Gesellschaft für Telematikanwendungen der Gesundheitskarte mbH

Einführung der Gesundheitskarte

Übergreifende Spezifikation

Verwendung kryptographischer Algorithmen in der Telematikinfrastruktur

Version:	2.4.0
Revision:	\main\rel_online\rel_ors1\27
Stand:	17.07.2015
Status:	öffentlich
Klassifizierung:	freigegeben
Referenzierung:	[gemSpec_Krypt]

Dokumentinformationen

Änderungen zur Vorversion

Änderungen bzgl. Errata 1.4.6 sind eingearbeitet.

Dokumentenhistorie

Version	Datum	Kap./ Seite	Grund der Änderung, besondere Hinweise	Bearbeitung
1.4.0	03.07.08		freigegeben (für Rel. 2.3.4)	gematik
1.9.0	26.06.12		Kommentierung	PL P77
1.10.0	13.09.12		Einarbeitung der Gesellschafterkommentare	PL P77
2.0.0	15.10.12		bQS Kommentare eingearbeitet	gematik
2.1.0	06.06.13		Erweiterung im Rahmen der PP-Erstellung Konnektor (kryptographische Vorgaben für die SAK); Anpassung an das fortgeschriebene PP Konnektor ORS1 (BSI-CC-PP-046), Konsistenz zur veränderten gemSpec_Kon herstellen	gematik
2.2.0	21.02.14		Losübergreifende Synchronisation	gematik
2.3.0	17.06.14		Entfernung des CBC-Modus bei der Dokumentenver- und -entschlüsselung gemäß P11-Änderungsliste	gematik
	09.07.15		Einarbeitung Änderungen aus Errata 1.4.6	gematik
2.4.0	17.07.15		freigegeben	gematik

© gematik – freigegeben

Inhaltsverzeichnis

Dokumentinformationen 2

Inhaltsverzeichnis 3

1 Einführung 5

1.1 Zielsetzung und Einordnung des Dokuments 5

1.2 Zielgruppe 5

1.3 Geltungsbereich 6

1.4 Abgrenzung des Dokuments 6

1.5 Methodik 6

1.5.1 Hinweis auf offene Punkte 6

2 Einsatzszenarioübergreifende Algorithmen 7

2.1 Identitäten 7

2.1.1 X.509-Identitäten 7

2.1.1.1 X.509-Identitäten für digitale nicht-qualifizierte elektronische Signaturen 8

2.1.1.2 X.509-Identitäten für qualifizierte elektronische Signaturen *9*

2.1.1.3 X.509-Identitäten für die TLS-Authentifizierung *10*

2.1.1.4 X.509-Identitäten für die IPsec-Authentifizierung *10*

2.1.1.5 X.509-Identitäten für digitale Signaturen durch TI-Komponenten *10*

2.1.1.6 X.509-Verschlüsselungszertifikate *10*

2.1.2 CV-Identitäten 10

2.1.2.1 CV-Zertifikate G1 ..... *10*

2.1.2.2 CV-Certification-Authority (CV-CA) Zertifikat G1 ..... *11*

2.1.2.3 CV-Zertifikate G2 ..... *11*

2.1.2.4 CV-Certification-Authority (CV-CA) Zertifikat G2 ..... *12*

2.2 Zufallszahlengeneratoren 12

2.3 Hilfestellung bei der Umsetzung (Zufallsgeneratoren) 13

2.4 Schlüsselerzeugung 14

2.5 Padding 14

2.5.1 Zufalls-Padding für Blockchiffren bei XML-Verschlüsselung 14

3 Konkretisierung der Algorithmen für spezifische Einsatzszenarien 15

3.1 Kryptographische Algorithmen für XML-Dokumente 15

3.1.1 XML-Signaturen für nicht-qualifizierte Signaturen 16

3.1.2 XML-Signaturen für qualifizierte elektronische Signaturen 17

3.1.3 Webservice Security Standard (WSS) 17

3.1.4 XML-Verschlüsselung – Symmetrisch 18

3.1.5 XML-Verschlüsselung – Hybrid 18

3.2 Karten-verifizierbare Authentifizierung und Verschlüsselung 18
3.2.1 Card-to-Card-Authentisierung G1 18
3.2.2 Card-to-Server (C2S) Authentisierung und Trusted Channel G1 19
3.2.3 Card-to-Card-Authentisierung G2 19
3.2.4 Card-to-Server (C2S) Authentisierung und Trusted Channel G2 19
3.2.5 Hinweis für die C2S-Authentisierung 20

3.3 Netzwerkprotokolle 20
3.3.1 IPsec-Kontext 21
3.3.2 TLS-Verbindungen 22
3.3.3 DNSSEC-Kontext 25

3.4 Masterkey-Verfahren (informativ) 25

3.5 Hybride Verschlüsselung binärer Daten 27
3.5.1 Symmetrischer Anteil der hybriden Verschlüsselung binärer Daten 27
3.5.2 Asymmetrischer Anteil der hybriden Verschlüsselung binärer Daten 27

3.6 Symmetrische Verschlüsselung binärer Daten 28

3.7 Signatur binärer Inhaltsdaten (Dokumente) 28

3.8 Signaturen innerhalb von PDF/A-Dokumenten 29

3.9 MAC im Rahmen der Personalisierung der eGK 30

3.10 Algorithmus im Rahmen der Bildung der pseudonymisierten Versichertenidentität 30

3.11 Spezielle Anwendungen von Hashfunktionen 31

3.12 kryptographische Vorgaben für die SAK des Konnektors 31

3.13 Migration kryptographischer Primitive für die Signatur im PKI-Bereich 32

3.14 Spezielle Anwendungen von kryptographischen Signaturen 32

4 Umsetzungsprobleme mit der TR-03116-1 33

4.1 XMLDSig und PKCS1-v2.1 33

4.2 XMLEnc: Die Nutzung von RSAES-OAEP und AES-GCM 34

4.3 XML Signature Wrapping und XML Encryption Wrapping 34

4.4 Güte von Zufallszahlen 34

Anhang A - Verzeichnisse 36

A1 – Abkürzungen 36

A2 – Glossar 36

A3 – Abbildungsverzeichnis 36

A4 – Tabellenverzeichnis 37

A5 - Referenzierte Dokumente 37
A5.1 – Dokumente der gematik 37
A5.2 – Weitere Dokumente 38

1 Einführung

1.1 Zielsetzung und Einordnung des Dokuments

Die vorliegende übergreifende Spezifikation definiert Anforderungen an Produkte der TI bezüglich kryptographischer Verfahren. Diese Anforderungen sind als übergreifende Regelungen relevant für Interoperabilität und Verfahrenssicherheit.

Für die TI ist die Technische Richtlinie 03116 Teil 1 [BSI-TR-03116-1] normativ, d. h. nur dort aufgeführte kryptographische Verfahren dürfen von Produkten in der TI verwendet werden. Wenn mehrere unterschiedliche Produkttypen der TI zusammenarbeiten ist es bez. der Interoperabilität nicht sinnvoll wenn jeder beteiligter Produkttyp alle dort aufgeführten Verfahren umsetzen muss, da er vermuten muss die Gegenstelle beherrscht nur eine Teilmenge der dort aufgeführten Verfahren. Um einen gemeinsamen Nenner zu definieren, legt dieses Dokument für bestimmte Einsatzzwecke ein Mindestmaß an verpflichtend zu implementierenden Verfahren aus [BSI-TR-03116-1] fest, oftmals mit spezifischen Parametern. Ein Produkttyp ist frei, weitere Verfahren aus der [BSI-TR-03116-1] optional zu implementieren, kann sich jedoch nicht ohne Weiteres darauf verlassen, dass sein potentieller Kommunikationspartner diese auch beherrscht.

Dieses Dokument folgt den Konventionen der TR. Diese hat einen Betrachtungszeitraum von sechs bzw. sieben Jahren. Analog zu Kapitel 1 [BSI-TR-03116-1] bedeutet eine Aussage „Algorithmus X ist geeignet bis Ende 2020+“ generell nicht, dass Algorithmus X nach Ende 2020 nicht mehr geeignet ist, sondern lediglich dass über die Eignung nach Ende 2020 in der TR keine explizite Aussage gemacht wird und dass aus heutiger Sicht die weitere Eignung nicht ausgeschlossen ist. Aussagen über den Betrachtungszeitraum hinaus sind „mit einem höheren Maß an Spekulation verbunden“.

Bei neuen Erkenntnissen über die verwendeten kryptographischen Algorithmen, die zu einer Änderung der TR-03116-1 führen, wird eine Anpassung dieses Dokumentes erfolgen. Für Verwendungszwecke, bei denen bereits eine Migration zu stärkeren Algorithmen in Planung ist oder die Verwendung von Algorithmen unterschiedlicher Stärke zulässig ist, wird ein Ausblick gegeben, bis wann welche Algorithmen ausgetauscht sein müssen. Bei den Migrationsstrategien für kryptographische Algorithmen ist darauf zu achten, dass hinterlegte Objekte umzuschlüsseln sind bzw. die älteren Algorithmen (unter der Bedingung, dass sie sicherheitstechnisch noch geeignet sind) für eine gewisse Übergangsphase weiter unterstützt werden müssen und danach zuverlässig in den Komponenten deaktiviert werden müssen.

1.2 Zielgruppe

Das Dokument richtet sich an Hersteller und Anbieter von Produkten der TI, die kryptographische Objekte verwalten.

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 Dokumenten (z. B. Dokumentenlandkarte, Produkttypsteckbrief, Leistungsbeschreibung) festgelegt und bekannt gegeben.

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

Aufgabe des Dokumentes ist es nicht, eine Sicherheitsbewertung von kryptographischen Algorithmen vorzunehmen. Dieser Gesichtspunkt wird in [BSI-TR-03116-1] behandelt. Es werden lediglich die dort vorgegebenen Algorithmen weiter eingeschränkt, um die Herstellung der Interoperabilität zu unterstützen.

Es ist nicht Ziel dieses Dokumentes, den Prozess zum Austauschen von Algorithmen zu definieren, sondern lediglich den zeitlichen Rahmen für die Verwendbarkeit von Algorithmen festzulegen und somit auf den Bedarf für die Migration hinzuweisen.

1.5 Methodik

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

Sie werden im Dokument wie folgt dargestellt:

⌦ **GS-A_0000 <Titel der Afo>**

Text / Beschreibung⌫

Dabei umfasst die Anforderung sämtliche innerhalb der Textmarken angeführten Inhalte.

1.5.1 Hinweis auf offene Punkte

Beschreibung des offenen Punktes.

2 Einsatzszenarioübergreifende Algorithmen

Nachfolgend werden grundlegende Festlegungen zur Verwendung von Algorithmen innerhalb der Telematikinfrastruktur getroffen. Diese Anforderungen sind unabhängig von den im nachfolgenden Kapitel definierten Einsatzszenarien und werden durch diese verwendet.

⌦ **GS-A_3080 asymmetrischen Schlüssel maximale Gültigkeitsdauer**

Die Lebensdauer von asymmetrischen Schlüsseln und somit die in einem Zertifikat angegebene Gültigkeitsdauer SOLL maximal 5 Jahre betragen.⌫

2.1 Identitäten

Der Begriff „kryptographische Identität“ (nachfolgend nur noch als Identität bezeichnet) bezeichnet einen Verbund aus Identitätsdaten und einem kryptographischen Objekt, das bspw. im Rahmen einer Authentisierung und Authentifizierung verwendet werden kann. Im Allgemeinen handelt es sich um Schlüsselpaare, bestehend aus öffentlichem und privatem Schlüssel, sowie einem Zertifikat, das die Kombination aus Attributen und öffentlichem Schlüssel durch eine übergeordnete Instanz (CA – Certification Authority) bestätigt.

Bei den Algorithmenvorgaben für Identitäten muss u. a. spezifiziert werden:

- für welche Algorithmen und für welchen Verwendungszweck die Schlüssel verwendet werden (bestimmte Verwendungszwecke schließen einander aus)[1],
- welche Algorithmen für die Signatur des Zertifikates verwendet werden,
- mit welchen Algorithmen die OCSP-Responses signiert werden und
- wie die Zertifikate des OCSP-Responders signiert sind.

2.1.1 X.509-Identitäten

Eine X.509-Identität ist eine Identität gemäß Abschnitt 2.1, bei der ein X.509-Zertifikat [RFC-5280] verwendet wird.

Bei der Aufteilung von X.509-Identitäten wurden die Identitäten zunächst nach Gruppen für verschiedene Einsatzzwecke des Schlüssels unterteilt und diese bei Bedarf um einen notwendigen Einsatzkontext erweitert. Aus dieser Aufteilung ergibt sich die nachfolgend tabellarisch dargestellte Übersicht der Arten von X.509-Identitäten. Der exemplarische

[1] Bspw. dürfen nicht Signaturschlüssel für die Sicherung von Authentizität und Integrität von Dokumenten als Signaturschlüssel für beliebige Challenges im Rahmen einer Authentisierung verwendet werden.

Einsatzort der Identitäten ist hierbei rein informativ, die Ausprägung wird in den Spezifikationen festgelegt, die eine kryptographische Identität benötigen.

Tabelle 1: Tab_KRYPT_001 Übersicht über Arten von X.509-Identitäten

Referenz	Gruppe	Kontext	Exemplarische Identitäten zur Verwendung (nicht vollständig)
2.1.1.1	Identitäten für die Erstellung von Signaturen	Identitäten für die Erstellung nicht-qualifizierter digitaler Signaturen	OSIG-Identität der SMC-B bzw. HSM-B
2.1.1.2		Identitäten für die Erstellung qualifizierter Signaturen	QES-Identität des HBA
2.1.1.5		Signaturidentitäten, die in den Diensten der TI-Plattform und den Fachdiensten zum Einsatz kommen.	Fachdienstsignatur Signatur durch zentrale Komponente der TI-Plattform Code-Signatur
2.1.1.3	Identitäten für die Client-Server-Authentifizierung	Identitäten für den Aufbau von TLS-Verbindungen	Fachdienst TLS – Server Fachdienst TLS – Client zentrale TI-Plattform TLS – Server zentrale TI-Plattform TLS – Client AUT-Identität der SMC-B AUT-Identität des Kartenterminals AUT-Identität des Anwendungs-konnektors AUT-Identität der SAK AUT-Identität der eGK AUTN-Identität der eGK AUT-Identität des HBA
2.1.1.4		Identitäten für den Aufbau von IPsec-Verbindungen	ID.NK.VPN ID.VPNK.VPN
2.1.1.6	Verschlüsselungs-zertifikate	Identitäten, für die medizinische Daten verschlüsselt werden	ENC-Identität der eGK ENCV-Identität der eGK ENC-Identität des HBA ENC-Identität der SMC-B

Für den Aufbau der X.509-Zertifikate gelten die Vorgaben aus den jeweiligen Spezifikationen der X.509-Zertifikate.

2.1.1.1 X.509-Identitäten für digitale nicht-qualifizierte elektronische Signaturen

⌦ **GS-A_4357 X.509-Identitäten für die Erstellung und Prüfung digitaler nicht-qualifizierter elektronischer Signaturen**

Alle Produkttypen, die X.509-Identitäten bei der Erstellung oder Prüfung digitaler nicht-qualifizierter elektronischer Signaturen verwenden, MÜSSEN die in Tab_KRYPT_002 aufgeführten Algorithmen unterstützen und die Tabellenvorgaben erfüllen.⌫

Tabelle 2: Tab_KRYPT_002 Algorithmen für X.509-Identitäten zur Erstellung nicht-qualifizierter Signaturen

Algorithmen Typ	Algorithmen	Schlüssellänge
Verwendung der Schlüssel	RSA (OID 1.2.840.113549.1.1.1)	2048 Bit bis Ende 2020+
Signatur des Endnutzer- und CA-Zertifikates	sha256withRSAEncryption (OID 1.2.840.113549.1.1.11)	2048 Bit bis Ende 2020+
Signatur der OCSP-Response	sha256withRSAEncryption (OID 1.2.840.113549.1.1.11)	2048 Bit bis Ende 2020+
Signatur des OCSP-Responder-Zertifikates	sha256withRSAEncryption (OID 1.2.840.113549.1.1.11)	2048 Bit bis Ende 2020+

Für die maximale Gültigkeitsdauer der Zertifikate gilt die Anforderung [GS-A_3080].

2.1.1.2 X.509-Identitäten für qualifizierte elektronische Signaturen

⌦ **GS-A_4358 X.509-Identitäten für die Erstellung und Prüfung qualifizierter elektronischer Signaturen**

Alle Produkttypen, die X.509-Identitäten für die Erstellung oder Prüfung von qualifizierten elektronischen Signaturen verwenden, MÜSSEN mindestens alle in Tabelle Tab_KRYPT_003 aufgeführten Algorithmen unterstützen und die Tabellenvorgaben erfüllen.⌫

Tabelle 3: Tab_KRYPT_003 Algorithmen für X.509-Identitäten zur Erstellung qualifizierter elektronischer Signaturen

Algorithmen Typ	Algorithmen	Schlüssellänge
Verwendung der Schlüssel	RSA (OID 1.2.840.113549.1.1.1)	2048 Bit bis Ende 2020+
Signatur des Endnutzer- und CA-Zertifikates	sha256withRSAEncryption (OID 1.2.840.113549.1.1.11)	2048 Bit bis Ende 2020+
Signatur der OCSP-Response	sha256withRSAEncryption (OID 1.2.840.113549.1.1.11)	2048 Bit bis Ende 2020+
Signatur des OCSP-Responder-Zertifikates	sha256withRSAEncryption (OID 1.2.840.113549.1.1.11)	2048 Bit bis Ende 2020+

Die Festlegung über die maximale Gültigkeitsdauer eines qualifizierten Zertifikats macht [SigV] in §14 Absatz (3).

(informativer Hinweis: Obere Schranke ist der Vorhersagezeitraum des Algorithmenkatalogs [ALGCAT].)

2.1.1.3 X.509-Identitäten für die TLS-Authentifizierung

⌦ **GS-A_4359 X.509-Identitäten für die Durchführung einer TLS-Authentifizierung**

Alle Produkttypen, die X.509-Identitäten für eine TLS-Authentifizierung verwenden, MÜSSEN alle in Tab_KRYPT_002 aufgeführten Algorithmen unterstützen und die Tabellenanforderungen erfüllen.⌫

2.1.1.4 X.509-Identitäten für die IPsec-Authentifizierung

⌦ **GS-A_4360 X.509-Identitäten für die Durchführung der IPsec-Authentifizierung**

Alle Produkttypen, die X.509-Identitäten für eine IPsec-Authentifizierung verwenden, MÜSSEN alle in Tab_KRYPT_002 aufgeführten Algorithmen unterstützen und die Tabellenanforderungen erfüllen.⌫

2.1.1.5 X.509-Identitäten für digitale Signaturen durch TI-Komponenten

⌦ **GS-A_4361 X.509-Identitäten für die Erstellung und Prüfung digitaler Signaturen**

Alle Produkttypen, die X.509-Identitäten verwenden, die zur Erstellung und Prüfung digitaler Signaturen in Bezug auf TI-Komponenten (technische X.509-Zertifikate) genutzt werden, MÜSSEN alle in Tab_KRYPT_002 aufgeführten Algorithmen unterstützen und die Tabellenanforderungen erfüllen. ⌫

2.1.1.6 X.509-Verschlüsselungszertifikate

⌦ **GS-A_4362 X.509-Identitäten für Verschlüsselungszertifikate**

Alle Produkttypen, die X.509-Identitäten für die Verschlüsselung (Verschlüsselungszertifikate) verwenden, MÜSSEN alle in Tab_KRYPT_002 aufgeführten Algorithmen unterstützten und die Tabellenanforderungen erfüllen.⌫

2.1.2 CV-Identitäten

CV-Identitäten werden für die Authentifizierung zwischen Karten verwendet.

2.1.2.1 CV-Zertifikate G1

⌦ **GS-A_4363 CV-Zertifikate G1**

Alle Produkttypen, die CV-Zertifikate der Kartengeneration G1 erstellen oder prüfen, MÜSSEN die in Tab_KRYPT_004 aufgeführten Algorithmen verwenden und die Tabellenanforderungen erfüllen.⌫

Tabelle 4: Tab_KRYPT_004 Algorithmen für CV-Zertifikate

Algorithmen Typ	Algorithmus	Schlüssellänge
über das Zertifikat bestätigtes Schlüsselpaar	authS_ISO9796-2 Withrsa_sha256_mutual (OID 1.3.36.3.5.2.4)	2048 Bit bis Ende 2017
Signatur des Endnutzerzertifikats	sigS_ISO9796-2Withrsa_sha256 (OID 1.3.36.3.4.2.2.4)	2048 Bit bis Ende 2017

Für die maximale Gültigkeitsdauer der Zertifikate gilt die Anforderung [GS-A_3080].

Das verwendete Signaturverfahren ISO-9796-2 DS1 ist nach [BSI-TR-03116-1] in der TI nur noch bis Ende 2017 zulässig. Damit ist Ende 2017 eine obere Schranke für das Ende der G1-Karten.

2.1.2.2 CV-Certification-Authority (CV-CA) Zertifikat G1

⌦ **GS-A_4364 CV-CA-Zertifikate G1**

Alle Produkttypen, die CV-CA-Zertifikate der Kartengeneration G1 erstellen oder prüfen, MÜSSEN die in Tab_KRYPT_005 aufgeführten Algorithmen verwenden und die Tabellenanforderungen erfüllen.⌫

Tabelle 5: Tab_KRYPT_005 Algorithmen für CV-CA-Zertifikate

Algorithmen Typ	Algorithmus	Schlüssellänge
über das Zertifikat bestätigtes Schlüsselpaar	sigS_ISO9796-2Withrsa_sha256 (OID 1.3.36.3.4.2.2.4)	2048 Bit bis Ende 2017
Signatur des CA-Zertifikates	sigS_ISO9796-2Withrsa_sha256 (OID 1.3.36.3.4.2.2.4)	2048 Bit bis Ende 2017

Für die maximale Gültigkeitsdauer der Zertifikate gilt die Anforderung [GS-A_3080].

Das verwendete Signaturverfahren ISO-9796-2 DS1 ist nach [BSI-TR-03116-1] in der TI nur noch bis Ende 2017 zulässig. Damit ist Ende 2017 eine obere Schranke für das Ende der G1-Karten.

2.1.2.3 CV-Zertifikate G2

⌦ **GS-A_4365 CV-Zertifikate G2**

Alle Produkttypen, die CV-Zertifikate der Kartengeneration G2 erstellen oder prüfen, MÜSSEN die in Tab_KRYPT_006 aufgeführten Algorithmen verwenden und die Tabellenanforderungen erfüllen.⌫

Tabelle 6: Tab_KRYPT_006 Algorithmen für CV-Zertifikate

Algorithmen Typ	Algorithmus	Schlüssellänge
über das Zertifikat bestätigtes Schlüsselpaar	**Authentisierung ohne Sessionkey-Aushandlung** [RFC-5639#3.4, brainpoolP256r1] ecdsa-with-SHA256 {OID 1.2.840.10045.4.3.2} **Authentisierung mit Sessionkey-Aushandlung** [RFC-5639#3.4, brainpoolP256r1] authS_gemSpec-COS-G2_ecc-with-sha256 {OID 1.3.36.3.5.3.1}	256 Bit bis Ende 2020+
Signatur des Endnutzerzertifikats	[RFC-5639#3.4, brainpoolP256r1] ecdsa-with-SHA256 {OID 1.2.840.10045.4.3.2}	256 Bit bis Ende 2020+

Für die maximale Gültigkeitsdauer der Zertifikate gilt die Anforderung [GS-A_3080].

2.1.2.4 CV-Certification-Authority (CV-CA) Zertifikat G2

⌦ **GS-A_4366 CV-CA-Zertifikate G2**

Alle Produkttypen, die CV-CA-Zertifikate der Kartengeneration G2 erstellen oder prüfen, MÜSSEN dien Tab_KRYPT_007 aufgeführten Algorithmen verwenden und die Tabellenanforderungen erfüllen.⌫

Tabelle 7: Tab_KRYPT_007 Algorithmen für CV-CA-Zertifikate

Algorithmen Typ	Algorithmus	Schlüssellänge
über das Zertifikat bestätigtes Schlüsselpaar	[RFC-5639#3.4, brainpoolP256r1] ecdsa-with-SHA256 {OID 1.2.840.10045.4.3.2}	256 Bit bis Ende 2020+
Signatur des CA-Zertifikates	[RFC-5639#3.4, brainpoolP256r1] ecdsa-with-SHA256 {OID 1.2.840.10045.4.3.2}	256 Bit bis Ende 2020+

Für die maximale Gültigkeitsdauer der Zertifikate gilt die Anforderung [GS-A_3080].

2.2 Zufallszahlengeneratoren

⌦ **GS-A_4367 Zufallszahlengenerator**

Alle Produkttypen, die Zufallszahlen generieren, MÜSSEN die Anforderungen aus [BSI-TR-03116-1#3.4 Erzeugung von Zufallszahlen] erfüllen.⌫

2.3 Hilfestellung bei der Umsetzung (Zufallsgeneratoren)[2]

Die Sicherheit eines deterministischen Zufallszahlengenerators (DRNGs) hängt maßgeblich von drei Faktoren ab:

- von der Entropie des Seeds,
- vom algorithmischen Anteil (generelles Design) und
- dem Schutz des inneren Zustands (und der zur Ausgabe vorgesehenen Zufallszahlen).

Der Nachweis, dass der algorithmische Anteil eines DRNGs den Anforderungen einer bestimmten Funktionalitätsklasse genügt, kann schwierig und aufwändig sein. Deshalb wurde das BSI gebeten, die DRNGs in [FIPS-186-2+CN1] und [ANSI-X9.31] in Bezug auf die kryptographische Güte ihres algorithmischen Anteils zu bewerten.

Das Ergebnis ist:

A) [FIPS-186-2+CN1]: Lässt man in dem DRNG aus Appendix 3.1 (S. 16f.) in Schritt 3c bzw. in dem DRNG aus Algorithmus 1 (Change Notice 1, S. 72f.) in Schritt 3.3 den Term "mod q" weg, so werden gleich verteilt 160-Bit Zufallszahlen bzw. 320-Bit Zufallszahlen erzeugt (vgl. Abschnitt „General Purpose Random Number Generation“ (Change Notice 1, S. 74)).
Beide DRNGs sind dann

(1) algorithmisch geeignet für die Klasse K4 [AIS-20-1999] und

(2) erfüllen die algorithmischen Anforderungen aus DRG.3 [AIS-20].

Ob eine konkrete Implementierung eines dieser DRNG bspw. Teil der Klasse DRG.3 ist, bleibt im Einzelfall zu prüfen, da dazu u. a. auch Fragen über die Initialisierung zu beantworten sind (vgl. (DRG.3.1) [KS-2011]).

Das BSI empfiehlt bei den Zufallsgeneratoren aus [FIPS-186-2+CN1] nach Möglichkeit SHA-256 [FIPS-180-4] anstatt SHA-1 zu verwenden. Folgt man der Empfehlung, so ist der Algorithmus dementsprechend zu adaptieren.

B) [ANSI-X9.31]: Der Zufallsgenerator aus Appendix A.2.4 ist

(1) algorithmisch geeignet für die Klasse K3 [AIS-20-1999] und

(2) erfüllt die algorithmischen Anforderungen aus DRG.2 [AIS-20].

[2] Hinweis: dies ist das ehemalige „Kapitel 5.2.4 Hilfestellung bei der Umsetzung der Anforderungen“. Der Text in diesem Abschnitt entstand in enger Abstimmung mit dem BSI auf Gesellschafterwunsch.

2.4 Schlüsselerzeugung

⌦ **GS-A_4368 Schlüsselerzeugung**

Alle Produkttypen, die Schlüssel erzeugen, MÜSSEN die Anforderungen aus [BSI-TR-03116-1#3.5 Schlüsselerzeugung] erfüllen.⌫

Hinweis: im Rahmen der Sicherheitszertifizierung von Komponenten, wie bspw. des Konnektors wird dies überprüft.

⌦ **GS-A_5021 Schlüsselerzeugung bei einer Schlüsselspeicherpersonalisierung**

Ein Herausgeber von Sicherheitsmodulen für kryptographisches Schlüsselmaterial, welche in der TI genutzt werden (also bspw. eGK, SMC-B, HSM-B, SMC-KT und HBA), MUSS sicherstellen, dass auf dem Sicherheitsmodul gespeicherten Schlüssel die Anforderungen aus [BSI-TR-03116-1#3.5 Schlüsselerzeugung] erfüllen.⌫

Hinweis: Dies ist eine Anforderung an Kartenherausgeber, die so sicherstellen müssen, dass das in den Sicherheitsmodulen (also auch HSM-B) zur Verfügung stehende kryptographische Schlüsselmaterial, geeignet ist, Daten mit sehr hohem Schutzbedarf schützen zu können. (siehe auch Kapitel 4.4)

2.5 Padding

2.5.1 Zufalls-Padding für Blockchiffren bei XML-Verschlüsselung

Nach dem Umstieg vom Betriebsmodus CBC auf den GCM (symmetrische Verschlüsselung) sind die ehemaligen Vorgaben in diesem Abschnitt obsolet.

© gematik – freigegeben

3 Konkretisierung der Algorithmen für spezifische Einsatzszenarien

In den nachfolgenden Abschnitten werden die kryptographischen Algorithmen für verschiedene Einsatzszenarien spezifiziert. In diesem Zusammenhang sind ausschließlich die kryptographischen Aspekte der Einsatzszenarien relevant.

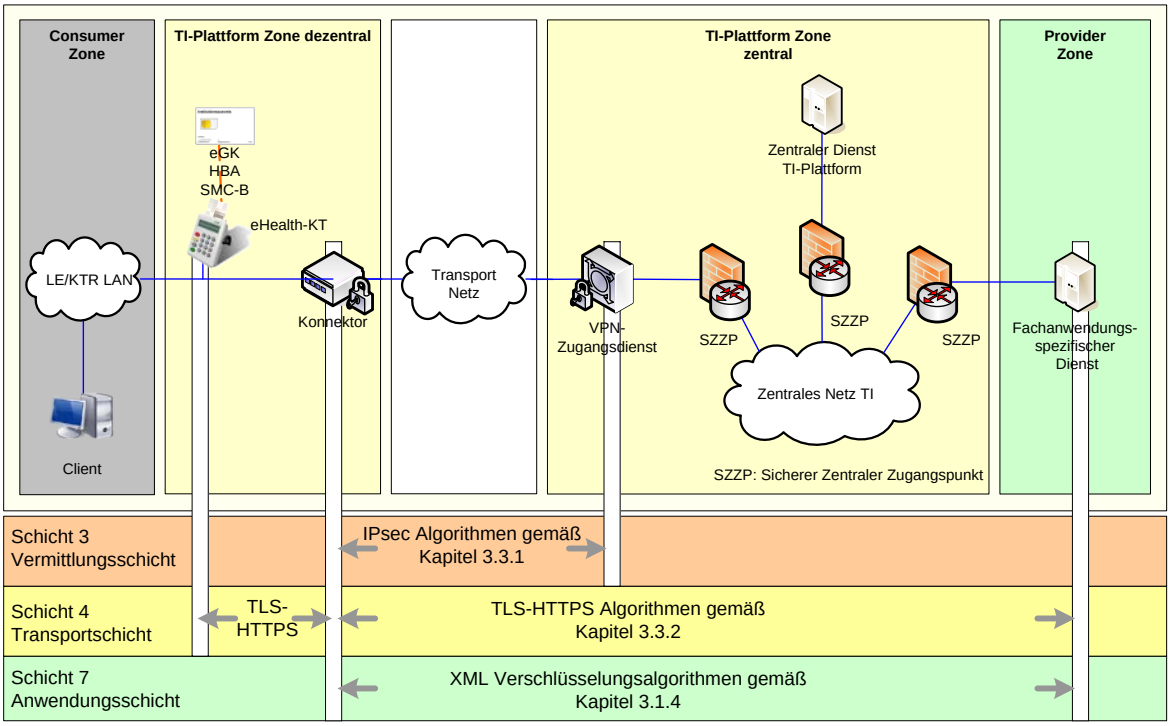

Abbildung 1: Verwendung von Algorithmen nach Zonen und OSI-Schicht

Abbildung 1 stellt beispielhaft die für die Vertraulichkeit von medizinischen Daten relevanten Algorithmen auf den verschiedenen OSI-Schichten in einer Übersicht dar. Es besteht in dieser Abbildung kein Anspruch auf Vollständigkeit.

3.1 Kryptographische Algorithmen für XML-Dokumente

⌦ GS-A_4370 Kryptographische Algorithmen für XML-Dokumente

Alle Produkttypen, die XML-Dokumente

- verschlüsseln, MÜSSEN dies mittels CMS (PKCS#7) oder XMLEnc durchführen,
- signieren, MÜSSEN dies mittels CMS (PKCS#7) oder XMLDSig durchführen. ⌫

XML-Signaturen sind bezüglich der verwendeten Algorithmen selbst beschreibend, die für die Erstellung einer Signatur verwendeten Algorithmen sind in der Signatur aufgeführt.

Zur vollständigen Spezifikation der Algorithmen für XML-Signaturen müssen für alle Signaturbestandteile Algorithmen spezifiziert werden. Die nachfolgenden Abschnitte wählen aus der Menge der zulässigen Algorithmen die jeweiligen Algorithmen für die einzelnen Einsatzszenarien aus.

Die Referenzierung von Algorithmen in XML-Signaturen und XML-Verschlüsselungen erfolgt nicht wie in Zertifikaten oder Signaturen binärer Daten über OIDs sondern über URIs. Die URIs der Algorithmen dienen als eindeutige Identifier und nicht dazu, dass unter der jeweils angegebenen URI die Beschreibung zu finden ist.

Tabelle 8: Tab_KRYPT_008 Beispiele für solche Algorithmen-URIs

Algorithmen Identifier	Erläutert in
http://www.w3.org/2001/04/xmlenc#aes256-cbc	[XMLEnc]
http://www.w3.org/2001/04/xmlenc#rsa-1_5	[XMLEnc]
http://www.w3.org/2001/04/xmlenc#sha256	[XMLDSig]
http://www.w3.org/2000/09/xmldsig#enveloped-signature	[XMLDSig]
http://www.w3.org/2001/04/xmldsig-more#rsa-sha256	[RFC-4051] bzw. [RFC-6931]
http://www.w3.org/2001/10/xml-exc-c14n#	[XMLCan_V1.0]
http://www.w3.org/2009/xmlenc11#aes256-gcm	[XMLEnc-1.1]
http://www.w3.org/2007/05/xmldsig-more#sha256-rsa-MGF1	[RFC-6931]

3.1.1 XML-Signaturen für nicht-qualifizierte Signaturen

⌦ GS-A_4371 XML-Signaturen für nicht-qualifizierte Signaturen

Alle Produkttypen, die XML-Signaturen für nicht-qualifizierte Signaturen erzeugen oder prüfen, MÜSSEN die Algorithmen und Vorgaben der Tabelle Tab_KRYPT_009 erfüllen.⌫

Tabelle 9: Tab_KRYPT_009 Algorithmen für die Erzeugung von nicht-qualifizierten elektronischen XML-Signaturen

Signaturbestandteil	Beschreibung	Algorithmus	Anmerkung
Signaturstandard	Signaturstandard	ETSI TS 101 903 V1.4.2 (2010-12) Electronic Signatures and Infrastructures (ESI); XML Advanced Electronic Signatures (XAdES) [ETSI-XAdES]	Die Verwendung des Standards ist für die Signatur von XML-Dokumenten verpflichtend, die nicht über CMS (PKCS#7) signiert werden.
kryptographisches Signaturverfahren	Algorithmus für die Berechnung des Nachrichten Digest und die Verschlüsselung mit dem privaten Schlüssel	RSASSA-PKCS1-v1_5 mit SHA256 Dieser Algorithmus ist nur noch bis Ende 2016 in der TI verwendbar, mit der Empfehlung ihn nicht mehr zu verwenden. RSASSA-PSS mit SHA256 bis nach Ende 2020+ verwendbar (Ende des Betrachtungshorizonts) (Hinweis: siehe Abschnitt 4.1)	Die Verwendung des Algorithmus ist verpflichtend. Es soll RSASSA-PSS verwendet werden. Alle hier aufgeführten Signaturverfahren müssen von einer Signaturprüfenden

Signaturbestandteil	Beschreibung	Algorithmus	Anmerkung
			Komponente überprüfbar sein.
DigestMethod	Methode zur Berechnung eines Digest der zu signierenden Bereiche	SHA-256 Die [XMLDSig] konforme Bezeichnung lautet: http://www.w3.org/2001/04/xmlenc#sha256	Die Verwendung des Algorithmus ist verpflichtend.
Kryptographisches Token	Kryptographisches Token für die Signatur, bestehend aus einem privaten Schlüssel und einem zugehörigen X.509-Zertifikat	Identitäten gemäß einem der folgenden Abschnitte 2.1.1.1	Die Auswahl des kryptographischen Tokens ist von dem jeweiligen Einsatzzweck abhängig.

3.1.2 XML-Signaturen für qualifizierte elektronische Signaturen

⌦ GS-A_4372 XML-Signaturen für qualifizierte elektronische Signaturen

Alle Produkttypen, die XML-Signaturen für qualifizierte elektronische Signaturen erzeugen oder prüfen, MÜSSEN die Vorgaben der Tabelle Tab_KRYPT_010 erfüllen.⌫

Tabelle 10: Tab_KRYPT_010 Algorithmen für qualifizierte XML-Signaturen

Signaturbestandteil	Beschreibung	Algorithmus	Anmerkung
Signaturstandard	Signaturstandard	ETSI TS 101 903 V1.4.2 (2010-12) Electronic Signatures and Infrastructures (ESI); XML Advanced Electronic Signatures (XAdES) [ETSI-XAdES]	Die Verwendung des Standards ist für die Signatur von XML-Dokumenten verpflichtend, die nicht über CMS (PKCS#7) signiert werden.
kryptographisches Signaturverfahren	Algorithmus für die Berechnung des Nachrichten-Digest und die Verschlüsselung mit dem privaten Schlüssel	RSASSA-PKCS1-v1_5 mit SHA256 Dieser Algorithmus ist nur noch bis Ende 2016 im qualifizierten Vertrauensraum (und damit auch in der TI) verwendbar, mit der Empfehlung ihn nicht mehr zu verwenden. RSASSA-PSS mit SHA256 bis nach Ende 2020+ verwendbar (Ende des Betrachtungshorizonts) (Hinweis: siehe Abschnitt 4.1)	Der Algorithmus muss für alle qualifizierten Signaturen verwendet werden. Es soll RSASSA-PSS verwendet werden. Alle hier aufgeführten Signaturverfahren müssen von einer Signaturprüfenden Komponente überprüfbar sein.
DigestMethod	Methode zur Berechnung eines Digest der zu signierenden Bereiche	SHA-256 Die [XMLDSig] konforme Bezeichnung lautet: http://www.w3.org/2001/04/xmlenc#sha256	Der Algorithmus muss für alle qualifizierten Signaturen verwendet werden.
Kryptographisches Token	Kryptographisches Token für die Signatur, bestehend aus einem privaten Schlüssel und einem zugehörigen X.509-Zertifikat	Identitäten gemäß dem folgenden Abschnitt 2.1.1.2	Es darf nur eine Identität, die den Ansprüchen qualifizierter Signaturen entspricht, verwendet werden.

3.1.3 Webservice Security Standard (WSS)

Nicht relevant für Online-Rollout (Stufe 1)

3.1.4 XML-Verschlüsselung – Symmetrisch

⌦ **GS-A_4373 XML-Verschlüsselung - symmetrisch**

Alle Produkttypen, die XML-Dokumente mittels [XMLEnc-1.1] verschlüsseln, MÜSSEN die folgenden Vorgaben umsetzen:

- Als symmetrische Block-Chiffre muss AES [FIPS-197] mit einer Schlüssellänge von 256 Bit im Galois/Counter Mode (GCM) gemäß [NIST-SP-800-38D] mit der Tag-Länge von 128 Bit verwendet werden.
- Die IVs dürfen sich bei gleichem Schlüssel nicht wiederholen (vgl. [NIST-SP-800-38D#S.25] und [BSI-TR-02102-1#S. 24]). Der IV soll eine Bitlänge von 96 Bit besitzen, seine Länge muss mindestens 96 Bit sein. Es wird empfohlen den IV zufällig zu wählen (vgl. [gemSpec_Krypt#GS-A_4367]).
- Hinweis: Im Normalfall ist davon auszugehen, dass für die Sicherung der Integrität und Authentizität der übertragenen Daten zudem noch eine Signatur der zu verschlüsselnden Daten notwendig ist.⌫

3.1.5 XML-Verschlüsselung – Hybrid

⌦ **GS-A_4374 XML-Verschlüsselung - Hybrid**

Alle Produkttypen, die XML-Dokumente mittels [XMLEnc-1.1] hybrid verschlüsseln, MÜSSEN das XML-Dokument gemäß [gemSpec_Krypt#GS-A_4373] symmetrisch verschlüsseln, wobei der eingesetzte symmetrischer Schlüssel (jeweils) für eine spezifische Person oder Komponente asymmetrisch verschlüsselt wird.
(Hinweis: Analog zum Hinweis in [gemSpec_Krypt#GS-A_4373] gilt auch hier, dass im Normalfall für die Sicherung der Integrität und Authentizität der übertragenen Daten zudem noch eine Signatur dieser Daten notwendig ist.) ⌫

⌦ **GS-A_4375 XML-Verschlüsselung - Hybrid, Schlüsseltransport**

Alle Produkttypen, die XML-Dokumente mittels [XMLEnc-1.1] hybrid verschlüsseln, MÜSSEN für die Verschlüsselung des symmetrischen Schlüssel den Algorithmus RSAES-OAEP gemäß RFC 3447 [PKCS#1] oder Algorithmus RSAES-PKCS1-v1_5 unter Berücksichtigung von speziellen Maßnahmen gegen Seitenkanalangriffe (vgl. [BSI-TR-03116-1] S. 16) verwenden.⌫

⌦ **GS-A_4376 XML-Verschlüsselung - Hybrid, Schlüsseltransport RSAES-OAEP**

Alle Produkttypen, die XML-Dokumente mittels [XMLEnc-1.1] hybrid verschlüsseln, SOLLEN für den Schlüsseltransport den Algorithmus RSAES-OAEP gemäß RFC 3447 [PKCS#1] verwenden.⌫

3.2 Karten-verifizierbare Authentifizierung und Verschlüsselung

3.2.1 Card-to-Card-Authentisierung G1

⌦ **GS-A_4377 Card-to-Card-Authentisierung G1**

Alle Produkttypen, die die Card-to-Card-Authentisierung für Karten der Generation G1 durchführen, MÜSSEN dabei eine CV-Identität gemäß [gemSpec_Krypt#GS-A_4363] verwenden.⌫

Das Verfahren zur Durchführung der Card-to-Card-Authentisierung wird in [gemSpec_eGK_ObjSys] festgelegt.

3.2.2 Card-to-Server (C2S) Authentisierung und Trusted Channel G1

⌦ **GS-A_4378 Card-to-Server (C2S) Authentisierung und Trusted Channel G1**

Alle Produkttypen, die die Card-to-Server-Authentisierung für Karten der Generation G1 durchführen, MÜSSEN die folgenden Vorgaben berücksichtigen:

- Die Authentisierung muss mit 3TDES analog [EN-14890-1#8.8] erfolgen und die Vorgaben der Tabelle Tab_KRYPT_011 berücksichtigen.
- Die Schlüsselvereinbarung muss analog zu [EN-14890-1#8.8.2] erfolgen.
- Das Verfahren zur Durchführung der Card-to-Server-Authentisierung erfolgt auf Grundlage von [EN-14890-1#8.8].⌫

Weitere Vorgaben finden sich in [gemSpec_SST_FD_VSDM].

C2S-Authentisierung bzw. der Trusted-Channel wird zwischen der eGK, dem zugeordneten CMS und dem zugeordneten VSDM-System verwendet.

Der Algorithmus 3TDES ist nach [BSI-TR-03116-1] in der TI nur noch bis Ende 2017 zulässig. Damit ist Ende 2017 eine obere Schranke für das Ende der G1-Karten.

Tabelle 11: Tab_KRYPT_011 Algorithmen für Card-to-Server-Authentifizierung

Algorithmen Typ	Algorithmus	Schlüssellänge
Authentifizierung und Verschlüsselung der Authentisierungsdaten	3TDES im CBC-Modus (OID 1.3.6.1.4.1.4929.1.8)	168 Bit zulässig bis Ende 2017

3.2.3 Card-to-Card-Authentisierung G2

⌦ **GS-A_4379 Card-to-Card-Authentisierung G2**

Alle Produkttypen, die die Card-to-Card-Authentisierung für Karten der Generation G2 durchführen, MÜSSEN dabei eine CV-Identität gemäß [gemSpec_Krypt#GS-A_4365] verwenden. ⌫

Das Verfahren zur Durchführung der Card-to-Card-Authentisierung wird in [gemSpec_COS] spezifiziert.

3.2.4 Card-to-Server (C2S) Authentisierung und Trusted Channel G2

⌦ **GS-A_4380 Card-to-Server (C2S) Authentisierung und Trusted Channel G2**

Alle Produkttypen, die die Card-to-Server-Authentisierung für Karten der Generation G2 durchführen, MÜSSEN die folgenden Vorgaben berücksichtigen:

- Die Authentisierung muss mit AES analog [EN-14890-1#8.8] erfolgen
- Die Schlüsselvereinbarung muss analog zu [EN-14890-1#8.8.2] erfolgen.⌫

Das Verfahren zur Durchführung der Card-to-Server-Authentisierung wird in [gemSpec_COS] spezifiziert.

C2S-Authentisierung bzw. der Trusted-Channel wird zwischen der eGK, dem zugeordneten CMS und dem zugeordneten VSDM-System verwendet.

Der Algorithmus AES ist nach [BSI-TR-03116-1] in der TI bis Ende 2020+ (meint bis Ende des Betrachtungsraums der TR) zulässig.

⌦ **GS-A_4381 Schlüssellängen Algorithmus AES**

Alle Produkttypen, die den Algorithmus AES nutzen, MÜSSEN die Schlüssellängen gemäß Tabelle Tab_KRYPT_012 nutzen.⌫

Tabelle 12: Tab_KRYPT_012 Algorithmen für Card-to-Server-Authentifizierung

Algorithmen Typ	Algorithmus	Schlüssellänge
Authentifizierung und Verschlüsselung der Authentisierungsdaten	AES im CBC-Modus (OID 2.16.840.1.101.3.4.1)	128 Bit zulässig bis Ende 2020+

3.2.5 Hinweis für die C2S-Authentisierung

Der in [NIST-SP-800-38B] definierte CMAC unterscheidet sich von dem in [gemSpec_COS#N002.800] definierten CMAC. Man beachte insbesondere (N002.800b): Im Gegensatz zum CMAC [NIST-SP-800-38B] wird beim CMAC gemäß [gemSpec_COS] erwartet, dass die Daten **vor** der CMAC-Berechnung gepaddet werden (siehe auch [gemSpec_COS#Hinweis(19)]).

3.3 Netzwerkprotokolle

Im Gegensatz zu kryptographischen Verfahren für den Integritätsschutz oder die Vertraulichkeit von Daten, bei denen keine direkte Kommunikation zwischen dem Sender bzw. dem Erzeuger und dem Empfänger stattfindet, kann bei Netzwerkprotokollen eine Aushandlung des kryptographischen Algorithmus erfolgen. Das Ziel der nachfolgenden Festlegungen ist es daher, jeweils genau einen verpflichtend zu unterstützenden Algorithmus festzulegen, so dass eine Einigung zumindest auf diesen Algorithmus immer möglich ist. Zusätzlich können aber auch optionale Algorithmen festgelegt werden, auf die sich Sender und Empfänger ebenfalls im Zuge der Aushandlung einigen können. Es darf jedoch durch keine der Komponenten vorausgesetzt werden, dass der Gegenpart diese optionalen Algorithmen unterstützt.

© gematik – freigegeben

3.3.1 IPsec-Kontext

⌦ **GS-A_4382 IPsec-Kontext - Schlüsselvereinbarung**

Alle Produkttypen, die die Authentifizierung, den Schlüsselaustausch und die verschlüsselte Kommunikation im IPsec-Kontext durchführen, MÜSSEN die Schlüsselvereinbarung mittels IKEv2 [RFC-5996] gemäß den folgenden Vorgaben durchführen:

- Als symmetrische Verschlüsselungsalgorithmen sind die Algorithmenverwendungen gemäß Tabelle Tab_KRYPT_013 normativ.
- Hashing und HMAC mittels SHA-1 müssen unterstützt werden. SHA-2 mit 256 Bit und größer kann optional unterstützt werden.
- Zur Authentisierung muss eine Identität mit einem X.509-Zertifikat gemäß [gemSpec_Krypt#GS-A_4360] verwendet werden.
- Die Verwendung von Diffie-Hellman-Gruppen für den Schlüsselaustausch gemäß Tabelle Tab_KRYPT_014 ist verpflichtend.
- Der DH-Exponent für den Schlüsselaustausch soll eine Länge von mindestens 240 Bit haben. Als begründete Ausnahme, die eine Exponentenlänge von weniger als 240 Bit erlaubt, wird derzeit ausschließlich erheblicher Aufwand für Änderungen an bestehenden, bereits bei der gematik angemeldeten, Implementierungen angesehen. Alle neuen Implementierungen müssen von Beginn an eine Exponentenlänge von mindestens 240 Bit berücksichtigen.
- Rekeying: die IKE-Lifetime darf maximal 86400 Sekunden betragen. Die IPsec-SA-Lifetime darf maximal 3600 Sekunden betragen. Der Initiator soll nach Möglichkeit vor Ablauf der Lifetime das Rekeying anstoßen. Ansonsten muss der Responder bei Ablauf der Lifetime das Rekeying von sich aus sicherstellen.
- Für die Schlüsselberechnung muss Forward Secrecy [BSI-TR-02102-1, S.ix] (in [RFC-5996] noch „Perfect Forward Secrecy“ genannt) gewährleistet werden. Meint die Wiederverwendung von zuvor schon verwendeten Diffie-Hellman-Schlüsseln ([RFC-5996#Abschnitt 2.12]) ist nicht erlaubt.⌫

⌦ **GS-A_4383 IPsec-Kontext – Verschlüsselte Kommunikation**

Alle Produkttypen, die die Authentifizierung, den Schlüsselaustausch und die verschlüsselte Kommunikation im IPsec-Kontext durchführen, MÜSSEN für die verschlüsselte Kommunikation die folgenden Vorgaben erfüllen:

- Als symmetrische Verschlüsselungsalgorithmen sind die Algorithmenverwendungen gemäß Tabelle Tab_KRYPT_013 normativ.
- Hashing und HMAC mittels SHA-1 müssen unterstützt werden. SHA-2 mit 256 Bit und größer kann optional unterstützt werden.
- Die Verwendung von Diffie-Hellman-Gruppen für den Schlüsselaustausch gemäß Tabelle Tab_KRYPT_014 ist verpflichtend.

- Der DH-Exponent für den Schlüsselaustausch soll eine Länge von mindestens 240 Bit haben. Als begründete Ausnahme, die eine Exponentenlänge von weniger als 240 Bit erlaubt, wird derzeit ausschließlich erheblicher Aufwand für Änderungen an bestehenden, bereits bei der gematik angemeldeten, Implementierungen angesehen. Alle neuen Implementierungen müssen von Beginn an eine Exponentenlänge von mindestens 240 Bit berücksichtigen.
- Rekeying: die IKE-Lifetime darf maximal 86400 Sekunden betragen. Die IPsec-SA-Lifetime darf maximal 3600 Sekunden betragen. Der Initiator soll nach Möglichkeit vor Ablauf der Lifetime das Rekeying anstoßen. Ansonsten muss der Responder bei Ablauf der Lifetime das Rekeying von sich aus sicherstellen.
- Für die Schlüsselberechnung muss Forward Secrecy [BSI-TR-02102-1, S.ix] (in [RFC-5996] noch „Perfect Forward Secrecy“ genannt) gewährleistet werden. Meint die Wiederverwendung von zuvor schon verwendeten Diffie-Hellman-Schlüsseln ([RFC-5996#Abschnitt 2.12]) ist nicht erlaubt.⌫

Tabelle 13: Tab_KRYPT_013 Algorithmen zur symmetrischen Verschlüsselung für IPsec

Algorithmen Typ	Algorithmus	Schlüssellänge
Symmetrische Verschlüsselung des IPsec-Transports	AES im CBC-Modus (OID 2.16.840.1.101.3.4.1.42)	256 Bit bis Ende 2020+

Tabelle 14: Tab_KRYPT_014 Diffie-Hellman-Gruppen für den Schlüsselaustausch im IPsec-Kontext

kryptographischer Parameter	Vorgabe
zu verwendende Diffie-Hellman-Gruppe	Gruppe 14 definiert in [RFC-3526], verwendbar bis Ende 2020+ (informativ: Die Ordnung der DH-Gruppe ist eine 2048-Bit-Primzahl.)

3.3.2 TLS-Verbindungen

Hinweis: eine Unterteilung in TLS/SSL-Verbindungen mit normalen und erhöhten Schutzbedarf gibt es seit der TR-03116 Version 3.04 (vom 11.06.2010) nicht mehr.

⌦ **GS-A_4384 TLS-Verbindungen**

Alle Produkttypen, die Übertragungen mittels TLS durchführen, MÜSSEN die folgenden Vorgaben erfüllen:

- Zur Authentifizierung muss eine X.509-Identität gemäß [gemSpec_Krypt#GS-A_4359] verwendet werden.
- Als Cipher Suite muss eine Cipher Suite gemäß der Tabelle Tab_KRYPT_015 verwendet werden.
- Die Verwendung von Diffie-Hellman-Gruppen für die Schlüsselaushandlung gemäß Tab_KRYPT_016 ist verpflichtend.

- Der DH-Exponent für den Schlüsselaustausch soll eine Länge von mindestens 240 Bit haben. Als begründete Ausnahme, die eine Exponentenlänge von weniger als 240 Bit erlaubt, wird derzeit ausschließlich erheblicher Aufwand für Änderungen an bestehenden, bereits bei der gematik angemeldeten, Implementierungen angesehen. Alle neuen Implementierungen müssen von Beginn an eine Exponentenlänge von mindestens 240 Bit berücksichtigen. Für bestehende Komponenten bei denen einer Ausnahmeregelung zugestimmt wird, gilt diese Anforderung ab der nächsten größeren Anpassung an der Komponente.⌫

⌦ **GS-A_4385 TLS-Verbindungen, Version 1.2**

Alle Produkttypen, die Übertragungen mittels TLS durchführen, SOLLEN die TLS-Version 1.2 verwenden.⌫

⌦ **GS-A_4386 TLS-Verbindungen, Version 1.1**

Alle Produkttypen, die Übertragungen mittels TLS durchführen, MÜSSEN die TLS-Version 1.1 unterstützen.⌫

⌦ **GS-A_4387 TLS-Verbindungen, Version 1.0**

Alle Produkttypen, die Übertragungen mittels TLS durchführen, DÜRFEN NICHT die TLS-Version 1.0 unterstützen.⌫

⌦ **GS-A_5035 Nichtverwendung des SSL-Protokolls**

Alle Produkttypen, die Daten über Datenleitungen übertragen wollen, DÜRFEN NICHT das SSL-Protokoll unterstützen (Hinweis: TLS Version 1.1 oder 1.2 verwenden).⌫

Tabelle 15: Tab_KRYPT_015 Algorithmen für TLS

Algorithmen Typ	Algorithmus	Symmetrische Schlüssellänge
TLS Cipher Suite	TLS_DHE_RSA_WITH_AES_128_CBC_SHA	128 Bit bis Ende 2020+
	TLS_DHE_RSA_WITH_AES_256_CBC_SHA	256 Bit bis Ende 2020+

Tabelle 16: Tab_KRYPT_016 Diffie-Hellman-Gruppen für die Schlüsselaushandlung bei TLS

kryptographischer Parameter	Vorgabe
zu verwendende Diffie-Hellman-Gruppe	Gruppe 14 definiert in [RFC-3526], verwendbar bis Ende 2020+ (informativ: Die Ordnung der DH-Gruppe ist eine 2048-Bit-Primzahl.)

Einen lesenswerten Abriss bekannter Angriffe auf TLS findet man in [TLS-Attacks], vgl. auch [Breaking-TLS].

Sicherheitsziel bei der Verwendung von TLS in der TI ist die Forward Secrecy [BSI-TR-02102-1, S. ix], was sich u. a. in den vorgegebenen CipherSuites (vgl. Tab_KRYPT_015 und Tab_KRYPT_016) widerspiegelt. Um dieses Ziel zu erreichen, muss sichergestellt werden, dass in regelmäßigen Abständen frisches Schlüsselmaterial über einen authentisierten Diffie-Hellman-Schlüsselaustausch gebildet wird, welches das alte

Material ersetzt, wobei das alte Material sowohl im Klienten als auch im Server sicher gelöscht wird. Insbesondere bei der Nutzung von TLS-Resumption (vgl. [RFC-5246, S. 36] oder [RFC-5077]) kann die Dauer einer TLS-Session deutlich länger sein als die Lebensdauer der TCP-Verbindung innerhalb welcher der initiale Schlüsselaustausch stattgefunden hat. Aus diesem Grunde werden analog zu den IPsec-Vorgaben (vgl. [gemSpec_Krypt#GS-A_4383]) Vorgaben für die maximale Gültigkeitsdauer dieses Schlüsselmaterials gemacht.

⌦ **GS-A_5322 Weitere Vorgaben für TLS-Verbindungen**

Alle Produkttypen, die Übertragungen mittels TLS durchführen, MÜSSEN u. a. folgende Vorgaben erfüllen:

- Falls der Produkttyp als Klient oder als Server im Rahmen von TLS an einer Session-Resumption mittels SessionID (vgl. [RFC-5246, Abschnitt 7.4.1.2]) teilnimmt, MUSS er sicherstellen, dass nach spätestens 24 Stunden das über den Diffie-Hellman-Schlüsselaustausch ausgehandelte Schlüsselmaterial und alles davon abgeleitete Schlüsselmaterial (vgl. [RFC-5246, Abschnitt 8.1 und 6.3]) bei ihm sicher gelöscht wird.

- Falls der Produkttyp als Klient im Rahmen von TLS an einer Session-Resumption nach [RFC-5077] teilnimmt, MUSS er sicherstellen, dass nach spätestens 24 Stunden das über den Diffie-Hellman-Schlüsselaustausch ausgehandelte Schlüsselmaterial und alles davon abgeleitete Schlüsselmaterial (vgl. [RFC-5246, Abschnitt 8.1 und 6.3]) bei ihm sicher gelöscht wird. Damit verbundene SessionTickets MUSS er ebenfalls sicher löschen.

- Falls der Produkttyp als Server im Rahmen von TLS an einer Session-Resumption nach [RFC-5077] teilnimmt, MUSS er sicherstellen, dass nach spätestens 24 Stunden das über den Diffie-Hellman-Schlüsselaustausch ausgehandelte Schlüsselmaterial und alles davon abgeleitete Schlüsselmaterial (vgl. [RFC-5246, Abschnitt 8.1 und 6.3]) bei ihm sicher gelöscht wird. Damit verbundene SessionTickets MUSS er, falls bei ihm vorhanden, sicher löschen. Das Schlüsselmaterial, dass bei der Erzeugung des SessionTickets (für die Sicherung von Vertraulichkeit und Authentizität der SessionTickets) verwendet wird, MUSS spätestens alle 48 Stunden gewechselt werden und das alte Material MUSS sicher gelöscht werden. Als kryptographische Verfahren zur Erzeugung/Sicherung der SessionTickets MÜSSEN ausschließlich nach [BSI-TR-03116-1] zulässige Verfahren verwendet werden und das Schlüsselmaterial muss die Entropieanforderungen gemäß [gemSpec_Krypt#GS-A_4368] erfüllen.

  Falls ein Produkttyp als Klient oder Server im Rahmen von TLS die Renegotiation unterstützt, so MUSS er dies ausschließlich nach [RFC-5746] tun. Ansonsten MUSS er die Renegotiation-Anfrage des Kommunikationspartners ablehnen. ⌫

Aktuell gibt es in der TI keine Anwendungsfälle für eine Session-Renegotiation im Rahmen von TLS.

Hinweis: Hintergrundinformationen sind in diesem Zusammenhang in [CM-2014] zu finden.

3.3.3 DNSSEC-Kontext

⌦ GS-A_4388 DNSSEC-Kontext

Alle Produkttypen, die DNSSEC verwenden, MÜSSEN die Algorithmen und Vorgaben gemäß Tabelle Tab_KRYPT_017 erfüllen.⌫

Tabelle 17: Tab_KRYPT_017 Algorithmen für DNSSEC

Algorithmen Typ	Algorithmus	Schlüssellänge
TSIG – symmetrischer Schlüssel zur Absicherung der Transaktionskanäle zwischen zwei Name-Server-Instanzen bei Zonentransfers, Änderungsbenachrichtigungen, dynamischen Updates und rekursiven Queries.	HMAC-SHA-256	256 Bit
DNSSEC ZSK Asymmetrische Schlüssel zur Wahrung der Authentizität und Integrität von Zonendatenobjekten.	RSA-SHA-256 [RFC-5702]	2048 Bit
DNSSEC KSK Asymmetrische Schlüssel zur Wahrung der Authentizität und Integrität von Zonendatenobjekten.	RSA-SHA-256 [RFC-5702]	2048 Bit

Hinweis: Nach [RFC-5702] ist die Verwendung von SHA-256 [FIPS-180-4] möglich. Schlüssellängen von RSA zwischen 512 bis 4096 Bit sind seit den Anfängen von DNSSEC möglich. Bei TSIG ist nach [RFC-4635] auch SHA-256 verwendbar und bspw. von bind seit der Version 9.5 unterstützt.

3.4 Masterkey-Verfahren (informativ)

Die gematik wurde aufgefordert, beispielhaft ein mögliches Ableitungsverfahren für einen versichertenindividuellen symmetrischen Schlüssel auf Grundlage eines Ableitungsschlüssels (Masterkey) aufzuführen. Ein Kartenherausgeber ist frei in der Wahl seines Ableitungsverfahrens. Jedoch müssen beim Einsatz eines Ableitungsverfahrens, um die Qualität der Ableitung zu garantieren, insbesondere folgende Punkte beachtet werden:

- Der Ableitungsprozess muss unumkehrbar und nicht-vorhersehbar sein, um sicherzustellen, dass die Kompromittierung eines abgeleiteten Schlüssels nicht den Ableitungsschlüssel oder andere abgeleitete Schlüssel kompromittiert.
- Bei einer Schlüsselableitung (im Sinne von [ISO-11770]) basiert die kryptographische Stärke der abgeleiteten Schlüssel auf der Ableitungsfunktion und der kryptographischen Stärke des geheimen Ableitungsschlüssels (insbesondere hier dessen Entropie). Die Entropie der abgeleiteten Schlüssel ist kleiner gleich der Entropie des geheimen Ableitungsschlüssels. Um die Entropie der abgeleiteten Schlüssel sicherzustellen, muss die Entropie des geheimen Ableitungsschlüssels (deutlich) größer sein als die zu erreichende Entropie der abgeleiteten Schlüssel.

- Der Betreiber eines Schlüsseldienstes muss im Falle des Einsatzes einer Schlüsselableitung (nach [ISO-11770]) in seinem Sicherheitskonzept Maßnahmen für das Bekanntwerden von Schwächen des kryptographischen Verfahrens, welche die Grundlage der Schlüsselableitung ist, darlegen.

Ein Kartenherausgeber hat auch die Freiheit, gar kein Ableitungsverfahren zu verwenden, sondern alle symmetrischen SK.CMS aller seiner Karten sicher in seinem RZ vorzuhalten.

Ziel des Masterkey-Verfahrens zur Ableitung eines versichertenindividuellen Schlüssels ist es, aus einem geheimen Masterkey und einem öffentlichen[3] versichertenindividuellen Merkmal einen geheimen symmetrischen Schlüssel abzuleiten, der zur Absicherung der Verbindung zwischen CMS und Smartcard verwendet wird. Die Vertraulichkeit der Daten muss durch die Geheimhaltung des Masterkeys gewährleistet sein. Das bedeutet, die Geheimhaltung anderer Daten als des Masterkeys darf für die Vertraulichkeit der Daten nicht notwendig sein. Die Durchführung dieses Verfahrens muss bei gleichen Eingangsparametern immer das gleiche Ergebnis generieren.

Für die Durchführung des Algorithmus wird neben dem Masterkey auch noch mindestens ein versichertenindividuelles Merkmal verwendet. Die Auswahl des Merkmals ist fachlich motiviert und wird daher in diesem Dokument nicht spezifiziert. Das in Tabelle 18 beispielhafte Verfahren besteht aus einer Kombination von AES-Verschlüsselung [FIPS-197] und Hashwert-Bildung. Die Schlüssel- bzw. Hashwert-Länge ergibt sich gemäß Tabelle 19.

Tabelle 18: Tab_KRYPT_018 Ablauf zur Berechnung eines versichertenindividuellen Schlüssels

Reihenfolge	Beschreibung	Formale Darstellung
1	Bildung eines Hashwertes über dem versichertenindividuellen Merkmal unter Verwendung eines statischen Padding-Verfahrens für den Fall, dass das versichertenindividuelle Merkmal in seiner Länge nicht der Blocklänge des Hash-Algorithmus entspricht. Im Ergebnis wird ein versichertenindividuelles Merkmal geeigneter Länge für den nächsten Schritt erzeugt.	HASH#1 = SHA-256(versichertenindividuelles Merkmal)
2	AES-Verschlüsselung des Resultats mit dem Masterkey. Durch die Verschlüsselung an dieser Stelle ist sichergestellt, dass der versichertenindividuelle Schlüssel nur durch den Besitzer des geheimen Masterkeys erzeugt werden kann.	ENC#1 = AES-256(HASH#1)
3	Bildung eines Hashwertes über dem Ergebnis des vorherigen Verarbeitungsschritts. Dies stellt sicher, dass ein Schlüssel geeigneter Länge erzeugt wird.	Versichertenindividueller Schlüssel = SHA-256(ENC#1)

In der nachfolgenden Tabelle werden Kürzel entsprechend der Definition aus Abschnitt 3.2.3 verwendet.

[3] Öffentlich bedeutet an dieser Stelle nicht, dass die Merkmale selbst nicht schützenswert sind, es soll jedoch ausdrücken, dass die Vertraulichkeit des versichertenindividuellen Schlüssels nicht von der Geheimhaltung dieser Merkmale abhängt.

Tabelle 19: Tab_KRYPT_019 eingesetzte Algorithmen für die Ableitung eines versichertenindividuellen Schlüssels

Algorithmen Typ	Algorithmus	Unterverfahren
Masterkey-Verfahren für die Generierung des versichertenindividuellen Schlüssel innerhalb eines CMS	AES basiertes Verfahren gemäß vorheriger Definition	AES-256 SHA-256 anwendbar bis Ende 2020+

3.5 Hybride Verschlüsselung binärer Daten

Für die hybride Verschlüsselung werden die Daten zunächst symmetrisch mittels eines zufällig gewählten geheimen symmetrischen Schlüssels verschlüsselt. Der geheime Schlüssel wird im Anschluss asymmetrisch für jeden Empfänger separat verschlüsselt.

Hinweis: unter binären Daten sind im gesamten Dokument beliebige Daten insbesondere beliebigen Typs (Text, HTML, PDF, JPG etc.) zu verstehen. Es gilt das Prinzip: das Spezielle vor dem Allgemeinen: gibt es weitere spezielle Vorgaben für bestimmte Datenformate, sind diese für die entsprechenden Daten verpflichtend (überschreiben oder ergänzen die allgemeinen Vorgaben).

3.5.1 Symmetrischer Anteil der hybriden Verschlüsselung binärer Daten

⌦ **GS-A_4389 Symmetrischer Anteil der hybriden Verschlüsselung binärer Daten**

Produkttypen, die die hybride Verschlüsselung binärer Daten durchführen, MÜSSEN für den symmetrischen Anteil der Verschlüsselung die folgenden Vorgaben berücksichtigen:

- Als symmetrische Block-Chiffre muss AES [FIPS-197] mit einer Schlüssellänge von 256 Bit im Galois/Counter Mode (GCM) gemäß [NIST-SP-800-38D] mit der Tag-Länge von 128 Bit verwendet werden.
- Die IVs dürfen sich bei gleichem Schlüssel nicht wiederholen (vgl. [NIST-SP-800-38D#S.25] und [BSI-TR-02102-1#S.24]). Der IV soll eine Bitlänge von 96 Bit besitzen, seine Länge muss mindestens 96 Bit sein. Es wird empfohlen den IV zufällig zu wählen (vgl. [gemSpec_Krypt#GS-A_4367]).
- Hinweis: Im Normalfall ist davon auszugehen, dass für die Sicherung der Integrität und Authentizität der zu verschlüsselnden Daten zudem noch eine Signatur dieser Daten notwendig ist.⌫

Hinweis: In [RFC-5084] findet man Informationen über die Verwendung von AES-GCM innerhalb von CMS [RFC-5652].

3.5.2 Asymmetrischer Anteil der hybriden Verschlüsselung binärer Daten

⌦ **GS-A_4390 Asymmetrischer Anteil der hybriden Verschlüsselung binärer Daten**

Produkttypen, die die hybride Verschlüsselung binärer Daten durchführen, MÜSSEN für den asymmetrischen Anteil der Verschlüsselung die folgenden Vorgaben berücksichtigen:

- Als asymmetrisches Verschlüsselungsverfahren soll RSAES-OAEP gemäß [PKCS#1, Kapitel 7.1] verwendet werden.
- Sofern eine Implementierung der Systeme mit RSAES-OAEP nicht möglich ist, muss RSAES-PKCS1-v1-5 gemäß [PKCS#1 Kapitel 7.2] verwendet werden. Die Gültigkeit dieses Verfahrens ist bis Ende 2017 beschränkt. Bei der Verwendung dieses Verfahrens ist besonders auf die zusätzliche Sicherung der Integrität und Authentizität der verschlüsselten Daten zu achten, da Angriffe bekannt sind bei denen ein Angreifer korrekt dekodierbare Chiffretexte erzeugen kann.
- Als Mask-Generation-Function für die Verwendung in RSAES-OAEP muss MGF 1 mit SHA-256 als Hash-Funktion gemäß [PKCS#1, Anhang B.2.1] verwendet werden.⌫

3.6 Symmetrische Verschlüsselung binärer Daten

⌦ **GS-A_5016 Symmetrische Verschlüsselung binärer Daten**

Produkttypen, die die symmetrische Verschlüsselung binärer Daten durchführen, MÜSSEN die folgenden Vorgaben berücksichtigen:

- Als symmetrische Block-Chiffre muss AES [FIPS-197] mit einer Schlüssellänge von 256 Bit im Galois/Counter Mode (GCM) gemäß [NIST-SP-800-38D] mit der Tag-Länge von 128 Bit verwendet werden.
- Die IVs dürfen sich bei gleichem Schlüssel nicht wiederholen (vgl. [NIST-SP-800-38D#S.25] und [BSI-TR-02102-1#S.24]). Der IV soll eine Bitlänge von 96 Bit besitzen, seine Länge muss mindestens 96 Bit sein. Es wird empfohlen den IV zufällig zu wählen (vgl. [gemSpec_Krypt#GS-A_4367]).
- Hinweis: Im Normalfall ist davon auszugehen, dass für die Sicherung der Integrität und Authentizität der übertragenen Daten zudem noch eine Signatur der zu verschlüsselnden Daten notwendig ist.⌫

Hinweis: In [RFC-5084] findet man Informationen über die Verwendung von AES-GCM innerhalb von CMS [RFC-5652].

3.7 Signatur binärer Inhaltsdaten (Dokumente)

⌦ **GS-A_5080 Signaturen binärer Daten (Dokumente)**

Alle Produkttypen, die CMS-Signaturen [RFC-5652] von Inhaltsdaten (wie bspw. Textdokumenten ungleich PDF/A) erzeugen oder prüfen, MÜSSEN die Algorithmen und Vorgaben der Tabelle Tab_KRYPT_020 erfüllen.⌫

Tabelle 20: Tab_KRYPT_020 Algorithmen für die Erzeugung und Prüfung von binären Daten im Kontext von Dokumentensignaturen

Signaturbestandteil	Beschreibung	Algorithmus	Anmerkung
Signaturstandard	Signaturstandard	ETSI TS 101 733 V1.7.4 (2008-07) Electronic Signatures and Infrastructures (ESI); CMS Advanced Electronic Signatures (CAdES) [ETSI-CAdES]	Die Verwendung des Standards ist für die Signatur von Dokumenten verpflichtend die mittels CMS (PKCS#7) erzeugt werden.
kryptographisches Signaturverfahren	Algorithmus für die Berechnung des Nachrichten Digest und die Verschlüsselung mit dem privaten Schlüssel	**RSASSA-PKCS1-v1_5 mit SHA256** Dieser Algorithmus ist nur noch bis Ende 2016 in der TI verwendbar, mit der Empfehlung ihn nicht mehr zu verwenden. oder **RSASSA-PSS mit SHA256** bis nach Ende 2020+ verwendbar (Ende des Betrachtungshorizonts) oder **ISO9796-2 DS2 [ISO-9796-2]**	Die Verwendung einer dieser Algorithmen ist verpflichtend. Es soll RSASSA-PSS verwendet werden. Alle hier aufgeführten Signaturverfahren müssen von einer Signaturprüfenden Komponente überprüfbar sein.
DigestMethod	Methode zur Berechnung eines Digest der zu signierenden Bereiche	SHA-256	Die Verwendung des Algorithmus ist verpflichtend.
Kryptographisches Token	Kryptographisches Token für die Signatur, bestehend aus einem privaten Schlüssel und einem zugehörigen X.509-Zertifikat	Identitäten gemäß einem der folgenden Abschnitte 2.1.1.1 2.1.1.2	Die Auswahl des kryptographischen Tokens ist von dem jeweiligen Einsatzzweck abhängig.

3.8 Signaturen innerhalb von PDF/A-Dokumenten

⌦ GS-A_5081 Signaturen von PDF/A-Dokumenten

Alle Produkttypen, die in PDF/A-Dokumenten [PDF/A-2] Signaturen einbetten/erzeugen oder diese Signaturen prüfen, MÜSSEN die Algorithmen und Vorgaben der Tabelle Tab_KRYPT_021 erfüllen.⌫

Tabelle 21: Tab_KRYPT_021 Algorithmen für die Erzeugung und Prüfung von PDF/A-Dokumentensignaturen

Signaturbestandteil	Beschreibung	Algorithmus	Anmerkung
Signaturstandard	Signaturstandard	ETSI TS 102 778-3 V1.2.1, PDF Advanced Electronic Signature Profiles; Part 3: PAdES Enhanced – PAdES-BES and PAdES-EPES Profiles Technical Specification, 2010 [PAdES-3]	Die Verwendung des Standards ist für die Signatur von PDF/A [PDF/A-2] Dokumenten verpflichtend, die mittels eingebetteter Signaturen signiert

Signaturbestandteil	Beschreibung	Algorithmus	Anmerkung
			werden.
kryptographisches Signaturverfahren	Algorithmus für die Berechnung des Nachrichten Digest und die Verschlüsselung mit dem privaten Schlüssel	**RSASSA-PKCS1-v1_5 mit SHA256** Dieser Algorithmus ist nur noch bis Ende 2016 in der TI verwendbar, mit der Empfehlung ihn nicht mehr zu verwenden. oder **RSASSA-PSS mit SHA256** bis nach Ende 2020+ verwendbar (Ende des Betrachtungshorizonts) oder **ISO9796-2 DS2 [ISO-9796-2]**	Die Verwendung einer dieser Algorithmen ist verpflichtend. Es soll RSASSA-PSS verwendet werden. Alle hier aufgeführten Signaturverfahren müssen von einer Signaturprüfenden Komponente überprüfbar sein.
DigestMethod	Methode zur Berechnung eines Digest der zu signierenden Bereiche	**SHA-256**	Die Verwendung des Algorithmus ist verpflichtend.
Kryptographisches Token	Kryptographisches Token für die Signatur, bestehend aus einem privaten Schlüssel und einem zugehörigen X.509-Zertifikat	Identitäten gemäß einem der folgenden Abschnitte 2.1.1.1 2.1.1.2	Die Auswahl des kryptographischen Tokens ist von dem jeweiligen Einsatzzweck abhängig.

3.9 MAC im Rahmen der Personalisierung der eGK

⌦ GS-A_4391 MAC im Rahmen der Personalisierung der eGK

Der Herausgeber der eGK MUSS sicherstellen, dass bei der Personalisierung der eGK die Daten bei der Übermittlung integritätsgeschützt werden. Für die Absicherung der Integrität ist in diesem Kontext der AES-256 CMAC nach [NIST-SP-800-38B] (vgl. [BSI-TR-03116-1#3.2.2, 4.5.2]) zu verwenden.

Die Länge des CMAC muss 128 Bit betragen.

Nach [NIST-SP-800-38B#S.13] sollen nicht mehr als 2^{48} Nachrichtenblöcke (2^{22} GByte) mit dem selbem Schlüssel verarbeitet werden. Nach [NIST-SP-800-38B#S.14] ist ein CMAC anfällig für Replay-Attacken, was bei der Anwendung des CMACs zu berücksichtigen ist.⌫

3.10 Algorithmus im Rahmen der Bildung der pseudonymisierten Versichertenidentität

⌦ GS-A_4392 Algorithmus im Rahmen der Bildung der pseudonymisierten Versichertenidentität

Alle Produkttypen, die pseudonymisierte Versichertenidentitäten berechnen, MÜSSEN den Hash-Algorithmus SHA-256 [FIPS-180-4] verwenden.⌫

3.11 Spezielle Anwendungen von Hashfunktionen

⌦ GS-A_4393 Algorithmus bei der Erstellung von Hashwerten von Zertifikaten oder öffentlichen Schlüsseln

Alle Produkttypen, die Fingerprints eines öffentlichen Schlüssels oder eines Zertifikates erstellen, MÜSSEN den Hash-Algorithmus SHA-256 [FIPS-180-4] dafür verwenden.⌫

Erläuterung: Alle CAs und der TSL-Dienst müssen im Rahmen ihrer Prozesse öffentliche Schlüssel oder Zertifikate (bspw. auf Webseiten) veröffentlichen. Dabei wird auch jeweils der SHA-256 Hashwert mit veröffentlicht. Hersteller einer gSMC-KT müssen den Hashwert des auf der Karte befindlichen Zertifikats in MF / DF.KT / EF.C.SMKT.AUT.R2048 entweder auf dem ID-1-Kartenkörper drucken (das ID-000-Modul ist dann herausbrechbar) oder ausgedruckt mitliefern. Der Konnektor muss den Hashwert des Zertifikats bei initialen Pairing mit dem KT berechnen und dem Administrator präsentieren.

⌦ GS-A_5131 Hash-Algorithmus bei OCSP / CertID

Alle Produkttypen, die OCSP-Anfragen stellen oder beantworten, MÜSSEN bei der Erstellung und Verwendung der CertID-Struktur (vgl. [RFC-6960, Abschnitt 4.1.1] oder [RFC-2560, Abschnitt 4.1.1]) den Hash-Algorithmus SHA-1 [FIPS-180-4] verwenden.⌫

3.12 kryptographische Vorgaben für die SAK des Konnektors

⌦ GS-A_5071 kryptographische Vorgaben für eine Signaturprüfung in der SAK-Konnektor

Die SAK des Konnektors MUSS bei der Prüfung von qualifizierten elektronischen Signaturen mindestens folgende Verfahren wie im Algorithmenkatalog [ALGCAT] benannt, unterstützen:

- SHA-256, SHA-512/256, SHA-384, SHA-512 nach FIPS-180-4 (März 2012) [FIPS-180-4] (jeweils Abschnitt 6.2, 6.7, 6.5 und 6.4 ebenda),
- RSASSA-PSS nach PKCS#1 (PKCS#1 v2.1: RSA Cryptographic Standard, 14.06.2002) Abschnitt 8.1 und 9.1,
- RSASSA-PKCS1-v1_5 nach PKCS#1 (PKCS#1 v2.1: RSA Cryptographic Standard, 14.06.2002) Abschnitt 8.2 und 9.2,
- „Digital signature scheme 2“ aus ISO/IEC 9796-2 [ISO-9796-2],
- bei RSA muss ein Modulus zwischen 1976 bis 4096 Bit verwendbar sein,
- ECDSA basierend auf E(F_p) (vgl. Technische Richtlinie 03111, Version 2.0) auf der Kurve P256r1 [RFC-5639]. ⌫

3.13 Migration kryptographischer Primitive für die Signatur im PKI-Bereich

Diese Vorgabe ist aus den Produkten TSP-CVC, TSP-X.509-nonQES, TSL-Dienst hier her verlagert worden (ehemals TIP1-A_2623).

⌦ **GS-A_5079 Migration von Algorithmen und Schlüssellängen bei PKI-Betreibern**

Der Anbieter einer Schlüsselverwaltung MUSS neue Vorgaben zu Algorithmen und/oder Schlüssellängen der gematik nach einer vorgegebenen Übergangsfrist umsetzen. Nach Ablauf der Übergangsfrist MÜSSEN ausschließlich diese geänderten Parameter bei der Erzeugung von Zertifikaten verwendet werden. ⌫

3.14 Spezielle Anwendungen von kryptographischen Signaturen

⌦ **GS-A_5207 Signaturverfahren beim initialen Pairing zwischen Konnektor und eHealth-Kartenterminal**

Alle Produkttypen, die beim initialen Pairing zwischen Konnektor und eHealth-Kartenterminal die Signatur des Shared-Secret (ShS.AUT.KT vgl. [gemSpec_KT#2.5.2.1, 3.7.2.1]) erzeugen oder prüfen, MÜSSEN dafür RSASSA-PSS [PKCS#1] verwenden.⌫

Erläuterung: Beim initialen Pairing zwischen Konnektor und eHealth-Kartenterminal wird vom Konnektor ein 16 Byte langes Geheimnis erzeugt, das bei späteren Verbindungsaufbauten zwischen Konnektor und KT im Rahmen eines Challenge-Response-Verfahrens ([gemSpec_KT#3.7.2]) verwendet wird. Dieses Geheimnis wird von der gSMC-KT des KT beim initialen Pairing signiert. Die Signatur wird vom KT zum Konnektor transportiert und dort vom Konnektor geprüft.

⌦ **GS-A_5208 Signaturverfahren für externe Authentisierung**

Der Konnektor MUSS an der Schnittstelle für die externe Authentisierung die Signaturverfahren RSASSA-PKCS1-v1_5 [PKCS#1] und RSASSA-PSS [PKCS#1] anbieten.⌫

Erläuterung: Der Konnektor erlaubt (bei entsprechender Berechtigung) die direkte Nutzung der privaten Schlüssel MF/ DF.ESIGN/ PrK.HP.AUT.* auf einem HBA oder MF/ DF.ESIGN/ PrK.HCI.AUT.* auf einer SMC-B durch ein Primärsystem. Dies wird fast immer für eine klientenseitige TLS-Authentisierung gegenüber einem TLS-Server (außerhalb der TI) verwendet. Dafür werden über die Schnittstelle RSASSA-PKCS1-v1_5-Signaturen von den entsprechenden Karten erzeugt und über den Konnektor an ein Primärsystem übergeben. Für unbenannte Anwendungen müssen auch RSASSA-PSS-Signaturen erzeugbar sein. Diese Signaturen sind nicht als Dokumentensignaturen verwendbar, der Verwendungszweck ist in den zu den privaten Schlüsseln gehörigen Zertifikaten kodiert (ExtendedKeyUsage: keyPurposeId = id-kp-clientAuth).

4 Umsetzungsprobleme mit der TR-03116-1

Das u. a. durch die TR-03116-1 [BSI-TR-03116-1] angestrebte Sicherheitsniveau soll persönliche medizinische Daten effektiv schützen. Dazu lehnt sie sich an die sehr starken kryptographischen Vorgaben für die qualifizierte elektronische Signatur [ALGCAT] an. Einige Formate (bspw. XMLDSig) oder Implementierungen (bspw. Standard-Java-Bibliotheken) können einige Vorgaben von Hause aus nicht erfüllen.

Dieses Kapitel weist auf Umsetzungsprobleme hin (ehemals Kapitel 3.3 aus dem Kryptographiekonzept des Basis-Rollouts).

4.1 XMLDSig und PKCS1-v2.1

Mit [XMLDSig] allein ist aktuell keine Nutzung von RSASSA-PSS [PKCS#1] möglich. Die Alternative für RSA-Signaturen RSASSA-PKCS1-v1_5 ist nach [BSI-TR-03116-1] nur noch bis Ende 2016 zulässig (insbesondere auch für digitale nicht-qualifizierte elektronische Signaturen).

Aus diesem Grund hat die gematik entschieden für die Signatur nach [XMLDSig] zusätzliche Identifier für RSASSA-PSS aus [RFC-6931] innerhalb der TI zu verwenden, welche auf der Lösung aus [XMLDSig-RSA-PSS] basieren. Der RFC-6931 [RFC-6931] ist die Aktualisierung von [RFC-4051]. Die in Abschnitt „2.3.9 RSASSA-PSS With Parameters“ und „2.3.10 RSASSA-PSS Without Parameters“ aufgeführten Identifier für RSASS-PSS-Signaturen müssen innerhalb von XMLDSig für solche Signaturen verwendet werden.

⌦ **GS-A_5091 Verwendung von RSASSA-PSS bei XMLDSig-Signaturen**

Produkttypen, die RSASSA-PSS-Signaturen [PKCS#1] innerhalb von XMLDSig erstellen oder prüfen, MÜSSEN die Identifier aus [RFC-6931] Abschnitt „2.3.9 RSASSA-PSS With Parameters“ und „2.3.10 RSASSA-PSS Without Parameters“ für die Kodierung dieser Signaturen verwenden. ⌫

Ein Beispiel aus [RFC-6931] Abschnitt „2.3.10 RSASSA-PSS Without Parameters“:

```
<SignatureMethod
     Algorithm=
     "http://www.w3.org/2007/05/xmldsig-more#sha256-rsa-MGF1"
   />
```

Vgl. [gemSpec_COS, (N003.000)]: Die Hashfunktion, auf der die Mask-generation-function basiert, ist SHA-256 [FIPS-180-4]. Die Länge des salt ist gleich der Ausgabelänge eben jener Hashfunktion (= 256 Bit).

4.2 XMLEnc: Die Nutzung von RSAES-OAEP und AES-GCM

Bei der Verschlüsselung mittels XMLEnc [XMLEnc] gibt es zwei Probleme in Bezug auf fehlende Identifier für kryptographische Verfahren, die in Abstimmung mit dem BSI für den Einsatz in der TI notwendig sind.

- Für die symmetrische Verschlüsselung mittels AES-GCM ([FIPS-197], [NIST-SP-800-38D]) gibt es keine Algorithmen-Identifier innerhalb von [XMLEnc]. Solche gibt es in [XMLEnc-1.1, Abschnitt 5.2.4].
- Bei der Verschlüsselung mittels [PKCS#1] gibt es zwei Varianten: RSAES-OAEP und RSAES-PKCS1-v1_5. Beide Varianten werden von den Smartcards der TI unterstützt und für die zweite Variante stehen innerhalb von [XMLEnc] ausreichend Identifier zur Verfügung. Diese Variante ist nach [BSI-TR-03116-1] nur bis Ende 2017 zulässig. Bei der Variante RSAES-OAEP fehlt in [XMLEnc] ein Identifier für RSAES-OAEP mit der MGF basierend auf SHA-256 (vgl. auch Kapitel 5.10 „MGF Mask Generation Function“ in [gemSpec_COS]). Einen solchen Identifier[4] gibt es in XMLEnc Version 1.1 [XMLEnc-1.1, Abschnitt 5.5.2].

Aus diesem Grund hat die gematik entschieden für die XML-Verschlüsselung die Vorgaben aus [XMLEnc-1.1] zu verwenden.

4.3 XML Signature Wrapping und XML Encryption Wrapping

Komplexität ist der natürliche Feind von Sicherheit. Die unten dem Sammelbegriff XML betitelten Formate und Protokolle sind sehr flexibel und leistungsfähig, aber auch sehr komplex. Noch dazu sind Sicherheitsmechanismen in diesem Bereich zum Teil nachträglich beigefügt worden und sind damit oft weniger leistungsfähig als im CMS-Bereich. XML-Daten effektiv zu schützen ist aktives Forschungsthema [XMLEnc-CM], [XSpRES]. Öfter als in anderen Bereichen werden neue Schwachstellen bekannt [BreakingXMLEnc], [XSW-Attack].

Aus diesem Grunde wird bei einer Sicherheitsevaluierung gesondert auf derartige Angriffe geachtet. Die gematik beobachtet neue Entwicklungen im Bereich der XML-Sicherheit und leitet falls notwendig Maßnahmen ein.

4.4 Güte von Zufallszahlen

Nach dem Kerckhoffs'schen Prinzip von 1883 [Ker-1883] darf die Sicherungsleistung von kryptographischen Verfahren alleinig auf der Geheimhaltung der geheimen oder privaten Schlüssel beruhen. Geheimhaltung inkludiert insbesondere, dass sie nicht erraten werden können. Wenn bei einer Schlüsselerzeugung zu wenig Entropie vorhanden ist, kann die Geheimhaltung nicht gewährleistet werden. Die kryptographischen Verfahren, welche mit diesen Schlüsseln dann arbeiten, können die von ihnen verlangten Sicherheitsleistungen nicht mehr erbringen. Aus diesem Grunde verlangt [BSI-TR-03116-

[4] „http://www.w3.org/2009/xmlenc11#mgf1sha256“

1] eine Mindestgüte der Zufallszahlerzeugung u. a. bei einer Schlüsselerzeugung. Die Basis für die Beurteilung der Güte stellt [AIS-20] und [AIS-31] dar.

Aktuell sind nicht alle Produkte in der TI bez. dieser Mindestgüte bewertet worden. Davon sind Smartcards nicht betroffen, da diese eine Sicherheitsevaluierung/-zertifizierung durchlaufen haben, bei der die Güte der Zufallszahlenerzeugung positiv beurteilt wurde. Probleme bereiten insbesondere VPN-Konzentratoren und HSMs.

Neben einer möglichen Common-Criteria-Zertifizierung dieser Produkte, bei der analog zu den Smartcards die Güte geprüfte wird, gibt es weitere mögliche Lösungen:

1. gesonderte Prüfung der Güte nach [AIS-20] und [AIS-31] ohne komplette Common-Criteria-Zertifizierung,
2. Herstellererklärung über die Güte (wie sie bspw. aktuell bei der Kartenproduktion üblich ist).

© gematik – freigegeben

Anhang A - Verzeichnisse

A1 – Abkürzungen

Kürzel	Erläuterung
C2C	Card to Card
C2S	Card to Server
CA	Certificate Authority
CBC	Cipher Block Chaining
DNS	Domain Name System
DNSSEC	Domain Name System Security Extensions
DRNG	Deterministic Random Number Generator
eGK	elektronische Gesundheitskarte
IV	Initialisierungsvektor
MAC	Message Authentication Code
OCSP	Online Certificate Status Protocol
OID	Object Identifier
OSI	Open Systems Interconnection
SAK	Signaturanwendungskomponente
TI	Telematikinfrastruktur
TLS	Transport Layer Security
TSIG	Transaction Signature
URI	Uniform Resource Identifier

A2 – Glossar

Das Glossar wird als eigenständiges Dokument, vgl. [gemGlossar] zur Verfügung gestellt.

A3 – Abbildungsverzeichnis

Abbildung 1: Verwendung von Algorithmen nach Zonen und OSI-Schicht 15

A4 – Tabellenverzeichnis

Tabelle 1: Tab_KRYPT_001 Übersicht über Arten von X.509-Identitäten 8

Tabelle 2: Tab_KRYPT_002 Algorithmen für X.509-Identitäten zur Erstellung nicht-qualifizierter Signaturen 9

Tabelle 3: Tab_KRYPT_003 Algorithmen für X.509-Identitäten zur Erstellung qualifizierter elektronischer Signaturen 9

Tabelle 4: Tab_KRYPT_004 Algorithmen für CV-Zertifikate.....11

Tabelle 5: Tab_KRYPT_005 Algorithmen für CV-CA-Zertifikate.....11

Tabelle 6: Tab_KRYPT_006 Algorithmen für CV-Zertifikate.....12

Tabelle 7: Tab_KRYPT_007 Algorithmen für CV-CA-Zertifikate.....12

Tabelle 8: Tab_KRYPT_008 Beispiele für solche Algorithmen-URIs.....16

Tabelle 9: Tab_KRYPT_009 Algorithmen für die Erzeugung von nicht-qualifizierten elektronischen XML-Signaturen 16

Tabelle 10: Tab_KRYPT_010 Algorithmen für qualifizierte XML-Signaturen 17

Tabelle 11: Tab_KRYPT_011 Algorithmen für Card-to-Server-Authentifizierung.....19

Tabelle 12: Tab_KRYPT_012 Algorithmen für Card-to-Server-Authentifizierung.....20

Tabelle 13: Tab_KRYPT_013 Algorithmen zur symmetrischen Verschlüsselung für IPsec 22

Tabelle 14: Tab_KRYPT_014 Diffie-Hellman-Gruppen für den Schlüsselaustausch im IPsec-Kontext 22

Tabelle 15: Tab_KRYPT_015 Algorithmen für TLS 23

Tabelle 16: Tab_KRYPT_016 Diffie-Hellman-Gruppen für die Schlüsselaushandlung bei TLS.....23

Tabelle 17: Tab_KRYPT_017 Algorithmen für DNSSEC.....25

Tabelle 18: Tab_KRYPT_018 Ablauf zur Berechnung eines versichertenindividuellen Schlüssels 26

Tabelle 19: Tab_KRYPT_019 eingesetzte Algorithmen für die Ableitung eines versichertenindividuellen Schlüssels.....27

Tabelle 20: Tab_KRYPT_020 Algorithmen für die Erzeugung und Prüfung von binären Daten im Kontext von Dokumentensignaturen 29

Tabelle 21: Tab_KRYPT_021 Algorithmen für die Erzeugung und Prüfung von PDF/A-Dokumentensignaturen.....29

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 der Telematikinfrastruktur
[gemSpec_COS]	gematik: Spezifikation des Card Operating System (COS)
[gemSpec_eGK_ObjSys]	gematik: Die Spezifikation der elektronischen Gesundheitskarte (eGK) – Objektsystem
[gemSpec_KT]	gematik: Spezifikation eHealth-Kartenterminal
[gemSpec_SST_FD_VSDM]	gematik: Schnittstellenspezifikation Fachdienste (UFS/VSDD/CMS)

A5.2 – Weitere Dokumente

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
[AIS-20-1999]	W. Schindler: Functionality Classes and Evaluation Methodology for Deterministic Random Number Generators. Version 1.0, 02.12.1999, ehemalige mathematisch technische Anlage zur AIS20, https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Zertifierung/Interpretation/AIS20_Functionality_Classes_Evaluation_Methodology_DRNG.pdf?__blob=publicationFile
[AIS-20]	AIS 20: Funktionalitätsklassen und Evaluationsmethodologie für deterministische Zufallszahlengeneratoren, Version 3, 15.05.2013, https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Zertifizierung/Interpretationen/AIS_20_pdf.pdf?__blob=publicationFile
[AIS-31]	AIS 31: Funktionalitätsklassen und Evaluationsmethodologie für physikalische Zufallszahlengeneratoren, Version 3, 15.05.2013, https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Zertifizierung/Interpretationen/AIS_31_pdf.pdf?__blob=publicationFile
[ALGCAT]	Bekanntmachung zur elektronischen Signatur nach dem Signaturgesetz und der Signaturverordnung (Übersicht über geeignete Algorithmen), Bundesnetzagentur für Elektrizität, Gas, Telekommunikation, Post und Eisenbahnen, vom 13.01.2014 (auch online verfügbar: https://www.bundesnetzagentur.de/SharedDocs/Downloads/DE/Sachgebiete/QES/Veroeffentlichungen/Algorithmen/2014Algorithmenkatalog.pdf)
[ANSI-X9.31]	National Institute of Standards and Technology, NIST-Recommended Random Number Generator Based on ANSI X9.31 Appendix A.2.4 Using the 3-Key Triple DES and AES Algorithms, January 31, 2005. http://csrc.nist.gov/groups/STM/cavp/documents/rng/931rngext.pdf
[BrainPool]	ECC Brainpool Standard Curves and Curve Generation v. 1.0 19.10.2005 http://www.teletrust.de/fileadmin/files/oid/oid_ECC-Brainpool-Standard-

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
	curves-V1.pdf
[Breaking-TLS]	Lucky Thirteen: Breaking the TLS and DTLS Record Protocols Nadhem J. AlFardan and Kenneth G. Paterson Information Security Group, Royal Holloway, University of London, Egham, Surrey TW20 0EX, UK, 6th February 2013
[BreakingXMLEnc]	How to Break XML Encryption, Tibor Jager, Juraj Somorovsky, 2011 http://www.nds.rub.de/media/nds/veroeffentlichungen/2011/10/22/HowToBreakXMLenc.pdf
[BSI-CC-PP-046]	BSI (in Zertifizierung): Common Criteria Schutzprofil (Protection Profile) für einen Konnektor im elektronischen Gesundheitswesen Schutzprofil 2: Anforderungen an den Gesamtkonnektor und den darin enthaltenen AK-EB (AK-EB-PP), BSI-CC-PP-0046
[BSI-TR-02102-1]	BSI TR-02102-1 Technische Richtlinie „Kryptographische Verfahren: Empfehlungen und Schlüssellängen“ Version 2014-01, Stand 10.02.2014 https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR02102/BSI-TR-02102_pdf.html
[BSI-TR-03116-1]	Technische Richtlinie BSI TR-03116-1 Kryptographische Vorgaben für Projekte der Bundesregierung, Version: 3.18 vom 30.01.2014 https://www.bsi.bund.de/DE/Publikationen/TechnischeRichtlinien/tr03116/index_htm.html
[CM-2014]	20 Years of SSL/TLS Research, An Analysis of the Internet's Security Foundation, Christopher Meyer, 9. February 2014 http://www-brs.ub.ruhr-uni-bochum.de/netahtml/HSS/Diss/MeyerChristopher/diss.pdf
[EN-14890-1]	DIN EN 14890-1:2008 Application Interface for smart cards used as Secure Signature Creation Devices - Part 1: Basic services
[ETSI-CAdES]	ETSI TS 101 733 V1.7.4 (2008-07), Electronic Signatures and Infrastructures (ESI); CMS Advanced Electronic Signatures (CAdES)
[ETSI-XAdES]	ETSI TS 101 903 V1.4.2 (2010-12), Electronic Signatures and Infrastructures (ESI); XML Advanced Electronic Signatures (XAdES)
[FIPS-180-4]	Federal Information, Processing Standards Publication 180-4, Secure Hash Standard (SHS), March 2012 http://csrc.nist.gov/publications/fips/fips180-4/fips180-4.pdf
[FIPS-186-2+CN1]	FIPS 186-2 - National Institute of Standards and Technology, Digital Signature Standard (DSS), Federal Information Processing Standards Publication 186-2, January 27, 2000 – Appendix 3.1 unter der Beachtung des Change Notice 1, vom 5. Oktober 2001 http://csrc.nist.gov/publications/fips/archive/fips186-2/fips186-2-change1.pdf
[FIPS-197]	Federal Information Processing Standards Publication 197, (FIPS–197), November 26, 2001, Announcing the ADVANCED ENCRYPTION STANDARD (AES) http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf
[ISO-11770]	ISO/IEC 11770: 1996, Information technology – Security techniques – Key

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
	management, Part 3: Mechanisms using asymmetric techniques
[ISO-9796-2]	ISO/IEC 9796-2: Information technology – Security techniques – Digital Signature schemes giving message recovery – Part 2: Integer Factorization based mechanisms, 2010.
[Ker-1883]	Auguste Kerckhoffs, "La cryptographie militaire", Journal des sciences militaires, vol. IX, Seite 5–83, Jan. 1883, Seite 161–191, Feb. 1883. siehe auch http://www.petitcolas.net/fabien/kerckhoffs/
[KS-2011]	W. Killmann, W. Schindler, „A proposal for: Functionality classes for random number generators“, Version 2.0, September 2011 https://www.bsi.bund.de/SharedDocs/Downloads/EN/BSI/Zertifierung/Interpretation/AIS31_Functionality_classes_for_random_number_generators.pdf?__blob=publicationFile
[NIST-SP-800-38A]	NIST Special Publication 800-38A, Recommendation for Block, Cipher Modes of Operation, Methods and Techniques, Morris Dworkin, December 2001 Edition, http://csrc.nist.gov/publications/nistpubs/800-38a/sp800-38a.pdf
[NIST-SP-800-38B]	NIST Special Publication 800-38B, Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication, Morris Dworkin, May 2005 Edition, http://csrc.nist.gov/publications/nistpubs/800-38B/SP_800-38B.pdf
[NIST-SP-800-38D]	NIST Special Publication 800-38D, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, Morris Dworkin, November, 2007
[Padding-Oracle-2005]	Padding Oracle Attacks on CBC-mode Encryption with Secret and Random IVs Arnold K. L. Yau, Kenneth G. Paterson and Chris J. Mitchell, FSE 2005 http://www.isg.rhul.ac.uk/~kp/secretIV.pdf
[PAdES-3]	ETSI TS 102 778-3 V1.2.1, PDF Advanced Electronic Signature Profiles; Part 3: PAdES Enhanced – PAdES-BES and PAdES-EPES Profiles Technical Specification, 2010
[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)
[RFC-2119]	RFC 2119 (März 1997): Key words for use in RFCs to Indicate Requirement Levels, S. Bradner, http://tools.ietf.org/html/rfc2119
[RFC-2590]	RFC 2590 (June 1999): X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP https://tools.ietf.org/html/rfc2560 (Obsoleted by [RFC-6960])
[RFC-3447], [PKCS#1]	"Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1", February 2003 https://tools.ietf.org/html/rfc3447
[RFC-3526]	RFC 3526 (Mai 2003: More Modular Exponential (MODP) Diffie-Hellman groups for Internet Key Exchange (IKE)

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
	http://tools.ietf.org/html/rfc3526
[RFC-4051]	Additional XML Security Uniform Resource Identifiers (URIs), April 2005 https://tools.ietf.org/html/rfc4051
[RFC-4635]	RFC 4635 (August 2006): HMAC SHA TSIG Algorithm Identifiers http://tools.ietf.org/html/rfc4635
[RFC-5077]	Transport Layer Security (TLS) Session Resumption without Server-Side State, January 2008, https://tools.ietf.org/html/rfc5246
[RFC-5084]	RFC 5084: Using AES-CCM and AES-GCM Authenticated Encryption in the Cryptographic Message Syntax (CMS), November 2007 https://tools.ietf.org/html/rfc5084
[RFC-5246]	The Transport Layer Security (TLS) Protocol Version 1.2, August 2008, https://tools.ietf.org/html/rfc5246
[RFC-5280]	RFC 5280: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, Mai 2008 https://tools.ietf.org/html/rfc5280
[RFC-5639]	RFC 5639 (March 2010): Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation http://www.ietf.org/rfc/rfc5639.txt
[RFC-5652]	RFC 5652 (September 2009): Cryptographic Message Syntax (CMS), R. Housley, http://tools.ietf.org/html/rfc5652
[RFC-5702]	RFC 5702 (October 2009): Use of SHA-2 Algorithms with RSA in DNSKEY and RRSIG Resource Records for DNSSEC http://tools.ietf.org/html/rfc5702
[RFC-5746]	Transport Layer Security (TLS) Renegotiation Indication Extension, February 2010, https://tools.ietf.org/html/rfc5246
[RFC-5996]	RFC 5996 (September 2010): Internet Key Exchange Protocol Version 2 (IKEv2), https://tools.ietf.org/html/rfc5996
[RFC-6931]	RFC 6931: Additional XML Security Uniform Resource Identifiers (URIs), Donald Eastlake, April 2013, https://tools.ietf.org/html/rfc6931
[RFC-6960]	RFC 6960 (June 2013): X.509 Internet Public Key Infrastructure Online Certificate Status Protocol – OCSP https://tools.ietf.org/html/rfc6960
[SigV]	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)
[TLS-Attacks]	Lessons Learned From Previous SSL/TLS Attacks - A Brief Chronology Of Attacks And Weaknesses, Christopher Meyer und Jörg Schwenk, 31. Januar 2013, http://eprint.iacr.org/2013/049
[XMLCan_V1.0]	Exclusive XML Canonicalization, Version 1.0 W3C Recommendation 18 July 2002 http://www.w3.org/TR/xml-exc-c14n/
[XMLDSig]	XML Signature Syntax and Processing (Second Edition) W3C Recommendation 10 June 2008

[Quelle]	Herausgeber (Erscheinungsdatum): Titel
	http://www.w3.org/TR/2008/PER-xmldsig-core-20080326/
[XMLDSig-Draft]	XML Signature Syntax and Processing Version 2.0 W3C Editor's Draft 04 February 2014 http://www.w3.org/2008/xmlsec/Drafts/xmldsig-core-20/
[XMLDSig-RSA-PSS]	RSA-PSS in XMLDSig, 25/26 September 2007 Konrad Lanz, Dieter Bratko, Peter Lipp http://www.w3.org/2007/xmlsec/ws/papers/08-lanz-iaik/
[XMLEnc]	XML Encryption Syntax and Processing W3C Recommendation 10 December 2002 http://www.w3.org/TR/xmlenc-core/
[XMLEnc-CM]	Technical Analysis of Countermeasures against Attack on XML Encryption - or - Just Another Motivation for Authenticated Encryption. Juraj Somorovsky, Jörg Schwenk. 2011 http://www.w3.org/2008/xmlsec/papers/xmlEncCountermeasuresW3C.pdf
[XMLEnc-1.1]	XML Encryption Syntax and Processing W3C Recommendation 11 April 2013 http://www.w3.org/TR/xmlenc-core1/
[XSpRES]	XML Spoofing Resistant Electronic Signature (XSpRES) -- Sichere Implementierung für XML-Signaturen Bundesamt für Sicherheit in der Informationstechnik 2012 https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/SOA/XSpRESS.pdf?__blob=publicationFile
[XSW-Attack]	On Breaking SAML: Be Whoever You Want to Be Juraj Somorovsky, Andreas Mayer, Jörg Schwenk, Marco Kampmann, Meiko Jensen, Usenix 2012 http://www.nds.rub.de/media/nds/veroeffentlichungen/2012/08/03/BreakingSAML.pdf
[Vaudenay-2002]	Security Flaws Induced by CBC Padding: Applications to SSL, IPsec, WTLS … , Serge Vaudenay, Eurocrypt 2002, LNCS 2332/2002, 535-545 https://www.iacr.org/cryptodb/data/paper.php?pubkey=2850