**3GPP TSG-SA3 Meeting #121 draft\_S3-251701-r1**

Goteborg, Sweden, 7 - 11 April 2025

|  |
| --- |
| *CR-Form-v12.1* |
| **CHANGE REQUEST** |
|  |
|  |  | **CR** |  | **rev** |  | **Current version:** |  |  |
|  |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* |
|  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME |  | Radio Access Network |  | Core Network | **X** |

|  |
| --- |
|  |
| ***Title:***  | Living document for Automatic Certificate Management Environment (ACME) for the Service Based Architecture (SBA) |
|  |  |
| ***Source to WG:*** | Cisco Systems, Huawei, US National Security Agency, Charter Communications, Google |
| ***Source to TSG:*** | S3 |
|  |  |
| ***Work item code:*** | ACME\_SBA |  | ***Date:*** | 2025-04-07 |
|  |  |  |  |  |
| ***Category:*** | **B** |  | ***Release:*** | Rel-19 |
|  | *Use one of the following categories:****F*** *(correction)****A*** *(mirror corresponding to a change in an earlier release)****B*** *(addition of feature),* ***C*** *(functional modification of feature)****D*** *(editorial modification)*Detailed explanations of the above categories canbe found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | *Use one of the following releases:Rel-8 (Release 8)Rel-9 (Release 9)Rel-10 (Release 10)Rel-11 (Release 11)…Rel-15 (Release 15)Rel-16 (Release 16)Rel-17 (Release 17)Rel-18 (Release 18)* |
|  |  |
| ***Reason for change:*** | This living draft-CR specifies security procedures and protocols for automated certificate management for 5G Core Network Functions using ACME. |
|  |  |
| ***Summary of change:*** | Specify procedures and protocols related to the automated certificate management for 5G Core Network Functions using ACME as per conclusions reached in TR 33.776.Additional procedures may be specified as corresponding WID (SP-241960) objectives are addressed. |
|  |  |
| ***Consequences if not approved:*** | Security procedures for automated certificate management for 5G Core Network Functions using ACME are not specified. |
|  |  |
| ***Clauses affected:*** |  |
|  |  |
|  | **Y** | **N** |  |  |
| ***Other specs*** |  | **X** |  Other core specifications  | TS/TR ... CR ...  |
| ***affected:*** |  | **X** |  Test specifications | TS/TR ... CR ...  |
| ***(show related CRs)*** |  | **X** |  O&M Specifications | TS/TR ... CR ...  |
|  |  |
| ***Other comments:*** |  |
|  |  |
| ***This CR's revision history:*** | Integrates all approved contributions for this work item into a living document the captures the proposed changes to 33.310. This includes S3-251089, S3-251090, S3-251091, S3-251092, S3-251093, S3-251094, and S3-251139 from SA3#120, and S3-25128, S3-251293, S3-251294, S3-251700, S3-251794, and S3-251795 from SA3#121. |

\* \* \* First Change \* \* \* \*

# 2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non‑specific.

- For a specific reference, subsequent revisions do not apply.

- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TS 33.210: "3rd Generation Partnership Project; Technical Specification Group Services and System Aspects; 3G Security; Network domain security; IP network layer security".

[2] IETF RFC 2986: "PKCS#10 Certification Request Syntax Specification Version 1.7".

[3] Void.

[4] IETF RFC 4210: "Internet X.509 Public Key Infrastructure Certificate Management Protocol".

[5] Void

[6] Void.

[7] "PKI basics – A Technical Perspective", November 2002, <http://www.oasis-pki.org/pdfs/PKI_Basics-A_technical_perspective.pdf>.

[8] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[9] 3GPP TS 33.203: "Access security for IP-based services".

[10] 3GPP TS 33.220: "Generic Authentication Architecture: Generic Bootstrapping Architecture".

[11] Void.

[12] Void.

[13] Void.

[14] IETF RFC 5280: "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile".

[15] IETF RFC 4945: "The Internet IP Security PKI Profile of IKEv1/ISAKMP, IKEv2, and PKIX".

[16] Void.

[17] Void.

[18] IETF RFC 6712: "Internet X.509 Public Key Infrastructure -- HTTP Transfer for the Certificate Management Protocol (CMP)".

[19] IETF RFC 4211: "Internet X.509 Public Key Infrastructure Certificate Request Message Format (CRMF)".

[20] IETF RFC 2818: "HTTP Over TLS".

[21] IETF RFC 5922: "Domain Certificates in the Session Initiation Protocol (SIP)".

[22] IETF RFC 5924: "Extended Key Usage (EKU) for Session Initiation Protocol (SIP) X.509 Certificates".

[23] Void.

[24] Void.

[25] IETF RFC 1035: "Domain Names - Implementation and Specification".

[26] Void.

[27] Void.

[28] Void.

[29] Void.

[30] Void.

[31] 3GPP TS 23.251: "Network sharing; Architecture and functional description".

[32] 3GPP TS 32.508: "Telecommunication management; Procedure flows for multi-vendor plug-and-play eNode B connection to the network".

[33] 3GPP TS 32.509: "Telecommunication management; Data formats for multi-vendor plug and play eNode B connection to the network".

[34] Void.

[35] Void.

[36] Void.

[37] Void.

[38] Void.

[39] Void.

[40] Void.

[41] Void.

[42] IETF RFC 7296: "Internet Key Exchange Protocol Version 2 (IKEv2)".

[43] IETF RFC 7427: "Signature Authentication in the Internet Key Exchange Version 2 (IKEv2)".

[44] Void.

[45] Void.

[46] Void.

[47] IETF RFC 6960: " X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP".

[48] IETF RFC 8201: "Path MTU Discovery for IP version 6".

[49] IETF RFC 8446: "The Transport Layer Security (TLS) Protocol Version 1.3".

[50] IETF RFC 9113: "HTTP/2".

[51] IETF RFC 6066: "Transport Layer Security (TLS) Extensions: Extension Definitions".

[52] Void

[53] IETF RFC 7633: "X.509v3 Transport Layer Security (TLS) Feature Extension".

[54] IETF RFC 5246: "The Transport Layer Security (TLS) Protocol Version 1.2".

[55] 3GPP TS 23.003: "Numbering, addressing and identification".

[56] 3GPP TS 29.510: "5G System; Network function repository services; Stage 3".

[57] 3GPP TS 29.571: "5G System; Common Data Types for Service Based Interfaces; Stage 3".

[58] IETF RFC 6979: " Deterministic Usage of the Digital Signature Algorithm (DSA) and Elliptic Curve Digital Signature Algorithm (ECDSA)".

[59] CA-Browser-Forum-BR-2.0.4, April 2024, <https://cabforum.org/working-groups/server/baseline-requirements/documents/TLSBRv2.0.4.pdf>.

[60] GSMA FS.34 Key Management for 4G and 5G inter-PLMN Security, <https://www.gsma.com/security/resources/fs-34-key-management-for-4g-and-5g-inter-plmn-security/>.

[61] IETF RFC 9310: "X.509 Certificate Extension for 5G Network Function Types".

[62] 3GPP TS 33.501: "Security architecture and procedures for 5G system".

[63] IETF RFC 9509: "X.509 Certificate Extended Key Usage (EKU) for 5G Network Functions".

[64] IETF RFC 4122:" A Universally Unique Identifier (UUID) URN Namespace".

[65] IETF RFC 9110: " HTTP Semantics".

[66] IETF RFC 9525: "Service Identity in TLS".

[67] IETF RFC 4510: "Lightweight Directory Access Protocol (LDAP): Technical Specification Road Map".

[68] IETF RFC 4517: "Lightweight Directory Access Protocol (LDAP): Syntaxes and Matching Rules".[69] IETF RFC 4523: "Lightweight Directory Access Protocol (LDAP): Schema Definitions for X.509 Certificates".

[70] IETF RFC 4512: " Lightweight Directory Access Protocol (LDAP): Directory Information Models".

[71] RFC 4754: "IKE and IKEv2 Authentication Using the Elliptic Curve Digital Signature Algorithm (ECDSA)".

[XX] IETF RFC 8555: "Automatic Certificate Management Environment (ACME)".

[XB] IETF RFC 9447: "Automated Certificate Management Environment (ACME) Challenges Using an Authority Token".

[XC] IETF RFC 7519: " JSON Web Token (JWT)".

[XD] IETF RFC 7515: "JSON Web Signature (JWS)".

[XE] IETF RFC 9448: "TNAuthList Profile of Automated Certificate Management Environment (ACME) Authority Token".

\* \* \* Next Change \* \* \* \*

## 10.3 Certificate enrolment and renewal for 5GC NFs

This clause describes the protocols and corresponding procedures for certificate enrolment and renewal for 5G Core Network Functions based on CMP protocol.

NOTE 1: At the time of writing IETF is working in a new version 3 of CMP protocol. The profiling of CMP for 5G Core Network Functions in 10.3.1 might be updated accordingly if required.

NOTE 2: Annex YY describes how to use ACME [XX] as an automated certificate management protocol for 5GC NFs.

\* \* \* Next Change \* \* \* \*

## I.2 NF Certificate Updates

The normal procedure of update and renewal of 5GC NF certificates is managed by CMP protocol as described in clause 10.3.1.

NOTE: Annex YY describes how to use ACME [XX] as an automated certificate management protocol for 5GC NFs.

Nevertheless, the certificate management framework can be severely impacted by special critical circumstances, which can derive in simultaneous updates of vast number of certificates, causing a potential partial or complete disruption of the service. For example, a compromised security algorithm, the disclosure of broken cryptographic primitives, the revocation of CA root certificates or multiple certificates with same expiration data, are some of the special circumstances triggering the certificate update procedure.

This clause lists a few practical recommendations to be considered in NF certificate update procedure with the aim of mitigating potential issues or disruptions due to outages or overload situations. These recommendations can be deployed and implemented via internal configuration, operator policies and other mechanisms and functionalities in the operator PKI infrastructure, OAM systems, orchestration systems, etc.

- The NF certificate updates can be configured in the operator PKI, and consequently the procedure can be initiated in advance before the certificate expiration time. For example, making use of different time interval/periodicity based on the NF type when configuring certificate update policies. Observe that the NF type is included in the certificates as per the profile in clause 6.1.3c and hence can be checked there while configuring such policies.

- The operator PKI does not have to update the certificates with the same or similar expiration time simultaneously. Furthermore, the certificate update policies can take into consideration the expiration time and the triggers of the procedure being configured in advance. Certificate updates policies can be configured, for example in the operator PKI, to create different batches of certificates to be updated sequentially or with certain prioritization criteria.

- Certificate expiry related alarms reported by network management systems, operator CA announcements for certificate revocations (e.g., via CRL, OCSP, etc.), and any other type of certificate related event, can be monitored with the purpose of mitigating the risk of service unavailability due to above mentioned special circumstances.

\* \* \* Next Change - all new text \* \* \* \*

Annex YY (informative):
Certificate management for 5GC NFs using ACME

## YY.1 Introduction

This annex describes the requirements to support ACME as an automated certificate management protocol for 5GC NFs. The NF acts as an ACME client and the operator CA/RA acts as an ACME server. The overall certificate management procedure follows RFC 8555 [XX]. It consists of Initial configuration (YY.2), Certificate enrolment and renewal (YY.3), and Certificate revocation (YY.4).

## YY.2 Initial configuration

### YY.2.1 NF instantiation and security assumptions

NF instantiation includes providing the NF with sufficient information for it to bootstrap its communication within the 5GC. It is assumed that the 5G NF has an authenticated channel to the OAM (e.g., established using an OAM-issued TLS client certificate or platform specific method) and is issued with the operator CA’s root certificate, which is used to validate the ACME server’s TLS certificate.

The communication between 5G NF and the operator CA is over HTTPS for authentication and confidentiality.

When a 5G NF fetches a resource from an ACME server, the server authenticates the requester and verifies any access control as described in RFC 8555 [XX]. JWS based integrity protection is used as described in RFC 8555 [XX]. Nonces are used to protect messages against replay-attacks as described in RFC 8555 [XX].

### YY 2.2 Account creation

In 5G SBA, it is recommended that the ACME account creation be authenticated and authorized. This prevents unauthorized entities from creating accounts and attempting to request certificates. Failure to do this could lead to resource exhaustion and an increased attack surface for obtaining mis-issued certificates.

The external account binding mechanism described in RFC 8555 [XX], section 7.3.4, allows the use of authorizations granted to the OAM for creating a new ACME account. The OAM acting as the external account verifier shares with the operator CA a symmetric MAC key bound to the ACME client and any other data which facilitates CA operation. This could include NF Instance ID, NF Type, or certificate profiles. The ACME server can use this data to validate a "newOrder" request as described in RFC 8555 [XX], section 7.4.

NOTE 1: Distribution of MAC keys and storage of MAC and ACME account keys is left to implementation.

NOTE 2: Account key renewal periods are left to operator policy.

NOTE 3: Use of external account binding in this way is consistent with the IAK option for Initial Trust as specified in clause 10.2. If future specifications create extensions for external account binding it could be possible to re-use other options for Initial Trust and this annex be updated to reflect this.

## YY.3 Certificate enrolment and renewal

### YY.3.1 Introduction

This clause describes the ACME certificate enrolment and renewal procedures for the 5GC SBA. The procedures are based on the ACME protocols specified in RFC 8555 [XX]. The steps for certificate enrolment and renewal procedures have been grouped as follows: Certificate issuance (YY.3.2) for order submission, and Challenge validation (YY.3.3) for proving control of identifiers in the certificate, CSR, certificate download.

### YY.3.2 Certificate issuance

Certificate issuance as defined in RFC 8555 [XX] is used for certificate enrolment and renewal.

Pre-authorization is not used in ACME for 5G SBA, therefore:

- The newAuthz resource is not supported.

- The newOrder resource is used for all enrolment and renewal.

Clause 6.1.3c defines the certificate profiles for 5GC SBA. X.509 version 3 certificates are used for all entities in 5GC SBA. ACME supports X.509 version 3 certificates and the necessary extensions. Table YY.3.2-1 lists the ACME challenge types, as described in Annex YY clause YY.3.3, used for each 5G SBA certificate profile.

Table YY.3.2.1: ACME challenge types per 5G SBA certificate profile

|  |  |
| --- | --- |
| 5G SBA certificate type | ACME challenge type |
| TLS client | tkauth-01 |
| TLS server | tkauth-01 |
| OAuth 2.0 access token | tkauth-01 |
| CCA token | tkauth-01 |

The policy-based certificate renewal can be used as described in the Annex I.2 of the present document.

### YY.3.3 Challenge validation

#### YY.3.3.1 Introduction

The ACME challenge-type used is the ACME Authority Token challenge type, "tkauth-01", as specified in RFC 9447 [XB]. The challenge-type assumes a trust relationship between a CA and a Token Authority, i.e., that a CA is willing to accept the attestation of a Token Authority for particular types of identifiers as sufficient proof to issue a credential. When using ACME, the OAM system acts as a Token Authority that is trusted by the operator CA/RA. As such, the OAM is trusted to act as the authority for the NF Instance ID namespace within the 5GC.

#### YY.3.3.2 "NfInstanceId" identifier type

A new ACME identifier type, "NfInstanceId", is defined in this clause. A NF uses its NF Instance ID as the value of the “NfInstanceId". The format of the value of the "NfInstanceId" is that of the NfInstanceId, as defined in TS 29.571 [57]:

- NfInstanceId: string: String uniquely identifying a NF instance. The format of the NF Instance ID shall be a Universally Unique Identifier (UUID) version 4, as described in RFC 4122 [64]. The hexadecimal letters should be formatted as lower-case characters by the sender, and they shall be handled as case-insensitive by the receiver.

- Example: "4ace9d34-2c69-4f99-92d5-a73a3fe8e23b"

An example of an ACME order object "identifiers" field containing a "NfInstanceId" is as follows:

- "identifiers": [{"type":"NfInstanceId","value":"4ace9d34-2c69-4f99-92d5-a73a3fe8e23b"}]

Editor's Note: This new ACME identifier type needs to be listed in a new registration in the ACME Identifier Types registry and the ACME Validation Methods registry maintained by IANA, per RFC 9447 [XB], section 3.

In NF certificates, both client and server, the subjectAltName extension contains the NfInstanceId as a "uniformResourceIdentifier" formatted as a URN as described in clause 5.3.2 of TS 29.571 [57]. For example, "urn:uuid:4ace9d34-2c69-4f99-92d5-a73a3fe8e23b" is the string representation of the NF Instance ID "4ace9d34-2c69-4f99-92d5-a73a3fe8e23b" as a URN.

When processing a certificate order containing an identifier of type "NfInstanceId", a CA uses the Authority Token challenge type of "tkauth-01" with a "tkauth-type" of "atc", as defined in RFC 9447 [XB], to verify that the requesting ACME client has authenticated and authorized control over the requested resources represented by the "NfInstanceId" value as well as any other NF profile parameters included in the certificate order.

The NF's ACME client responds to the challenge by posting the Authority Token, as received from the OAM system, to the challenge URL identified in the returned ACME authorization object, an example of which follows:

POST /acme/chall/prV\_B7yEyA4 HTTP/1.1

Host: boulder.example.com

Content-Type: application/jose+json

{

 "protected": base64url({

 "alg": "ES256",

 "kid": "https://example.com/acme/acct/evOfKhNU60wg",

 "nonce": "Q\_s3MWoqT05TrdkM2MTDcw",

 "url": "https://boulder.example.com/acme/authz/asdf/0"

 }),

 "payload": base64url({

 "tkauth": "DGyRejmCefe7v4N...vb29HhjjLPSggwiE"

 }),

 "signature": "9cbg5JO1Gf5YLjjz...SpkUfcdPai9uVYYQ"

}

The "tkauth" field is, as defined in RFC 9448 [XE], a field in the challenge object specific to the tkauth-01 challenge type that contains an Authority Token as defined in the next clause.

#### YY.3.3.3 NF Certificate Authority Token

A new Authority Token profile, NF Certificate Authority Token, is defined in this clause. The NF Certificate Authority Token is a profile instance of the ACME Authority Token defined in RFC 9447 [XB].

The NF Certificate Authority Token protected header meets the requirements for "Request Authentication", as specified in RFC 8555 [XX], section 6.2.

The NF Certificate Authority Token payload includes the mandatory claims "exp", "jti", and "atc":

- "exp" claim, defined in RFC 7519 [XC], section 4.1.4, is included and contains the DateTime value of the date and time that the NF Certificate Authority Token expires.

- "jti" claim, defined in RFC 7519 [XC], section 4.1.7, is included and contains a unique identifier for this NF Certificate Authority Token transaction.

- "atc" claim, defined in RFC 9447 [XB], is included and contains a JSON object with the following mandatory elements:

- "tktype" key with a string value equal to "NfInstanceId" to identify this as a NF Instance ID claim.

- "tkvalue" key with a string value equal to value of the "NfInstanceId".

- "fingerprint" key constructed as defined in RFC 8555 [XX], section 8.1, corresponding to the computation of the "Thumbprint" step using the ACME account key credentials.

Additional elements for additional NF profile parameters can optionally be included, per RFC 9447 [XB], section 4. These include the following:

- "nftype" key with a string value equal to an NF Type as defined in RFC 9310 [XF],

- "sans" key with value of an array of identifiers, as defined in clause 6.1.3c, to be included as SANs in addition to the NF Instance ID.

An example of the NF Certificate Authority Token is as follows:

{

 "protected": base64url({

 "typ":"JWT",

 "alg":"ES256",

 "x5u":"https://authority.example.org/cert"

 }),

 "payload": base64url({

 "exp":1640995200,

 "jti":"id6098364921",

 "atc":{"tktype":"NfInstanceId",

 "tkvalue":"4ace9d34-2c69-4f99-92d5-a73a3fe8e23b",

 "fingerprint":"SHA256 56:3E:CF:AE:83:CA:4D:15:B0:29:FF:1B:71:

 D3:BA:B9:19:81:F8:50:9B:DF:4A:D4:39:72:E2:B1:F0:B9:38:E3",

 "nftype" : "AMF",

 "sans" : ["amf1234.mcc.mnc.3ggp.org"]}

 }),

 "signature": "9cbg5JO1Gf5YLjjz...SpkUfcdPai9uVYYQ"

}

The Authority Token is acquired by the NF using a RESTful HTTP POST transaction as follows:

POST /at/account/:id/token HTTP/1.1

Host: authority.example.org

Content-Type: application/json

The request includes the account identifier as a string in the request parameter "id". This string is managed as an identifier specific to the Token Authority's relationship with an operator CA.

The body of the POST request contains a JSON object with key value pairs corresponding to values that are requested as the content of the claims in the issued token. An example is as follows:

{

 "tktype":"NfInstanceId",

 "tkvalue":"4ace9d34-2c69-4f99-92d5-a73a3fe8e23b",

 "fingerprint":"SHA256 56:3E:CF:AE:83:CA:4D:15:B0:29:FF:1B:71:D3

 :BA:B9:19:81:F8:50:9B:DF:4A:D4:39:72:E2:B1:F0:B9:38:E3",

 "nftype" : "AMF",

 "sans" : ["amf1234.mcc.mnc.3ggp.org"]}

}

If successful, the response to the POST request returns a 200 (OK) with a JSON body that contains, at a minimum, the NF Certificate Authority Token as a JSON object with a key of "token" and the base64url-encoded string representing the atc token. An example of a successful response is as follows:

HTTP/1.1 200 OK

Content-Type: application/json

{"token": "DGyRejmCefe7v4N...vb29HhjjLPSggwiE"}

If the request is not successful, the response indicates the error condition. Specifically, for the case that the authorization credentials are invalid or if the account identifier provided does not exist, the response code 403 (Forbidden) is returned. Other 4xx and 5xx responses follow standard HTTP error condition conventions, as described in RFC 9110 [65].

When creating the NF Certificate Authority Token, the Token Authority validates that the information contained in the token accurately represents the NF Instance ID and additional NF profile parameters the requesting party is authorized to represent based on their pre-established, verified, and secure relationship. Note that the fingerprint in the token request is not meant to be verified by the Token Authority but rather is meant to be signed as part of the token so that the party that requests the token can, as part of the challenge response, allow the ACME server to validate that the token requested and used came from the same party that controls the ACME client.

#### YY.3.3.4 Validation of NF Certificate Authority Token

Upon receiving a response to the challenge, the operator CA's ACME server performs the following steps to determine the validity of the response.

- Verify that the value of the "atc" claim is a well-formed JSON object containing the mandatory key values.

- If there is an "x5u" parameter, verify the "x5u" parameter is an HTTPS URL with a reference to a certificate representing the trusted issuer of Authority Tokens for the ecosystem (i.e., the OAM system), as described in RFC 7515 [XD], section 4.1.5.

- If there is an "x5c" parameter, verify the certificate array contains a certificate representing the trusted issuer of Authority Tokens for the ecosystem (i.e., the OAM system), as described in RFC 7515 [XD], section 4.1.6.

- Verify the NF Certificate Authority Token signature using the public key of the certificate referenced by the token's "x5u" or "x5c" parameter.

- Verify that an "atc" claim contains a "tktype" identifier with the value "NfInstanceId", a "tkvalue" identifier with an "NfInstanceId" value matching the identifier specified in the original challenge, a "fingerprint" that is valid and matches the account key of the client making the request, and optional elements corresponding to any addition NF profile parameters included in the certificate order (e.g., NF Type or SAN).

- Verify that the remaining claims are valid (e.g., verify that token has not expired).

#### YY.3.3.5 Use of JSON Web Signature

JSON Web Signature (JWS) objects, as defined in RFC 7515 [XD], can include an "x5u" header parameter to refer to a certificate that is used to validate the JWS signature. The URLs used in "x5u" are expected to provide the required certificate in response to a GET request, not a POST-as-GET, as required for the "certificate" URL in the ACME order object. This generally requires the ACME client to download the certificate and host it on a public URL to make it accessible to relying parties. RFC 9448, section 7 [XE], defines an optional mechanism for the certification authority (CA) to host the certificate directly and provide a URL that the ACME client owner can directly reference in the "x5u" of their signed NfInstanceId.

The following is an example of the use of "x5u" in the response when the certificate status is "valid".

HTTP/1.1 200 OK

Content-Type: application/json

Replay-Nonce: CGf81JWBsq8QyIgPCi9Q9X

Link: <https://example.com/acme/directory>;rel="index"

Location: https://example.com/acme/order/TOlocE8rfgo

{

 "status": "valid",

 "expires": "2024-05-20T14:09:07.99Z",

 "notBefore": "2024-05-01T00:00:00Z",

 "notAfter": "2024-05-08T00:00:00Z",

 "identifiers": [

 "type":"NfInstanceId",

 "value":"4ace9d34-2c69-4f99-92d5-a73a3fe8e23b"

 ],

 "authorizations": ["https://sti-ca.com/acme/authz/1234"],

 "finalize": "https://example.com/acme/order/TOlocE8rfgo/finalize",

 "certificate": "https://example.com/acme/cert/mAt3xBGaobw",

 "x5u": "https://example.com/cert-repo/giJI53km23.pem"

}

## YY.4 Certificate revocation

### YY.4.1 Introduction

The ACME protocol [XX] supports an automated revocation of ACME enrolled and renewed certificates using established authenticated and authorized credentials (i.e., key pair) verified during ACME client account activation and certificate issuance. An end entity (e.g., ACME client in the NF) can use its account key pair or the key pair of the issued certificate to request revocation of its certificate by the CA (i.e., ACME server).

There are several practices associated with certificate revocation that are recommended.

- This client-side certificate revocation procedure does not impact existing CA or OAM initiated revocation mechanisms, which are based on operator’s implementation and outside the scope of this solution. The CA operator will continue to have the ability to revoke certificates that have been issued. In addition, production and distribution of certificate revocation status (i.e., via CRL or OCSP) is solely dependent on CA operator’s implementation.

- The ability to revoke certificates is limited to the original enrolling ACME client or if the ACME client has knowledge of the certificate private key.

- When the ACME client is co-located with the NF in 5G SBA, the ACME client should not have privileges to request certificate revocation for other NFs.

- The ACME client should maintain the valid account key pair for the NF Instance ID for which the certificate was issued and/or access to the key pair of the issued certificate being requested for revocation to properly sign the revocation request. Maintenance of the account key pair in a secure manner is left to implementation.

- The certificate being requested for revocation should not have expired. Since expired certificates cannot be used, there is no justification to revoke expired certificates.

- If the NF has been compromised and the ACME client is co-located, access to the ACME client may not be possible. In such instances, certificate revocation would use existing server-side operator’s implementation.

### YY.4.2 ACME Certificate Revocation Procedure

Prerequisites of the ACME certificate revocation procedure:

- ACME client has been established per clause YY.2.2 (Account creation)

- Certificate has been enrolled or renewed by the CA (i.e., ACME server) and has not expired

The revocation request contains the certificate to be revoked and, optionally, the reason code for certificate revocation. Valid revocation reason codes are defined in CRLReason in RFC 5280 [14].

NOTE 1: To deny or accept revocation requests based on which reasonCode is left to the operator’s implementation. Use of the reasonCode is strongly encouraged.

NOTE 2: RFC 8555 [XX] includes optional revocation reason codes, such as superseded. These codes could provide an indication to the CA and further to the OAM in case that the CA is under control of the OAM.

To reduce potential impact of a DoS attack using compromised credentials, the CA should be configured to deny the ACME request for certificate revocation if the ACME client’s credentials have been compromised or are believed to be compromised.

\* \* \* End of Changes \* \* \* \*