Änderung in gemSpec_Kon

TIP1-A_4649-03 - TUC_KON_152 „Signaturvoraussetzungen für QES prüfen"
Der Konnektor MUSS den technischen Use Case TUC_KON_152 „Signaturvoraussetzungen für QES prüfen" umsetzen.

Tabelle 1: 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

Standardablauf	1. Abhängig von seinem Typ werden für jedes Dokument die typabhängigen Dokumentvalidierungsschritte durchgeführt (Aufruf TUC_KON_080 „Dokument validieren"). Wird der Aufruf von TUC_KON_080 mit einem Fehler beendet, wird die Prüfung im laufenden TUC mit diesem Fehler abgebrochen.
	2. Durch Aufruf von TUC_KON_216 „Lese Zertifikat" wird das Signaturzertifikat von der Signaturkarte gelesen.
	3. Das Signaturzertifikat wird durch Aufruf von TUC_KON_037 „Zertifikat prüfen"{
	certificate = Zertifikatsreferenz;
	qualifiedCheck = required;
	offlineAllowNoCheck = true;
	validationMode = NONE;
	getOCSPResponses = includeRevocationInfo}
	geprüft. Das Signaturzertifikat muss zum

	Signaturzeitpunkt zeitlich gültig sein. Andere Zertifikatsprüfergebnisse können aus dem Cache genommen werden. a. ~~Prüfung der cached OCSP-Antwort des Signaturzertifikats~~
Varianten/Alternativen	(->2,3) Es dürfen alternativ die vollständigen cached Zertifikatsprüfungsdaten ohne Aufruf von TUC_KON_216 und TUC_KON_037 verwendet werden.
Fehlerfälle	(->3) Für MGM_LU_ONLINE=Enabled gilt: Liefert die Zertifikatsprüfung (OCSP-Abfrage) die Warnung CERT_REVOKED oder CERT_UNKNOWN gemäß [gemSpec_PKI#Tab_PKI_274], dann wird der TUC mit Fehler 4123 abgebrochen.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 2: TAB_KON_588 Fehlercodes TUC_KON_152 „Signaturvoraussetzungen für QES prüfen“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können keine weiteren Fehlercodes auftreten.			

[<=]

TIP1-A_4647-04 - TUC_KON 165 „Signaturvoraussetzungen für nonQES prüfen“
Der Konnektor MUSS den technischen Use Case „Signaturvoraussetzungen für nonQES prüfen” umsetzen.

Tabelle 3: 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. 3. Das Signaturzertifikat wird durch Aufruf von TUC_KON_037 „Zertifikat prüfen"{ certificate = Zertifikatsreferenz; qualifiedCheck = not_required; offlineAllowNoCheck = true; validationMode = NONE} geprüft. Das Signaturzertifikat muss zum Signaturzeitpunkt zeitlich gültig sein. Andere Zertifikatsprüfergebnisse können aus dem Cache genommen werden. a. ~~Prüfung der cached OCSP-Antwort des Signaturzertifikats~~
Varianten/Alternativen	(->2,3) Es dürfen alternativ die vollständigen cached Zertifikatsprüfungsdaten ohne Aufruf von TUC_KON_216 und TUC_KON_037 verwendet werden.
Fehlerfälle	Keine
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 4: TAB_KON_587 Fehlercodes TUC_KON_165 „Signaturvoraussetzungen für nonQES prüfen"

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können keine weiteren Fehlercodes auftreten.			

[<=]

A_23536-01 - TUC_KON_159 - "Signaturdatenelemente nachbereiten"
Der Konnektor MUSS den technischen Use Case TUC_KON_159 "Signaturdatenelemente nachbereiten" umsetzen.

Tabelle 5 : TAB_KON_892 – TUC_KON_159 „Signaturdatenelemente nachbereiten"

Element	Beschreibung
Name	TUC_KON_159 „Signaturdatenelemente nachbereiten“
Beschreibung	Es wird für das verwendete Signaturzertifikat die Statusauskunft eingeholt, überprüft und falls gefordert, in die vorab erstellte Signatur eingebettet.
Auslöser	TUC_KON_150 „Dokumente QES signieren“, TUC_KON_170 Dokumente mit Komfort signieren", TUC_KON_160 „Dokumente nonQES signieren“
Vorbedingungen	keine
Eingangsdaten	• signatureMode (Signaturart: QES | nonQES) • Signierte Dokumente / signiertes Dokument • signatureType (URI für den Signaturtyp XML-, CMS-, S/MIME- oder PDF-Signatur) • Zertifikatsreferenz (zu verwendende Signatur-Identität) • includeRevocationInfo [Boolean] - optional; Default: true (Dieser Parameter steuert die Einbettung von OCSP-Responses in die Signatur. true: Die Sperrinformationen werden in die Signatur eingebettet.)
Komponenten	Konnektor, Kartenterminal, Signaturkarte
Ausgangsdaten	• Prüfergebnis für das Zertifikat • Signiertes Dokument/ Dokumente mit eingebetteter OCSP-Antwort optional/nur wenn includeRevocationInfo = true

Standardablauf	1. Das Signaturzertifikat wird durch Aufruf von TUC_KON_037 „Zertifikat prüfen“{
	certificate = Zertifikatsreferenz;
	qualifiedCheck = if_QC_present;
	offlineAllowNoCheck = true;
	validationMode = OCSP;
	getOCSPResponses = includeRevocationInfo}
	geprüft. Eine OCSP-Auskunft muss eingeholt werden. Andere Zertifikatsprüfergebnisse können aus dem Cache genommen werden.
	2. Falls includeRevocationInfo== true wird die OCSP-Antwort gemäß des signatureType in die Signatur für jedes Dokument eingebettet.
	
	signatureType = XMLDSig (XAdES)

	Einbettung der OCSP-Response im Sinne vom AdES-X-L; die base-64 kodierte OCSP-Response wird im Feld `QualifyingProperties/UnsignedProperties /UnsignedSignatureProperties/RevocationValues /OCSPValues/EncapsulatedOCSPValue` (selbst DER-kodiert) gespeichert. signatureType = CMS (CAdES) Ist die Einbettung von OCSP-Responses gefordert, wird die für die Offline-Prüfung notwendige OCSP-Antwort des EE-Zertifikats im Attribut `SignedData.crls.other` abgelegt. signatureType = PDF/A (PAdES) OCSP-Responses werden bei PAdES nicht eingebettet.
Varianten/Alternativen	keine
Fehlerfälle	(->1) Für MGM_LU_ONLINE=Enabled gilt: Liefert die Zertifikatsprüfung (OCSP-Abfrage) die Warnung CERT_REVOKED oder CERT_UNKNOWN gemäß [gemSpec_PKI#Tab_PKI_274], dann wird der TUC mit Fehler 4123 abgebrochen.
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 6 : TAB_KON_893 Fehlercodes TUC_KON_159 „Signaturdatenelemente nachbereiten

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4123	Security	Error	Fehler bei Signaturerstellung

[<=]

TIP1-A_4696-04 - TUC_KON_037 „Zertifikat prüfen"
Der Konnektor MUSS den technischen Use Case „Zertifikat prüfen" gemäß TUC_KON_037 „Zertifikat prüfen" umsetzen.

Tabelle 7: TAB_KON_769 TUC_KON_037 „Zertifikat prüfen“

Element	Beschreibung
Name	TUC_KON_037 „Zertifikat prüfen“
Beschreibung	Der TUC beschreibt • die Prüfung eines X.509-Zertifikats gegen den Vertrauensraum
Auslöser	• Aufruf in einem Fachmodul oder • technischen Use Case
Vorbedingungen	• aktuelle TSL-Informationen im Truststore vorhanden • für QES X.509-Prüfung: eine aktuell gültige BNetzA-VL
Eingangsdaten	• certificate (ein X.509-Zertifikat (nonQES- oder QES-X.509-Zertifikat)) • EECertificateContainedInTSL - *optional; default: false* (true: Prüfung, ob ein EE-Zertifikat in der TSL vorhanden und zeitlich gültig ist; EE-Zertifikat wird in der TSL innerhalb eines "TSPService"-Eintrags ServiceTypeIdentifier " http://uri.etsi.org/TrstSvc/Svctype/unspecified " erwartet. false: vollständige Prüfung eines X.509-Zertifikats mit TUC_PKI_018 bzw. TUC_PKI_030) • qualifiedCheck [not_required | required | if_QC_present] – (Art der Zertifikatsprüfung) • baseTime – *optional/verpflichtend, wenn ein Zeitpunkt zur Prüfung vorgegeben werden soll; default: Verwendung der Systemzeit des Konnektors* (Referenzzeitpunkt: Zeitpunkt, für den das Zertifikat geprüft werden soll) • offlineAllowNoCheck [Boolean] – *optional; default: false* (Angabe, ob es als Fehler (false) oder als Warnung (true) interpretiert werden soll, wenn eine OCSP-Prüfung nicht durchgeführt werden konnte.) • intendedKeyUsage – *optional/verpflichtend, wenn certificate ein nonQES-X.509-Zertifikat ist; wird bei QES nicht ausgewertet* (Vorgesehene KeyUsage) • gracePeriod – *optional~~/nur für nonQES-X.509-Zertifikat und wenn vom Standard abgewichen werden soll; wird bei QES nicht ausgewertet~~*; *default*: *CERT_OCSP_~~DEFAULT_~~GRACE_PERIOD~~_NONQES~~* (OCSP-GracePeriod: maximal zulässiger Zeitraum, den letzte OCSP-Antwort aus dem Cache bezüglich des Referenzzeitpunkts zurückliegen darf;) • validationMode [OCSP | CRL | NONE] – *optional/verpflichtend, wenn certificate ein nonQES-X.509-*

	Zertifikat ist (Prüfmodus: • OCSP: Es wird mittels OCSP geprüft. Dabei wird, falls die OCSP-GracePeriod noch nicht abgelaufen ist, die OCSP-Antwort aus dem Cache des Konnektors verwendet. ~~Für QES einzig erlaubter validationMode.~~ • CRL: Es wird gegen die aktuelle CRL auf dem Konnektor geprüft. • NONE: Keine Prüfung von Statusinformationen) • nur für nonQES-Zertifikate: • policyList (Liste der zugelassenen Zertifikatstyp-OIDs gemäß [gemSpec_OID#GS-A_4445]) • intendedExtendedKeyUsage – *optional/verpflichtend, wenn certificate ein nonQES-X.509-Zertifikat ist; wird bei QES nicht ausgewertet* *(*Vorgesehene ExtendedKeyUsage) • ~~gracePeriod – *optional/nur für nonQES-X.509-Zertifikat und wenn vom Standard abgewichen werden soll; wird bei QES nicht ausgewertet; default: CERT_OCSP_DEFAULT_GRACE_PERIOD_NONQES* (OCSP-GracePeriod: maximal zulässiger Zeitraum, den letzte OCSP-Antwort aus dem Cache bezüglich des Referenzzeitpunkts zurückliegen darf;)~~ • ~~validationMode [OCSP | CRL | NONE] – *optional/verpflichtend, wenn certificate ein nonQES-X.509-Zertifikat ist* (Prüfmodus:~~ • ~~OCSP: Es wird mittels OCSP geprüft. Dabei wird, falls die OCSP-GracePeriod noch nicht abgelaufen ist, die OCSP-Antwort aus dem Cache des Konnektors verwendet. Für QES einzig erlaubter validationMode.~~ • ~~CRL: Es wird gegen die aktuelle CRL auf dem Konnektor geprüft.~~ • ~~NONE: Keine Prüfung von Statusinformationen)~~ • ocspResponse – *optional* (OCSPResponse des EE-Zertifikats) • getOCSPResponses [Boolean]– *optional; default: false* (true – OCSPResponse des geprüften Zertifikats soll an den Aufrufer zurückgegeben werden)
Komponenten	Konnektor
Ausgangsdaten	• Status und Liste von Warnungen/Fehlern bei der Zertifikatsprüfung • role (aus dem Zertifikate ermittelte Rolle oder Berufsgruppe;

	siehe „Tab_PKI_406 OID-Festlegung technische Rolle in X.509-Zertifikaten“ oder „Tab_PKI_402 OID-Festlegung Rolle im X.509-Zertifikat für Berufsgruppen“ oder Tab_PKI_403 OID-Festlegung Institutionen im X.509-Zertifikat der SMC-B [gemSpec_OID]) • qcStatement – *optional/verpflichtend, wenn certificate ein QES-X.509-Zertifikat ist*; *nicht relevant bei EECertificateContainedInTSL=true.* (QCStatements des Zertifikats) • ocspResponsesRenewed – *optional/verpflichtend, wenn Eingabeparameter getOCSPResponses = true* oder wenn im Ablauf spezifiziert; *nicht relevant bei EECertificateContainedInTSL=true.* (OCSP-Response des geprüften Zertifikats)

Standardablauf	1. Falls EECertificateContainedInTSL=false: a. Wenn das X.509-Zertifikat von einem CA-Zertifikat ausgestellt wurde, das in CERT_IMPORTED_CA_LIST enthalten ist, erfolgt eine Zertifikatsprüfung analog zu den Festlegungen in TUC_PKI_018 „Zertifikatsprüfung". Dabei sind zu prüfen: - Zeitliche Gültigkeit, - Gültigkeit des EE-Zertifikats nach Kettenmodell (analog zu z. B. [gemKPT_PKI_TIP#2.4.3]) - 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. Wenn das zum X.509-Zertifikat gehörende CA-Zertifikat nicht in CERT_IMPORTED_CA_LIST enthalten ist, werden, abhängig vom Parameter *qualifiedCheck* folgende TUCs unter Weitergabe aller Eingangsparameter sowie der Negation des Werts von MGM_LU_ONLINE als Parameter „Offline-Modus" aufgerufen: i. Für qualifiedCheck = not_required: TUC_PKI_018 „Zertifikatsprüfung in der TI" Ist der Eingangsparameter ocspResponses mit einer OCSP-Antwort gefüllt, so wird dieser übergeben. Die aktuell aus der OCSP-Abfrage resultierte OCSP-Antwort, falls vorhanden, wird an den Aufrufer weitergegeben. ii. Für qualifiedCheck = required: TUC_PKI_030 „QES-Zertifikatsprüfung" Dabei wird das Basiszertifikat übergeben. Ist Eingangsparamter ocspResponses mit einer OCSP-Response gefüllt, so wird dieser übergeben. Die aktuell aus der OCSP-Abfrage resultierende OCSP-Response, falls vorhanden, wird an den Aufrufer weitergegeben. iii. Für qualifiedCheck = 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". Wenn der Eingangsparameter validationMode („Prüfmodus") den Wert NONE hat: iv. werden diese Eingangsparameter des ~~die~~ TUC_PKI_018 ~~Eingangsparameter~~ folgendermaßen gesetzt

- „Offline-Modus“ unabhängig von MGM_LU_ONLINE auf „ja“
- „Prüfmodus“ auf „OCSP“

v. wird dieser Eingangsparameter des TUC_PKI_030 folgendermaßen gesetzt

- „Offline-Modus“ unabhängig von MGM_LU_ONLINE auf „ja“

Als Timeout wird beim Aufruf von TUC_PKI_018 der Wert von CERT_OCSP_TIMEOUT_NONQES bzw. beim Aufruf von TUC_PKI_030 der Wert von CERT_OCSP_TIMEOUT_QES übergeben (siehe auch Eingangsdaten von diesen TUCs in [gemSpec_PKI]).

Für die QES-Zertifikatsprüfung wird das zu prüfende QES-Zertifikat an TUC_PKI_030 „QES-Zertifikatsprüfung“ übergeben.

Wird im Aufruf der Eingangsparameter getOCSPResponses = false mit übergeben, wird keine OCSP-Response an den Aufrufer zurückgegeben.

Wird von TUC_PKI_030 mit dem Result "Valid" die Warnung "PROVIDED_OCSP_RESPONSE_NOT_VALID" zurückgemeldet, so wird die von TUC_PKI_030 zurückgemeldete OCSP-Response unabhängig von getOCSPResponses an den Aufrufer zurückgemeldet.

Als TOLERATE_OCSP_FAILURE wird beim Aufruf von TUC_PKI_018 offlineAllowNoCheck verwendet.
~~Wenn der Eingangsparameter validationMode („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. Falls EECertificateContainedInTSL=true

c. Prüfe, ob das in certificate übergebene X.509-Zertifikat in der TSL innerhalb eines "TSPService"-Eintrags mit dem ServiceTypeIdentifier " http://uri.etsi.org/TrstSvc/Svctype/unspecified" aufgeführt ist.

d. Prüfe zeitliche Gültigkeit von certificate zum Prüfzeitpunkt aktuelle Systemzeit durch Aufruf von TUC_PKI_002.

e. Ermittle role von certificate durch Aufruf von TUC_PKI_009.

3. Die Parameter `CARD.CERTSTATUS` und `CARD.CERTOCSPRESPONSE` werden befüllt.
~~3~~4. Der Status der Prüfung und die ermittelten Ausgangsdaten werden zurückgegeben.

Varianten/ Alternativen	
Fehlerfälle	TUC_KON_037 im kritischen Betriebszustand EC_TSL_Out_Of_Date_Beyond_Grace_Period aufgerufen: Fehlercode 4002. -> 2a) certificate ist nicht in der TSL enthalten
Nichtfunktionale Anforderungen	Der Konnektor MUSS unter Einhaltung aller anderen 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 8: TAB_KON_601 Fehlercodes TUC_KON_037 „Zertifikat prüfen"

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases treten folgende Fehlercodes auf.			
4002	Security	Fatal	Der Konnektor befindet sich in einem kritischen Betriebszustand
4260	Security	Error	Zertifikat nicht vorhanden in TSL

[<=]

TIP1-A_4689-01 - Caching von OCSP-Antworten
Der Zertifikatsdienst MUSS die beim Signieren und Entschlüsseln erhaltene OCSP-Antworten für eine durch `CERT_OCSP_~~DEFAULT_~~GRACE_PERIOD~~_NONQES~~` angegebene ~~Anzahl an Minuten (nonQES-Zertifikate)~~ Zeit zwischenspeichern.
[<=]

TIP1-A_4690-01 - 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_4579 - TUC_KON_018 „eGK-Sperrung prüfen"
Der Konnektor MUSS den technischen Use Case „eGK-Sperrung prüfen" gemäß TUC_KON_018 umsetzen.

Tabelle 9: TAB_KON_110 - TUC_KON_018 „eGK-Sperrung prüfen“

Element	Beschreibung
Name	TUC_KON_018 „eGK-Sperrung prüfen“
Beschreibung	Es wird geprüft, dass DF.HCA (Health Care Application) der eGK nicht gesperrt ist und optional, dass das AUT-Zertifikat im DF.ESIGN gültig ist. Für eine Karte ab der Generation G2.1 wird das AUT-Zertifikat (ECC) geprüft. Für eine Karte der Generation G2.0 wird das AUT-Zertifikat (RSA) geprüft.
Auslöser	Aufruf durch Fachmodul im Konnektor
Vorbedingungen	keine
Eingangsdaten	• cardSession • checkHcaOnly [Boolean] - *optional; default = false* (Prüfung auf die Frage beschränken, ob auf DF.HCA zugegriffen werden kann)
Komponenten	Konnektor, Kartenterminal, eGK

Ausgangsdaten	• Karte gesperrt: true | false
	• Status – *optional/wenn checkHcaOnly = false*
	  • DF.HCA gesperrt: true | false
	  • 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 = false
	  b. In allen anderen Fällen war das Selektieren nicht fehlerfrei möglich: DF.HCA gesperrt = true
	4. Wenn checkHcaOnly = true
	Beende TUC, liefere Status.
	5. Ermittle Zertifikatsobjekt (fileIdentifier und folder) für C.AUT der Karte unter Berücksichtigung des kryptographischen Verfahrens crypt gemäß TAB_KON_858.
	Für eine Karte ab der Generation G2.1 setze crypt=ECC.

	Für eine Karte der Generation G2.0 setze crypt=RSA. Rufe Cert = TUC_KON_216 „LeseZertifikat“ {cardSession; fileIdentifier; folder} 6. Bestimme per Aufruf von TUC_KON_037 „Zertifikat prüfen“ (mit gracePeriode=20min) a. das Ergebnis der Offline-Prüfung des C.CH.AUT-Zertifikats (gültig | ungültig) sowie b. den Sperrstatus des C.CH.AUT-Zertifkats (gut | gesperrt | nicht ermittelbar). 7. Die Karte ist gesperrt = true, wenn a. DF.HCA gesperrt = true oder b. Ergebnis der Offline-Prüfung des C.CH.AUT-Zertifikats = ungültig oder c. Sperrstatus des C.CH.AUT-Zertifkats = gesperrt. In allen anderen Fällen ist die Karte gesperrt = false.
Varianten/ Alternativen	keine
Fehlerfälle	(→2) Karte ist fremd reserviert, Fehlercode 4093
Nichtfunktionale Anforderungen	keine
Zugehörige Diagramme	keine

Tabelle 10: TAB_KON_239 Fehlercodes TUC_KON_018 „eGK-Sperrung prüfen“

Fehlercode	ErrorType	Severity	Fehlertext
Neben den Fehlercodes der aufgerufenen technischen Use Cases können folgende weitere Fehlercodes auftreten:			
4093	Technical	Error	Karte wird in einer anderen Kartensitzung exklusiv verwendet

[<=]

TIP1-A_4702-05 - 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 11: TAB_KON_606 Konfiguration des Zertifikatsdienstes

ReferenzID	Belegung	Bedeutung

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 Stunden	~~Default~~ Grace Period OCSP ~~für nonQES~~ in Stunden. Der Wert MUSS zwischen 0 und 24 Stunden liegen. Default-Wert = 24 Stunden
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 = 10 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 = 10 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-	Der Administrator MUSS CA-Zertifikate importieren, anzeigen und löschen können. Der Konnektor DARF CA-Zertifikate zur Ableitung von QES-

	Zertifikaten	Zertifikaten NICHT importieren. Default-Wert = leere Liste
CERT_BNETZA_VL_UPDATE_INTERVAL	X Stunden	Intervall, in dem die BNetzA VL auf Aktualität geprüft werden muss. Der Wert MUSS zwischen 1 Stunde und 168 Stunden (7 Tage) liegen. Default-Wert = 24 Stunden
CERT_TSL_DOWNLOAD_ADDRESS_INTERNET_BU	1 URI	Konfigurierbare Backup Adresse der TSL im Internet
CERT_TSL_IP_ADDRESS_INTERNET_BU	1 URI	Konfigurierbare Backup Adresse der TSL im Internet (enthält IP-Adresse des Hosts statt FQDN). Wird verwendet, falls Auflösen der FQDN mittels DNS bei CERT_TSL_DOWNLOAD_ADDRESS_INTERNET_BU fehlschlägt.

Tabelle 12: TAB_KON_733 Einsehbare Konfigurationsparameter des Zertifikatsdienstes

ReferenzID	Belegung	Bedeutung
CERT_CRL_DOWNLOAD_ADDRESS	2 URIs	Download-Adressen für die CRL
CERT_OCSP_FORWARDER_ADDRESS	2 FQDNs	Adressen der OCSP-Forwarder (HTTPS-Proxy) beim Zugangsdienstprovider Der Administrator muss in geeigneter Weise einen Test auslösen können, ob einer der Server per ICMP-Echo (ping) erreichbar ist und ob ein (beliebiger) OCSP-Request zu einer erhaltenen OCSP-Antwort führt.
CERT_OCSP_FORWARDER_PORT	TCP-Port	TCP-Port des OCSP-Forwarders (HTTPS-Proxy) beim Zugangsdienstprovider
CERT_TSL_DOWNLOAD_ADDRESS_INTERNET	1 URI	Adresse der TSL im Internet gemäß gemSpec_TSL
CERT_TSL_IP_ADDRESS_INTERNET	1 URI	Adresse der TSL im Internet gemäß gemSpec_TSL (enthält IP-Adresse des Hosts statt FQDN). Wird verwendet, falls Auflösen der FQDN mittels DNS bei

		CERT_TSL_DOWNLOAD_ADDRESS_INTERNET fehlschlägt.

[<=]

TIP1-A_4676-10 - Basisdienst Signaturdienst (nonQES und QES)
Der Konnektor MUSS Clientsystemen den Basisdienst Signaturdienst (nonQES und QES) anbieten.

Tabelle 13: TAB_KON_197 Basisdienst Signaturdienst (nonQES und QES)

Name	SignatureService	
Version (KDV)	~~7.4.0 (WSDL-Version), 7.4.2 (XSD-Version)~~ ~~7.4.2 (WSDL-Version), 7.4.4 (XSD-Version)~~ ~~7.5.5 (WSDL- und XSD-Version)~~ 7.4.3 (WSDL-Version), 7.4.5 (XSD-Version) 7.5.6 (WSDL- und XSD-Version)	
Namensraum	Siehe GitHub	
Namensraum-Kürzel	SIG für Schema und SIGW für WSDL	
Operationen	**Name**	**Kurzbeschreibung**
	SignDocument	Dokument signieren
	VerifyDocument	Signatur verifizieren
	StopSignature	Signieren eines Dokumentenstapels abbrechen
	GetJobNumber	Liefert eine Jobnummer für den nächsten Signiervorgang
	ActivateComfortSignature	Aktiviert die Komfortsignatur für einen HBA
	DeactivateComfortSignature	Deaktiviert die Komfortsignatur für einen oder mehrere HBA
	GetSignatureMode	Liefert den Status der Komfortsignaturfunktion und Informationen zur Komfortsignatursession eines HBA
WSDL	SignatureService_V7_5_6.wsdl SignatureService_V7_4_3.wsdl ~~SignatureService_V7_5_5.wsdl~~ ~~SignatureService_V7_4_2.wsdl~~ ~~SignatureService.wsdl (WSDL-Version 7.4.0)~~	
Schema	SignatureService_V7_5_6.xsd SignatureService_V7_4_5.xsd	

	~~SignatureService_V7_5_5.xsd~~ ~~SignatureService_V7_4_4.xsd~~ ~~SignatureService.xsd (XSD-Version 7.4.2)~~

[<=]

Tabelle 14: TAB_KON_531 Parameterübersicht des Kartendienstes

(...)

CARD. CERTSTATUS	Valid Invalid Inconclusive NotAvailable	Prüfungsergebnis aus TUC_KON_037 für • C.HCI.OSIG von SMC-B • C.HP.QES von HBAx Default ist NotAvailable
CARD. CERTOCSPRESPONSE	Good Revoked Unknown NotAvailable	OCSP-Response (TUC_KON_037) für • C.HCI.OSIG von SMC-B • C.HP.QES von HBAx Default ist NotAvailable

(...)

Tabelle 15: TAB_KON_777 Events Interne Mechanismen

(...)

CERT /CARD /STATUS	Op	Warning	-	x	CARD_TYPE=$Type; ICCSN=$ICCSN; CARD_HANDLE=$CardHandle;CardHolderName=$CardHolderName; ZertName=<Name des Zertifikatsobjekts>; ExpirationDate=$validity" CARD_CERTSTATUS= $CARD.CERTSTATUS~~|NotAvailable~~		

(...)

Änderungen an den Schema-Files

Signaturdienst Version	Änderung zu Vorversion(en)	Zu unterstützen	eIDAS-konform	abkündigen	Bemerkung

7.5.6 SignatureService_V7_5_6.wsdl SignatureService_V7_5_6.xsd	Einbetten von OCSP-Antworten bei nonQES Bug Fix eIDAS-Konformität	x	x	-	
7.5.5	Komfortsignatur, Bug Fixes, Härtung	-	-	x	
7.4.3 SignatureService_V7_4_3.wsdl SignatureService_V7_4_5.xsd	Bug Fix eIDAS-Konformität	x	x	-	Benötigt für NFDM
7.4.2	ECC-Migration	-	-	x	
7.4.0	Signaturproxy statt xTV	-	-	x	