Clients	Parameter	Bedeutung	Auslöser (TUC/Op)
							Konnektor=$MGM_KONN_HOSTNAME; ExpirationDate=$validity; DAYS_LEFT=$validity-$Today	DAYS_LEFT Tagen ab	f prüfen"
CERT	CARD	EXPIRATION	Op	Info	-	x	CARD_TYPE=$Type; ICCSN=$ICCSN; CARD_HANDLE=$CardHandle; CardHolderName=$CardHolderName; ExpirationDate=$validity; DAYS_LEFT=$validity-$Today	Sonstige Karte läuft innerhalb von DAYS_LEFT Tagen ab	TUC_KON_033 "Zertifikatsablauf prüfen"
Protokollierungsdienst									
LOG	ERROR		$ErrorType	$Severity	-	-	Error=$Fehlercode	Im Protokollierungsdienst auftretende Fehler werden verteilt	TUC_KON_271 TUC_KON_271
LOG	CRYPTO_OP		Sec	Info	x	-	Operation=$Operationsname;<für alle betroffenen Schlüssel:>Karte=$ICCSN;Keyref=<Referenz auf den Schlüssel>; CARD_HANDLE=$CardHandle; CardHolderName=$CardHolderName		
TLS-Dienst									
Anbindung LAN/WAN									
NETWORK	LAN	IP_CHANGED	Op	Warning	x	-	IP=$dieNeueIP	Wenn der LAN-Adapter eine neue IP oder Netzwerk bekommen hat	DHCP, Management-schnittstelle

Topic Ebene1	Topic Ebene2	Topic Ebene3	Typ	Schwere	Prot	An Clients	Parameter	Bedeutung	Auslöser (TUC/Op)
NETWORK	WAN	IP_CHANGED	Op	Info	x	-	IP=$dieNeueIP	Wenn der WAN-Adapter eine neue IP oder Netzwerk bekommen hat	DHCP, Management-schnittstelle
ANLW	FW	DENY	Op	Info	-	-	BlockedPacket=Von $FromIP:$FromPort nach $ToIP:$ToPort mit Protokoll $Protokoll	Jedes IP-Paket, dass durch die Firewall des Konnektor verworfen wurde	Statefull Inspection Firewall des Konnektors
ANLW	FW	SECURITY	Sec	Warn	-	-	BlockedPacket=Von $FromIP:$FromPort nach $ToIP:$ToPort mit Protokoll $Protokoll	Jedes IP-Paket, dass durch die Firewall des Konnektor verworfen wurde	Statefull Inspection Firewall des Konnektors
DHCP-Server									
DHCP	SERVER	STATECHANGED	Op	Info	x	x	STATE=$DHCP_SERVER_STATE		Administrator
DHCP Client									
DHCP	LAN_CLIENT	RENEW	Op	Info	x	x	IP_ADDRESS=<Belegung>		TUC_KON_341
DHCP	WAN_CLIENT	RENEW	Op	Info	x	x	IP_ADDRESS=<Belegung>		TUC_KON_341
DHCP	LAN_CLIENT	STATECHANGED	Op	Info	x	x	STATE=$DHCP_CLIENT_LAN_STATE		
DHCP	WAN_CLIENT	STATECHANGED	Op	Info	x	x	STATE=$DHCP_CLIENT_WAN_STATE		
VPN-Client									
NETWORK	VPN_TI	UP	Op	Info	x	x		Wenn der VPN-Tunnel zur TI erfolgreich aufgebaut worden ist.	
NETWORK	VPN_TI	DOWN	Op	Info	x	x		Wenn der VPN-Tunnel zur TI nicht mehr zur Verfügung steht.	AFO

Topic Ebene1	Topic Ebene2	Topic Ebene3	Typ	Schwere	Prot	An Clients	Parameter	Bedeutung	Auslöser (TUC/Op)
NETWORK	VPN	CONFIG_CHANGED	Op	Info	x	-		Wenn die Konfiguration des VPN-Clients angepasst wurde.	Manage-mentschnittstelle
NETWORK	VPN_SIS	UP	Op	Info	x	x		Wenn der VPN-Tunnel zum SIS erfolgreich aufgebaut worden ist.	
NETWORK	VPN_SIS	DOWN	Op	Info	x	x		Wenn der VPN-Tunnel zum SIS nicht mehr zur Verfügung steht.	AFO
Zeitdienst									
NTP	ENTERCRITICALSTATE		Op	FATAL	x	-	MESSAGE="CRITICALTIMEDEVIATION"	Zeitabweichung von mehr als einer Stunde entdeckt	
Namensdienst und Dienstlokalisierung									
Leistungsumfänge und Standalone-Szenarios									
MGM	ADMINCHANGES		Op	Info	x	-	User=$AdminUsername; RefID=$ReferenzID; NewVal=$NeuEingestellterWert"	Änderungen die der Admin vornimmt werden protokolliert	
MGM	CONFIG_EXIMPORT		Op	Info	x	-	User=$AdminUsername; Mode=[Export / Import]	Dokumentiert (via Mode), dass die Konnektorkonfiguration exportiert oder importiert wurde.	
MGM	FACTORYSETTINGS		Op	Info	x	-	User=$AdminUsername	Ein ausgelöster Werksreset wird protokolliert	
MGM	REMOTE_SESSION		Op	Info	x	-	InitUser=$AdminUsername; RemoteID=<Kennung der	Protokollierung des Versuchs, des Begins und	

Topic Ebene1	Topic Ebene2	Topic Ebene3	Typ	Schwere	Prot	An Clients	Parameter	Bedeutung	Auslöser (TUC/Op)
							Gegenstelle>; Mode=[InitSuccess / InitFail / Exit]	des Endes einer Remote-Management Session	
MGM	LU_CHANGED	LU_ONLINE	Op	Info	x	x	Active=$MGM_LU_ONLINE	Leistungsumfang Online wurde aktiviert / deaktiviert	Administrator
MGM	LU_CHANGED	LU_SAK	Op	Info	x	x	Active=$MGM_LU_SAK	Leistungsumfang Signaturanwendungskomponente wurde aktiviert / deaktiviert	Administrator
MGM	STANDALONE_CHANGED		Op	Info	x	x	Active=$MGM_STANDALONE_KON	Festlegung des Konnektors als "Alleinstehend" wurde geändert	Administrator
MGM	LOGICAL_SEP_CHANGED		Op	Info	x	x	Active=$MGM_LOGICAL_SEPARATION	Festlegung des Konnektors zur logischen Separierung wurde geändert	Administrator
In- und Außerbetriebnahme									
MGM	TI_ACCESS_GRANTED		Op	Info	x	-	Active=$MGM_TI_ACCESS_GRANTED	Der Konnektor wurde erfolgreich freigeschaltet	Administrator
Software-Aktualisierungsdienst (KSR-Client)									
KSR	ERROR		$ErrorType	$Severity	x	x	Target=Konnektor Name= <MGM_KONN_HOSTNAME> Error=$Fehlercode Bedeutung=$Fehlertext	Während der Konnektoraktualisierung ist ein Fehler aufgetreten	TUC_KON_280
KSR	ERROR		$ErrorType	$Severity	x	x	Target=KT Name=<KT-FriendlyName> CTID=$ctID Error=$Fehlercode Bedeutung=$Fehlertext	Während einer Kartenterminalaktualisierung ist ein Fehler aufgetreten	TUC_KON_281

Topic Ebene1	Topic Ebene2	Topic Ebene3	Typ	Schwere	Prot	An Clients	Parameter	Bedeutung	Auslöser (TUC/Op)
KSR	ERROR		$Error Type	$Severity	x	x	Error=$Fehlercode Bedeutung=$Fehlertext	Im KSR-Client ist ein Fehler aufgetreten	TUC_KON_282
KSR	UPDATE	START	Sec	Info	x	x	für TUC_KON_280 Target=Konnektor Name=<MGM_KONN_HOSTNAME> für TUC_KON_281 Target=KT CTID=$CtID	Ein Updateprozess im Konnektor wird gestartet, Ziel Konnektor oder Kartenterminal	TUC_KON_280 TUC_KON_281
KSR	UPDATE	SUCCESS	Sec	Info	x	x	für TUC_KON_280 Target=Konnektor Name= <MGM_KONN_HOSTNAME> NewFirmwareversion=<UpdateInformation.FirmwareVersion> ConfigurationChanged=<Ja/Nein> ManualInputNeeded=<Ja/Nein> für TUC_KON_281 Target=KT Name=<KT-FriendlyName> CTID=$ctID NewFirmwareversion=<UpdateInformation.FirmwareVersion>	Die Firmware des Konnektors / eines Kartenterminals wurde erfolgreich aktualisiert	TUC_KON_280 TUC_KON_281
KSR	UPDATE	END	Sec	Info	x	x	für TUC_KON_280 Target=Konnektor Name=<MGM_KONN_HOSTNAME> für TUC_KON_281 Target=KT	Ein Updateprozess im Konnektor wurde beendet	TUC_KON_280 TUC_KON_281

Topic Ebene1	Topic Ebene2	Topic Ebene3	Typ	Schwere	Prot	An Clients	Parameter	Bedeutung	Auslöser (TUC/Op)
							CTID=$CtID		
KSR	UPDATES_AVAILABLE		Op	Info	-	x	ManufacturerID=$UpdateInformation/ManufacturerID ProductShortName=$UpdateInformation/ProductShortName ProductName=$UpdateInformation/ProductName FirmwareVersion=$UpdateInformation/FirmwareVersion	Ein oder mehrere Updates auf neuere Versionen sind verfügbar	TIP1-A_4836

Anhang G - Fehlercodes

Tabelle 343 - TAB_KON_691 Allgemeine Fehlercodes

Fehlercode	ErrorType	Severity	Fehlertext
4000	Technical	Error	Syntaxfehler
4001	Technical	Error	Interner Fehler
4002	Security	Error	Der Konnektor befindet sich in einem kritischen Betriebszustand

Tabelle 344 - TAB_KON_692 Fehlercodes Zugriffsberechtigungsdienst

Fehlercode	ErrorType	Severity	Fehlertext
4003	Technical	Error	Keine User-Id angegeben, die zur Identifikation der Kartensitzung_HBA 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 SMC-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	SMC-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 HBA hat mindestens eine Kartensitzung zu einer anderen UserId, deren Sicherheitszustand erhöht ist.
4019	Technical	Error	Zu den Parametern konnte keine Regel ermittelt werden.

Fehlercode	ErrorType	Severity	Fehlertext
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.

Tabelle 345 - TAB_KON_693 Fehlercodes Dokumentenvalidierungsdienst

Fehlercode	ErrorType	Severity	Fehlertext
4022	Technical	Error	XML-Dokument nicht wohlgeformt
4023	Technical	Error	XML-Dokument nicht valide in Bezug auf XML-Schema
4024	Technical	Error	Formatvalidierung fehlgeschlagen (%Dokumentformat%) Der Parameter Dokumentformat kann die Werte XML, PDF/A, TIFF, MIME und Text annehmen.
4026	Technical	Error	XML-Schema nicht valide
4193	Security	Warning	Kein XML-Schema für XML-Dokument vorhanden
4195	Security	Error	Fehler bei XSL-Transformation

Tabelle 346 - TAB_KON_694 Fehlercodes Dienstverzeichnisdienst

Fehlercode	ErrorType	Severity	Fehlertext
4027	Technical	Error	Die Endpunktinformationen konnten nicht übernommen werden.

Tabelle 347 - TAB_KON_695 Fehlercodes Kartenterminaldienst

Fehlercode	ErrorType	Severity	Fehlertext
4028	Technical	Error	Fehler beim Versuch eines Verbindungsaufbau zum KT
4029	Security	Error	Fehler bei der KT-Authentisierung. KT möglicher Weise manipuliert
4030	Security	Error	Admin-Werte für KT fehlerhaft
4031	Technical	Error	Interner Fehler
4032	Technical	Error	Verbindung zu HSM konnte nicht aufgebaut werden
4033	Technical	Error	Kartenterminal antwortet nicht, Zufügen fehlgeschlagen

Fehlercode	ErrorType	Severity	Fehlertext
4034	Technical	Error	Kartenterminal mit gleichem Hostname bereits in der Liste der Kartenterminals vorhanden. Bitte Hostname des Kartenterminals ändern.
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
4038	Security	Error	Zur Anzeige vorgesehene Meldung enthält eine nicht erlaubte Zeichenfolge
4039	Technical	Error	Kartenterminal durch andere Nutzung aktuell belegt
4040	Security	Error	Fehler beim Versuch eines Verbindungsaufbau zu KT
4041	Technical	Error	Fehler im Pairing, SICCT-Fehler: <SICCT-Fehler>
4042	Technical	Error	Die Version des Kartenterminals wird nicht unterstützt
4044	Technical	Error	Fehler beim Zugriff auf das Kartenterminal
4202	Technical	Error	Timeout. Es wurde keine Karte innerhalb der angegebenen Zeitspanne gesteckt.
4203	Technical	Error	Karte deaktiviert, aber nicht entnommen.

Tabelle 348 - TAB_KON_696 Fehlercodes Kartendienst

Fehlercode	ErrorType	Severity	Fehlertext
4043	Technical	Warning	Timeout bei der PIN-Eingabe
4045	Technical	Error	Fehler beim Zugriff auf die Karte
4046	Technical	Error	Kartenapplikation existiert nicht
4047	Technical	Error	Karten-Handle ungültig
4048	Technical	Error	Fehler bei der C2C-Authentisierung
4049	Technical	Error	Abbruch durch den Benutzer
4050	Technical	Error	Öffnen eines weiteren Kanals zur Karte nicht möglich
4051	Technical	Error	Falscher Kartentyp
4052	Security	Error	Kartenzugriff verweigert
4053	Security	Error	Remote-PIN nicht möglich
4054	Security	Error	Fehler beim Secure Messaging, Zielkarte
4055	Security	Error	Fehler beim Secure Messaging, Quellkarte
4056	Technical	Error	Fehler bei der C2C-Authentisierung, Quellkarte

Fehlercode	ErrorType	Severity	Fehlertext
4057	Technical	Error	Fehler bei der C2C-Authentisierung, Zielkarte
4058	Security	Error	Aufruf nicht zulässig
4060	Technical	Error	Ressource belegt
4061	Security	Warning	Falsche alte PIN, verbleibende Eingabeversuche <x>
4062	Security	Warning	Falsche PIN (hier: PUK) verbleibende Eingabeversuche <x>
4063	Security	Error	PIN bereits gesperrt (BLOCKED)
4064	Security	Error	Alte PIN bereits blockiert (hier: PUK)
4065	Technical	Warning	PIN ist transportgeschützt, Änderung erforderlich
4066	Technical	Error	PIN Pad nicht verfügbar
4067	Security	Error	Neue PIN nicht identisch
4068	Security	Error	Neue PIN zu kurz/lang
4069	Technical	Error	Korruptes Chiffrat bei asymmetrischer Entschlüsselung
4070	Technical	Error	Autorisierende Karte oder Kartensitzung fehlt
4071	Technical	Error	Keine Karte für C2C Auth gesetzt
4072	Technical	Error	Ungültige PIN-Referenz
4073	Technical	Error	Adressiertes Passwort konnte nicht gefunden werden
4074	Technical	Error	Formatfehler der übergebenen PIN
4075	Technical	Error	Formatfehler der übergebenen neuen PIN
4076	Technical	Error	Formatfehler im übergebenen PUK
4077	Security	Error	Setzen der neuen PIN nicht zulässig
4078	Security	Error	PIN-Eingabe über das Clientsystem ist nicht zugelassen
4079	Technical	Error	Schlüsseldaten fehlen
4080	Technical	Error	Schlüssel unterstützt den geforderten Algorithmus nicht
4081	Technical	Error	Kein Signierschlüssel ausgewählt
4082	Security	Error	PIN durch diese Fehleingabe blockiert (nowblocked)
4084	Technical	Warning	Datei deaktiviert
4085	Technical	Warning	Zugriffsbedingungen nicht erfüllt
4086	Technical	Error	Verzeichnis deaktiviert

Fehlercode	ErrorType	Severity	Fehlertext
4087	Technical	Error	Datei nicht vorhanden
4088	Technical	Error	Datensatz zu groß
4089	Technical	Error	Datei ist vom falschen Typ
4091	Technical	Error	Record nicht vorhanden
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
4096	Technical	Error	Ungültige Kartenterminal ID
4097	Technical	Error	Ungültige Kartenslot ID
4191	Technical	Error	Kartenzugriff verweigert
4221	Technical	Error	Kartenterminal nicht aktiv
4222	Technical	Error	Kartenterminal ist nicht verbunden
4228	Technical	Error	Das benötigte Cross-CV-Zertifikat ist nicht vorhanden

Tabelle 349 - TAB_KON_697 Fehlercodes Systeminformationsdienst

Fehlercode	ErrorType	Severity	Fehlertext
4095	Technical	Error	Fehler bei der Auswertung eines XPath-Ausdruck
4096	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
4100	Technical	Error	Ungültige HSM ID
4101	Technical	Error	Karten-Handle ungültig
4102	Technical	Error	Ungültige SubscriptionId

Tabelle 350 - TAB_KON_698 Fehlercodes Verschlüsselungsdienst

Fehlercode	ErrorType	Severity	Fehlertext
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
4106	Technical	Error	Falscher Schlüssel

Fehlercode	ErrorType	Severity	Fehlertext
4107	Technical	Error	Hybride Entschlüsselung konnte nicht durchgeführt werden
4108	Technical	Error	Symmetrische Verschlüsselung konnte nicht durchgeführt werden
4109	Technical	Error	Symmetrische Entschlüsselung konnte nicht durchgeführt werden
4200	Security	Error	Schlüssel erlaubt keinen zugelassenen Verschlüsselungsalgorithmus
4201	Technical	Error	Kryptographischer Algorithmus vom Konnektor nicht unterstützt

Tabelle 351 - TAB_KON_699 Fehlercodes Signaturdienst und SAK

Fehlercode	ErrorType	Severity	Fehlertext
4023	Technical	Error	XML-Dokument nicht valide in Bezug auf XML-Schema
4041	Technical	Error	Fehler im Pairing, SICCT-Fehler: <SICCT-Fehler>
4058	Technical	Error	Aufruf nicht zulässig
4060	Technical	Error	Ressource belegt
4110	Technical	Error	Ungültiges Dokumentformat (%Format%) Der Parameter Format enthält das übergebene Dokumentformat.
4111	Technical	Error	Ungültiger Signaturyp oder Signaturvariante
4112	Technical	Error	Dokument nicht konform zu Regeln für nonQES
4115	Security	Error	Signatur des Dokuments ungültig. Der SignatureValue des Dokumentes ist falsch oder für mindestens eine Reference ist der DigestValue falsch.
4116	Technical	Warning	Timeout (Benutzer)
4117	Technical	Error	Kein Extended Trusted Viewer zugeordnet
4118	Technical	Error	Stapelsignaturen werden nur für den HBA unterstützt. Mit HBA-Vorläuferkarten sind nur Einzelsignaturen möglich.
4119	Technical	Error	Darstellungs-Alias unbekannt
4120	Security	Error	Kartenfehler
4121	Security	Error	Kurztext für Listenanzeige konnte nicht gebildet werden
4122	Security	Error	Fehler bei Anzeige im xTV
4123	Security	Error	Fehler bei Signaturerstellung
4124	Security	Error	Dokument nicht konform zu Regeln für QES

Fehlercode	ErrorType	Severity	Fehlertext
4125	Security	Error	LU_SAK nicht aktiviert
4126	Security	Error	Kartentyp nicht zulässig für Signatur
4197	Technical	Warning	Parameter SignaturePlacement wurde ignoriert
4205	Technical	Error	Es ist nicht genügend Speicherplatz im PDF-Dokument verfügbar
4206	Technical	Error	Signaturzertifikat ermitteln ist fehlgeschlagen
4207	Technical	Error	Referenzzeitpunkt bestimmen ist fehlgeschlagen
4208	Technical	Error	Dokument nicht konform zu Profilierung der Signaturformate

Tabelle 352 - TAB_KON_700 Fehlercodes Zertifikatsdienst

Fehlercode	ErrorType	Severity	Fehlertext
4127	Security	Error	TSL nicht gültig
4128	Technical	Error	Der manuelle Import der TSL-Datei schlägt fehl
4130	Security	Error	Signaturprüfung der CRL fehlgeschlagen
4131	Technical	Fatal	Zum angegebenen CardHandle keine Karte gefunden
4132	Security	Error	Extraktion des Ablaufsdatums fehlschlägt
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
4149	Technical	Error	Ungültige Zertifikatsreferenz
4090	Security	Error	Zugriff auf eGK nicht gestattet
4196	Technical	Error	Fehler bei der CV-Zertifikatsprüfung

Tabelle 353 - TAB_KON_701 Fehlercodes Protokollierungsdienst

Fehlercode	ErrorType	Severity	Fehlertext
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
4153	Security	Error	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

Tabelle 354 - TAB_KON_702 Fehlercodes TLS-Dienst

Fehlercode	ErrorType	Severity	Fehlertext
4156	Security	Error	Server konnte bei TLS-Verbindungsaufbau nicht authentisiert werden
4157	Security	Error	Clientauthentisierung bei TLS-Verbindungsaufbau fehlgeschlagen
4158	Technical	Error	Adressierte TLS-Verbindung nicht vorhanden

Tabelle 355 - TAB_KON_703 Fehlercodes Anbindung LAN/WAN

Fehlercode	ErrorType	Severity	Fehlertext
4159	Technical	Fatal	Public-IP: DNS Server antwortet nicht
4160	Technical	Fatal	Public-IP: Zu einem DNS Namen konnte keine IP-Adresse gefunden werden
4161	Technical	Fatal	Public-IP: Ein oder mehrere IP-Adressen sind ungültig
4162	Technical	Error	Es liegt eine fehlerhafte LAN IP-Konfiguration vor.
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.
4165	Technical	Fatal	gSMC-K Konfiguration: Keine Netzwerk-Konfiguration gefunden.
4166	Technical	Fatal	gSMC-K Konfiguration: Ein oder mehrere Netzwerk-Adressen sind ungültig.
4167	Technical	Fatal	CreateRoutes: Ein oder mehrere Adressen sind ungültig.

Tabelle 356 - TAB_KON_704 Fehlercodes DHCP-Server

Fehlercode	ErrorType	Severity	Fehlertext
4168	Technical	Error	DHCP-Server konnte nicht gestartet werden

Tabelle 357 - TAB_KON_705 Fehlercodes DHCP-Client

Fehlercode	ErrorType	Severity	Fehlertext
4169	Technical	Error	Konnektor erhält keine DHCP-Informationen

4170	Technical	Error	Konnektor besitzt identische IP-Adressen am WAN und LAN Interface

Tabelle 358 - TAB_KON_706 Fehlercodes VPN-Client

Fehlercode	ErrorType	Severity	Fehlertext
4171	Technical	Fatal	Der VPN-Tunnel zur TI konnte nicht beendet werden.
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: Verbindung konnte nicht aufgebaut werden
4175	Technical	Fatal	Der VPN-Tunnel zum SIS konnte nicht beendet werden.
4176	Technical	Fatal	SIS VPN-Tunnel: Verbindung konnte nicht aufgebaut werden.

Tabelle 359 - TAB_KON_708 Fehlercodes Zeitdienst

Fehlercode	ErrorType	Severity	Fehlertext
4177	Technical	Warning	Der NTP-Server des Konnektors konnte nicht synchronisiert werden.
4178	Technical	Error	Das Fachmodul konnte die aktuelle Systemzeit des Konnektors nicht abrufen“.

Tabelle 360 - TAB_KON_709 Fehlercodes Namensdienst und Dienstlokalisierung

Fehlercode	ErrorType	Severity	Fehlertext
4179	Technical	Error	DNS: Anfrage wurde abgebrochen, da der Timeout von ANLW_SERVICE_TIMEOUT Sekunden überschritten wurde.
4180	Technical	Fatal	"DNS: Es ist ein Fehler bei der Namensauflösung aufgetreten" Die Fehlerdetails sind gemäß [gemSpec_Net] zu ergänzen.

Tabelle 361 - TAB_KON_710 Fehlercodes Software-Aktualisierungsdienst (KSR-Client)

Fehlercode	ErrorType	Severity	Fehlertext
4181	Security	Error	Integritätsprüfung UpdateInformation fehlgeschlagen.
4182	Security	Error	Download nicht aller UpdateFiles möglich.
4183	Security	Error	Integritätsprüfung UpdateFiles fehlgeschlagen.

Fehlercode	ErrorType	Severity	Fehlertext
4184	Security	Error	Anwendung der UpdateFiles fehlgeschlagen (<Details>).
4185	Security	Error	Firmware-Version liegt außerhalb der gültigen Firmware-Gruppe
4186	Security	Error	Download nicht aller UpdateFiles möglich.
4187	Security	Error	KT-Update fehlgeschlagen (<Fehlerinfo gemäß SICCT>)
4188	Technical	Error	Konfigurationsdienst nicht erreichbar, konfigurierte Adresse kontrollieren.
4189	Security	Fatal	Konfigurationsdienst liefert falsches Zertifikat
4190	Technical	Error	Fehler beim Beziehen der Updatelisten
4198	Technical	Error	Beim Übernehmen der Bestandsnetze ist ein Fehler aufgetreten

Anhang H - Mapping von „Architektur der TI-Plattform “ auf Konnektorspezifikation

Tabelle 362 - TAB_KON_711 Architektur der TI-Plattform, Berechtigt Fachmodule

Interface	Operation	→	Funktionsmerkmal	Interface	Operation / TUC
I_Cert_Verification	verify_Certificate	→	Zertifikatsdienst		TUC_KON_037 "Zertifikat prüfen"
I_Crypt_Operations	decrypt_Document	→	Verschlüsselungsdienst		TUC_KON_071 "Daten hybrid entschlüsseln"
	encrypt_Document	→	Verschlüsselungsdienst		TUC_KON_070 "Daten hybrid verschlüsseln"
I_DNS_Name_Information	get_FQDN	→	Namensdienst und Dienstlokalisierung		TUC_KON_364 „DNS Reverse Lookup durchführen“
	get_IP_Address	→	Namensdienst und Dienstlokalisierung		TUC_KON_361 „DNS Namen auflösen“
I_DNS_Service_Informatio n	get_Service_Information	→	Namensdienst und Dienstlokalisierung		TUC_KON_362 „Liste der Dienste abrufen“ TUC_KON_363 „Dienstdetails abrufen“
I_IP_Transport	send_Data_TI	→			
I_KT_Operations	interact_with_User	→	Kartenterminaldienst		TUC_KON_051 "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 "Entschlüssele"
	do_Reset	→	Kartendienst		TUC_KON_024 "Karte zurücksetzen"
	extract_card_data	→	Zertifikatsdienst		TUC_KON_034 "Zertifikatsinformationen extrahieren"
	read_Card_Data	→	Kartendienst		TUC_KON_202 "LeseDatei"

Interface	Operation	→	Funktionsmerkmal	Interface	Operation / TUC
		→	Kartendienst		TUC_KON_209 "LeseRecord"
		→	Kartendienst		TUC_KON_215 "SucheRecord"
	read_KVK	→	Kartendienst		TUC_KON_202 "Lese Datei"
	send_APDU	→	Kartendienst		TUC_KON_200 "SendeAPDU"
	sign_Data	→	Kartendienst		TUC_KON_218 "Signiere"
	verify_eGK	→	Kartendienst		TUC_KON_018 "eGK-Sperrung prüfen"
	write_Card_Data	→	Kartendienst		TUC_KON_203 "SchreibeDatei"
		→	Kartendienst		TUC_KON_210 "SchreibeRecord"
		→	Kartendienst		TUC_KON_214 "FügeHinzuRecord"
	write_eGK_Protocol	→	Kartendienst		TUC_KON_006 "Datenzugriffsaudit eGK schreiben"
I_KV_Card_Reservation	handle_Session	→	Kartendienst		TUC_KON_023 "Karte reservieren"
I_KV_Card_Unlocking	authorize_Card	→	Kartendienst		TUC_KON_005 "Card-to-Card authentisieren"
	change_PIN	→	Kartendienst		TUC_KON_019 "PIN ändern"
	do_C2C	→	Kartendienst		TUC_KON_005 "Card-to-Card authentisieren"
	get_PIN_Status	→	Kartendienst		TUC_KON_022 "Liefere PIN-Status"
	initialize_PIN	→	Kartendienst		TUC_KON_019 "PIN ändern"
	unblock_PIN	→	Kartendienst		TUC_KON_021 "PIN entsperren"
	verify_PIN	→	Kartendienst		TUC_KON_012 "PIN verifizieren"
I_Notification_From_FM	notify	→	Systeminformationsdienst		TUC_KON_256 "Systemereignis absetzen"
I_Poll_System_Informatio n	get_Ressource_Information	→	Systeminformationsdienst		TUC_KON_254 "Liefere Ressourcendetails"

Interface	Operation	→	Funktionsmerkmal	Interface	Operation / TUC
	get_Ressource_List	→	Systeminformationsdienst		TUC_KON_252 "Liefere KT_Liste"
		→	Systeminformationsdienst		TUC_KON_253 "Liefere Karten_Liste"
I_Reg_Notification	register_for_Notifications	→	---		*--- keine Umsetzung notwendig. Erfolgt implizit*
I_SAK_Operations	sign_Document_QES	→	Signaturdienst und SAK		TUC_KON_150 „Dokumente QES signieren“
	verify_Document_QES	→	Signaturdienst und SAK		TUC_KON_151 "QES Dokumentensignatur prüfen"
I_Sign_Operations	sign_Document	→	Signaturdienst und SAK		TUC_KON_160 „Dokumente nonQES signieren“
	external_Authenticate	→	Signaturdienst und SAK		TUC_KON_160 „Dokumente nonQES signieren“
	verify_Document	→	Signaturdienst und SAK		TUC_KON_161 „nonQES Dokumentsignatur prüfen“
	get_Certificate	→	Kartendienst		TUC_KON_216 „LeseZertifikat“
I_Symm_Crypt_Operations	decrypt_Document_Symmetric	→	Verschlüsselungsdienst		TUC_KON_073 "Daten symmetrisch entschlüsseln"
	encrypt_Document_Symmetric	→	Verschlüsselungsdienst		TUC_KON_072 "Daten symmetrisch verschlüsseln"
I_Synchronised_System_Time	get_Time	→	Zeitdienst		TUC_KON_351 " Liefere Systemzeit"
I_TLS_Client	send_Secure	→	TLS-Dienst		TUC_KON_110 "Kartenbasierte TLS-Verbindung aufbauen"
		→	TLS- Dienst		TUC_KON_111 "Kartenbasierte TLS-Verbindung abbauen"
		→	Anbindung LAN/WAN		AFOs: Routing der IP-Pakete von Fachmodul(=Konnektor intern) --> VPN_TI

Interface	Operation	→	Funktionsmerkmal	Interface	Operation / TUC
I_Directory_Query	search_Directory	→	LDAP-Proxy		TUC_KON_290 „LDAP-Verbindung aufbauen“
		→	LDAP-Proxy		TUC_KON_291 „Verzeichnis abfragen“
		→	LDAP-Proxy		TUC_KON_292 „LDAP-Verbindung trennen"
		→	LDAP-Proxy		TUC_KON_293 „Verzeichnisabfrage abbrechen"

Tabelle 363 - TAB_KON_712 Architektur der TI-Plattform, Berechtigt Clientsysteme

Interface	Operation	→	Funktionsmerkmal	Interface / TUC	Operation
I_Crypt_Operations	decrypt_Document	→	Verschlüsselungsdienst	EncryptionService	DecryptDocument
	encrypt_Document	→	Verschlüsselungsdienst	EncryptionService	EncryptDocument
I_DNS_Name_Resolution	get_FQDN	→	Namensdienst und Dienstlokalisierung	GetFQDN	I_DNS_Name_Resolution
	get_IP_Address	→	Namensdienst und Dienstlokalisierung	GetIPAddress	
I_IP_Transport	send_Data_External	→	Anbindung LAN/WAN		AFOs: Routing der IP-Pakete von Client --> VPN_SIS
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_Unlocking	authorize_Card	→	Kartendienst	CardService	AuthorizeSMC
	change_PIN	→	Kartendienst	CardService	ChangePin
	get_PIN_Status	→	Kartendienst	CardService	GetPinStatus

© gematik – öffentlich

Interface	Operation	→	Funktionsmerkmal	Interface / TUC	Operation
	initialize_PIN	→	Kartendienst	CardService	ChangePin
	unblock_PIN	→	Kartendienst	CardService	UnblockPin
	verify_PIN	→	Kartendienst	CardService	VerifyPin
I_Poll_System_Information	get_Ressource_Information	→	Systeminformationsdienst	EventService	GetResourceInformation
	get_Ressource_List	→	Systeminformationsdienst	EventService	GetCardTerminals
	get_Ressource_List	→	Systeminformationsdienst	EventService	GetCards
	get_Ressource_List	→	Systeminformationsdienst	EventService	GetHSMs
I_Reg_Notification	register_for_Notifications	→	Systeminformationsdienst	EventService	Subscribe
		→	Systeminformationsdienst	EventService	Unsubscribe
		→	Systeminformationsdienst	EventService	GetSubscription
I_SAK_Operations	sign_Document_QES	→	Signaturdienst und SAK	SignatureService	SignDocument
	verify_Document_QES	→	Signaturdienst und SAK	SignatureService	VerifyDocument
I_Sign_operations	sign_Document	→	Signaturdienst und SAK	SignatureService	SignDocument
	external_Authenticate	→	Signaturdienst und SAK	SignatureService	ExternalAuthenticate
	verify_Document	→	Signaturdienst und SAK	SignatureService	VerifyDocument
	get_Certificate	→	Zertifikatsdienst	CertificateService	ReadCardCertificate
I_TV_Management	configure_TV	→	Signaturdienst und SAK	SignatureService	SetXTVView
		→	Signaturdienst und SAK	SignatureService	GetXTVView
I_TV_User_Interaction	display_Document	→	Signaturdienst und SAK		Im Rahmen von TUC_KON_153
	display_Metadata	→	Signaturdienst und SAK		Im Rahmen von TUC_KON_153

Interface	Operation	→	Funktionsmerkmal	Interface / TUC	Operation
	request_Confirmation	→	Signaturdienst und SAK		Im Rahmen von TUC_KON_153
I_NTP_Time_Information	sync_Time	→	Zeitdienst		TIP1-A_2331 „I_NTP_Time_Information“
I_Directory_Query	search_Directory	→	LDAP-Proxy	LDAP-Operation	TIP1-A_5521

Tabelle 364 - TAB_KON_713 Architektur der TI-Plattform, Berechtigt eHealth-KT

Interface	Operation	→	Funktionsmerkmal	Interface / TUC	Operation
I_Notification	notify	→	SICCT	Ereignisdienst	SICCT-Ereignisnachrichten
		→	SICCT	Ereignisdienst	Service Announcement

Tabelle 365 - TAB_KON_714 Architektur der TI-Plattform, Berechtigt Administrator

Interface	Operation	→	Funktionsmerkmal	Interface / TUC	Operation
I_Change_System_Time	set_System_Time	→	Zeitdienst		TIP1-A_4793 Konfigurierbarkeit des Konnektor NTP Servers
I_Facade_Access_Configuration	add_Clientsystem	→	Anbindung Clientsysteme		TIP1-A_4518 Konfiguration der Anbindung Clientsysteme
	remove_Clientsystem	→	Anbindung Clientsysteme		TIP1-A_4518 Konfiguration der Anbindung Clientsysteme
	set_CS_Access_Mode	→	Anbindung Clientsysteme		TIP1-A_4518 Konfiguration der Anbindung Clientsysteme
I_KSRC_Local_Management	do_local_Update	→	Software-Aktualisierung	TUC_KON_280	I_KSRC_Local_Management

Interface	Operation	→	Funktionsmerkmal	Interface / TUC	Operation
			(KSR-Client)	"Konnektoraktualisierung durchführen"	
I_KSRC_Management	do_Update	→	Software-Aktualisierung (KSR-Client)	TUC_KON_280 "Konnektoraktualisierung durchführen"	I_KSRC_Management
			Software-Aktualisierung (KSR-Client)	TUC_KON_281 "Kartenterminalaktualisierung anstoßen"	
	list_available_Updates	→	Software-Aktualisierung (KSR-Client)	TUC_KON_282 "UpdateInformationen beziehen"	
I_KTV_Management	configure_KTs	→	Kartenterminalverwaltung	Managementschnittstelle	TIP1-A_4555 Manuelles Hinzufügen eines Kartenterminals
		→	Kartenterminalverwaltung	Managementschnittstelle	TIP1-A_4540 Reaktion auf KT Service Announcement
		→	Kartenterminalverwaltung	Managementschnittstelle	TIP1-A_4556 Pairing mit Kartenterminal durchführen
		→	Kartenterminalverwaltung	Managementschnittstelle	TIP1-A_4557 Ändern der Korrelationswerte eines Kartenterminals

Tabelle 366 - TAB_KON_715 Architektur der TI-Plattform, Berechtigt LE

Interface	Operation	→	Funktionsmerkmal	Interface / TUC	Operation
I_SAK_Self_Test	check_SAK	→	Signaturdienst und SAK	-	-

Anhang I - Umsetzungshinweise (informativ)

In diesem Anhang finden sich Darstellungen und Informationen, die ein Konnektorhersteller zur Umsetzung der normativen Anforderungen in ein konkretes Produkt berücksichtigen kann. Sie wurden im Rahmen der Erhebung der normativen Anforderungen erarbeitet, um die Umsetzbarkeit der Anforderungen zu bestätigen.

Dieser Anhang soll als Unterstützung für eine Umsetzung verstanden werden und erhebt keinen Anspruch auf Korrektheit und Vollständigkeit.

I1 - Systemüberblick

I1.1 - Hinweise zur Sicherheitsevaluierung nach Common Criteria.

Gemäß dem Sicherheitskonzept des Konnektors [gemKPT_Sich_Kon] muss die Software des Konnektors nach Common Criteria (CC) evaluiert und geprüft werden.

Diese Software erbringt Sicherheitsleistungen in zwei wesentlichen Funktionsblöcken. Durch diese Aufteilung ist es möglich, dass die einzelnen Funktionsblöcke zeitlich voneinander unabhängig bzw. sogar von unterschiedlichen Herstellern implementiert, evaluiert und geprüft werden können. Es werden zwei Schutzprofile (Protection Profile) für die Funktionsblöcke des Konnektors erstellt. Es handelt sich dabei um die Schutzprofile des Netzkonnektors (KONN.NK) sowie des Anwendungskonnektors (KONN.AK) inklusive der Signaturanwendungskomponente. Das Schutzprofil des Sicherheitsmoduls für den Konnektor (SM-K) wird in diesem Kapitel nicht betrachtet.

Diese Schutzprofile definieren eine implementierungsunabhängige Menge von Sicherheitsanforderungen für die einzelnen Konnektorfunktionsblöcke bzw. Konnektorbestandteile. Anhand dieser Schutzprofile werden von den Herstellern der Konnektoren die Sicherheitsvorgaben (Security Targets) für die konkreten Umgebungen erstellt, welche als Eingangsdokumente für den Zertifizierungsprozess der jeweiligen konkreten Komponenten eingesetzt werden. Diese zu evaluierenden Komponenten werden als Evaluierungsgegenstand (EVG) bezeichnet.

I1.1.1 - Separationsmechanismen des Konnektors

Damit es nach einer erfolgreichen Evaluierung eines Konnektors auch weiterhin möglich bleibt, Software oder Daten, die keinen direkten Einfluss auf Sicherheitsfunktionen der EVGs aufweisen, ohne eine Re-Evaluierung definiert auszutauschen, hinzuzufügen oder zu erweitern, ist eine Separation der Komponenten des EVG dringend anzuraten.

Implementiert der Hersteller keine bzw. nicht ausreichende Separationsmechanismen, so ist bei bestimmten Update-Arten von einer aufwändigen Re-Evaluierung des entsprechenden EVGs auszugehen. Die Separation dient also der Trennung zwischen ausführbarem Code des EVG, welcher Sicherheitsfunktionen umsetzt, und zusätzlichem ausführbarem Code auf dem Konnektor, welcher keine Sicherheitsfunktionen umsetzt.

Die Wahl der Separationsmechanismen steht dem Hersteller frei und muss in den Sicherheitsvorgaben für den EVG beschrieben und als solcher evaluiert werden. Aus

diesen Sicherheitsvorgaben ergibt sich auch, welche Update-Arten bei welchen Separationsmechanismen eine Re-Evaluierung des EVG erfordern und wie aufwendig diese Re-Evaluierung ausfällt.

Unter diese Update-Arten können beispielsweise – je nach Konnektorarchitektur, CC-Dokumentation oder Konnektorimplementierung – Bestandteile des unter dem Konnektor arbeitenden Betriebsystems, die Installation dezentraler Komponenten von Fachlogik oder Konfigurationsdaten des Konnektors fallen.

Als Beispiel für Separationsmechanismen sei auf die folgende informative Aufzählung verwiesen, welche jedoch keinen Anspruch auf Vollständigkeit besitzen kann und nur mögliche Alternativen aufzeigt:

- Java-Sandbox Konzept,
- Interpreter mit restriktiver Laufzeitprüfung,
- vom Betriebssystem bereitgestellte Prozess- und Speichertrennung,
- virtuelle Maschinen,
- physische Trennung durch separierte Hardware.

Je nach gewähltem Architekturansatz des Herstellers sind nicht alle hier genannten Alternativen für die Separation des EVG auf dem Konnektor anwendbar.

Insbesondere sollte der Hersteller den eigentlichen Update-Prozess und die dafür verantwortliche Komponente mit besonderer Sorgfalt beschreiben, spezifizieren und implementieren. Bei einer fehlerhaften Implementierung dieser Komponente besteht die Gefahr einer Schwächung oder des Ausschaltens von Sicherheitsfunktionen des EVG. Die Update-Komponente muss eine sichere Zuweisung der Updates zu den separierten Bestandteilen des EVGs gewährleisten. Auch ist zu betonen, dass der EVG immer die Integrität der Daten des Updates und die Authentizität des Absenders prüfen muss, bevor ein Update akzeptiert wird. Der Update-Komponente muss somit besondere Beachtung geschenkt werden.

I1.1.2 - Granularität der TSF

Die TSF (TOE Security Functionality) eines EVG besteht aus Subsystemen und Modulen, wobei ein Modul die genaueste Beschreibung einer Funktionalität darstellt und unterhalb der Subsysteme angesiedelt ist. Subsysteme beschreiben das Design des EVG und können wiederum – je nach Komplexität eines EVGs – aus weiteren Subsystemen bestehen. Ein Entwickler sollte außer der Modulbeschreibung keine weiteren Informationen zur Implementierung der dort beschriebenen Funktionalität benötigen.

Die Subsysteme und Module der TSF gliedern sich in drei Klassen:

(a) SFR-Enforcing Subsysteme und Module. Hierunter fallen die Subsysteme und Module, welche eine funktionale Sicherheitsanforderung direkt durchsetzen.

(b) SFR-Supporting Subsysteme und Module. Hierunter fallen die Subsysteme und Module, welche bei der Durchsetzung einer funktionalen Sicherheitsanforderung unterstützend wirken.

(c) SFR-Non-Interfering Subsysteme und Module. Hierunter fallen die Subsysteme und Module, welche keine Leistung bei der Erfüllung einer funktionalen Sicherheitsanforderung erbringen.

Sollte nach einer erfolgreichen CC-Evaluierung eines Konnektors die Notwendigkeit zur Änderung der Software des Konnektors gegeben sein, so ist unter Umständen eine Re-Evaluierung des EVG erforderlich. Diese Notwendigkeit kann sich aus der Behebung von nachträglich erkannten Fehlern, aufgetretenen Sicherheitslücken, Schwächen eines Standardverfahrens oder einer erforderlichen Erweiterung der Funktionalität ergeben.

Im Rahmen der Aufzählung der Anforderungen an die Beschreibung des EVG-Design (ADV_TDS) wird bereits die Aufteilung der TSF auf Subsysteme und Module beschrieben. Trotzdem soll hiermit ausdrücklich geraten werden, die Aufteilung der TSF auf die Subsysteme und Module selbst und die Aufteilung der Subsysteme und Module auf die drei o. g. Klassen möglichst feingranular durchzuführen.

Denn so

1. können einfacher umfassende Tests durchgeführt und die Testabdeckung sichergestellt werden,
2. kann bei der Veränderung von Programmcode der Evaluator die Auswirkungen auf SFR-Enforcing, Supporting oder Non-Interfering SFRs einfacher herausfinden und damit die Kosten und den zeitlichen Aufwand einer Re-Evaluierung senken.
3. kann bei der Veränderung von Programmcode, welcher als SFR-Non-Interfering eingestuft wird, das Maintenance-Verfahren anstelle einer Re-Evaluierung angewandt werden, welches einen erheblichen zeitlichen und damit auch monetären Vorteil gegenüber dem Re-Evaluierungsverfahren darstellt.

I2 - Übergreifende Festlegungen

I2.1 - Interne Mechanismen

I2.1.1 - Zufallszahlen und Schlüssel

Der Konnektor kann zur Erzeugung von Zufallszahlen und Einmalschlüsseln einen Hardware- oder Software-Generator verwenden. Als Quelle für Zufallszahlen kann der Konnektor die gSMC-K verwenden.

I3 - Funktionsmerkmale

I3.1 - Anwendungskonnektor

I3.1.1 - Administration des Informationsmodells

Wie die Administration der persistenten Entitäten und Beziehungen des Informationsmodells im Detail über die bereitzustellende Administrationsoberfläche erfolgt, entscheidet der Hersteller.

Es wird folgende Reihenfolge für die Pflege des Informationsmodells empfohlen.

1. Mandantenübergreifende Administration:
   - Es werden die Entitäten Arbeitsplätze, ExtendedTrustedViewer, Clientsysteme mit Authentifizierungsmerkmalen CS-AuthMerkmal und SMC-B_Verwaltet erfasst.
     Die Eingabe der Kartenterminals erfolgt über die Kartenterminalverwaltung.
   - Es wird die Beziehungen zwischen Arbeitsplatz und Kartenterminals „lokal“ und „entfernt (zentral)“ eingepflegt.
   - Es wird die Beziehung zwischen ExtendedTrustedViewer und Arbeitsplatz eingepflegt.
2. Mandantenbezogene Administration:
   - Die Definition bzw. Auswahl eines Mandanten bildet den Einstiegpunkt.
   - 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.

I3.1.2 - 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:

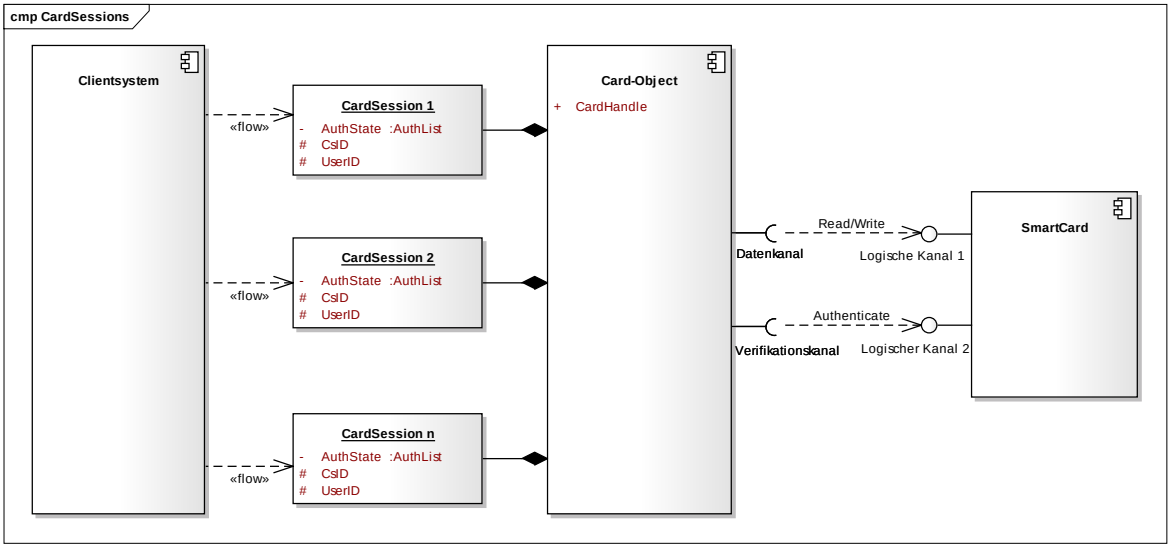


Abbildung 21: PIC_KON_120 Abbildung von CardSessions auf logische Kanäle

Der Kartendienst fungiert als Multiplexer. Er spiegelt die Zugriffsrechte der Karte und wendet deren Regeln selbständig gegen die unterschiedlichen Zugriffe durch Clientsysteme an.

Für jedes Card-Objekt wird „in Richtung Clientsystem“ pro Kontext genau eine CardSession erzeugt und verwaltet. Zugriffe des Clientsystems erfolgen somit „im Kontext“ einer CardSession.

In Richtung Karte verwendet das Card-Object genau zwei Kanäle (zwei logische oder einen logischen und einen Basiskanal):

- Einen Datenkanal, über den die Datenbewegungen und kryptographischen Operationen laufen und
- Einen Verifikationskanal, der ausschließlich für Authentisierungszwecke verwendet wird

In jeder CardSession werden die in ihrem Kontext erreichten Sicherheitszustände vermerkt. Das Vorgehen für die Durchführung der Verifikationen und des Vermerkens der erreichten Sicherheitszustände, sowie der Datenzugriffe folgt folgenden Regeln (hier für PIN-Verifikation, sinngleich auch für C2C):

- Soll über eine CardSession eine PIN-Verifikation für PinRef_A gegen eine Karte durchgeführt werden und der erhöhte Sicherheitszustand für PinRef_A ist noch nicht erreicht (bsp. direkt nach einem Karten-Reset), dann leite die Verifikation über den Datenkanal (initiale Freischaltung des Datenkanals für folgende Datenzugriffe).
- Soll über eine CardSession eine PIN-Verifikation für PinRef_A gegen eine Karte durchgeführt werden und der erhöhte Sicherheitszustand für PinRef_A ist auf dem Datenkanal bereits erreicht, dann leite die Verifikation über den Verifikationskanal.

- Wurde durch eine CardSession eine Verifikation für PinRef_A erfolgreich durchgeführt, wird dieser erreichte Sicherheitszustand für PinRef_A in der zugreifenden CardSession vermerkt
- Datenzugriffe auf oder Kryptooperationen mit Karten werden durch den Kartendienst nur zugelassen, wenn die zugreifende CardSession über einen für diese Zugriffe benötigten erhöhten Sicherheitszustand verfügt. Ist der benötigte Vermerk für die zugreifende CardSession nicht vorhanden, beantwortet der Kartendienst die Anfrage mit der passenden Kartenfehlermeldung. Es erfolgt keine Interaktion mit der Karte.

Diese Regeln führen dazu, dass eine durch CardSession Y fehlgeschlagene Verifikation für PinRef_A die zuvor erfolgreich durch CardSession X durchgeführt Verifikation nicht beeinflusst. Kartenzugriffe auf dem Datenkanal sind für CardSession X weiterhin möglich, da der Verlust des erhöhten Sicherheitszustands durch fehlerhafte Verifikation immer nur im Verifikationskanal erfolgt.

Dieser Mechanismus funktioniert mit zwei Kanälen zu einer Karte für beliebig viele CardSessions.

I3.1.3 - Darstellung von Terminal-Anzeigen auf einem Kartenterminal

Die folgenden Ausführungen dienen der Klarstellung für die korrekte Verwendung der zur Verfügung stehenden Datenobjekte (DO) zur Darstellung von Terminal-Anzeigen an einem Kartenterminal nach SICCT- und eHealth-Kartenterminal-Spezifikation.

Die SICCT-Spezifikation enthält eine Liste von Datenobjekten (DO), die von den Kartenterminals unterstützt werden müssen oder können. Dabei gibt es zwei Datenobjekte zur Anzeige von Terminal-Anzeigen: APPL DO und SMTBD DO.

Kartenterminals müssen APPL DO (steht für Application Label Data Object) unterstützen. APPL DOs müssen immer eine 7 Bit ISO646DE / DIN66003-Codierung enthalten [DIN66003].

Kartenterminals können SMTBD DO (steht für SICCT Message-To-Be-Displayed Data Object) unterstützen, müssen dieses aber nicht. Über SMTBD DOs können weitere Zeichensätze am Display angezeigt werden.

Der Konnektor soll APPL DOs verwenden. Er kann SMTBD DOs verwenden, wenn er sicherstellt, dass das angesteuerte Kartenterminal diese unterstützt und die dargestellte Meldung der Klartextmeldung entspricht, die mittels APPL DO erreicht worden wäre.

Um dem Kartenterminal den Umbruch längerer Texte über das Zeilenende hinaus zu erleichtern, enthalten die Terminal-Anzeigen das Zeichen 0x0B als „Soll-Zeilenumbrüche“. Die „Soll-Zeilenumbrüche“ werden nicht als Textzeichen gezählt. Sie zeigen einen potentiellen Zeilenumbruch an. Diese müssen vom Kartenterminal herausgefiltert werden und werden nicht durch andere Zeichen ersetzt.

Die Maximallänge für Terminal-Anzeigen beträgt ohne PIN-Eingabe (OUTPUT [O]) 48 Zeichen.

Besonderheit bei Terminal-Anzeigen, die zu einer PIN-Eingabe (INPUT [I]) auffordern:

Für die PIN-Eingabe wird eine strukturierte Terminal-Anzeige übergeben, aufgeteilt auf maximal 40 Zeichen für die Terminal-Anzeige plus maximal 10 Zeichen für den sog. PIN-

Prompt (bei Platz für zusätzliche 6 Zeichen für die PIN-Eingabe). Ein gültiger String hat die Form: <Terminal-Anzeige>0x0F<PIN-Prompt>. Auch die Terminal-Anzeige für Eingaben soll mit „Soll-Zeilenumbrüchen“ versehen werden.

Bei der Übertragung der Terminal-Anzeige ist auf die korrekte Codierung der Zeichenkette zu achten. Der einzige Zeichensatz, der von allen Kartenterminals unterstützt werden MUSS, ist (7 Bit) ISO646DE / DIN66003 [DIN66003]. Dadurch darf eine Terminal-Anzeige auch deutsche Sonderzeichen enthalten.

Hex Code	**…0**	**…1**	**…2**	**…3**	**…4**	**…5**	**…6**	**…7**	**…8**	**…9**	**…A**	**…B**	**…C**	**…D**	**…E**	**…F**
0…	*diverse Steuerzeichen - nicht verwendet -*															
1…	*diverse Steuerzeichen - nicht verwendet -*															
2…	*space*	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
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	*del*

Abbildung 22: PIC_KON_007 Übersicht Zeichensatz ISO646DE / DIN66003

I3.2 – Signaturanwenungskomponente

Adressatengerechte und benutzerfreundliche Präsentation

Der Extended Trusted Viewer (xTV) muss dem Benutzer die relevanten Informationen in einer verständlichen und adressatengerechten Form präsentieren. Hierbei sollen zunächst die wesentlichen Informationen in einer übersichtlichen und benutzerfreundlichen Weise dargestellt werden, so dass die zu signierenden Daten bzw. das Ergebnis einer Signaturprüfung auch für technisch nicht-versierte Benutzer intuitiv interpretiert werden kann.

Beispielsweise soll bei der Darstellung des Signaturergebnisses die Metapher einer Ampel (rot, gelb, grün) genutzt werden. Darüber hinaus sollen neben der Übersichts-darstellung auch bei Bedarf (z.B. im Fall eines Fehlers) oder auf Wunsch des Benutzers weitere technische Details bzgl. der zu signierenden Daten dargestellt werden. Dies kann z.B. das Ergebnis der Validierung eines Dokumentenformats, der Dokumenteninhalt oder das Prüfungsergebnis (d.h. vollständiger Prüfbericht mit Einzelergebnissen der jeweiligen Prüfschritte) sein. Um diese technischen Details auch für technisch wenig versierte Benutzer zugänglich zu machen, sollen zusätzlich erläuternde Angaben zu den einzelnen Elementen in Form einer kontext-sensitiven Hilfe angeboten werden.

Schnittstelle zwischen SAK_{KON} xTV

Die SAK_{KON} kann das Schlüsselmaterial und das Zertifikat der X.509-Identität des xTVerzeugen. Der Aufbau des X.509-Zertifikates des xTV ist herstellerspezifisch. Der Konnektor muss die X.509-Zertifikate der Instanzen des xTV in der SAK_{KON} hinterlegen. Die SAK_{KON} kann das erzeugte Schlüsselmaterial und das Zertifikat bei der Einrichtung des xTV auf einem Arbeitsplatz durch ein herstellerspezifisches Verfahren an die Instanz des xTV sicher übertragen.

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.

Szenario 1: Einfache Installation ohne spezielle Anforderungen und ohne bestehende Infrastruktur

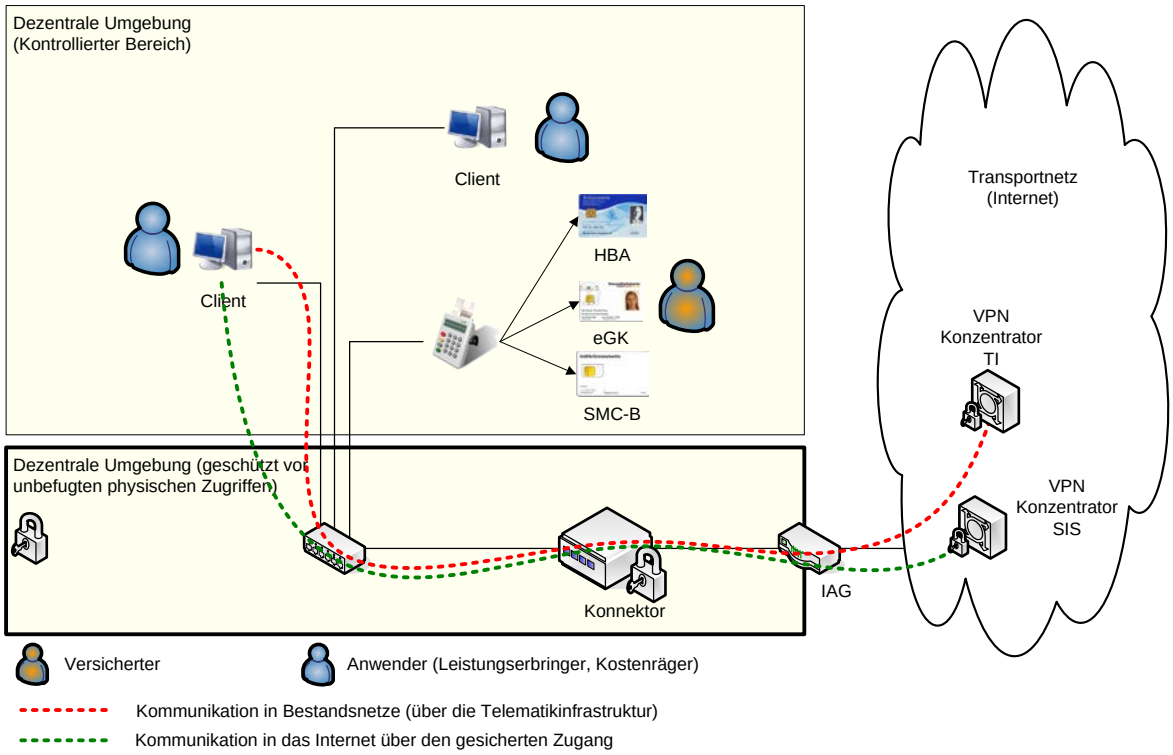


Abbildung 23 - Szenario einer einfachen Installation

Beschreibung des Szenarios

Abbildung 23 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 den Internetzugang ermöglichen und typischerweise vom Internet Service Provider (ISP) zur Verfügung gestellt werden (z.B. DSL Router und DSL Modem).

Ein oder mehrere Clientsysteme können über den Konnektor Anwendungsfälle der Telematikinfrastruktur initiieren und über den Konnektor und die zentrale TI-Plattform in Bestandsnetze kommunizieren (rote gestrichelte Linie). Dabei ist die Nutzung der Anwendungsfälle der Telematikinfrastruktur je nach Konfiguration des Konnektors entweder nur authentifizierten Clients möglich oder beliebigen Clients.

In diesem einfachen Szenario werden über ein einziges Kartenterminal die SMC-B, der HBA und auch die eGK des Versicherten gelesen, es können dazu alternativ auch mehrere Kartenterminals genutzt werden.

Darüber hinaus können die Clientsysteme über den SIS (Secure Internet Service) auf Dienste des Internets zugreifen.

Voraussetzungen

- Anbindung der bestehenden Clientsysteme an ein zum Konnektor kompatibles LAN muss möglich sein.
- Konfiguration des Konnektors als Default-Gateway in den Clientsystemen und Konfiguration der notwendigen VPN-Tunnel im Konnektor, um in die verschiedenen Netze zu routen.
- Verfügbarkeit einer SMC-B

Auswirkungen

- Die Clientsysteme können über den Konnektor Anwendungsfälle der TI initiieren
- Die Clientsysteme können über den Konnektor auf das Internet und Bestandsnetze zugreifen

Szenario 2: Installation mit mehreren Behandlungsräumen

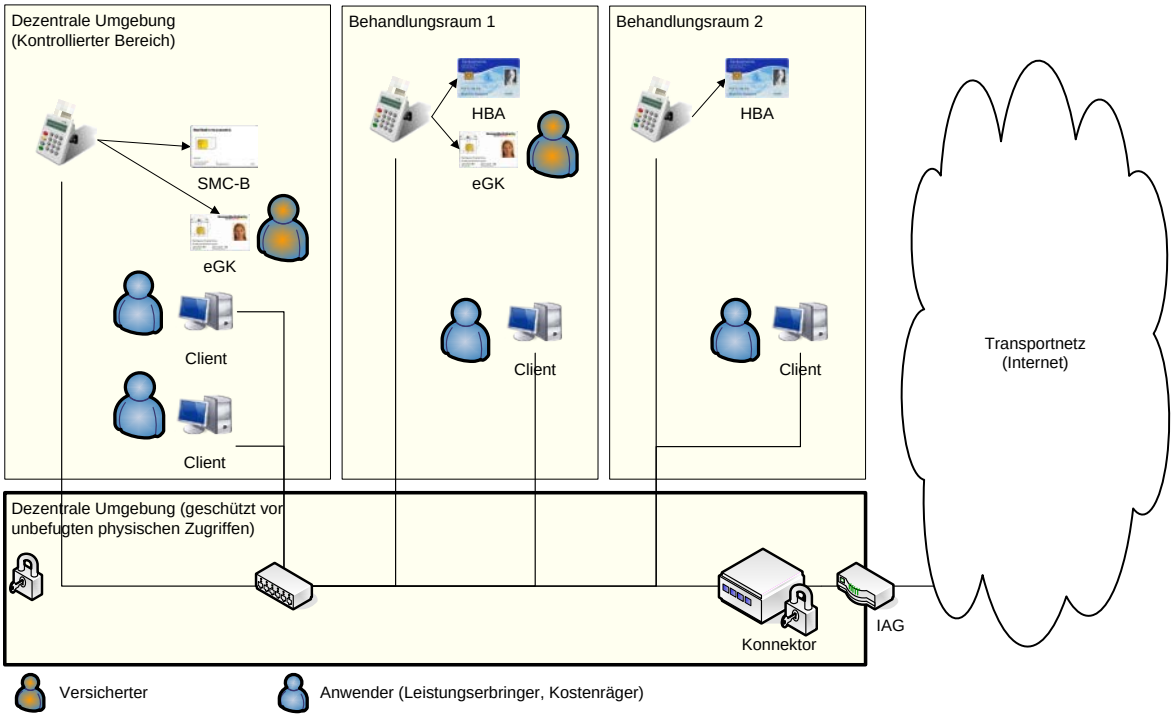


Abbildung 24 - Szenario einer Installation mit mehreren Behandlungsräumen

Beschreibung des Szenarios

Mit der in Szenario 1 skizzierten Topologie kann auch ein Szenario bedient werden, bei dem mehrere Behandlungsräume unterstützt werden (siehe Abbildung 24). 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 24 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.

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

- Die Clientsysteme und Kartenterminals und deren Relationen sind dem Konnektor über Konfiguration bekannt gemacht worden.

Auswirkungen

- Die Clientsysteme können über den Konnektor Anwendungsfälle der TI initiieren
- Die Clientsysteme können über den Konnektor auf das Internet (über den SIS) und Bestandsnetze zugreifen
- Der HBA-Inhaber muss seinen HBA mit sich führen und kann diesen in den einzelnen Kartenterminals der Behandlungsräume nutzen.

Szenario 3: Integration in bestehende Infrastruktur ohne Netzsegmentierung

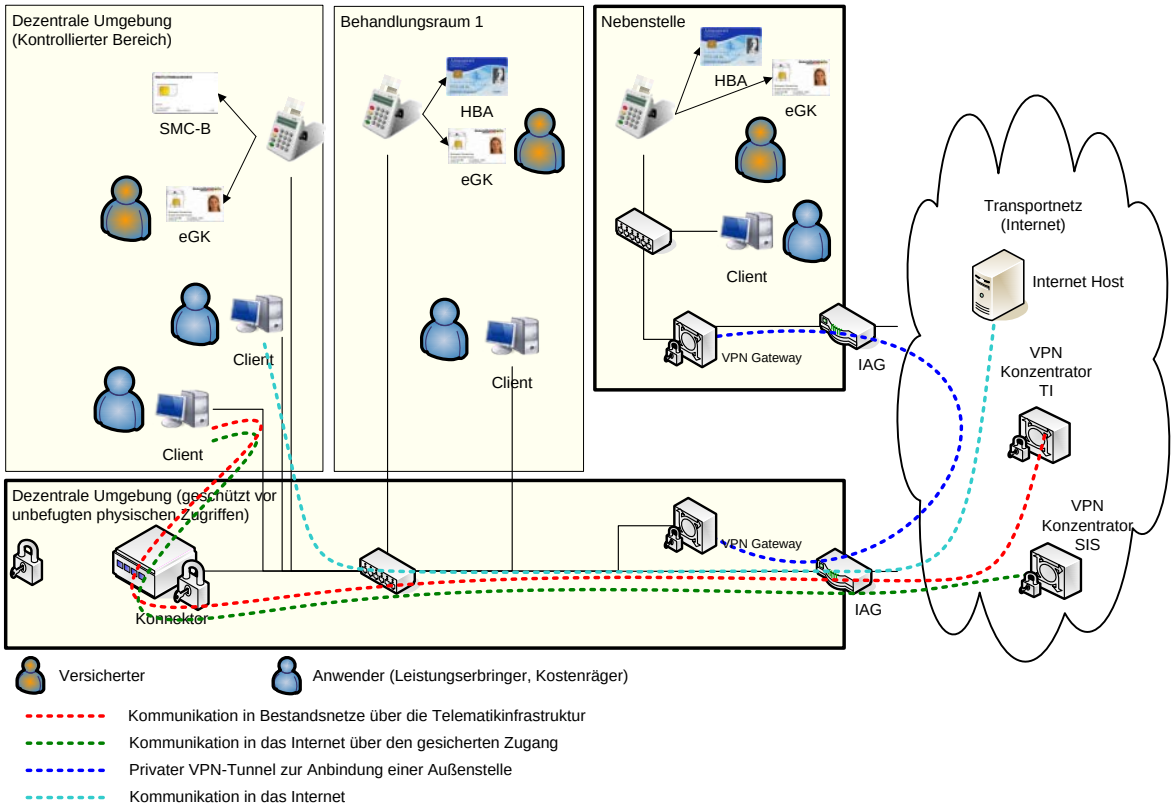


Abbildung 25 - Szenario einer Integration der TI Produkte in eine bestehende Infrastruktur

Beschreibung des Szenarios

Im Falle einer bereits vorhandenen Infrastruktur im dezentralen Bereich können die Produkte der TI, insbesondere der Konnektor, so in die Infrastruktur integriert werden, dass Bestandsanwendungen bereits erprobte Kommunikationswege weiter nutzen können.

Wie in Abbildung 25 beispielhaft dargestellt, existiert bereits eine Infrastruktur, die sowohl einen Internetzugang für die Arbeitsplätze ermöglicht (gestrichelte Linie in türkis), als auch eine Nebenstelle über VPN anbindet (gestrichelte Linie in blau). In diesem Fall wird der Konnektor als zusätzliches Gerät an das bestehende Netzwerk angeschlossen und nutzt den bereits vorhandenen Internetanschluss zur Kommunikation in die TI.

Für die Clientsysteme muss in diesem Szenario je nach individuellem Anforderungsprofil entschieden werden, ob das jeweilige Clientsystem über die Telematikinfrastruktur kommunizieren können soll und den gesicherten Internetzugang (SIS) nutzen soll oder nicht.

Soll ein Clientsysteme nicht über die Telematikinfrastruktur kommunizieren, bleibt der IAG als Default-Gateway dieses Clientsystems konfiguriert. In diesem Fall routet der IAG die eingehenden IP-Pakete mit öffentlichen Zieladressen weiter in das Internet. Die gestrichelte Linie in türkis zeigt beispielhaft einen Zugriff in das Internet.

Soll ein Clientsystem über die Telematikinfrastruktur kommunizieren oder den gesicherten Internetzugang (SIS) nutzen, muss der Konnektor als Default-Gateway konfiguriert werden. In diesem Fall routet der Konnektor die eingehenden IP-Pakete, die nicht für ihn bestimmt sind, entweder durch den VPN-Tunnel der TI über die Telematikinfrastruktur in ein angeschlossenes Bestandsnetz, (gestrichelte Linie in rot) oder durch den VPN-Tunnel zum SIS (Secure Internet Service) in das Internet (gestrichelte Linie in grün). Sollte kein sicherer Internetzugang konfiguriert sein, so würde der Konnektor den Traffic verwerfen und ggf. per ICMP dem Client eine anderes Gateway (IAG) vorschlagen. Alternativ können die von den Clients benötigten Routing-Informationen manuell oder per DHCP konfiguriert werden.

Voraussetzungen

- Konnektor ist kompatibel zur bestehenden Infrastruktur (Vernetzung)
- Die bestehende Infrastruktur verfügt über einen Internetzugang
- Verfügbarkeit einer SMC-B und mehrerer Kartenterminals
- Die Clientsysteme und Kartenterminals und deren Relationen sind dem Konnektor über Konfiguration bekannt gemacht worden.

Auswirkungen

- Produkte der Telematik können „minimal-invasiv“ in die bestehende Infrastruktur integriert werden. Bestehende Kommunikationswege können weiter genutzt werden.
- Für Clients kann je nach individuellen Anforderungsprofil der sichere Internetzugang über den Konnektor genutzt werden oder der direkte Internetzugang über den bestehenden IAG

Szenario 4: Integration in bestehende Infrastruktur mit Netzsegmentierung

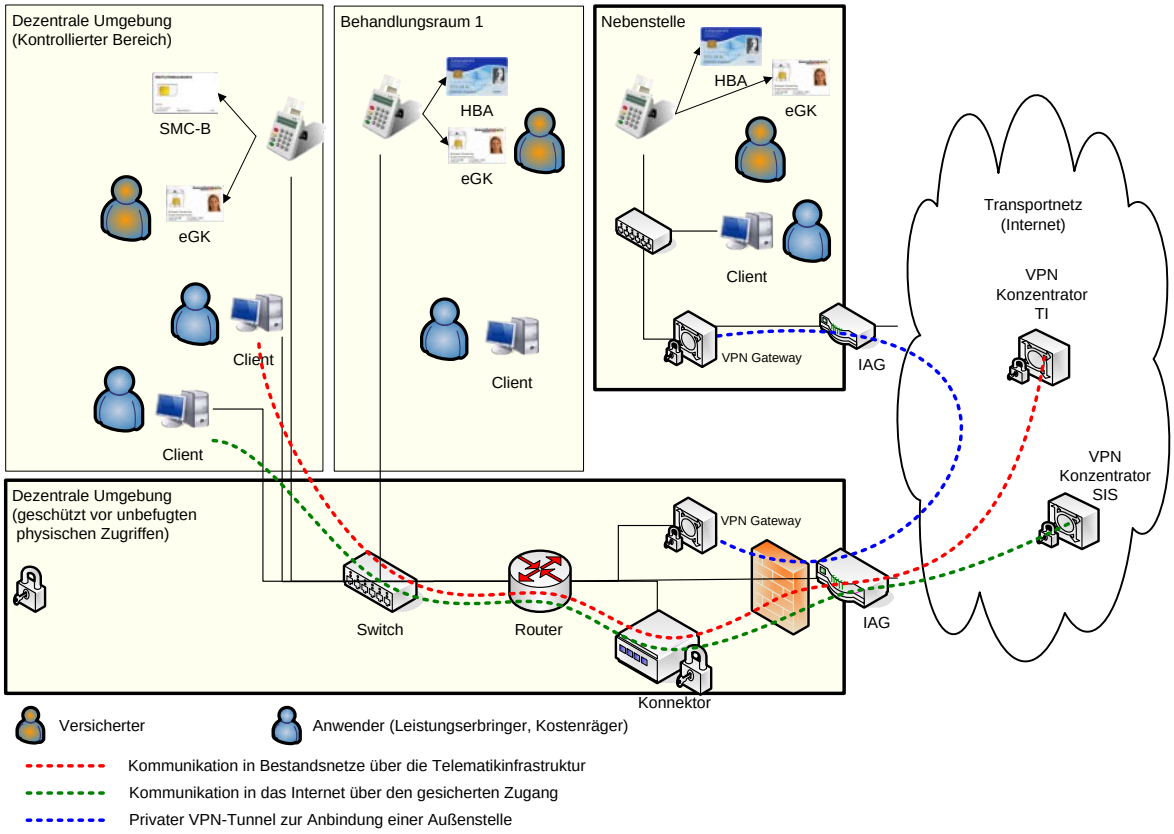


Abbildung 26 - Szenario einer Integration der TI Produkte in eine bestehende Infrastruktur mit existierendem Router

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.

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.

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.

Szenario 5: Zentral gesteckter HBA

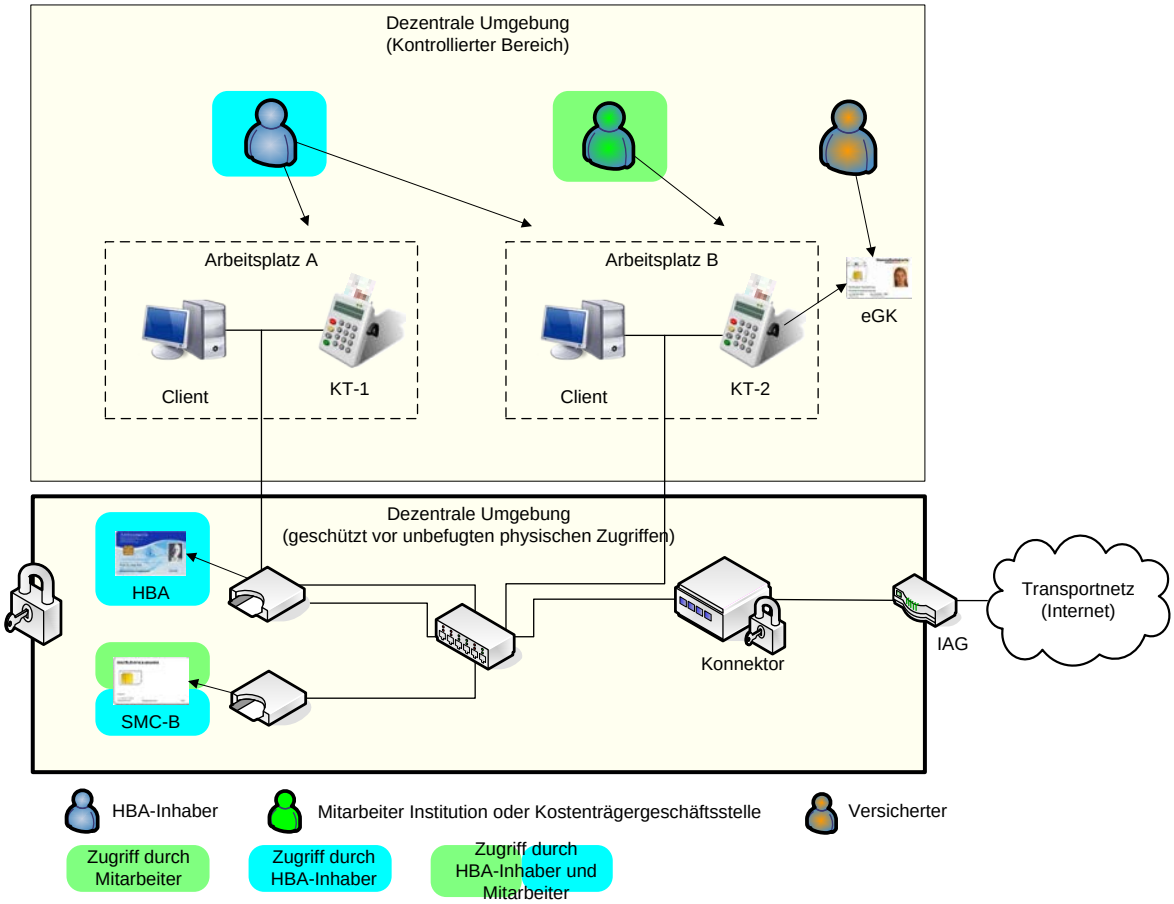


Abbildung 27 – Szenario mit zentral gesteckten HBA und SMC-B

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.

Voraussetzungen

Folgende zusätzliche Punkte müssen erfüllt sein, um dieses Szenario umzusetzen:

- Stecken der zentral gesteckten Karten HBA und SMC-B (ohne direkte Aufsicht) und Sicherstellung des Schutzes vor unbefugtem physischen Zugriff
- Konfiguration im Konnektor: Lokales eHealth-Kartenterminals als lokales eHealth-Kartenterminal für eine Remote-PIN-Eingabe eines bestimmten Arbeitsplatzes.
  Im abgebildeten Beispiel KT-1 für Arbeitsplatz A und KT-2 für Arbeitsplatz B.
- Konfiguration im Konnektor: Assoziation der gewünschten Arbeitsplätze zum jeweiligen Kartenterminal mit zentral gesteckter Karte.
  Im abgebildeten Beispiel Arbeitsplatz A assoziiert mit dem eHealth-Kartenterminal des HBAs und Arbeitsplatz B mit eHealth-Kartenterminal des HBAs und dem eHealth-Kartenterminal der SMC-B.

Auswirkung

- HBA muss nicht mehr durch seinen Inhaber mitgeführt werden
- SMC-B muss nicht mehr unter ständiger Aufsicht eines Mitarbeiters einer medizinischen Institution oder einer Kostenträgergeschäftsstelle sein.

Szenario 6: Installation mit zentralem PS

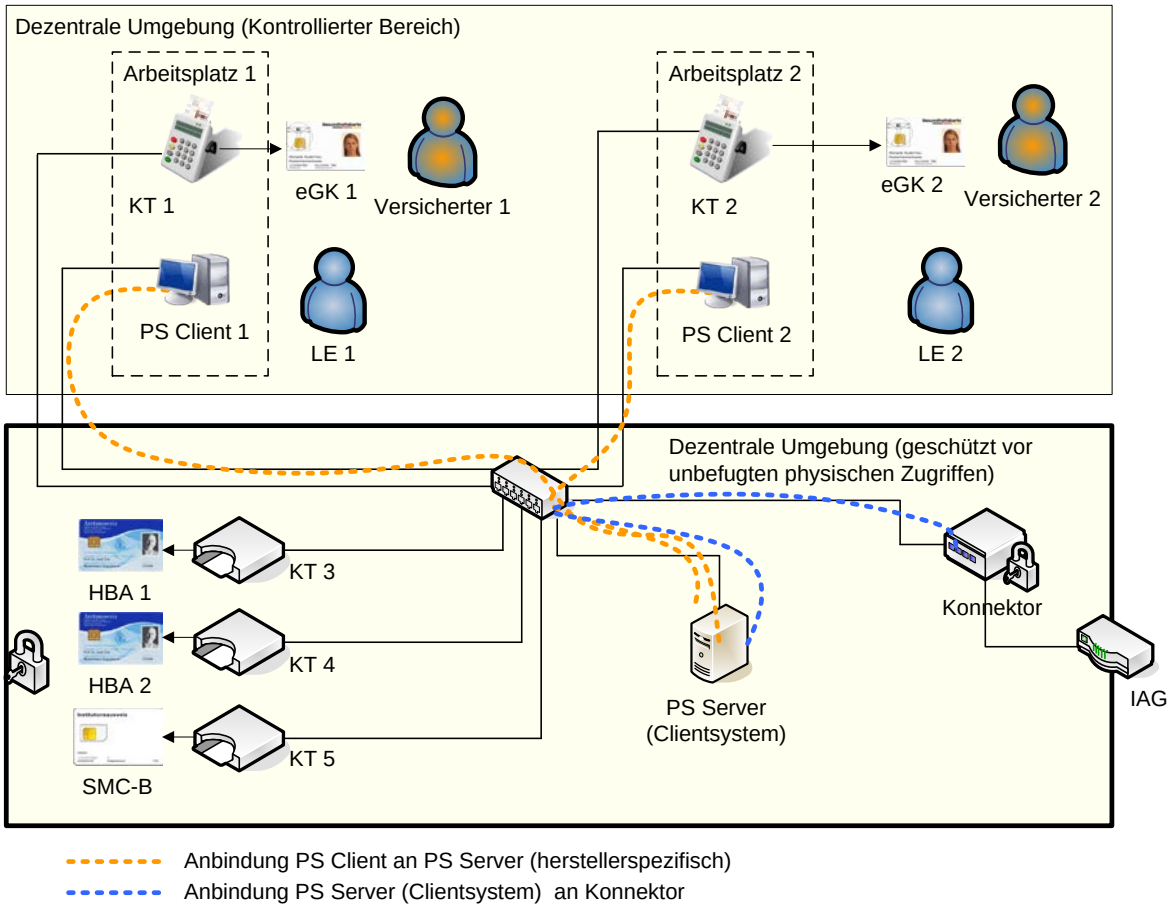


Abbildung 28 - Szenario mit zentralem Primärsystem als Clientsystem

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.

Voraussetzungen

- Netzanbindung aller Komponenten (u. a. KT, PS Client, PS Server, Konnektor) in der dezentralen Umgebung bis einschließlich zur Netzwerkschicht (IP-Ebene)
- Konfiguration des Primärsystems mit seinen Anteilen „PS Server“ und ggf. mehreren „PS Clients“ passend zum Informationsmodell des Konnektors (herstellerspezifisch).
- Konfiguration des Konnektors. U. a.:
    - Informationsmodel:
      Beim Beispielszenario u.a Entitäten „Clientsystem“ für „PS Server“, „Arbeitsplatz“ für „Arbeitsplatz 1“ und Arbeitsplatz 2“, „Kartenterminal“ und „KT-Slot“ für „KT 1“ – „KT 5“, „Mandat“ für die vorgesehene Anzahl von Mandaten, „SM-B_Verwaltet“ sowie entsprechende Entitätenbeziehungen.
    - Anbindung PS Server (ggf. über TLS)
    - Pairing der Kartenterminals
- Gesteckte Karten (SMC-B, HBA, eGK)
- Anmeldung Leistungserbringer am „PS Client“

Auswirkungen

- An den verschiedenen Arbeitsplätzen können für die definierten Mandaten und Nutzer Anwendungsfälle der TI initiiert werden.
- HBA-Inhaber müssen entsprechen der gewählten HBA-Deployment-Varianten
    - ihren HBA zentral stecken und über das Remote-PIN-Verfahren zugreifen
    - ihren HBA mit sich führen und lokal in Kartenterminal der Arbeitplätze stecken

Szenario 7: Mehrere Mandanten

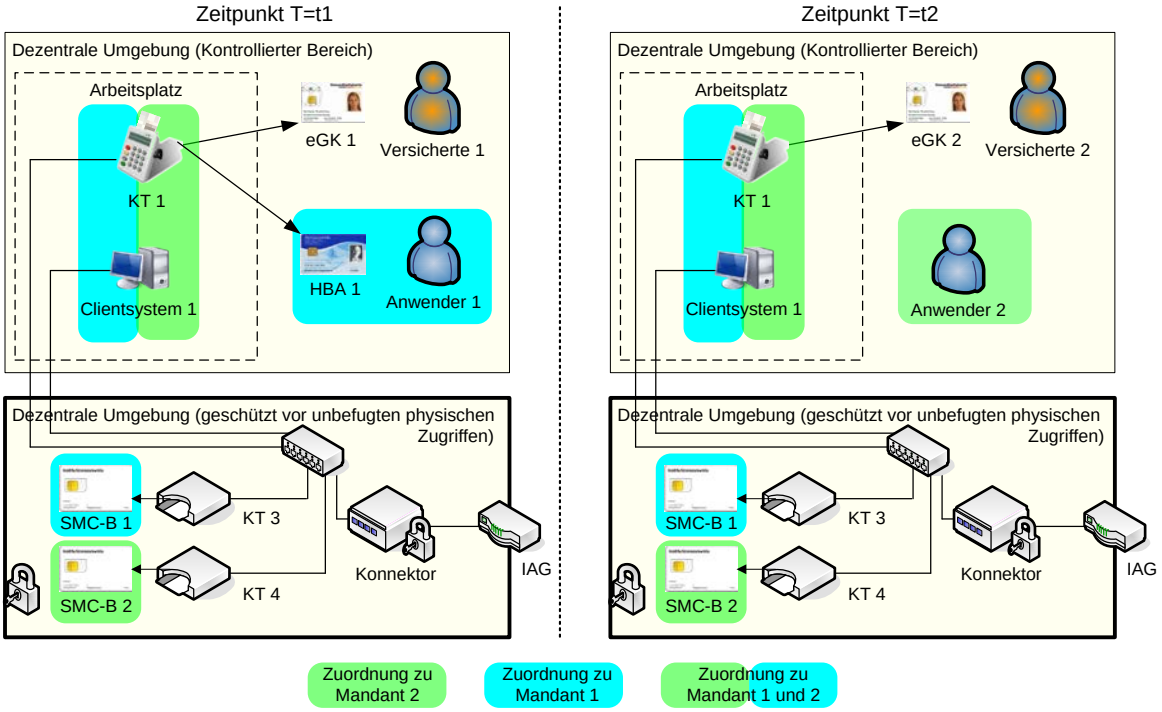


Abbildung 29 - Szenario für den Zugriff

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 Arbeitplätze, Clientsysteme und Kartenterminals müssen eine Zuordnung zum Mandanten haben, wobei Zuordnungen zu mehreren Mandaten möglich sind. Das Beispiel zeigt einen Arbeitplatz mit „Clientsystem 1“ und „KT 1“, der zu unterschiedlichen Zeiten durch verschiedene Mandaten 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.

Voraussetzungen

- Netzwerkanbindung 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).

- Konfiguration des Konnektors. U. a.:
    - Konfiguration Konnektor:
      Beim Beispielszenario u.a Entitäten „Clientsystem“ für „Clientsystem 1“, „Arbeitsplatz“ für „Arbeitsplatz 1“, „Kartenterminal“ und „KT-Slot“ für „KT 1“ – „KT 4“, „Mandat“ für „Mandant 1“ und „Mandant 2“, „SM-B_Verwaltet“ für „SMC-B 1“ und SMC-B 2“ sowie entsprechende Entitätenbeziehungen
    - Anbindung „Clientsystem 1“ (ggf. über TLS)
    - Pairing der Kartenterminals
- Gesteckte Karten (SMC-B 1, SMC-B 2, HBA 1, eGK 1, eGK 2)
- Anmeldung eines Anwenders mit Mandantenbezug am Clientsystem

Auswirkungen

- An den verschiedenen Arbeitsplätzen können für die definierten Mandaten und Anwender Anwendungsfälle der TI initiiert werden.
- HBA-Inhaber müssen entsprechen der gewählten HBA-Deployment-Varianten
    - ihren HBA zentral stecken und über das Remote-PIN-Verfahren zugreifen
    - ihren HBA mit sich führen und lokal in Kartenterminal der Arbeitplätze stecken

Szenario 8: Standalone Konnektor - Logische Trennung

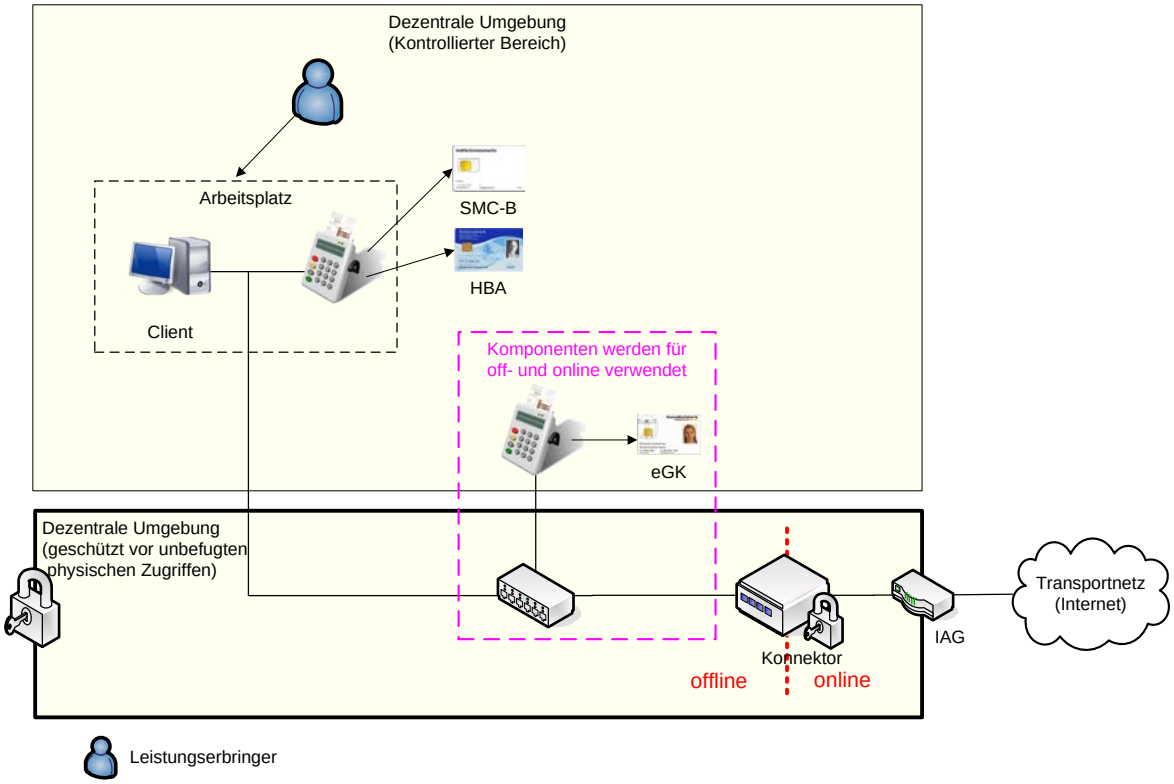


Abbildung 30 Standalone-Szenario mit logischer Trennung im Konnektor

Beschreibung des Szenarios

Dieses Szenario stellt eine Variante des Standalone-Szenarios dar, bei dem eine logische Trennung im Konnektors konfiguriert wurde.

Im Standalone-Szenario besteht eine Trennung zwischen den Praxissystemen der dezentralen Umgebung, welche offline (also, ohne Anbindung an die zentrale TI-Plattform) betrieben werden und den für das Update der eGK durch die Fachanwendung VSDM notwendigen Komponenten, welche online (also, mit Verbindung in die zentrale TI-Plattform) betrieben werden.

Die logische Trennung im Standalone-Szenario zeichnet sich dadurch aus, dass ein Standalone-Szenario ohne einen separaten Konnektor und ohne ein separates eHealth-Kartenterminal realisiert werden kann. Der Konnektor stellt dabei sicher, dass nur die Fachlogik von VSDM auf die Online-Funktionen der TI zugreifen kann. Den Praxissystemen steht alleinig die Offline-Funktionalität des Konnektors zur Verfügung.

Im Offline-Modus sind einzelne Funktionen des Konnektors nicht verfügbar, andere haben einen eingeschränkten Funktionsumfang. So kann z.B. eine QES erzeugt oder geprüft aber dabei keine aktuelle Statusauskunft (OCSP-Response) für die eingesetzten Zertifikate eingeholt werden. Dies hat zur Folge, dass bei Erzeugung einer QES keine Statusauskunft für das Signaturzertifikat in die Signatur eingebettet werden kann und bei einer Prüfung der QES nur eine eventuell in die Signatur eingebettet Statusauskunft des Zertifikats berücksichtigt werden kann.

Der Leistungserbringer muss in diesem Fall selber entscheiden, ob der gebotene Funktionsumfang für seinen Anwendungsfall ausreichend ist.

Voraussetzungen

Folgende zusätzliche Punkte müssen erfüllt sein, um dieses Szenario umzusetzen:

- Konfiguration im Konnektor: Logische Trennung
- Konfiguration im Konnektor: eHealth-Kartenterminal, welches für von der Fachlogik von VSDM für das Update der eGK mit Online-Verbindung verwendet werden soll.

Auswirkung

- Standalone-Szenario mit minimalem Systemausbau
- Eingeschränkte Funktionalität der TI für Praxissysteme (nur Offline-Funktionalität)
- Notwendige Prüfung des Leistungserbringers, ob eingeschränkte Funktionalität (insbesondere bei Sicherheitsfunktionen) akzeptabel ist.
- Keine Verfügbarkeit des sicheren Internetzugangs der TI (als Teil der Einschränkungen im Offline-Modus).

Szenario 9: Standalone Konnektor - Physische Trennung

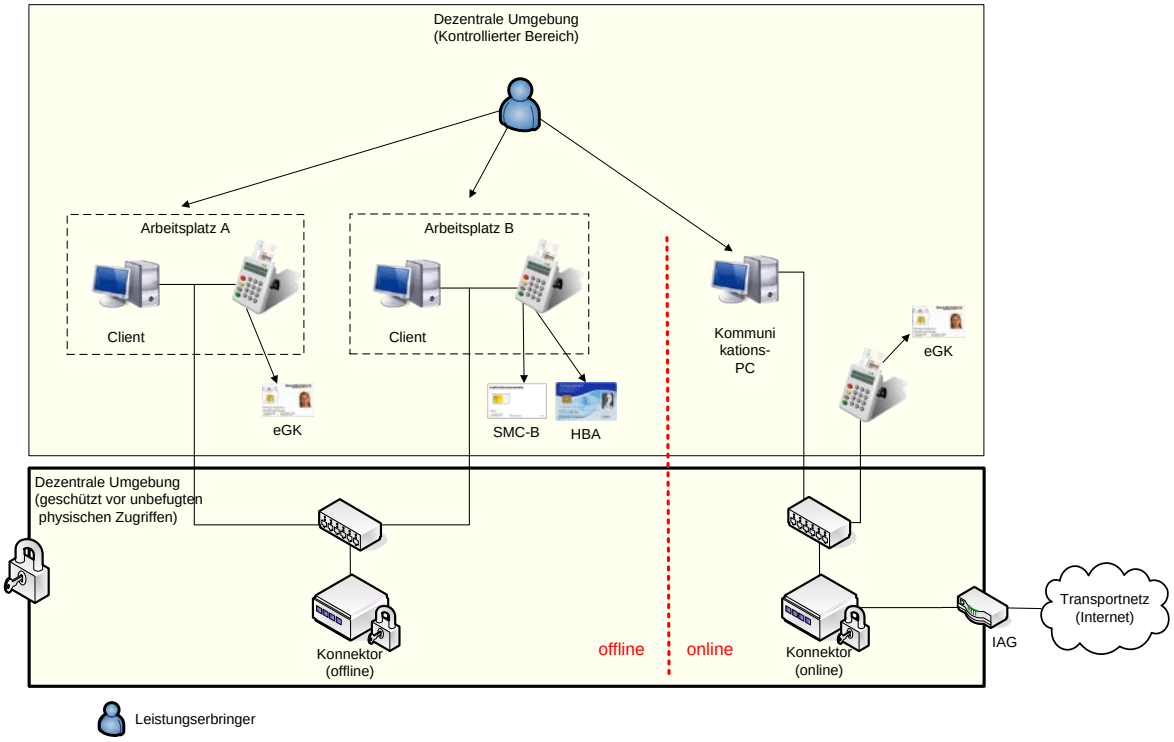


Abbildung 31 Standalone-Szenario mit physischer Trennung im Konnektor

Beschreibung des Szenarios

Dieses Szenario stellt eine Variante des Standalone-Szenarios dar, bei dem eine physische Trennung der Konnektoren eingesetzt wurde.

Im Standalone-Szenario besteht eine Trennung zwischen den Praxissystemen der dezentralen Umgebung, welche offline (also, ohne Anbindung an die zentrale TI-Plattform) betrieben werden und den für das Update der eGK durch die Fachanwendung VSDM notwendigen Komponenten, welche online (also, mit Verbindung in die zentrale TI-Plattform) betrieben werden.

Die physische Trennung im Standalone-Szenario zeichnet sich dadurch aus, dass getrennte Komponenten zum Einsatz kommen. Der Online-Konnektor ist mit der zentralen TI-Plattform verbunden und ermöglicht das VSDM Update der eGKs. Ein am Online-Konnektor angebundener Kommunikations-PC kann darüber hinaus über den sicheren Internetzugang der TI auf das Internet zugreifen.

Sollten die Online-/Offline-Systeme nicht netztechnisch voneinander getrennt sein, so obliegt es dem Administrator der Praxissysteme sicherzustellen, dass die netztechnische Verbindung keine Gefährdung für die Praxissysteme zur Folge hat.

Im Offline-Konnektor sind einzelne Funktionen nicht verfügbar, andere haben einen eingeschränkten Funktionsumfang. So kann z.B. eine QES erzeugt oder geprüft aber dabei keine aktuelle Statusauskunft (OCSP-Response) für die eingesetzten Zertifikate eingeholt werden. Dies hat zur Folge, dass bei Erzeugung einer QES keine Statusauskunft für das Signaturzertifikat in die Signatur eingebettet werden kann und bei einer Prüfung der QES

nur eine eventuell in die Signatur eingebettet Statusauskunft des Zertifikats berücksichtigt werden kann.

Der Leistungserbringer muss in diesem Fall selber entscheiden ob der gebotene Funktionsumfang für seinen Anwendungsfall ausreichend ist.

Voraussetzungen

Folgende zusätzliche Punkte müssen erfüllt sein, um dieses Szenario umzusetzen:

- Konfiguration im Konnektor: Es muss konfiguriert werden, welche Komponenten von welchem Konnektor (online/offline) verwendet werden dürfen.
- Ein eHealth-Kartenterminal oder ein Arbeitsplatz darf immer nur mit einem der Konnektoren verbunden sein.
- Konfiguration im Konnektor: Im Offline-Konnektor wird kein VPN-Kanal konfiguriert.
- Clients bzw. Kommunikations-PC müssen sicherstellen, dass sie nur den jeweils richtigen Konnektor ansprechen.
- Es sollte eine netztechnische Trennung des Online- und Offline-Segmentes erfolgen. Wird dies nicht umgesetzt, dann obliegt es dem Administrator der Praxissysteme sicherzustellen, dass die netztechnische Verbindung keine Gefährdung für die Praxissysteme zur Folge hat.
  Sollte keine netztechnische Trennung erfolgen, so kann nur einer der Konnektoren als DHCP-Server agieren. Es wird empfohlen hier den Offline-Konnektor zu verwenden, da dort tendenziell mehr Systeme angeschlossen sind. Die am Online-Konnektor angeschlossenen Systeme müssen dann direkt konfiguriert werden.

Auswirkung

- Erhöhter Aufwand durch separate Konnektoren und separate eHealth-Kartenterminals.
- Trennung der Praxissysteme von der zentralen TI-Plattform ist für den Leistungserbringer nachweislich sichergestellt.
- Eingeschränkte Funktionalität der TI für Praxissysteme (nur Offline-Funktionalität)
- Notwendige Prüfung des Leistungserbringers, ob eingeschränkte Funktionalität (insbesondere bei Sicherheitsfunktionen) akzeptabel ist.
- Sicherer Internetzugangs der TI nur über den Kommunikations-PC nutzbar.