edi@energy
Datenformate Strom & Gas

Konsolidierte Lesefassung mit Fehlerkorrekturen

Stand: 27.06.2025

Regelungen zum Übertragungsweg für API-Webdienste

Regelungen zum sicheren Austausch von Übertragungsdateien

Version:	1.1
Ursprüngliches Publikationsdatum:	01.10.2025
Anzuwenden ab:	04.04.2025
Autor:	BDEW

Inhalt

1 Einleitung

Dieses Dokument regelt die Sicherheits- und Schutzmechanismen, die im Rahmen des elektronischen Datenaustauschs für regulierte Prozesse zwischen den Marktpartnern der deutschen Energiewirtschaft für den Übertragungsweg[1] API-Webdienste in der Marktkommunikation einzuhalten sind.

Gemäß BNetzA-Beschlüsse[2] sind grundsätzlich die kryptographischen Vorgaben der BSI TR 03116-3 anzuwenden und einzuhalten, sowie die Nutzung der Smart Metering-PKI des BSI, nach § 52 Abs. 4 MsbG vorzusehen.

Dieses Dokument benennt nicht die ggf. existierenden rechtlichen Folgen, wenn aufgrund eines abweichenden Vorgehens kein gesicherter elektronischer Datenaustausch stattfinden kann.

2 Terminologie

2.1 Schlüsselwörter

Die Schlüsselwörter „MÜSSEN“ (Englisch „MUST“), „DÜRFEN NICHT“ (Englisch „MUST NOT“), „ERFORDERLICH“ (Englisch „REQUIRED“), „SOLL“ (Englisch „SHALL“), „SOLL NICHT“ (Englisch „SHALL NOT“), „SOLLTE“ (Englisch „SHOULD“), „SOLLTE NICHT“ (Englisch „SHOULD NOT“), „EMPFOHLEN“ (Englisch „RECOMMENDED“), „DÜRFEN“ (Englisch „MAY“), and „FREIWILLIG“ (Englisch „OPTIONAL“) in diesem Dokument sind zu interpretieren gemäß [RFC2119]. Dabei spielt die Groß- und Kleinschreibung keine Rolle.

3 Technische Vorgaben der API-Webdienste

3.1 Netzwerk

MUST Alle Kommunikationsendpunkte werden durch DNS-Namen und nicht durch IP-Adressen bestimmt. Die beschriebenen Webservices müssen IPv4 implementieren und SOLLTEN IPv6 implementieren (Dual Stack).

Der API-Nutzer des API-Webservice MUSS in der Lage sein, die TLS-Verbindung über IPv4 herzustellen.

3.2 Transportebene

MUST Zum Aufbau der http-Verbindung MUSS das Protokoll Transport Layer Security (TLS) nach den Regeln in [TR03116-3] (Kapitel TLS-Kommunikation im WAN und in der Marktkommunikation) eingesetzt werden.

---

[1] Mit „Übertragungsweg“ wird in diesem Dokument das bezeichnet, was auch als „Kommunikationskanal“, „Kommunikationsweg“ „Transportprotokoll“ oder „Übertragungsprotokoll“ bezeichnet wird.

[2] Vgl. [BK6-21-282] und Fortentwicklung mit [BK6-22-128] und [BK6-22-024].

MUST Für den Aufbau der TLS-Verbindung ist die Erweiterung Server Name Identification (SNI) gemäß [RFC6066] bzw. [RFC8449] zu unterstützen und zu verwenden.

3.3 Inhaltsdatensicherungsebene

MUST Jede API-Interaktion und der übertragene Payload wird signiert. Dies gilt sowohl für http-Requests als auch synchrone Responses, in denen ein Payload enthalten ist.

MUST Die dabei anzuwendenden Algorithmen (Hash- und Signatur-Algorithmus) sowie die Hinweise zur Umsetzung, sind in [TR03116-3] in den Kapiteln 8 und 9 identisch beschrieben und soweit hier anzuwenden.

Signiert wird die angesprochene Ressource (URI) inklusive aller Query-Parameter, die http-Kopfzeilen creationDateTime, transactionId und soweit vorhanden initialTransactionId, und die übermittelte Payload (JSON-Objekt)[3].

Vor der Berechnung des Hashwerts eines JSON-Objekts muss dieses kanonisiert werden gemäß [RFC8785].

Die zu signierende Bitfolge M ist wie folgt zu berechnen, wobei der Hash für die HTTP-Kopfzeile „initialTransactionId“ nur dann einzubeziehen und in die Gesamtberechnung aufzunehmen ist, wenn diese vorhanden ist, und das „+“ als Konkatenation der Ergebnisse der Hash-Funktionen zu verstehen ist:

› M = Hash(URI) + Hash(Payload) + Hash(creationDateTime) + Hash(transactionId) + Hash(initialTransactionId)

- Beispiel: Mit dem Signaturalgorithmus ECDSA gemäß [FIPS-186-5] und dem Hashalgorithmus SHA256 würde die vollständige Signatur wie folgt berechnet:

  ECDSA(M, PK_S, SHA256), wobei

  – M die wie oben beschriebene, zu signierende Bitfolge mit Hash=SHA256,
  – PK_S den privaten Signaturschlüssel des Senders und
  – SHA256 die zu verwendende Hash-Funktion bezeichnet.

Die Signatur ist im Plain Format gemäß [TR03111] zu kodieren. Der Plain-kodierte Signaturwert ist im HTTP-Header X-BDEW-SIGNATURE und der signierte Digest Hash(M) im HTTP-Header X-BDEW-DIGEST als Base64-kodierter Text gemäß Kapitel 4 von [RFC4648] zu übertragen.

Das Zertifikat mit dem zum Signieren verwendeten öffentlichen Schlüssel wird in dem HTTP-Header X-BDEW-CERT hinterlegt. Die Kodierung des Zertifikats in einem HTTP-Header Feld ist in Kapitel 2.1 in [RFC9440] für TLS-Zertifikate beschrieben und ist hier in gleicher Weise zu verwenden.

[3] Alle für die fachliche Verarbeitung eines Requests erforderlichen Parameter müssen in der Signatur enthalten sein. Insbesondere bedeutet dies, dass alle weiteren http-Header rein informatorischen Charakter haben und keinen Einfluss auf die fachliche Verarbeitung eines Requests haben dürfen.

Die mittels des API-Aufrufs übertragenen Parameter und Payload werden nicht verschlüsselt.

4 Zertifikate und PKI

Die Kommunikation wird durch Verwendung der Smart Metering PKI (SM-PKI) des BSI abgesichert. Die Vorgaben der Certificate Policy (CP) der SM-PKI müssen eingehalten werden. Die Kommunikationspartner, API-Nutzer und Anbieter, sind nach dem Rollenkonzept der SM-PKI in der Rolle passiver EMT, sofern der Kommunikationspartner nicht durch Regelungen außerhalb dieses Regelwerks zur API die Rolle eines aktiven EMT wahrnehmen muss.

4.1 Vertrauensdiensteanbieter

Die Vertrauensdiensteanbieter müssen eine Sub-CA-Instanz gemäß der CP der SM-PKI sein.

4.2 Verzeichnisdienst

Neben den von den Sub-CA gestellten Verzeichnisdiensten muss jeder Marktpartner einen dezentralen Verzeichnisdienst (siehe EDI@Energy-Dokument „Regelungen zum Verzeichnis-dienst“) beauftragen oder betreiben, der Metadaten zu seinen verwendeten Zertifikaten und insbesondere zur API-Kennung den zughörigen Endpunkt (URL) der API veröffentlicht.

Hinweis: Die möglichen API-Kennungen sind im EDI@Energy-Dokument “Anwendungsübersicht der Prüfindetifikatoren“ zu entnehmen.

4.3 Zertifikate: Parameter und Anforderungen

Die Anforderungen an die Zertifikate ergeben sich aus der CP der genutzten SM-PKI; insbesondere gilt:

- der CommonName muss nach dem Schema <*org*>.EMT.**API**<*extension*> gebildet werden.
- Das Feld Organisational Unit („OU“) des Subject muss die MP-ID enthalten.
- Der Parameter im Feld „Alternativer Antragstellername“ mit der Ausprägung UniformResourceIdentifier muss vorhanden sein und die Adresse des genutzten Verzeichnisdiensts (siehe Kapitel 4.2) beinhalten.
- ein EMT.API-Zertifikat kann für einen oder mehrere API-Webdienste genutzt werden.

4.4 Übergangsregelung

Bereits ausgestellte und in der Marktkommunikation genutzte Zertifikate für die Nutzung der API-Webdienste zur prozessualen Abwicklung von Steuerungshandlungen in Verbindung mit intelligenten Messsystemen (iMS), die nicht den genannten Anforderungen entsprechen, dürfen bis zum Gültigkeitsende weiter genutzt werden.

4.5 Zertifikatswechsel

Spätestens 10 Werktage, bevor Zertifikate ungültig werden, muss der Inhaber dieser Zertifikate die Nachfolgezertifikate zur Verfügung gestellt haben. Somit entsteht ein

Überlappungszeitraum von mindestens 10 Werktagen, in dem noch die bisherigen und die neuen Zertifikate gleichzeitig gültig sind. Für diesen Zeitraum gilt: Innerhalb dieses Überlappungszeitraums kann bei allen Marktpartnern die Umstellung von den bisher genutzten auf die neuen Zertifikate erfolgen.

4.6 Rückruf und Sperrlisten

Will ein Zertifikatsinhaber sein Zertifikat vor Ablauf der Gültigkeitsfrist nicht mehr verwenden oder für ungültig erklären, so muss er sein Zertifikat über die Sperrlisten (CRL) seines CA-Anbieters zurückziehen lassen. Die Vorgaben und Regelungen für die Sperrung von Zertifikaten, Verarbeitung von Sperrlisten und der Aktualisierungs- und Prüfungszeiten ergeben sich aus der Certificate Policy (CP) der SM-PKI.

Wenn die CRL einer CA über die im Zertifikat eingetragene Certificate Revocation List Distribution Point (CRL-DP) von einer CA über 3 Tage nicht abrufbar ist oder im Gültigkeitszeitraum nicht verlängert wurde, dann ist der ausstellenden CA und aller darunter gelisteten Zertifikate bis zur Veröffentlichung einer aktuellen CRL nach den Regelungen der CP zu misstrauen. Die konkreten, möglichen Konsequenzen sind Kapitel 9 zu entnehmen

5 Quellen

[BK6-21-282] Beschluss (BK6-21-282) und Anlagen zur Absicherung der elektronischen Marktkommunikation Strom, Bundesnetzagentur, 31.03.2022.

[BK6-22-128] Beschluss (BK6-22-128) und Anlagen zur prozessualen Abwicklung von Steuerungshandlungen in Verbindung mit intelligenten Messsystemen (iMS) (Universalbestellprozess), 21.11.2022.

[BK6-22-024] Beschluss (BK6-22-024) und Anlagen zu Regelungen für einen beschleunigten werktäglichen Lieferantenwechsel in 24 Stunden (LFW24), 21.03.2024.

[CP-SM-PKI] Certificate Policy der Smart Metering PKI. Version 1.1.2, 25.01.2023. https://www.bsi.bund.de/dok/7615484

[FIPS-186-5] Federal Information Processing Standards Publication. (FIPS PUB) 186-5, Digital Signature Standard (DSS), 03.02.2023. https://doi.org/10.6028/NIST.FIPS.186-5

[RFC2119] Key words for use in RFCs to Indicate Requirement Levels. IETF RFC 2119. March 1997. https://www.rfc-editor.org/rfc/rfc2119

[RFC4648] The Base16, Base32, and Base64 Data Encodings. IETF RFC 4648. October 2006. https://www.rfc-editor.org/rfc/rfc4648

[RFC6066] Transport Layer Security (TLS) Extensions: Extension Definitions, ITEF RFC 6066. Januar 2011. https://www.rfc-editor.org/rfc/rfc6066

[RFC8449] Record Size Limit Extension for TLS IETF RFC 8449. August 2018. https://www.rfc-editor.org/rfc/rfc8449

[RFC8785] JSON Canonicalization Scheme (JCS). IETF RFC 8785. June 2020. https://www.rfc-editor.org/rfc/rfc8785

[RFC9440] Client-Cert HTTP Header Field. IETF RFC 9440. July 2023. https://www.rfc-editor.org/rfc/rfc9440

[TR03111] Technical Guideline BSI TR-03111 Elliptic Curve Cryptography. 01.06.2018. https://www.bsi.bund.de/dok/TR-03111

[TR03116-3] Technische Richtlinie BSI TR-03116 Kryptographische Vorgaben für Projekte der Bundesregierung. Teil 3: Intelligente Messsysteme. 13.12.2024. https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03116/BSI-TR-03116-3.html

6 Änderungshistorie

Änd-ID	Ort	Änderungen		Grund der Anpassung	Status
		Bisher	Neu		
14031	Deckblatt	Version: 1.1 Publikationsdatum: 01.10.2024 Anzuwenden ab: 03.04.2025	**Konsolidierte Lesefassung mit Fehlerkorrekturen** **Stand: 28.05.2025** Version: 1.1 **Ursprüngliches** Publikationsdatum: 01.10.2024 Anzuwenden ab: 03.04.2025	Version aktualisiert. Zusätzlich wurden im gesamten Dokument Schreibfehler, Layout, Internetadressen etc. korrigiert, die keinen Einfluss auf die inhaltliche Aussage haben.	Fehler (28.05.2025).
14031	Kapitel 3.3 Inhaltsdatensicherungsebene	Der zu signierende Digest wird dann berechnet (Der Hash für die http-Kopfzeile „initialTransactionId“ wird nur berechnet und in die Gesamtberechnung aufgenommen, wenn sie vorhanden ist!): › Hash(Hash(URI) + Hash(Payload) + Hash(CreationDateTime) + Hash(transactionId) + Hash(initialTransactionId)) Digest und Signatur werden in den http-Header Feldern X-BDEW-DIGEST und X-BDEW-SIGNATURE Base64-kodiert abgelegt.	**Die** zu signierende ~~Digest wird dann berechnet (Der~~ **Bitfolge M ist wie folgt zu berechnen, wobei der** Hash für die HTTP-Kopfzeile „initialTransactionId“ ~~wird nur berechnet und in die Gesamtberechnung aufgenommen, wenn sie vorhanden ist!):~~ **nur dann einzubeziehen und in die Gesamtberechnung aufzunehmen ist, wenn diese vorhanden ist, und das „+“ als Konkatenation der Ergebnisse der Hash-Funktionen zu verstehen ist:** › **M =** ~~Hash(~~Hash(URI) + Hash(Payload) + Hash(creationDateTime) + Hash(transactionId) + Hash(initialTransactionId)~~)~~ • **Beispiel: Mit dem Signaturalgorithmus ECDSA gemäß [FIPS-186-5] und dem Hashalgorithmus SHA256 würde die vollständige Signatur wie folgt berechnet:** **ECDSA(M, PKS, SHA256), wobei** **– M die wie oben beschriebene, zu signierende Bitfolge mit Hash=SHA256,** **– PKS den privaten Signaturschlüssel des Senders und** **– SHA256 die zu verwendende Hash-Funktion bezeichnet.** **Die Signatur ist im Plain Format gemäß [TR03111] zu kodieren. Der Plain-kodierte Signaturwert ist im HTTP-Header X-BDEW-SIGNATURE und der signierte Digest**	Präzisierung der Berechnung und Kodierung der Signatur.	Fehler (28.05.2025).

Änd-ID	Ort	Änderungen		Grund der Anpassung	Status
		Bisher	**Neu**		
			Hash(M) im HTTP-Header X-BDEW-DIGEST als Base64-kodierter Text zu übertragen. ~~Digest und Signatur werden in den http-Header Feldern X-BDEW-DIGEST und X-BDEW-SIGNATURE Base64-kodiert abgelegt.~~		
14032	Kapitel 5 Quellen	[…] [TR03116-3] Technische Richtlinie BSI TR-03116 Kryptographische Vorgaben für Projekte der Bundesregierung. Teil 3: Intelligente Messsysteme. 06.12.2022. https://[...].	**[FIPS-186-5] Federal Information Processing Standards Publication. (FIPS PUB) 186-5, Digital Signature Standard (DSS), 03.02.2023. https://doi.org/10.6028/NIST.FIPS.186-5** […] **[TR03111] Technical Guideline BSI TR-03111 Elliptic Curve Cryptography. 01.06.2018. https://www.bsi.bund.de/dok/TR-03111** [TR03116-3] Technische Richtlinie BSI TR-03116 Kryptographische Vorgaben für Projekte der Bundesregierung. Teil 3: Intelligente Messsysteme. 13.12.20**24**. https://[...].	Ergänzungen gemäß 14031. Aktualisierung Versionsdatum TR gemäß Festlegung.	Fehler (28.05.2025).
14033	Deckblatt	Konsolidierte Lesefassung mit Fehlerkorrekturen Stand: 28.05.2025 Version: 1.1 Ursprüngliches Publikationsdatum: 01.10.2024 Anzuwenden ab: 03.04.2025	Konsolidierte Lesefassung mit Fehlerkorrekturen Stand**: 27.06.2025** Version: 1.1 Ursprüngliches Publikationsdatum: 01.10.2024 Anzuwenden ab: 03.04.2025	Version aktualisiert. Zusätzlich wurden im gesamten Dokument Schreibfehler, Layout, Internetadressen etc. korrigiert, die keinen Einfluss auf die inhaltliche Aussage haben.	Fehler (27.06.2025).
14034	Kapitel 3.3 Inhaltsdatensicherungsebene	Die Signatur ist im Plain Format gemäß [TR03111] zu kodieren. Der Plain-kodierte Signaturwert ist im HTTP-Header X-BDEW-SIGNATURE und der signierte Digest Hash(M) im HTTP-Header X-BDEW-DIGEST als Base64-kodierter Text zu übertragen.	Die Signatur ist im Plain Format gemäß [TR03111] zu kodieren. Der Plain-kodierte Signaturwert ist im HTTP-Header X-BDEW-SIGNATURE und der signierte Digest Hash(M) im HTTP-Header X-BDEW-DIGEST als Base64-kodierter Text **gemäß Kapitel 4 von [RFC4648]** zu übertragen.	Präzisierung Base64-kodierter Text nach RFC 3648.	Fehler (27.06.2025).

Änd-ID	Ort	Änderungen		Grund der Anpassung	Status
		Bisher	Neu		
14035	Kapitel 5 Quellen	[…]	**[RFC4648] The Base16, Base32, and Base64 Data Encodings. IETF RFC 4648. October 2006. https://www.rfc-editor.org/rfc/rfc4648** […] **[RFC9440] Client-Cert HTTP Header Field. IETF RFC 9440. July 2023. https://www.rfc-editor.org/rfc/rfc9440**	Ergänzung gemäß 14034. Ergänzung fehlender RFC-Eintrag aus Kapitel 3.3.	Fehler (27.06.2025).