Deutsche Telekom Security GmbH

Magenta Security Qualified.ID

Schnittstellenbeschreibung OCSP-Responder

Version: 03.00
Datum: 21.05.2026
Status: Freigegeben
Geheimhaltungsvermerk: Öffentlich

Connecting
your world.

Impressum

Herausgeber		
Deutsche Telekom Security GmbH Koblenzer Str. 87-93 57072 Siegen Deutschland - nachfolgend Telekom Security oder DT Security -		
Dateiname	Dokumentennummer	Dokumentenbezeichnung
Qualified.ID OCSP-Responder v03.00.docx		
Version	Stand	Status
03.00	21.05.2026	Freigegeben
Autor*in		
Deutsche Telekom Security GmbH		
Siegen, 21.05.2026		
Kurzinfo		
Das vorliegende Dokument beschreibt die Schnittstellen des Qualified.ID OCSP-Responders.		

Connecting your world.

Inhaltsverzeichnis

Tabellenverzeichnis

1 Einleitung

Zur Überprüfung der Existenz und des Status eines Zertifikates ist es notwendig, eine vertrauenswürdige Instanz nach diesem Zertifikat zu befragen. Diese Schnittstelle einer Zertifizierungsstelle wird allg. als Verzeichnisdienst bezeichnet. Er hat die Aufgabe, dem Anwender Informationen über Zertifikate über öffentliche Kommunikationsverbindungen zur Verfügung zu stellen. Damit dient der Verzeichnisdienst dem Kunden als Auskunftsdienst, ob und ab welchem Zeitpunkt ein bestimmtes Zertifikat gesperrt ist oder nicht.
In diesem Dokument werden die Funktion des OCSP-Responders beschrieben.

2 Der OCSP-Responder

Der OCSP-Responder übernimmt im Einzelnen folgende Aufgaben:

- Bereitstellen von Statusinformationen zu Zertifikaten.
- Erstellung einer signierten Antwort. In diese Antwort wird die gesetzlich gültige Zeit eingebunden.
- Rücksendung der Antwort an den Kunden.
- Erzeugung von Fehlermeldungen bei Problemen.

2.1 Das OCSP-Protokoll

Alle Anfragen an den OCSP-Responder werden mit Hilfe des http-Protokolls übertragen und müssen in einem definierten Format vorliegen (OCSP-Request) siehe dazu auch [COMMON-PKI] und [RFC6960].
Die Antwort des OCSP-Responders liefert für jedes Zertifikat eine der drei folgenden Statusinformationen:

- das Zertifikat ist nicht gesperrt und im Verzeichnisdienst vorhanden,
- das Zertifikat ist gesperrt,
- das Zertifikat befindet sich nicht in der Liste der von der Zertifizierungsstelle ausgestellten Zertifikate.

In die Antworten des OCSP-Responders an den Kunden wird jeweils die gesetzlich gültige Zeit eingebunden. Bei Fehlermeldungen erfolgt nur die Angabe des entsprechenden Fehlercodes ohne Einbindung eines Zeitstempels. Zur Gewähr der Integrität der Antwort als auch zur Angabe der Identität des Ausstellers wird die Antwort mit Hilfe eines privaten Schlüssels elektronisch signiert. Das zugehörige Signatur-Zertifikat des Verzeichnisdienstes wird der Antwort beigefügt.

2.1.1 HTTP als Transportprotokoll

In diesem Kapitel wird auf die Einbettung von OCSP-Anfragen und Antworten in HTTP eingegangen. Die Kommunikation zum Transport von OCSP-Anfragen über HTTP erfolgt über die beschriebenen Datenformate, die als Nachrichten über das Hypertext Transfer Protokoll (HTTP) an den OCSP-Responder übermittelt werden. Die Verwendung von HTTP [RFC9112] erlaubt eine einfache Erstellung von Software für den Zugriff auf Verzeichnisdienste unter Verwendung erprobter Programmbibliotheken. Weiterhin sind für HTTP Übergangsmöglichkeiten von firmeninternen Netzen in öffentliche Netze, sogenannten Firewalls, definiert und in der Form von HTTP Proxies bereits erprobt.

2.1.2 Definitionen für die Anfragen

Eine Anfrage wird mittels der Methode `POST` übertragen. Das HTTP Headerfeld „Content-Type“ mit dem Wert „application/ocsp-request“ muss verwendet werden. Die Länge der Anfrage muss im HTTP Headerfeld „Content-Length“ mit der Länge des Inhaltes der HTTP Anfrage belegt werden. Als Inhalt der HTTP Anfrage wird die DER-kodierte Anfrage an den Verzeichnisdienst verwendet. Anfragen an den Verzeichnisdienst mit der Zugriffsmethode `POST` verwenden folgendes Format. Die Anfrage enthält eine ASN.1 Struktur vom Typ ***OCSPRequest***:

```
POST {url}
Content-Type: application/ocsp-request
Content-Length: ...
{binär kodierte Anfrage}
```

Außer den hier aufgeführten HTTP Header Feldern werden keine weiteren Felder ausgewertet, sondern nur stillschweigend ignoriert.

2.1.3 Definitionen für die Antworten

Eine HTTP Antwort mit Status 200 enthält eine ASN.1-Struktur vom Typ OCSPResponse. Der Transport erfolgt binär.
Im HTTP Header „Content-Type“ wird der Wert „application/ocsp-response“ angegeben, im Header „Content-Length“ ist die Länge der Antwort eingetragen. Andere HTTP Header können vorhanden sein.

2.1.4 OCSP-Datenformate

2.1.4.1 OCSP-Request

Der Verzeichnisdienst erlaubt das Abrufen von Statusinformationen. Zur Abfrage werden durch den anfragenden Applikationsnutzer die nachfolgend genannten Informationen an den Verzeichnisdienst übergeben:

- Angabe über den verwendeten Hashalgorithmus (***hashAlgorithm***), folgende Algorithmen werden unterstützt
  - SHA1 (1.3.14.3.2.26)
  - SHA256 (2.16.840.1.101.3.4.2.1)
- Angabe über den Namen des Ausstellers (***issuerNameHash***),
- Angabe über das Ergebnis der Anwendung der Hashfunktion auf den Wert des Feldes ***subjectPublicKey*** (ohne den ASN.1-Tag und die Längenbeschreibung) aus dem Zertifikat des Ausstellers (***issuerKeyHash***),
- Angabe über die Seriennummer des betreffenden Zertifikats (***serialNumber***),
- Statusanfragen zu mehreren Zertifikaten des gleichen Ausstellers können gleichzeitig in einem Request übermittelt werden

Die formelle Struktur der Anfrage im ASN1-Standard sieht wie folgt aus (unvollständige Darstellung der Struktur; vollständige Beschreibung ist in [COMMON-PKI] nachzulesen):

```
Request ::= SEQUENCE{
reqCert CertID,
singleRequestExtensions [0] EXPLICIT Extensions OPTIONAL }

CertID ::= SEQUENCE{
hashAlgorithm AlgorithmIdentifier,
issuerNameHash OCTET STRING,
issuerKeyHash OCTET STRING,
serialNumber CertificateSerialNumber }
```

2.1.4.2 OCSP-Response

Die Antworten des Verzeichnisdienstes werden an den Applikationsnutzer zurückgesendet. Die Antwort des Dienstes kann für jedes angefragte Zertifikat grundsätzlich drei verschiedene Ergebnisse liefern. Die Antworten tragen eine elektronische Signatur, enthalten den Antwortzeitpunkt (Zeitstempel) und eine *Statusinformation*.
Ist das angeforderte Zertifikat in der Datenbank vorhanden und nicht gesperrt, lautet die zugehörige *Statusinformation* „good".
Ist das angeforderte Zertifikat in der Datenbank vorhanden und gesperrt, lautet die zugehörige *Statusinformation* „revoked".
Ist das angeforderte Zertifikat nicht in der Datenbank vorgehalten, lautet die zugehörige *Statusinformation* „unknown".
Die Ergebnismeldungen des OCSP-Verzeichnisdienstes lauten:

Name	**Bedeutung**
successful	erfolgreiche Bearbeitung einer Anfrage
malformedRequest	nicht erfolgreiche Bearbeitung einer Anfrage wegen fehlerhaftes Anfrage-Format
internalError	Auftreten eines internen Fehlers des Verzeichnisdienst
tryLater	temporäre Nichtverfügbarkeit des Verzeichnisdienst

Tabelle 1 - OSCP-Ergebnismeldungen

Anfragen zu unbekannten Ausstellerzertifikaten werden mit dem http Statuscode UNAUTHORIZED beantwortet.
Signierte Anfragen werden nicht unterstützt.
Die formelle Struktur der Anwort gem. [COMMON-PKI] im ASN1-Standard sieht wie folgt aus (unvollständige Darstellung der Struktur; vollständige Beschreibung ist in [COMMON-PKI] nachzulesen):

```
OCSPResponse ::= SEQUENCE {
responseStatus OCSPResponseStatus,
responseBytes [0] EXPLICIT ResponseBytes OPTIONAL }

OCSPResponseStatus ::= ENUMERATED {
succesful (0),
malformedRequest (1),
internalError (2),
tryLater (3),
sigRequired (5),
unauthorized (6) }
```

Mögliche Fehlerwerte einer Antwort sind `malformedRequest`, `internalError` und `try-Later`.

```
ResponseBytes ::= SEQUENCE {
responseType OBJECT IDENTIFIER,
response OCTET STRING }

id-pkix-ocsp OBJECT IDENTIFIER ::= { id-ad-ocsp }
id-pkix-ocsp-basic OBJECT IDENTIFIER ::= { id-pkix-ocsp 1 }

BasicOCSPResponse ::= SEQUENCE {
tbsResponseData ResponseData
signatureAlgorithm AlgorithmIdentifier,
signature BIT STRING,
certs [0] EXPLICIT SEQUENCE OF Certificate
OPTIONAL }

ResponseData ::= SEQUENCE {
version [0] EXPLICIT Version DEFAULT v1,
responderID ResponderID,
producedAt GeneralizedTime,
responses SEQUENCE OF SingleResponse,
responseExtensions [1] EXPLICIT Extensions OPTIONAL }
```

```
ResponderID ::= CHOICE {
byName [1] EXPLICIT Name,
byKey [2] EXPLICIT KeyHash }

KeyHash ::= OCTET STRING

SingleResponse ::= SEQUENCE {
certID CertID,
certStatus CertStatus,
thisUpdate GeneralizedTime,
nextUpdate [0] EXPLICIT GeneralizedTime OPTIONAL,
singleExtensions [1] EXPLICIT Extensions OPTIONAL }

CertStatus ::= CHOICE {
good [0] EXPLICIT NULL,
revoked [1] IMPLICIT RevokedInfo,
unknown [2] IMPLICIT UnknownInfo }
RevocedInfo ::= SEQUENCE {
revocationTime     GeneralizedTime,
revocationReason [0] EXPLICIT CRLReason OPTIONAL}

UnknownInfo ::= NULL

Extensions ::= SEQUENCE SIZE (1..MAX) OF Extension

Extension ::= SEQUENCE {
extnId       OBJECT IDENTIFIER
critical           BOOLEAN DEFAULT FALSE
extnValue          OCTET STRING }
```

Folgende Extension wird benutzt um die Positivauskunft abzubilden:

```
id-isismtt-at-certHash OBJECT IDENTIFIER ::= {1 3 36 8 3 13 }
CertHash ::= SEQUENCE {
HashAlgorithm      AlgorithmIdentifier,
certificateHash    OCTET STRING}
```

2.1.5 Übersicht zu den OCSP-Protokoll-Elementen der OCSP Anfrage

(Sub-) Feld	Common-PKI	generiert	geparst	ausge-wertet	Kommentar
OCSPRequest	tbsRequest		X	X	
	optionalSignature	MUST-NOT	X	-	
tbsRequest	version		X		Muss v1 sein
	requestorName	MUST-NOT	X	-	
	requestList		X	X	
	requestExtensions	MUST	X	X	
Request	reqCert		X	X	Element of requestList
	singleRequestExtensions	MUST	X	X	
reqCert	hashAlgorithm		X	X	muss SHA1 oder SHA256 sein
	issuerNameHash		X	X	
	issuerKeyHash		X	X	
	serialNumber		X	X	
requestExtensions	Nonce	MAY, non-critical	X	X	wird transparent in der Antwort durchgereicht

Tabelle 2 - OSCP-Protokoll-Elemente in der Anfrage

2.1.6 Übersicht zu den OCSP-Protokoll-Elementen der OCSP Antwort

(Sub-) Feld	Common-PKI	generiert	Kommentar
OCSPResponse	responseStatus		
	responseBytes	MAY	Je nach Status
responseStatus	successful	MUST	
	malformedRequest	MUST	
	internalError	MUST	
	tryLater	MUST	
	sigRequired	MUST	tritt nicht auf
	unauthorized	MUST	falls eine Anfrage zu unbekannten Ausstellerzertifikat
responseBytes	responseType	(MUST)	nur id-pkix-ocsp-basic
	response	MUST	demnach Basic-Response
response	tbsResponseData		
	signatureAlgorithm		ecdsa-with-SHA256 (1.2.840.10045.4.3.2)
	signature		
	certs	MAY	enthält Zertifikate des kompletten Pfades
responseData	version		immer v1
	responderID		
	producedAt		
	responses		
	responseExtensions	MAY	
responderID	byName	MAY	Immer byName, Subject aus Signaturzertifikat
	byKey	MAY	

singleResponse	certID		Element of responses
	certStatus		
	thisUpdate		gleich producedAt
	nextUpdate	MAY	
	singleExtensions	MAY	
certStatus	good	MUST	certHash wird mitgeliefert
	revoked	MUST	certHash wird mitgeliefert
	unknown	MUST	
revokedInfo	revocationTime		
	revocationReason	MAY	
unknownInfo	NULL	(MUST)	
revocationReason	MAY		
responseExtensions	Nonce	MAY, non-critical	nur wenn im Request vorhanden
singleExtensions	CRL entry extensions	MAY, non-critical	
	CertHash	MAY, non-critical	MUST, nur wenn Status = good/revoked

Tabelle 3 - OSCP-Protokoll-Elemente in der Antwort

3 Quellenverzeichnis

[COMMON-PKI]
Common-PKI COMMON PKI SPECIFICATIONS FOR INTEROPERABLE APPLICATIONS VERSION 2.0 – 20 JANUARY 2009

[RFC 6960]
S. Santesson, M. Myers, R. Ankney, A. Malpani, S. Galperin, C. Adams: X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP, Juni 2013

[RFC 9112]
R. Fielding, M. Nottingham, J. Reschke: HTTP/1.1, Juni 2022