Bundesamt
für Sicherheit in der
Informationstechnik

Technische Richtlinie BSI TR-03116

Kryptographische Vorgaben für Projekte der Bundesregierung

Teil 3: Intelligente Messsysteme

Stand 2017

Datum: 23. Januar 2017

Bundesamt für Sicherheit in der Informationstechnik
Postfach 20 03 63
53133 Bonn
Tel.: +49 22899 9582-0
E-Mail: smartmeter@bsi.bund.de
Internet: https://www.bsi.bund.de

Inhaltsverzeichnis

Tabellenverzeichnis

1 Einleitung

Die Technische Richtlinie BSI TR-03116 stellt eine Vorgabe für Projekte des Bundes dar. Die Technische Richtlinie ist in vier Teile gegliedert:

- Teil 1 der Technischen Richtlinie beschreibt die Sicherheitsanforderungen für den Einsatz kryptographischer Verfahren im Gesundheitswesen für die elektronische Gesundheitskarte (eGK), den Heilberufeausweis (HBA) und der technischen Komponenten der Telematikinfrastruktur.
- Teil 2 der Technischen Richtlinie beschreibt die Sicherheitsanforderungen für den Einsatz kryptographischer Verfahren in hoheitlichen Ausweisdokumenten, zur Zeit für den elektronischen Reisepass, den elektronischen Personalausweis, den elektronischen Aufenthaltstitel und den Ankunftsnachweis.
- Im vorliegenden Teil 3 werden die Sicherheitsanforderungen für den Einsatz kryptographischer Verfahren in die Infrastruktur intelligenter Messsysteme im Energiesektor beschrieben.
- Teil 4 der Technischen Richtlinie beschreibt die Sicherheitsanforderungen für den Einsatz der Kommunikationsverfahren SSL/TLS, S/MIME, SAML und OpenPGP in Anwendungen des Bundes.

Die Vorgaben des vorliegenden Teil 3 der Technischen Richtlinie basieren auf Prognosen über die Sicherheit der verwendeten kryptographischen Verfahren und Schlüssellängen über einen Zeitraum von 7 Jahren, zur Zeit bis einschließlich 2023. Eine weitere Verwendung des Verfahrens über diesen Zeitraum hinaus ist nicht ausgeschlossen und wird mit 2023+ gekennzeichnet.

Anforderungen als Ausdruck normativer Festlegungen werden durch die in Großbuchstaben geschriebenen deutschen Schlüsselworte MUSS/MÜSSEN, DARF NICHT/DÜRFEN NICHT, SOLLTE/SOLLTEN, SOLLTE NICHT/SOLLTEN NICHT, EMPFOHLEN, KANN/KÖNNEN, und OPTIONAL entsprechend [RFC2119], gekennzeichnet.

1.1 Smart Meter Gateways

Die Anforderungen an die Funktionalität, Interoperabilität und Sicherheit der Komponenten von Smart-Metering-Systemen werden in der Technischen Richtlinie TR-03109 [1] spezifiziert.

Basierend auf den Technischen Richtlinien TR-02102-1 [2], TR-02102-2 [3] und TR-03111 [4] werden in diesem Dokument verbindlich die einzusetzenden kryptographischen Verfahren und Primitive sowie zu verwendenden Schlüssellängen für die Absicherung der Infrastruktur von Messsystemen vorgegeben.

Tabelle 1 gibt einen Überblick über die verwendeten Verfahren und ihren Einsatzzweck.

Einsatzzweck	***Verfahren***
Sicherstellung der Authentizität von öffentlichen Schlüsseln (vgl. [5])	Public Key Infrastruktur (vgl. Abschnitt 3)
Absicherung der Kommunikation zwischen Kommunikationspartnern im WAN auf Transportebene (vgl. [1])	TLS (vgl. Abschnitt 4)
Absicherung der Kommunikation zwischen Smart Meter Gateway und Teilnehmern im HAN (vgl. [1])	TLS (vgl. Abschnitt 5)
Absicherung der Kommunikation von Zählern mit dem Smart Meter Gateway im LMN (vgl. [1])	TLS (vgl. Abschnitt 6)

Einsatzzweck	*Verfahren*
Absicherung der Kommunikation von Zählern mit dem Smart Meter Gateway im LMN für Daten mit niedrigem Schutzbedarf (vgl. [1])	Symmetrische Kryptographie (vgl Abschnitt 7)
Vertrauliche, authentische Ende-zu-Ende-Übertragung von Daten über das WAN an den Endempfänger (vgl. auch [1])	Inhaltsdatenverschlüsselung, MAC-Sicherung und Inhaltsdatensignatur (vgl. Abschnitt 8)
Gegenseitige Authentisierung zwischen Smart-Meter Gateway und Sicherheitsmodul sowie Aufbau eines sicheren Kanals	PACE (vgl. Abschnitt 9)
Vertrauliche, authentische Kommunikation zwischen Smart-Meter Gateway und Sicherheitsmodul	Secure Messaging (vgl. Abschnitt 9)

Tabelle 1: Verfahren zur Absicherung der Infrastruktur von Messsystemen

2 Kryptographische Algorithmen

2.1 Kryptographische Basisverfahren

Tabelle 2 gibt eine Übersicht über die kryptographischen Primitive, die in diesem Dokument verwendet werden.

Digitale Signatur	ECDSA [4]
Schlüsseleinigung	ECKA-DH [4]
Schlüsseltransport	ECKA-EG [4]
Blockchiffre	AES [6] • CBC-Mode [7] • CMAC-Mode [8] • GCM-Mode [9]
Hashfunktionen	SHA-2 Familie [10]

Tabelle 2: Kryptographische Primitive

2.2 Domainparameter für Elliptische Kurven

Für kryptographische Algorithmen und Protokolle basierend auf Elliptischen Kurven (d.h. TLS, ECDSA und ECKA) werden NIST-Domain-Parameter über Primkörpern [11] bzw. Brainpool-Domain-Parameter [12] in den entsprechenden Bitlängen verwendet.

Um eine einfache Migration auf andere Verfahren zu ermöglichen, MUSS das Sicherheitsmodul eines Smart-Meter-Gateways

- die Schlüsselerzeugung
- PACE, ECKA-DH, ECKA-EG, ECDSA Signaturerzeugung und -verifikation

gemäß den Vorgaben in [4] für alle Domain-Parameter aus Tabelle 3 unterstützen. Die Vorgaben beziehen sich auf die Herstellung des Smart Meter Gateways.

EC-Domain-Parameter	***Verwendung von***	***Verwendung bis***
BrainpoolP256r1 [12]	2015	2023+
BrainpoolP384r1 [12]	2015	2023+
BrainpoolP512r1 [12]	2015	2023+
NIST P-256 (secp256r1) [11]	2015	2023+
NIST P-384 (secp384r1) [11]	2015	2023+

Tabelle 3: Vom Sicherheitsmodul zu unterstützende Domain-Parameter

Als Encoding für die Punkte der elliptischen Kurven MUSS das Uncompressed Encoding gemäß [4] verwendet werden.

2.3 Zufallszahlen

Für die Erzeugung von Zufallszahlen und kryptographischen Schlüsseln (inkl. Ephemerschlüsseln) MUSS in jedem der verwendeten kryptographischen Protokolle ein Zufallszahlengenerator aus einer der folgenden Klassen (siehe [13]) verwendet werden:

- DRG.3,
- DRG.4,
- PTG.3,
- NTG.1.

Bei der Erzeugung von unvorhersagbaren Initialisierungsvektoren für symmetrische Verschlüsselungsverfahren im CBC-Mode MÜSSEN die Anforderungen aus [2] beachtet werden, sofern nicht explizit Abweichendes genannt wird.

2.4 Umgang mit Ephemerschlüsseln

Ephemerschlüssel und Sitzungsschlüssel MÜSSEN nach ihrer Verwendung unwiderruflich gelöscht werden. Ephemer- bzw. Sitzungsschlüssel DÜRFEN NICHT für mehr als *eine* Sitzung benutzt werden oder persistent abgespeichert werden. Dies gilt insbesondere für Master-Secret und Pre-Master-Secret bei TLS (vgl. Kap. 4) sowie Content bzw. Key Encryption Keys bei CMS (vgl Kap. 8).

3 Public Key Infrastruktur

Die Authentizität der öffentlichen Schlüssel von Kommunikationspartnern im WAN, welche im WAN zur gegenseitigen Authentisierung und zum Aufbau eines verschlüsselten, integritätsgesicherten TLS-Kanals bzw. zur Verschlüsselung oder Signatur von Daten auf Inhaltsebene eingesetzt werden, wird durch die Smart Metering Public Key Infrastruktur (SM-PKI) sichergestellt. Die SM-PKI wird in [5] und [14] spezifiziert.

Die SM-PKI besteht aus einer *Root-CA* als nationale Wurzelinstanz, *Sub-CAs* für die Ausstellung der Endnutzerzertifikate sowie den *Endnutzerzertifikaten* und wird in [5] spezifiziert. Zu den Endnutzern gehören insbesondere die Marktteilnehmer, die Gateway-Administratoren und die Smart Meter Gateways (vgl. [14]).

Als Signaturverfahren, mit dem die X.509-Zertifikate und -Sperrlisten signiert werden, MUSS das Verfahren ECDSA gemäß [4], 5.2.2 verwendet werden.

Tabelle 4 enthält die Hashfunktionen und Kurvenparameter, die von CAs für die Ausstellung von Zertifikaten verwendet werden MÜSSEN. Die Verwendungszeiträume beziehen sich auf die Erstellung der Zertifikate.

Verfahren/Parameter	***Vorgaben***	***Verwendung von***	***Verwendung bis***
Root-CA			
Signaturalgorithmus	`ecdsa-with-SHA384` [4]	2015	2023+
EC-Domain-Parameter	brainpoolP384r1 [12]	2015	2023+
Sub-CAs			
Signaturalgorithmus	`ecdsa-with-SHA256` [4]	2015	2023+
EC-Domain-Parameter	brainpoolP256r1 [12]	2015	2023+

Tabelle 4: Signatur der Zertifikate

Die Laufzeiten der Zertifikate werden in [5] verbindlich vorgegeben.

4 TLS-Kommunikation im WAN

Das TLS-Protokoll dient im WAN zum Aufbau eines verschlüsselten/integritätsgesicherten und gegenseitig authentisierten Kanals zwischen Kommunikationspartnern der Smart-Meter-PKI. Dieses Kapitel legt die hierbei einzusetzenden kryptographischen Parameter verbindlich fest..

4.1 Allgemeine Vorgaben

4.1.1 TLS-Version und Sessions

Das TLS-Protokoll MUSS nach Version 1.2 [15] implementiert werden. Ein Fallback auf eine ältere TLS-Version DARF NICHT möglich sein. Das Smart Meter Gateway SOLLTE beim TLS-Handshake, im Client-Hello, anstelle der eigenen Zeit, eine Zufallszahl als `gmt_unix_time` verwenden.

Kommunikationspartner im WAN MÜSSEN eine TLS-Session (inklusive eventueller Session Resumptions) auf einen Wert begrenzen, der 48 Stunden nicht überschreitet. Beim Smart Meter Gateway SOLLTE dieser Wert durch den Gateway Administrator konfigurierbar sein. Insbesondere MUSS das Smart Meter Gateway bestehende TLS-Verbindungen nach Ablauf dieser Zeit beenden und für eine neue Verbindung einen neuen TLS-Handshake durchführen.

Innerhalb der erlaubten Session-Lebensdauer KANN Session-Resumption verwendet werden. Hierbei KANN eine Stateless Resumption nach [16] unterstützt und genutzt werden. In diesem Falle MÜSSEN die Anforderungen aus [16], insbesondere Kap. 4 und 5, beachtet werden. Die Server-Schlüssel für die Verschlüsselung und Sicherung der Authentizität von Tickets für eine Session-Resumption MÜSSEN vom Server sicher gespeichert, verarbeitet und regelmäßig gewechselt werden. Nach Ablauf der Nutzungszeit MÜSSEN die Schlüssel unverzüglich vernichtet werden.

Session Renegotiation DARF NICHT möglich sein.

4.2 Cipher Suites und Kurvenparameter

Die TLS-Implementierung MUSS gemäß [17] mit ephemerem ECDH erfolgen. Dabei stehen grundsätzlich folgende Cipher Suites zur Verfügung:

- `TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256`
- `TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384`
- `TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256`
- `TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384`

Die Wahl der Cipher Suite legt folgende Bereiche des TLS-Protokolls fest:

- Schlüsselaustausch
- Authentifizierung
- Hashfunktion
- Verschlüsselung und Message Authentication Code (MAC)

Tabelle 5 enthält die Cipher Suites und elliptischen Kurven, die für die TLS-Kommunikation von Kommunikationspartnern im WAN mindestens unterstützt werden MÜSSEN.

Vorgaben		*Verwendung von*	*Verwendung bis*
Cipher Suite	`TLS_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256`	2015	2023+
EC-Parameter	NIST P-256 (secp256r1) [11] BrainpoolP256r1 [18]	2015 2015	2023+ 2023+

Tabelle 5: Mindestens zu unterstützende Verfahren

Um eine langfristige Nutzung zu ermöglichen, SOLLTEN für TLS zusätzlich auch die in Tabelle 6 genannten Cipher Suites und elliptischen Kurven unterstützt werden.

Vorgaben	*Verwendung von*	*Verwendung bis*
Cipher Suites		
`TLS_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384`	2015	2023+
`TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256`	2015	2023+
`TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384`	2015	2023+
Elliptische Kurven		
BrainpoolP384r1 [18]	2015	2023+
BrainpoolP512r1 [18]	2015	2023+
NIST P-384 (secp384r1) [11]	2015	2023+

Tabelle 6: Zusätzlich Verfahren, deren Unterstützung empfohlen wird

Die unterstützten Verfahren und Kurven MÜSSEN vom Client hierbei in den entsprechenden Datenfeldern[1] angezeigt werden.

Andere Cipher Suites oder elliptische Kurven, als die in Tabelle 5 oder Tabelle 6 genannten, DÜRFEN NICHT für die Kommunikation im WAN unterstützt werden.

4.3 Authentifizierung und TLS-Zertifikate

Zur gegenseitigen Authentifizierung, benötigt jede Partei ein TLS-Zertifikat aus der SM-PKI für ein Schlüsselpaar, das zur Erzeugung von Signaturen mit ECDSA (gemäß [4]) geeignet ist.

Hierbei MÜSSEN die Kurvenparameter aus Tabelle 7 verwendet werden. Der Verwendungszeitraum bezieht sich auf die Erstellung der Zertifikate.

1 Vgl. [15], Kap. 7.4.1.2 (`ClientHello` Message), und [19], Kap 5.1.1 (`supported_elliptic_curves` Extension).

Verfahren	*Vorgaben*	*Verwendung von*	*Verwendung bis*
EC-Domain-Parameter	BrainpoolP256r1 [12]	2015	2023+

Tabelle 7: Kurvenparameter in TLS-Zertifikaten

4.4 Weitere Vorgaben und Empfehlungen

4.4.1 Signaturalgorithmen

Digitale Signaturen während des TLS Handshakes MÜSSEN mit ECDSA erstellt werden. Hierbei MÜSSEN die Hashfunktionen aus Tabelle 8 unterstützt werden.

Vorgaben	*Verwendung von*	*Verwendung bis*
SHA-256	2015	2023+
SHA-384	2015	2023+

Tabelle 8: Verpflichtend zu unterstützende Hashfunktionen für Signaturen während des TLS-Handshakes

Zusätzlich SOLLTEN die Hashfunktionen aus Tabelle 9 unterstützt werden.

Vorgaben	*Verwendung von*	*Verwendung bis*
SHA-512	2015	2023+

Tabelle 9: Optional zu unterstützende Hashfunktionen für Signaturen während des TLS-Handshakes

Die unterstützten Verfahren MÜSSEN dem Kommunikationspartner dabei in den entsprechenden Datenfeldern[2] angezeigt werden. Andere als die in Tabelle 8 und Tabelle 9 angegebenen Hashfunktionen DÜRFEN NICHT verwendet werden.

4.4.2 Extensions

Für Extensions gelten ergänzend zu [20] folgende Regeln:

- Eine Verkürzung der Ausgabe des HMAC DARF NICHT verwendet bzw. akzeptiert werden[3].
- Im Allgemeinen werden bei TLS gemäß [15] Klartextdaten zunächst integritätsgesichert (MAC) und anschließend werden Klartext und MAC verschlüsselt (MAC-then-Encrypt). Grundsätzlich ist aber die Verwendung von Encrypt-then-MAC oder Authenticated Encryption vorzuziehen ([22]). Daher wird die Verwendung von Encrypt-then-MAC gemäß [23] EMPFOHLEN, d.h.
  - Clients SOLLTEN die Encrypt-then-MAC-Extension im Client-Hello anbieten und
  - Server SOLLTEN entweder eine GCM Cipher Suite auswählen oder die Encrypt-then-MAC-Extension im Server-Hello verwenden.

2 Vgl. hierzu [15], Kap 7.4 (`ServerKeyExchange`, `CertificateVerify`, `CertificateRequest` Messages und `supported_signature_Algorithms` Extension)

3 Vgl. hierzu [21], Kap.7 (`truncated_hmac`-Extension).

- Im Allgemeinen erfolgt bei TLS gemäß [15] die Berechnung des Master Secrets so, dass nicht alle kryptographischen Parameter aus dem TLS-Handshake in die Berechnung einbezogen werden. Je nach verwendeten kryptographischen Parametern kann die fehlende Einbeziehung dieser Daten zu Angriffen auf eine TLS-Session führen (vgl. etwa Triple-Handshake-Angriff [24]). Auch grundsätzlich ist es empfohlen, kontextspezifische Daten in die Berechnung von Session-Schlüsseln einzubeziehen. Daher SOLLTEN Kommunikationspartner im WAN die Extended Master Secret Extension gemäß [25] verwenden. Hierbei fließen die kryptographischen Parameter in Form eines *Session Hash* (Hashwert über alle Nachrichten des TLS-Handshakes) in die Berechnung des Master Secrets ein.

5 TLS-Kommunikation im HAN

Das TLS-Protokoll dient im HAN zum Aufbau eines authentisierten sicheren Kanals zwischen Smart Meter Gateway und Komponenten im HAN (wie etwa die Anzeigeeinheit oder CLS-Systeme) (vgl. [20]).

5.1 Allgemeine Vorgaben

Das TLS-Protokoll MUSS nach Version 1.2 [15] implementiert werden. Ein Fallback auf eine ältere TLS-Version DARF NICHT möglich sein.

Das Smart Meter Gateway MUSS eine TLS-Session (inklusive eventueller Session Resumptions) auf einen Wert begrenzen, der 48 Stunden nicht überschreitet. Dieser Wert SOLLTE durch den Gateway Administrator konfigurierbar sein. Insbesondere MUSS das Smart Meter Gateway bestehende TLS-Verbindungen mit Ablauf dieser Zeit beenden und für eine neue Verbindung einen neuen TLS-Handshake durchführen.

Innerhalb der erlaubten Session-Lebensdauer KANN Session-Resumption verwendet werden. Im Falle einer Stateless Resumption MÜSSEN die Anforderungen aus [16], insbesondere Kap. 5, zu beachtet werden.

Session Renegotiation DARF NICHT möglich sein.

5.2 Cipher Suites und Kurvenparameter

Smart-Meter Gateway und HAN-Komponenten MÜSSEN die Anforderungen an Cipher Suites und Kurvenparameter aus Kapitel 4.2 auch für die Kommunikation im HAN einhalten.

5.3 Authentisierung und TLS-Zertifikate

Für Zertifikate im HAN MÜSSEN Kurvenparameter aus Tabelle 3 dieser Technischen Richtlinie verwendet werden. Hierbei wird grundsätzlich die Verwendung von Brainpool-Kurven empfohlen.

Die maximalen Zertifikatslaufzeiten hängen vom konkreten Umsetzungsszenario (PKI-basiert vs. selbst-signiert) ab und werden von [20] festgelegt.

5.3.1 Signaturalgorithmen

Smart Meter Gateways und Komponenten im HAN MÜSSEN bzgl. Unterstützung und Verwendung von Signaturalgorithmen währen des TLS-Handshakes die Anforderungen aus Kapitel 4.4.1 auch für die Kommunikation im HAN einhalten.

5.3.2 Extensions

Smart Meter Gateways und HAN Komponenten MÜSSEN bzgl. der Unterstützung und Verwendung von Extensions die Anforderungen aus Kapitel 4.4.2 auch für die Kommunikation im HAN einhalten.

Für Extensions gelten ergänzend zu [20] folgende Regeln:

- Eine Verkürzung der Ausgabe des HMAC DARF NICHT verwendet bzw. akzeptiert werden[4].

4 Vgl. hierzu [21], Kap.7 (`truncated_hmac`-Extension).

- Im Allgemeinen werden bei TLS gemäß [15] Klartextdaten zunächst integritätsgesichert (MAC) und anschließend werden Klartext und MAC verschlüsselt (MAC-then-Encrypt). Grundsätzlich ist aber die Verwendung von Encrypt-then-MAC oder Authenticated Encryption vorzuziehen ([22]). Daher wird die Verwendung von Encrypt-then-MAC gemäß [23] EMPFOHLEN, d.h.
  - Clients SOLLTEN die Encrypt-then-MAC-Extension im Client-Hello anbieten und
  - Server SOLLTEN entweder eine GCM Cipher Suite auswählen oder die Encrypt-then-MAC-Extension im Server-Hello verwenden.
- Im Allgemeinen erfolgt bei TLS gemäß [15] die Berechnung des Master Secret ohne die direkte Einbeziehung kryptographischer Parameter aus dem TLS-Handshake. Je nach verwendeten kryptographischen Parametern kann dies zu Angriffen auf eine TLS-Session führen. Daher ist es ratsam, die Extended Master Secret Extension gemäß [25] zu unterstützen und zu verwenden. Hierbei fließen die kryptographischen Parameter in Form eines Session Hashs in die Berechnung des Master Secret ein.

5.3.3 Migration kryptographischer Verfahren und Schlüssel

Es wird EMPFOHLEN, Komponenten im HAN mit der Möglichkeit auszustatten, in Zukunft neue Schlüssel einzuspielen bzw. zu erzeugen und ggf. per Firmware-Update neue kryptographische Verfahren einzuspielen, um so eine weitere Verwendbarkeit der Komponenten auch nach einer erforderlichen Migration kryptographischer Verfahren zu ermöglichen.

6 TLS-Kommunikation im LMN

Sofern nicht die Voraussetzungen von Kap. 7 erfüllt sind, MUSS die Kommunikation zwischen Zählern und Smart Meter Gateway per TLS erfolgen. Insbesondere MUSS für die bidirektionale Kommunikation zwischen Smart Meter Gateway und Zähler mindestens in folgenden Einsatzszenarien TLS unterstützt werden:

- Wechsel des gemeinsamen, zählerindividuellen Schlüssels gemäß 7.1.1,
- Auslesen von Zählerdaten,
- Auswahl der auszulesenden Daten.

6.1 Allgemeine Vorgaben

Das TLS-Protokoll MUSS nach Version 1.2 [15] implementiert werden. Ein Fallback auf eine ältere TLS-Version DARF NICHT möglich sein.

Das Smart Meter Gateway MUSS die Länge einer TLS-Session (inklusive eventueller Session Resumptions) auf einen Wert begrenzen, der 31 Tage nicht überschreitet. Dabei DÜRFEN NICHT mehr als 5 MB (5.000.000 Bytes) an Daten innerhalb einer Session ausgetauscht werden[5]. Insbesondere MUSS das Smart Meter Gateways bestehende TLS-Verbindungen mit Ablauf dieser Zeit beenden und für eine neue Verbindung einen neuen TLS-Handshake erzwingen.

Innerhalb der erlaubten Session-Lebensdauer KANN Session-Resumption verwendet werden. Im Falle einer Stateless Resumption MÜSSEN die Anforderungen aus [16], insbesondere Kap. 5, beachtet werden.

Session Renegotiation DARF NICHT möglich sein.

6.2 Cipher Suites und Kurvenparameter

Smart-Meter Gateway und TLS-Zähler MÜSSEN die Anforderungen an Cipher Suites und Kurvenparameter aus Kapitel 4.2 auch für die Kommunikation im LMN einhalten.

6.3 Authentisierung und TLS-Zertifikate

Die Zertifikate für die Kommunikation zwischen Smart Meter Gateway und TLS-Zähler sind selbst-signiert [20]. Insbesondere, MUSS das Smart Meter Gateway also für die TLS-Kommunikation im WAN und im LMN separate Zertifikate verwenden.

Hierbei MÜSSEN die Kurvenparameter aus Tabelle 7 verwendet werden. Der Verwendungszeitraum bezieht sich auf die Erstellung der Zertifikate.

Für Zertifikate im LMN MÜSSEN Kurvenparameter aus Tabelle 3 dieser Technischen Richtlinie verwendet werden. Hierbei wird die Verwendung von Brainpool-Kurven empfohlen.

Die maximalen Zertifikatslaufzeiten MÜSSEN den Anforderungen aus Tabelle 10 entsprechen.

5 Die Datenmenge bezieht sich auf das Gesamtvolumen der ausgetauschten Nachrichten (ohne die Nachrichten des TLS-Handshakes).

Zertifikat	*Gültigkeitszeit*	*Private Key Usage*
Zählerzertifikat	Maximal 7 Jahre	Maximal 7 Jahre
SMGW-Zertifikat	Maximal 7 Jahre	Maximal 7 Jahre

Tabelle 10: Laufzeiten der LMN-Zertifikate

6.3.1 Initialer Austausch und Update der LMN-Zertifikate

Für Zähler, die TLS unterstützen, MUSS das initiale Schlüsselpaar vom Hersteller oder vom Smart Meter Gateway erzeugt und authentisch in den Zähler eingebracht werden.

Wird das initiale Schlüsselpaar von Hersteller erzeugt und in den Zähler eingebracht, so MUSS bei erstmaligem Anschluss an ein neues Smart Meter Gateway ein authentischer Austausch der TLS-Zertifikate mit dem Smart Meter Gateway erfolgen.

Wird das initiale Schlüsselpaar vom Smart Meter Gateway erzeugt, so MUSS dieses zusammen mit den Zertifikaten des Smart Meter Gateways an den Zähler verschlüsselt und authentisch an den Zähler übertragen werden.

Die initiale Übertragung der Zertifikate bzw. die Einbringung des initialen Schlüsselmaterials vom Smart Meter Gateway auf dem Zähler MUSS mit dem in Kapitel 7 beschriebenen symmetrischen Verfahren erfolgen[6].

Unmittelbar nach Austausch/Einbringung der TLS-Zertifikate MUSS ein TLS-Kanal aufgebaut und der zählerindividuelle Schlüssel für die Kommunikation auf Basis symmetrischer Kryptographie gemäß den Vorgaben von Kap. 7.1.1 gewechselt werden.

Für das Update eines Smart-Meter-Gateway-Zertifikats MUSS ein neues Schlüsselpaar erzeugt werden und anschließend MUSS das neue selbst-signierte Zertifikat über den aufgebauten TLS-Kanal an den Zähler gesendet werden.

Für das Update eines Zähler-Zertifikats MUSS das Smart-Meter-Gateway ein neues Schlüsselpaar erzeugen und anschließend MUSS das neue selbst-signierte Zertifikat mit dem zugehörigen privaten Schlüssel über den aufgebauten TLS-Kanal an den Zähler gesendet werden.

6.4 Weitere Vorgaben und Empfehlungen

6.4.1 Signaturalgorithmen

Smart Meter Gateways und TLS-Zähler MÜSSEN bzgl. Unterstützung und Verwendung von Signaturalgorithmen währen des TLS-Handshakes die Anforderungen aus Kapitel 4.4.1 auch für die Kommunikation im LMN einhalten.

6.4.2 Extensions

Für Extensions gelten ergänzend zu [20] folgende Regeln:

6 Es ist geplant, den Austausch der Zertifikate auch durch Aufbau eines verschlüsselten Kanals via Passworteingabe und PACE (siehe auch [26], [27]) zu ermöglichen.

- Im Allgemeinen werden bei TLS gemäß [15] Klartextdaten zunächst integritätsgesichert (MAC) und anschließend werden Klartext und MAC verschlüsselt (MAC-then-Encrypt). Grundsätzlich ist aber die Verwendung von Encrypt-then-MAC vorzuziehen ([22]). Daher wird die Verwendung von Encrypt-then-MAC gemäß [23] EMPFOHLEN.
- Im Allgemeinen erfolgt bei TLS gemäß [15] die Berechnung des Master Secret ohne die direkte Einbeziehung kryptographischer Parameter aus dem TLS-Handshake. Je nach verwendeten kryptographischen Parametern kann dies zu Angriffen auf eine TLS-Session führen. Daher ist es ratsam,die Extended Master Secret Extension gemäß [25] zu unterstützen und zu verwenden. Hierbei fließen die kryptographischen Parameter in Form eines Session Hashs in die Berechnung des Master Secrets ein.

6.4.3 Migration kryptographischer Verfahren und Schlüssel

Die gewünschte Verwendungszeit von TLS-Zählern kann deutlich über den Prognose-Zeitraum hinausgehen. Daher wird EMPFOHLEN, Zähler mit der Möglichkeit auszustatten, neue Schlüssel einzuspielen bzw. zu erzeugen und ggf. per Firmware-Update neue kryptographische Verfahren einzuspielen, um so eine weitere Verwendbarkeit des TLS-Zählers auch in Zukunft zu ermöglichen.

7 Kommunikation im LMN auf Basis symmetrischer Kryptographie

Für Zähler, die nur unidirektional kommunizieren können, ist die Möglichkeit von TLS nicht gegeben. Da außerdem Bandbreite und Verfügbarkeit des Kommunikationskanals im LMN auch für bidirektional kommunizierende Zähler unter Umständen starken Einschränkungen unterliegt, sind Zähler nicht immer imstande gemäß den zeitlichen Anforderungen einen TLS-Kanal mit einem Smart Meter Gateway aufzubauen.

Daher MUSS das Smart Meter Gateway diesen Zählern eine alternative Möglichkeit zum Senden von Mess- und Zähldaten bereitstellen. Diese Möglichkeit wird im Folgenden beschrieben. Diese Art der Kommunikation KANN für die Auswahl, den Abruf oder die Übertragung von Daten verwendet werden, falls eine Verbindung per TLS nicht möglich ist. Ansonsten DARF diese Art der Kommunikation NICHT verwendet werden.

7.1 Voraussetzungen

Zähler und Smart Meter Gateway verfügen über einen gemeinsamen geeigneten, symmetrischen Schlüssel *MK*. Dieser Schlüssel *MK* MUSS eine Länge von 128 Bit besitzen und für jeden Zähler individuell zufällig (gemäß den Vorgaben von Kap. 2.3) erzeugt werden. Dieser Schlüssel wird im Folgenden meist kurz zählerindividueller Schlüssel *MK* genannt. Die Erzeugung von *MK* MUSS gemäß den Anforderungen aus Kapitel 2.3

- durch den Hersteller vorgenommen werden, der *MK* in den Zähler einbringt, oder
- durch den Zähler erfolgen, der *MK* an den Hersteller ausgibt.

Vor dem Anschluss des Zählers an ein Smart Meter Gateway MUSS der Eigentümer des Zählers den initialen zählerindividuellen Schlüssel *MK* vertraulich und authentisch an den Administrator des Gateways übertragen. Der Gateway-Administrator MUSS den initialen zählerinidividuellen Schlüssel dann wie in [20] beschrieben, gesichert in das Gateway einbringen[7].

Handelt es sich bei den Zähler um einen bidirektional kommunizierenden Zähler, so MÜSSEN Zähler und Smart Meter Gateway unmittelbar nach dem Anschluss des Zählers an das Smart Meter Gateway TLS-Zertifikate austauschen und den zählerinidviduellen Schlüssel *MK* wechseln (vgl. hierzu Kapitel 6.3.1 und 7.1.1).

Jeder Zähler MUSS über einen Transmission Counter *C* mit einer Länge von 32 Bit verfügen. Erfolgt die Kommunikation zwischen Zähler und Smart Meter Gateway bidirektional, d.h. auf den Eingang eines Datensatzes erfolgt die Versendung einer Antwortnachricht, so MUSS auch das Smart Meter Gateway über einen zählerindividuellen Transmission Counter *C'* mit einer Länge von 32 Bit verfügen. Transmission Counter DÜRFEN NICHT überlaufen oder zurückgesetzt werden. Das Zurücksetzen der Transmission Counter ist ausnahmsweise nur direkt nach der Erzeugung eines neuen zählerindiviuellen Schlüssels *MK* und vor dessen erster Verwendung erlaubt.

Vor der Versendung einer Nachricht MUSS der Counterwert *C* bzw. *C'* gegenüber dem Counterwert der zuletzt authentisch empfangenen bzw. gesendeten Nachricht erhöht werden (vgl. Kap. 7.2).

7 Hierbei eingesetzte kryptographische Verfahren MÜSSEN den allgemeinen Empfehlungen aus [2] entsprechen.

7.1.1 Wechsel des gemeinsamen, zählerindividuellen Schlüssels *MK* für bidirektionale Zähler

Alle Zähler, die bidirektional kommunizieren können, MÜSSEN mindestens einmal innerhalb von 2 Jahren erfolgreich einen TLS-Kanal aufbauen, um den gemeinsamen, für jeden Zähler individuell zufällig erzeugten Schlüssel *MK* für das symmetrische Verfahren zu wechseln.

Zur Berechnung des neuen zählerindividuellen Schlüssels *MK'* MUSS das Smart Meter Gateway eine Zufallszahl z_1 von 128 Bit erzeugen und diese innerhalb des TLS-Kanals an den Zähler senden. Der Schlüssel ist dann $MK'=MAC(MK, z_1)$. Optional KANN zusätzlich der Zähler eine Zufallszahl z_2 von 128 Bit erzeugen und diese an das Smart Meter Gateway übertragen. Der neue gemeinsame, zählerindividuelle Schlüssel wird dann als $MK'=MAC(MK, z_1||z_2)$ gesetzt. Hierbei MUSS der MAC-Algorithmus aus Tabelle 11 verwendet werden.

7.2 Schlüsselableitung

Vor jeder Übertragung eines neuen Datensatzes MÜSSEN aus dem Schlüssel *MK* die Schlüssel K_{Enc} (für die Verschlüsselung) und K_{MAC} (für die MAC-Berechnung) abgeleitet werden.

Die Berechnung von K_{Enc} bzw. K_{MAC} MUSS jeweils durch MAC-Bildung des aktuellen Counterwertes mit dem im Folgenden beschriebenen Verfahren unter Verwendung der Primitive aus Tabelle 11 geschehen. Hierbei MUSS stets sichergestellt werden, dass der in die Schlüsselableitung für einen zu sendenden Datensatz eingehende Counterwert größer ist als der zugehörige Counterstand der zuletzt authentisch empfangenen bzw. gesendeten Nachricht.

Verfahren	***Mode***	***Länge***	***Verwendung von***	***Verwendung bis***
Berechnung von MK' bzw. K_{Enc}, K_{MAC}, L_{Enc} *oder* L_{MAC}				
AES	CMAC gemäß [8]	128	2015	2023+

Tabelle 11: Berechnung der abgeleiteten Schlüssel

Die Berechnung von K_{Enc} bzw. K_{MAC} für einen Datensatz erfolgt durch

- $K_{Enc} = MAC(MK,0x00||C||Zähler\text{-}ID)$ bzw.
- $K_{MAC} = MAC(MK,0x01||C||Zähler\text{-}ID)$,

wobei *0x00* bzw. *0x01* jeweils von der Länge 1 Byte sind. Der Input des MAC MUSS dabei vor Eingang in den MAC stets wie folgt auf Blocklänge aufgefüllt werden, wobei *l* jeweils die Bytelänge des Inputs und *I2OS()* die Konvertierungsfunktion von Integers nach Oktetts gemäß [4], Kap. 3.1.2, bezeichnet:

- Anzahl der aufzufüllenden Oktetts: $16-(l \bmod 16)$ Oktetts;
- Wert je aufzufüllender Oktett: $I2OS(16-(l \bmod 16))$.

Erfolgt die Kommunikation bidirektional, so MUSS auch das Smart Meter Gateway für die Versendung jedes Datensatzes neue Schlüssel L_{Enc} und L_{MAC} via

- $L_{Enc} = MAC(MK,0x10||C'||Zähler\text{-}ID)$ bzw.
- $L_{MAC} = MAC(MK,0x11||C'||Zähler\text{-}ID)$

und dem gleichen Padding ableiten. Hierbei MUSS das Smart Meter Gateway einen Transmission Counters *C'* verwenden, der größer ist als Counterstand des zuletzt authentisch empfangenen bzw. gesendeten Datensatzes.

7.3 Übertragung von Zählerdaten

Die Übertragung der Daten MUSS stets verschlüsselt und MAC-gesichert erfolgen. Die übertragenen Daten MÜSSEN hierbei zuerst (mit dem abgeleiteten Schlüssel K_{Enc} *bzw.* L_{Enc}) verschlüsselt und danach werden die verschlüsselten Daten (mit K_{MAC} *bzw.* L_{MAC}) MAC-gesichert werden.

Hierbei MÜSSEN die Verfahren ausTabelle 12 verwendet werden. Der Verwendungszeitraum bezieht sich auf die Herstellung des Zählers.

Verfahren	***Mode***	***Länge***	***Verwendung von***	***Verwendung bis***
Verschlüsselung (mit K_{Enc} bzw. L_{Enc})				
AES	CBC gemäß [7] (IV=0)[8]	128	2015	2023+
Authentizität und Integritätssicherung (mit K_{MAC} bzw. L_{MAC})				
AES	CMAC gemäß [8]. Der MAC-Wert kann optional auf die ersten 64 Bit gekürzt werden.	128/64	2015	2023+

Tabelle 12: Symmetrische Absicherung der Datenübertragung

Anmerkung:

1. Der Stand des Transmission Counters, der für die Ableitung der Schlüssel K_{Enc} und K_{MAC} bzw. L_{Enc} und L_{MAC} benötigt wird, MUSS unverschlüsselt aber MAC-gesichert übertragen werden. Zudem MUSS ein Zähler die Zähler-ID unverschlüsselt an das Smart Meter Gateway übertragen.
2. Die Wahl von IV=0 in der obigen Tabelle 12 ist möglich, da bei jeder Datenübertragung aus dem jeweiligen Counter insbesondere ein neuer Schlüssel K_{Enc} bzw. L_{Enc} abgeleitet wird.
3. Die Verschlüsselung und MAC-Sicherung MUSS stets über einen vollständigen Datensatz von zu schützenden Daten erfolgen. Der Transport des Datensatzes KANN in mehreren Paketen erfolgen.
4. Zur Detektion von Replay-Attacken MUSS der Empfänger einer Nachricht bei jedem empfangenen Datensatz prüfen, dass der Transmission Counter des empfangenen Datensatzes größer als der des letzten empfangenen Datensatzes ist.
5. Um Replay-Attacken auch im Falle eines Stromausfalls zu vermeiden, MUSS stets sichergestellt werden, dass auch nach einem Stromausfall des Smart Meter Gateways ein Transmission Counter *C* vorliegt, der einen Wert aufweist, der nicht kleiner als 12 Stunden vor Beginn des Stromausfalls ist.

8 Das Padding MUSS so gewählt werden, dass hierdurch keine Angriffe auf die Verschlüsselung möglich sind. Eine Möglichkeit ist das Padding aus Kap. 7.2.

8 Inhaltsdatenverschlüsselung und -signatur

Im Weitverkehrsnetz (WAN) erfolgt die Übermittlung von Daten nicht immer über einen direkten Transportkanal zwischen Sender und Endempfänger, sondern teilweise über dritte Parteien (z.B. den Gateway-Administrator). Daher geschieht der Austausch von Daten zwischen Kommunikationspartnern im WAN innerhalb eines TLS-Kanals stets auf der Basis von für den Endempfänger verschlüsselten und signierten Nachrichten im Cryptographic Message Syntax-Format gemäß [28].

Hierbei muss das im Folgenden vorgestellte Schema implementiert werden.

8.1 Authenticated-Enveloped-data Content Type

Für die Inhaltsdatenverschlüsselung MUSS der Authenticated-Enveloped-Data Content Type (vgl. [29]) unter Verwendung eines ephemer-statischem Diffie-Hellman nach den Vorgaben von [20] verwendet werden.

8.1.1 Content-Authenticated-Encryption

Die Inhaltsdaten werden symmetrisch verschlüsselt und die verschlüsselten Daten werden MAC-gesichert (Content-Authenticated-Encryption). Hierbei MÜSSEN Verfahren aus Tabelle 13 für die Verschlüsselung und die MAC-Sicherung der Inhaltsdaten verwendet werden. Alle Verfahren der Tabelle 13 MÜSSEN unterstützt werden.

Verfahren	*Vorgaben*	*Länge*	*Verwendung von*	*Verwendung bis*
AES-GCM				
Verschlüsselung und Authentizität	AES-GCM gemäß [30]	128	2015	2023+
AES-CBC-CMAC				
Verschlüsselung	AES-CBC (IV=0) gemäß [7] mit Padding gemäß [28], Abschnitt 6.3	128	2015	2023+
Authentizität	AES-CMAC gemäß [8]	128	2015	2023+

Tabelle 13: Inhaltsdatenverschlüsselung

Die symmetrischen Schlüssel für die Verschlüsselung und MAC-Sicherung der Inhaltsdaten MÜSSEN (unmittelbar vor ihrer Verwendung) zufällig erzeugt werden. Ein Schlüssel DARF NICHT für die Versendung mehrerer Nachrichten verwendet werden.

Bemerkung: Die Wahl von IV=0 in der obigen Tabelle ist möglich, da bei jeder erneuten Inhaltsdatenverschlüsselung, d.h. für jedes neue Authenticated-Enveloped-Data-Paket, die symmetrischen Schlüssel neu generiert werden.

8.1.2 Schlüsselableitung und Key Encryption

Die zufällig erzeugten Schlüssel für die Verschlüsselung und MAC-Sicherung der Inhaltsdaten sind verschlüsselt im CMS-Container enthalten (Key Encryption).

Der Schlüssel *K* für die Key Encryption MUSS per ECKA-EG [4] berechnet werden. Die Ableitung von *K* MUSS mittels der X9.63 Key Derivation Function erfolgen (vgl. [4], Kap. 4.3.3). ECKA-EG MUSS dazu gemäß [4], Kap. 4.3.2.2 bzw. 5.3.1 (OIDs mit X9.63-KDF) zu implementiert werden.

Tabelle 14 enthält die Hashfunktionen und Kurvenparameter, die für ECKA-EG verwendet werden MÜSSEN. Die Verwendungszeiträume beziehen sich auf die Erstellung des zugrundeliegenden Verschlüsselungszertifikats.

Verfahren	*Vorgaben*	*Verwendung von*	*Verwendung bis*
Hash	SHA-256	2015	2023+
EC-Domain-Parameter	BrainpoolP256r1	2015	2023+

Tabelle 14: Schlüsseltransport für die Inhaltsdatenverschlüsselung

Die Key Encryption MUSS mit dem abgeleiteten Schlüssel *K* auf Basis symmetrischer Kryptographie. Hierbei MÜSSEN die Verfahren aus Tabelle 15 verwendet werden.

Verfahren	*Vorgaben*	*Verwendung von*	*Verwendung bis*
Verschlüsselung	id-aes128-wrap [31]	2015	2023+

Tabelle 15: Algorithmen für die CMS Key Encryption

8.2 Signed-Data Content Type

Die verschlüsselten und MAC-gesicherten Inhaltsdaten (Authenticated-Enveloped-Data Content Type, siehe 8.1) müssen anschließend signiert werden. Hierzu MUSS ECDSA, implementiert nach [4], zu verwendet werden.

Tabelle 16 enthält die Hashfunktionen und Kurvenparameter, die für die Signatur verwendet werden MÜSSEN. Die Verwendungszeiträume beziehen sich auf die Erstellung des zugrundeliegenden Signaturzertifikats.

Verfahren	*Vorgaben*	*Verwendung von*	*Verwendung bis*
Hash	SHA-256	2015	2023+
EC-Domain-Parameter	BrainpoolP256r1 [12]	2015	2023+

Tabelle 16: Signatur der verschlüsselten Inhaltsdaten

9 PACE und Secure Messaging

Für den Zugriff des Smart-Meter Gateways auf das Sicherheitsmodul erfolgt eine gegenseitige Authentisierung beider Komponenten gemäß [26] mittels des PACE-Protokolls. PACE (Password Authenticated Connection Establishment) (vgl. [26] bzw. [27], [32]) ist ein passwort-basiertes Authentisierungs- und Schlüsseleinigungsverfahren, bei dem aus einer gemeinsamen PIN[9] Sitzungsschlüssel hoher Entropie für das anschließende Secure Messaging abgeleitet werden. Secure Messaging liefert einen verschlüsselten, authentisierten Kanal zwischen den Smart Meter Gateway und dem Sicherheitsmodul (vgl. [26]).

Tabelle 17 enthält die kryptographischen Verfahren sowie die Anzahl der dezimalen Zeichen der PIN, die für PACE verwendet werden MÜSSEN. Die angegebenen Verwendungszeiträume beziehen sich auf die Herstellung des Sicherheitsmoduls.

Vorgaben		***Verwendung von***	***Verwendung bis***
Algorithmus	`id-PACE-ECDH-GM-AES-CBC-CMAC-128` vgl. [26] bzw. [27], [32]	2015	2023+
EC-Domain-Parameter	BrainpoolP256r1	2015	2023+
PACE-PIN	Mindestens 10 Dezimalziffern	2015	2023+

Tabelle 17: PACE und Secure Messaging

Da die gewünschte Verwendungszeit der Komponenten eines Smart-Metering-Sytstems deutlich über den Verwendungszeitraum hinausgeht, wird (insbesondere für Komponenten ohne Update-Möglichkeit) EMPFOHLEN, zusätzlich auch die folgenden weiteren PACE-Algorithmen mit den Elliptischen Kurven der entsprechenden Bitlängen zu unterstützen:

- `id-PACE-ECDH-GM-AES-CBC-CMAC-192` vgl. [26] bzw. [27], [32]
- `id-PACE-ECDH-GM-AES-CBC-CMAC-256` vgl. [26] bzw. [27], [32]

Das Smart Meter Gateway MUSS eine Secure Messaging Session auf maximal 48 Stunden begrenzen.

9 Die PIN, die bei PACE zur Authentisierung des Smart-Meter Gateways gegenüber dem Sicherheitsmodul verwendet wird, MUSS im Smart Meter Gateway geeignet geschützt werden, vgl. [20]

10 Zertifizierung

Die Smart Meter Gateways und Sicherheitsmodule MÜSSEN nach den Common Criteria zertifiziert sein.

10.1 Smart Meter Gateway

Im Rahmen der erforderlichen Zertifizierung MUSS die Konformität des Smart Meter Gateways zum Schutzprofil BSI-CC-PP-0073 [33] nachgewiesen werden.

Das Common Criteria Zertifikat MUSS einen Hinweis enthalten, dass die Anforderungen dieser Technischen Richtlinie an das Smart Meter Gateway (siehe auch die entsprechenden Application Notes des PPs [33]) berücksichtigt wurden.

10.2 Sicherheitsmodul

Im Rahmen der erforderlichen Zertifizierung MUSS die Konformität des Sicherheitsmoduls zum Schutzprofil BSI-CC-PP-0077 [34] nachgewiesen werden.

Das Common Criteria Zertifikat MUSS einen Hinweis enthalten, dass die Anforderungen dieser Technischen Richtlinie an das Sicherheitsmodul (siehe auch die entsprechenden Application Notes des PPs [34]) berücksichtigt wurden.

Literaturverzeichnis

[1] BSI TR-03109, Technische Richtlinie BSI-TR-03109, 2013

[2] BSI TR-02102-1, Kryptographische Verfahren: Empfehlungen und Schlüssellängen, Version 2017-01, 2017

[3] BSI TR-02102-2, Kryptographische Verfahren: Empfehlungen und Schlüssellängen, Teil 2 - Verwendung von Transport Layer Security (TLS), 2017

[4] BSI TR-03111, Elliptic Curve Cryptography (ECC), Version 2.0, 2012

[5] BSI TR-03109-4, Smart Metering PKI - Public Key Infrastruktur für Smart Meter Gateways, 2016

[6] NIST FIPS 197, Advanced Encryption Standard (AES), 2001

[7] ISO/IEC 10116:2006, Information technology -- Security techniques -- Modes of operation for an n-bit block cipher, 2006

[8] IETF RFC 4493, JH. Song, R. Poovendran, J. Lee, T. Iwata: The AES-CMAC Algorithm, 2006

[9] NIST SP800-38D, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, 2007

[10] NIST FIPS 180-4, Secure Hash Standard (SHS), 2015

[11] IETF RFC 5114, M. Lepinski, S. Kent: Additional Diffie-Hellman Groups for Use with IETF Standards, 2008

[12] IETF RFC 5639, M. Lochter, J. Merkle: Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation, 2010

[13] BSI AIS 20/31, A proposal for: Functionality classes for random number generators, Version 2.0, 2011

[14] BSI, Certificate Policy der Smart Metering PKI, Version 1.0.1, 2015

[15] IETF RFC 5246, T. Dierks, E. Rescorla: Transport Layer Security (TLS) Version 1.2, 2008

[16] IETF RFC 5077, J. Salowey, H. Zhou, P. Eronen, H. Tschofenig: Transport Layer Security (TLS) Session Resumption without Server-Side State, 2008

[17] IETF RFC 5289, E. Rescorla: TLS Elliptic Curve Cipher Suites with SHA-256/384 and AES Galois Counter Mode (GCM), RFC 5289, 2008

[18] IETF RFC 7027, J. Merkle, M. Lochter, Elliptic Curve Cryptography (ECC) Brainpool Curves for Transport Layer Security (TLS), 2013

[19] IETF RFC 4492, S. Blake-Wilson, N. Bolyard, V. Gupta, C. Hawk, B. Möller, Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS), 20016

[20] BSI TR-03109-1, Anforderungen an die Interoperabilität der Kommunikationseinheit eines intelligenten Messsystems, 2013

[21] IETF RFC 6066, D. Eastlake 3rd, Transport Layer Security (TLS) Extensions: Extension Definitions, 2011

[22] M. Bellare, C. Namprempre, Authenticated Encryption: Relations among notions and analysis of the generic composition paradigm; in Advances in Cryptology - Asiacrypt 2000 Proceedings, Lecture Notes in Computer Science Vol. 1976, T. Okamoto ed, Springer-Verlag, 2000

[23] IETF RFC 7366, P. Gutman, Encrypt-then-MAC for Transport Layer Security (TLS) and Datagram Transport Layer Security (DTLS), 2014

[24] K. Bhargavan, A. Delignat-Lavaud, C. Fournet, A. Pironti, P.-Y. Strub, Triple Handshake and Cookie Cutters: Breaking and Fixing Authentication over TLS, IEEE Symposium on Security and Privacy, 2014,

[25] IETF RFC 7627, K. Bhargavan, Ed., A. Delignat-Lavaud, A. Pironti, A. Langley, M. Ray, Transport Layer Security (TLS) Session Hash and Extended Master Secret Extension, 2015

[26] BSI TR-03109-2, Smart Meter Gateway – Anforderungen an die Funktionalität und Interoperabilität des Sicherheitsmoduls, 2014

[27] BSI TR-03110-2, Advanced Security Mechanisms for Machine Readable Travel Documents - Part 2, Version 2.10, 2012

[28] IETF RFC 5652, R. Housley: Cryptographic Message Syntax (CMS), 2009
[29] IETF RFC 5083, R. Housley, Cryptographic Message Syntax (CMS) Authenticated-Enveloped-Data Content Type, 2007
[30] IETF RFC 5084, R. Housley, Using AES-CCM amd AES-GCM Authenticated Encryption in the Cryptographic Message Syntax (CMS), 2007
[31] IETF RFC 3565, J. Schaad, Use of AES Encryption Algorithm in CMS, 2003,
[32] BSI TR-03110-3, Advanced Security Mechanisms for Machine Readable Travel Documents - Part 3, Version 2.11, 2013
[33] BSI CC-PP-0073, Protection Profile for the Gateway of a Smart Metering System, 2014
[34] BSI CC-PP-0077, Protection Profile for the Security Module of a Smart Metering System, 2014