|  |
| --- |
| 3GPP TR 33.776 V0.4.0 (2024-08) |
| Technical Report |
| 3rd Generation Partnership Project;Technical Specification Group Services and System Aspects;Study of Automatic Certificate Management Environment (ACME) for the Service Based Architecture (SBA) (Release 19) |
|   |
|  |  |
|  |
| The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP.The present document has not been subject to any approval process by the 3GPPOrganizational Partners and shall not be implemented.This Specification is provided for future development work within 3GPPonly. The Organizational Partners accept no liability for any use of this Specification.Specifications and Reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices. |

|  |
| --- |
|  |
| ***3GPP***Postal address3GPP support office address650 Route des Lucioles - Sophia AntipolisValbonne - FRANCETel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16Internethttp://www.3gpp.org |
| ***Copyright Notification***No part may be reproduced except as authorized by written permission.The copyright and the foregoing restriction extend to reproduction in all media.© 2024, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).All rights reserved.UMTS™ is a Trade Mark of ETSI registered for the benefit of its members3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational PartnersLTE™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational PartnersGSM® and the GSM logo are registered and owned by the GSM Association |

Contents

Foreword 6

Introduction 7

1 Scope 8

2 References 8

3 Definitions of terms, symbols and abbreviations 9

3.1 Terms 9

3.2 Symbols 9

3.3 Abbreviations 9

4 Assumptions 10

5 Key issues 10

5.1 Key issue #1: ACME initial trust framework 10

5.1.1 Key issue details 10

5.1.2 Security threats 10

5.1.3 Potential security requirements 10

5.2 Key issue #2: Secure transport of messages 10

5.2.1 Key issue details 10

5.2.2 Security threats 10

5.2.3 Potential security requirements 10

5.3 Key issue #3: Aspects of challenge validation 10

5.3.1 Key issue details 10

5.3.2 Security threats 11

5.3.3 Potential security requirements 11

5.4 Key issue #4: Certificate enrolment 11

5.4.1 Key issue details 11

5.4.2 Security threats 11

5.4.3 Potential security requirements 11

5.5 Key issue #5: Certificate renewal 12

5.5.1 Key issue details 12

5.5.2 Security threats 12

5.5.3 Potential security requirements 12

5.6 Key Issue #6: Certificate revocation 12

5.6.1 Key issue details 12

5.6.2 Security threats 12

5.6.3 Potential security requirements 12

5.7 Key issue #7: Supporting all 5G SBA certificate types 12

5.7.1 Key issue details 12

5.7.2 Security threats 13

5.7.3 Potential security requirements 13

5.X Key issue #X: <Title> 13

5.X.1 Key issue details 13

5.X.2 Security threats 13

5.X.3 Potential security requirements 13

6 Solutions 13

6.0 Mapping of solutions to key issues 13

6.1 Solution #1: Using NF FQDN as ACME identifier 13

6.1.1 Introduction 13

6.1.2 Solution Details 14

6.1.2.1 Procedure 14

6.1.3 Evaluations 15

6.2 Solution #2: Automated validation of certificate signing requests for network functions 16

6.2.1 Introduction 16

6.2.2 Solution details 16

6.2.2.1 Initial trust 16

6.2.2.2 New identifier type 17

6.2.2.3 Certificate issuance 17

6.2.2.4 NF Certificate Authority Token 20

6.2.2.5 Validation of NF Certificate Authority Token 21

6.2.2.6 Use of JSON Web Signature 21

6.2.3 Evaluation 22

6.3 Solution #3: Using NF instance ID as ACME identifier 22

6.3.1 Introduction 22

6.3.2 Solution details 22

6.3.2.1 Initial trust 23

6.3.2.2 Procedure 23

6.3.3 Evaluation 24

6.4 Solution #4: Reuse solution about policy-based certificate renewal 24

6.4.1 Introduction 24

6.4.2 Solution details 24

6.4.3 Evaluation 24

6.5 Solution #5: Using ACME protocol for certificate enrolment 25

6.5.1 Introduction 25

6.5.2 Solution details 25

6.5.2.1 Initial Trust 25

6.5.2.2 Certificate enrolment 25

6.5.3 Evaluation 27

6.6 Solution #6: ACME automated revocation of certificates 27

6.6.1 Introduction 27

6.6.2 Solution Details 27

6.6.3 Evaluation 28

6.7 Solution #7: Using ACME protocol for secure transport of messages 29

6.7.1 Introduction 29

6.7.2 Solution details 29

6.7.3 Evaluation 29

6.Y Solution #Y: <Title> 29

6.Y.1 Introduction 29

6.Y.2 Solution details 29

6.Y.3 Evaluation 29

7 Conclusions 30

Annex <X> : Change history 31

# Foreword

This Technical Report has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.

z the third digit is incremented when editorial only changes have been incorporated in the document.

In the present document, modal verbs have the following meanings:

**shall** indicates a mandatory requirement to do something

**shall not** indicates an interdiction (prohibition) to do something

The constructions "shall" and "shall not" are confined to the context of normative provisions, and do not appear in Technical Reports.

The constructions "must" and "must not" are not used as substitutes for "shall" and "shall not". Their use is avoided insofar as possible, and they are not used in a normative context except in a direct citation from an external, referenced, non-3GPP document, or so as to maintain continuity of style when extending or modifying the provisions of such a referenced document.

**should** indicates a recommendation to do something

**should not** indicates a recommendation not to do something

**may** indicates permission to do something

**need not** indicates permission not to do something

The construction "may not" is ambiguous and is not used in normative elements. The unambiguous constructions "might not" or "shall not" are used instead, depending upon the meaning intended.

**can** indicates that something is possible

**cannot** indicates that something is impossible

The constructions "can" and "cannot" are not substitutes for "may" and "need not".

**will** indicates that something is certain or expected to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

**will not** indicates that something is certain or expected not to happen as a result of action taken by an agency the behaviour of which is outside the scope of the present document

**might** indicates a likelihood that something will happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

**might not** indicates a likelihood that something will not happen as a result of action taken by some agency the behaviour of which is outside the scope of the present document

In addition:

**is** (or any other verb in the indicative mood) indicates a statement of fact

**is not** (or any other negative verb in the indicative mood) indicates a statement of fact

The constructions "is" and "is not" do not indicate requirements.

# Introduction

5G Service Based Architecture (SBA) is secured using X.509 certificates across the large number of SBA components and corresponding Network Functions (NFs). Virtualization and increased modularity of NFs has resulted in multi-vendor environments becoming more prevalent. It is now common for NFs to come from different vendors and for the cloud native environment in which they run to come from yet another vendor and for all of these to be independent of the Certificate Authority that is authoritative for the certificates used to secure communications. In such deployments, it is impractical to manage certificates manually.

Automated Certificate Management Environment (ACME) [2] was defined specifically for automated certificate management and is particularly well suited for some scenarios. Infrastructure deployment such as NFs deployed on cloud native platforms often have built-in support for ACME, so it is a natural fit. Another important benefit of ACME is automated validation of authority to represent an identifier (i.e., to be authoritative for the resource for which the certificate is issued). This is particularly helpful for multi-vendor environments and in cross-carrier scenarios.

Additional work is required to determine the feasibility of the use of ACME in 5G SBA.

# 1 Scope

The scope of this document is to identify key issues and study solutions addressed using ACME for automated certificate management in SBA.

Areas of study include:

- Automated certificate management protocol and procedures for certificate life cycle events (i.e., enrolment,  renewal, and revocation) within 5G SBA (i.e., to be used by operator CAs and all 5GC NFs including NRF,  SCP, SEPP, etc.), including the following:

- ACME transport and request/response messages for 5G SBA use cases

- ACME certificate profiles for all 5G SBA entities

- Mechanisms for establishing initial trust and chain of trust of Certificate Authority hierarchies, including the  following:

- Existing ACME challenge types and if any new challenge types are needed for 3GPP use cases:

- Creation, deletion, rotation, revocation and storage of the certificates

- Ability to automate ACME challenge validation

- Suitability of existing mechanisms when 5G SBA is for standalone NPN (SNPN)

- Call flow of the messages exchanged between different entities in the chain of trust.

NOTE: Certificate management for the external interface of the SEPP is out of scope.

# 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 TR 21.905: "Vocabulary for 3GPP Specifications".

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

[3] 3GPP TS 33.310: "Network Domain Security (NDS); Authentication Framework (AF) ".

[4] IETF RFC 8738: "Automated Certificate Management Environment (ACME) IP Identifier Validation Extension".

[5] IETF RFC 8739: "Support for Short-Term, Automatically Renewed (STAR) Certificates in the Automated Certificate Management Environment (ACME)".

[6] IETF RFC 8823: "Extensions to Automatic Certificate Management Environment for End-User S/MIME Certificates".

[7] SP-231787: "New Study of ACME for Automated Certificate Management in SBA".

[8] 3GPP TS 33.501: "Security architecture and procedures for 5G System".

[9] [IETF RFC 9447](https://datatracker.ietf.org/doc/html/rfc9447): "Automated Certificate Management Environment (ACME) Challenges Using an Authority Token".

[10] [IETF RFC 9448](https://datatracker.ietf.org/doc/html/rfc9448): "TNAuthList Profile of Automated Certificate Management Environment (ACME) Authority Token".

[11] [3GPP TS 23.502](https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3145): "Procedures for the 5G System (5GS)".

[12] [IETF RFC 7519](https://datatracker.ietf.org/doc/html/rfc7519): " JSON Web Token (JWT)".

[13] [3GPP TS 29.571](https://portal.3gpp.org/desktopmodules/Specifications/SpecificationDetails.aspx?specificationId=3347): "5G System; Common Data Types for Service Based Interfaces; Stage 3".

[14] [IETF RFC 9110](https://datatracker.ietf.org/doc/html/rfc9110): "HTTP Semantics".

[15] [IETF RFC 7515](https://datatracker.ietf.org/doc/html/rfc7515): "JSON Web Signature (JWS)".

[16] [IETF RFC 4122](https://datatracker.ietf.org/doc/html/rfc4122): "Universally Unique IDentifier (UUID) URN Namespace".

[17] 3GPP TS 23.003: "Numbering, addressing and identification".

[18] [IETF RFC 5280](https://datatracker.ietf.org/doc/html/rfc5280): “Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile”.

# 3 Definitions of terms, symbols and abbreviations

This clause and its three subclauses are mandatory. The contents shall be shown as "void" if the TS/TR does not define any terms, symbols, or abbreviations.

## 3.1 Terms

For the purposes of the present document, the terms given in 3GPP TR 21.905 [1] and the following apply. A term defined in the present document takes precedence over the definition of the same term, if any, in 3GPP TR 21.905 [1].

Definition format (Normal)

**<defined term>:** <definition>.

**example:** text used to clarify abstract rules by applying them literally.

## 3.2 Symbols

For the purposes of the present document, the following symbols apply:

<symbol> <Explanation>

## 3.3 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [1].

CA Certificate Authority

NPN Non-Public Network

NRF Network Repository Function

SCP Service Communication Proxy

SEPP Security Edge Protection Proxy

SNPN Stand-Alone Non-Public Network

# 4 Assumptions

Clause 10 of TS 33.310 [3] specifies a framework for certificate provisioning and managements for 5G NFs. Though the enrolment protocol is CMPv2, many of the procedures, such as those for initial trust establishment and for certificate revocation status verification, are independent of the enrolment protocol. Therefore, many of the procedures are expected to be re-used.

# 5 Key issues

Editor’s Note: This clause contains all the key issues identified during the study.

## 5.1 Key issue #1: ACME initial trust framework

### 5.1.1 Key issue details

For automated certificate management in SBA, ACME requires the operator root certificates to be pre-installed and trusted. Solutions should take this into account.

ACME’s initial trust framework for asserting the certificate requesting client’s identity before issuing security credential is to be studied in this key issue.

### 5.1.2 Security threats

Not applicable.

### 5.1.3 Potential security requirements

Not applicable.

## 5.2 Key issue #2: Secure transport of messages

### 5.2.1 Key issue details

The ACME automated certificate management protocol provides procedures and recommendations to support different aspects of the certificate lifecycle [2]. Using ACME for automated certificate management in SBA, would require messages to be integrity protected, confidentiality protected, replay protected, and mutually authenticated.

### 5.2.2 Security threats

Not applicable.

### 5.2.3 Potential security requirements

Not applicable.

## 5.3 Key issue #3: Aspects of challenge validation

### 5.3.1 Key issue details

The objective of this key issue is to identify and evaluate suitable ACME challenge types for use within the 5G SBA. This includes new challenge types to address different NF types, and when challenges are not necessary.

Challenges require the client to have an identifier. The ACME protocol supports the issuance of certificates with domain names, IP addresses, or email address as subject identifiers. More precisely, according to the current ACME protocol specifications [2][4][5][6], the protocol can be used for the following purposes: Issuance of Web PKI certificates attesting to domain name or IP addresses, issuance of Short-Term Automatically Renewed (STAR) X.509 certificates, issuance of certificates for use by email users (S/MIME), issuance of STI (Secure Telephone Identity) certificates, and issuance of end user client and code signing certificates. However, in SBA, the NF instance ID is used as the unique identifier for NF instances. In addition, based on the current provisions of TS 33.310 [3], the use of IP addresses only is not allowed.

In the ACME protocol, RFC 8555 [2], the DNS challenge is specified when the ACME identifier is a domain name. The ACME client is required to show control of a given domain by updating the corresponding domain name directory on the DNS server with content specified by the ACME server. However, in the core network of a 3GPP system, a DNS server is managed by the operator. An NF, if taking the role of the ACME client, is not authorized to make changes to a DNS server.

As noted, ACME is tailored to automated certificate validation for server-side certificates. ACME challenges suitable for TLS client certificates will require study.

Editor’s Note: The requirement to include ACME challenges for other certificate types is FFS

### 5.3.2 Security threats

Not applicable.

### 5.3.3 Potential security requirements

Not applicable.

## 5.4 Key issue #4: Certificate enrolment

### 5.4.1 Key issue details

The ACME automated certificate management protocol provides procedures and identifies solutions to support authentication to the enrolment server CA and secure message protocol to protect ACME message exchanges during the certificate enrolment process against replay and confidentially protection. To address the objectives of this study [7] there is a requirement to identify procedures and solutions to use ACME across the 5GC SBA for different scenarios (e.g., multi-vendor integration) and use cases (e.g., authentication of domain names, HTTPS, mutual TLS authentication). Procedures and solutions for automated certificate enrolment to consider for this key issue include:

- Support for ACME client and authentication

- Certificate signing request (CSR) – content and creation of request

- CSR submission

- Certificate issuance

This KI is to identify ACME certificate enrolment procedures and solutions for different use cases for the 5GC SBA.

### 5.4.2 Security threats

Not applicable.

### 5.4.3 Potential security requirements

Not applicable.

## 5.5 Key issue #5: Certificate renewal

### 5.5.1 Key issue details

The ACME automated certificate management protocol provides procedures and recommendations to support different aspects of the certificate lifecycle [2]. Certificate renewal is the process of issuing a new digital certificate for an existing certificate that needs to be reissued (e.g., when a certificate is about to expire or if the certificate has been compromised). Certificate renewal may be conducted for a variety of other reasons, such as if a certificate needs to be changed or updated due to changes in the NF or network domain. In addition, the certificate that was replaced is revoked to prevent the potential for unauthorized use.

This KI is to identify ACME certificate renewal procedures and solutions in the 5GC SBA. In addition, the certificate expiration period and renewal interval need to be set appropriately against potential security threats while reducing certificate management overhead and associated risk (e.g., certificates expiring prior to being renewed).

### 5.5.2 Security threats

Not applicable.

### 5.5.3 Potential security requirements

Not applicable.

## 5.6 Key Issue #6: Certificate revocation

### 5.6.1 Key issue details

The ACME automated certificate management protocol [2] provides procedures and recommendations to support automated certificate revocation. Certificate revocation is the process of revoking a digital certificate so that it can no longer be used prior to expiration. ACME will use existing certification revocation status checking profiles that have been specified in TS 33.310 [3] such as CRL specified in clause 6.1a and OCSP specified in clause 6.1b. Revocation may be conducted for a variety of reasons, such as a compromise of the certificate’s private key or changes to underlying parameters such as the domain name. This KI is to study the ACME automated certificate revocation procedures, namely certificate revocation requests from the ACME client, as part of the management lifecycle in the 5GC SBA.

NOTE: Study on new certification revocation status procedure profiles beyond the existing set in clause 6.1 in TS 33.310 [3] are out of scope.

### 5.6.2 Security threats

Not applicable.

### 5.6.3 Potential security requirements

Not applicable.

## 5.7 Key issue #7: Supporting all 5G SBA certificate types

### 5.7.1 Key issue details

According to RFC 8555 [2], the ACME protocol was originally designed for the provisioning and management of TLS/SSL certificates for web servers. It is worth noticing that in the 5G Core, there are other types of certificates, such as TLS client certificates and OAuth 2.0 token signing certificates according to TS 33.310 [3]. Extensions beyond Web Server TLS already exist for ACME, and further extensions make sense to support 5G core certificates.

The scope of the key issue is to address the extension of ACME to support 5G core certificates.

### 5.7.2 Security threats

Not applicable.

### 5.7.3 Potential security requirements

Not applicable.

## 5.X Key issue #X: <Title>

### 5.X.1 Key issue details

### 5.X.2 Security threats

### 5.X.3 Potential security requirements

# 6 Solutions

Editor’s Note: This clause contains the proposed solutions addressing the identified key issues.

## 6.0 Mapping of solutions to key issues

Table 6.0.1: Mapping of solutions to key issues

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Solution | KI#1 | KI#2 | KI#3 | KI#4 | KI#5 | KI#6 | KI#7 |
| Solution #1: Using NF FQDN as ACME identifier |  |  | X |  |  |  |  |
| Solution #2: Automated validation of certificate signing requests for network functions | X |  | X |  |  |  |  |
| Solution #3: Using NF instance ID as ACME identifier |  |  | X |  |  |  |  |
| Solution #4: Reuse solution about policy-based certificate renewal |  |  |  |  | X |  |  |
| Solution #5: Using ACME protocol for certificate enrolment |  |  |  | X |  |  |  |
| Solution #6: ACME automated revocation of certificates |  |  |  |  |  | X |  |
| Solution #7: Using ACME protocol for secure transport of messages |  | X |  |  |  |  |  |
|  |  |  |  |  |  |  |  |

## 6.1 Solution #1: Using NF FQDN as ACME identifier

### 6.1.1 Introduction

This solution addresses the key issue #3.

The ACME protocol defined in the RFC 8555 [2] was designed to help a web server to get a domain name certificate from a CA automatically. However, in the current operator networks, an NF instance ID certificate is preferred since the NF instance ID is used to uniquely identify an NF. In this solution, the NF FQDN is linked to the NF instance ID so that the ACME protocol with domain name can be re-used for NF certificate management.

### 6.1.2 Solution Details

In 5G SBA, an NF is uniquely identified by an NF instance ID. The NF profile can also contain a FQDN. In order to link the NF instance ID with its FQDN and re-use the ACME protocol based on a domain name, it is proposed to use an NF instance ID to form part of the NF’s domain name, e.g. NF\_instance\_ID. NF\_types.operators\_name.3gpp.org etc. The format of NF\_instance\_ID is as defined in TS 29.571 [13].

For example, according to TS 23.003 [17], an FQDN for an NF, e.g. AMF, can be constructed as:

- AMF: <AMF-id>.amf.5gc.mnc<MNC>.mcc<MCC>.3gppnetwork.org

Similarly, the FQDN of an NF in this solution with NF instance ID can be constructed the same way as:

 - < NF\_instance\_ID >. <NFType>.5gc.mnc<MNC>.mcc<MCC>.3gpp.org. e.g. for an AMF, <AMF’s NF\_instance\_ID>.amf.5gc.mnc123.mcc456.3gpp.org.

Through the challenge and response procedure of the ACME protocol RFC 8555 [2], the ACME client is able to prove to the ACME Server that it is authoritative for its FQDN.

#### 6.1.2.1 Procedure

When the http-01 challenge type is used, it is required that the ACME client is authorized to control the */.well-known/acme-challenge/* directory on the http server corresponding to the domain name.

Figure 6.1.2.1 shows a high-level procedure for NF to obtain certificates from CA with ACME procedures (for simplicity, it is assumed that an NF takes the role of the ACME client). The procedure is as follows:

1. After the NF is deployed, it starts the ACME client and performs following the steps for certificate issuing based on RFC 8555 [2].

2. The ACME client on the NF chooses a CA as configured by the OAM and creates an ACME account as in RFC 8555 [2].

3. The ACME client creates a certificate order on the CA. To confirm that the ACME client is authorized to delegate the identifiers, the ACME server at the CA side generates challenges for the ACME client to complete.

4. The ACME client downloads the challenge from the ACME server, choose one of the challenge types, e.g. http-01 and complete the challenge according to RFC 8555 [2].

5. After the ACME client completes the challenge successfully, the ACME client is authorized to request and receive a certificate for its FQDN. To receive the certificate, the ACME client needs to send a Certificate Signing Request (CSR) to the ACME server.

6. After receiving the CSR, CA issues the certificates and puts it under the relevant directory on the ACME server. The FQDN in the certificate contains the NF\_instance\_ID.

7. The ACME client downloads the certificate from the ACME server.

ACME Server/CA

NF

2. ACME client creates an account

3. ACME client creates a certificate order for the domain name

1. NF starts the ACME client

5. ACME client sends the CSR to the server

Standard procedures from RFC 8555

4. ACME client completes the authorizations (http-01/dns-01) of the order

6. CA issues the certificate

7. ACME client download the certificate from server

Figure 6.1.2.1.1: ACME procedure for NF certificate management

### 6.1.3 Evaluations

This solution addresses they key issues #3.

The solution is limited to NF producers if it assumes control over HTTP resources for the http-01 challenge. The solution can be extended to support the dns-01 challenge if the DNS resource record can be updated by the NF or OAM.

This solution with the dns-01 challenge is feasible only if the DNS resource record can be updated by the ACME client and that DNS server should be accessible (i.e., record can be retrieved) by the ACME server. In order for an NF to modify its DNS records, the NF needs to be granted such privileges.

In order to not impact ACME, the solution requires changes to the current SBA certificate profiles so that an FQDN formed based on the NF instance ID can be used as an identifier value for the challenge. Observe that the standard impact is not only limited to the profile since there are also requirements for NF instance ID checks based on what is included in the certificate for example in TS 33.501 [8].

## 6.2 Solution #2: Automated validation of certificate signing requests for network functions

### 6.2.1 Introduction

This contribution proposed a solution that addresses the following key issues:

- Key Issue #1 - ACME initial trust framework, and

- Key Issue #3 - Aspects of challenge validation.

### 6.2.2 Solution details

This solution enables a 5GC network function (NF) to use ACME [2] to obtain certificates it can use to establish secure connections within the Service Based Architecture (SBA).

#### 6.2.2.1 Initial trust

Automated certificate management using ACME reuses the initial trust schema defined in TS 33.310 [3], as shown in Figure 6.2.2.1.1.



Figure 6.2.2.1.1: Initial trust schema

The Operations, Administration and Maintenance (OAM) system instantiates the NF, providing it with the initial trust needed for certificate enrollment with the operator CA/RA. The NF instance ID, which uniquely identifies the NF within the 5GC, is assigned to the NF by the OAM system as part of its NF profile, as specified in section 4.17 of TS 23.502 [11].

According to TS 33.310 [3], initial trust for certificate management of 5GC NFs may be provided using any of the following:

a) OAM issued certificate,

b) Initial Authentication Key (IAK), or

c) OAM issued signature of certain NF profile parameters, at least including the NF instance ID.

This solution assumes that when using ACME, option (c) is supported, and it serves as the basis of the Authority Token used for ACME challenge validation. As shown in Figure 6.2.2.1.2, the NF acts as the ACME client, the Operator CA/RA acts as the ACME server, and the OAM system acts as a Token Authority. The set of NF profile parameters signed by the OAM and included in the Authority Token includes the NF instance ID. Including additional NF profile parameters that the NF is authorized to include in its certificate can simplify interaction between the OAM and Operator CA/RA.



Figure 6.2.2.1.2: Initial trust schema with ACME

An ACME client authenticates to the ACME server by means of an "account key pair", as defined in [2]. The client uses the private key of this key pair to sign all messages sent to the server. The server uses the public key to verify the authenticity and integrity of messages from the client. The NF can generate its own private/public key combination for use as an ACME client account key. Alternatively, this can be assigned by the OAM system.

The ACME challenge-type used is the ACME Authority Token challenge type, "tkauth-01", as specified in RFC 9447 [9]. The architecture associated with this 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.

#### 6.2.2.2 New identifier type

A new ACME identifier type, "nf-instance-id", is defined in this document. A NF uses its NF instance ID as the value of the “nf-instance-id". The format of the value of the "nf-instance-id" is defined to match that of the NfInstanceId, as defined in TS 29.571 [13]:

- 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 [16]. 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 "nf-instance-id" is as follows:

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

This new ACME identifier type needs to be listed in a new registration in the ACME Validation Methods registry maintained by IANA, per RFC 9447 [9], clause 3.

#### 6.2.2.3 Certificate issuance

Figure 6.2.2.3.1 provides a simplified message flow for certificate issuance using the ACME Authority Token challenge type as described in this solution.



Figure 6.2.2.3.1 ACME message flow for certificate issuance

The NF begins the certificate issuance process by sending a POST request to the CA's newOrder resource. The body of the POST is a JWS object whose JSON payload contains fields that describe the certificate to be issued, including the ACME identifiers.

In NF certificates, both client and server, subjectAltName contains the NfInstanceId as a "uniformResourceIdentifier" formatted as a URN as described in clause 5.3.2 of TS 29.571 [13]. 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.

A full ACME new-order request would look as follows:

POST /acme/new-order HTTP/1.1

Host: example.com

Content-Type: application/jose+json

{

 "protected": base64url({

 "alg": "ES256",

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

 "nonce": "5XJ1L3lEkMG7tR6pA00clA",

 "url": "https://example.com/acme/new-order"

 }),

 "payload": base64url({

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

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

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

 }),

 "signature": "H6ZXtGjTZyUnPeKn...wEA4TklBdh3e454g"

}

On receiving a valid new-order request, the CA's ACME server creates an authorization object, per RFC8555 [2], clause 7.1.4, containing the challenge that the NF's ACME client needs to satisfy to demonstrate authority for the identifiers specified by the new order (in this case, the "nf-instance-id"). The CA adds the authorization object URL to the "authorizations" field of the order object and returns the order object to the NF in the body of a 201 (Created) response.

HTTP/1.1 201 Created

Content-Type: application/json

Replay-Nonce: MYAuvOpaoIiywTezizk5vw

Location: https://example.com/acme/order/1234

{

 "status": "pending",

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

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

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

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

 "authorizations": [

 "https://example.com/acme/authz/1234"

 ],

 "finalize": "https://example.com/acme/order/1234/finalize"

}

On receiving the new-order response, the NF queries the referenced authorization object to obtain the challenges for the identifier contained in the new-order request, as shown in the following example request and response.

POST /acme/authz/1234 HTTP/1.1

 Host: example.com

 Content-Type: application/jose+json

 {

 "protected": base64url({

 "alg": "ES256",

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

 "nonce": "uQpSjlRb4vQVCjVYAyyUWg",

 "url": "https://example.com/acme/authz/1234"

 }),

 "payload": "",

 "signature": "nuSDISbWG8mMgE7H...QyVUL68yzf3Zawps"

 }

HTTP/1.1 200 OK

Content-Type: application/json

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

{

 "status": "pending",

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

 "identifier": {

 "type":"nf-instance-id",

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

 },

 "challenges": [

 {

 "type": "tkauth-01",

 "tkauth-type": "atc",

 "token-authority": "https://authority.example.org",

 "url": "https://example.com/acme/chall/prV\_B7yEyA4",

 "token": "IlirfxKKXAsHtmzK29Pj8A"

 }

 ]

}

When processing a certificate order containing an identifier of type "nf-instance-id", a CA uses the Authority Token challenge type of "tkauth-01" with a "tkauth-type" of "atc", as defined in RFC 9447 [9], to verify that the requesting ACME client has authenticated and authorized control over the requested resources represented by the "nf-instance-id" value.

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 [10], a field in the challenge object specific to the tkauth-01 challenge type that should contain an Authority Token as defined in the next section.

#### 6.2.2.4 NF Certificate Authority Token

A new Authority Token profile, NF Certificate Authority Token, is defined in this document. The NF Certificate Authority Token is a profile instance of the ACME Authority Token defined in RFC9447 [9].

The NF Certificate Authority Token protected header meets the requirements for "Request Authentication", as specified in RFC 8555 [2], clause 6.2.

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

- "exp" claim, defined in RFC7519 [12], clause 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 RFC7519 [12], clause 4.1.7, is included and contains a unique identifier for this NF Certificate Authority Token transaction.

- "atc" claim, defined in RFC 9447 [9], is included and contains a JSON object with the following 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 "nf-instance-id".

- "fingerprint" key constructed as defined in RFC8555 [2], clause 8.1, corresponding to the computation of the "Thumbprint" step using the ACME account key credentials.

Additional "atc" claims for additional NF profile parameters can be included, but an "atc" claim for the NF instance ID needs to be included.

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"}

 }),

 "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 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"

}

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 [14].

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.

#### 6.2.2.5 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 [15], clause 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 [15], clause 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 "nf-instance-id" value matching the identifier specified in the original challenge, and a "fingerprint" that is valid and matches the account key of the client making the request.

- Verify that the remaining claims are valid (e.g., verify that token has not expired and any additional "atc" claims are valid).

#### 6.2.2.6 Use of JSON Web Signature

JSON Web Signature (JWS) objects, as defined in RFC 7515 [15], 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 [10], 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 nf-instance-id.

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":"nf-instance-id",

 "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"

}

### 6.2.3 Evaluation

This solution addresses the following key issues:

- Key issue #1 - ACME initial trust framework, and

- Key issue #3 - Aspects of challenge validation.

The solution uses the existing initial trust schema, as defined in TS 33.310 [3], and illustrates how the components of the initial trust schema map to the corresponding components of ACME [2]. This minimizes the impact of adding support for ACME [2].

This solution relies on support for the third initial trust option in 33.310 [3], i.e., OAM issued signature of certain NF profile parameters. It does not support the other two options, i.e., OAM issued certificate or Initial Authentication Key (IAK).

The solution uses the definition and format of NfInstanceId, as defined in TS 29.571 [13], and describes how it can be used as an ACME identifier. The solution also describes how the NF instance ID can be used with the existing Authority Token challenge type, as defined in RFC 9447 [9]. This eliminates the need for any work within IETF.

The OAM system that acts as a Token Authority and interfaces with the NF/ACME client will have more exposure to NFs and increased load.

The solution recommends the inclusion of all NF profile parameters in both the Authority Token and the OAM issued signature. This approach would simplify the interaction between the OAM and Operator CA/RA.

The registration of a new ACME identifier can be done directly with IANA. The definition of the NF Certificate Authority Token as a profile instance of the ACME Authority Token can be provided directly within a 3GPP technical specification, i.e., TS 33.310 [3].

NOTE: Authority Token challenge type, "tkauth-01", is one of multiple validation methods used in ACME".

## 6.3 Solution #3: Using NF instance ID as ACME identifier

### 6.3.1 Introduction

This solution addresses the key issue #3.

The ACME protocol defined in the RFC 8555 [2] uses domain names or IP addresses as the ACME identifier. In this solution, the NF instance ID, which is the unique identifier of an NF, is used as the ACME identifier. The ACME procedure is amended accordingly.

### 6.3.2 Solution details

In this solution, the initial trust is used to prove ownership of resources to ACME server. It supports all three optional initial trust mechanisms specified in TS 33.310 [3].

### 6.3.2.1 Initial trust

In this solution, the initial trust schema defined in clause 10.2.2 of TS 33.310 [3] is reused. As shown in Figure 6.3.2.1, the NF acts as the ACME client, the Operator CA/RA acts as the ACME server, and the OAM system acts as a validation information Authority.

A new identifier "NF instance ID" is introduced in this solution. A new ACME challenge-type is also introduced, named as "NF instance ID". In this challenge type, the initial trust is used to prove ownership of resources to ACME server. The trust relationship between a CA and OAM (validation information authority) is assumed.



Figure 6.3.2.1 Initial trust schema

### 6.3.2.2 Procedure

Prerequisites of the procedure: the same as the prerequisites stated in clause 10.2.2 of TS 33.310 [3].

Figure 6.3.2.2 shows the amended ACME procedure when using an NF instance ID as the ACME identifier. It is assumed that the NF takes the role of an ACME client for simplicity (i.e. the ACME client may be a separate entity).

NOTE 1: If NF and ACME client are separate entities, communications between the NF and the ACME client shall be protected, e.g. TLS. This may require reuse of mechanisms defined in TS 33.310 [3] for the initial trust setup and communications between the end entity (NF) and OAM.

For simplicity, the CA is assumed to be co-located with the ACME server. It is also assumed that the communication between the NF and the ACME server is protected, e.g. TLS.

The amended ACME procedure is as follows:

1. An NF creates its account on the ACME server as described in RFC 8555 [2].

2. The NF sends a newOrder request as in RFC 8555 [2]. In addition, the request message includes the NF instance ID as the identifier.

NOTE 2: A new identifier “NF instance ID” is introduced as opposed to RFC 8555 [2].

3. The ACME server sends a challenge to the NF with the challenge type "NF instance ID".

NOTE 3: A new challenge type “NF instance ID” is introduced as opposed to RFC 8555 [2]. More details are described in step 4.

4. The NF sends the challenge response to the ACME server, which includes the NF instance ID and validation information for the ACME server to validate the NF (i.e. to prove the NF has control over the NF instance ID). The validation information can be any of the three options of NF initial trust information as in clause 10.2 of TS 33.310 [3], i.e., 1) OAM issued certificates, 2) an Initial Authentication Key (IAK), or 3) OAM issued signature of certain NF profile parameters, at least including the NF instance ID. All the mandatory parameters, e.g. NF Type etc as defined in the NF certificate profile in 33.310 [3] are included in the validation information.

5. The ACME server validates the challenge response as in RFC 8555 [2]. In addition, the ACME server validates the validation information in the challenge response. The ACME server validates the validation information the same way as verifying the initial trust options specified in clause 10.2 of TS 33.310 [3].

NOTE 4: The steps 2 to 5 contain changes to the RFC 8555 [2] (e.g. NF instance ID identifier, NF instance challenge, proof of NF control over the NF instance ID based on verifying the initial trust) that are not possible as of now.

6. NF sends to the ACME server a CSR request for its certificate. The ACME server verifies the CSR based on the outcome in step 5 and, if successful, issues the NF certificate including the NF instance ID.

**

Figure 6.3.2.2 ACME procedure with an NF instance ID as the ACME identifier

### 6.3.3 Evaluation

This solution requires a new ACME identifier "NF instance ID", a new challenge type for the "NF instance ID" identifier, and proof of NF control over the NF instance ID. This is currently not possible in RFC 8555 [2] as described from the step 2 to the step 5. Therefore, additional work is required in IETF.

## 6.4 Solution #4: Reuse solution about policy-based certificate renewal

### 6.4.1 Introduction

This contribution addresses key issue #5, and in particular the requirement that the certificate expiration period and renewal interval need to be set appropriately against potential security threats while reducing certificate management overhead and associated risk (e.g., certificates expiring prior to being renewed).

### 6.4.2 Solution details

This solution reuses the list of practical recommendations defined in Annex I.2 of TS 33.310 [3] for NF certificate renewal expiration period and renewal interval. An NF can act as an ACME client and use ACME [2] to interact with an operator CA that acts as an ACME server to renew its certificate.

### 6.4.3 Evaluation

This contribution addresses key issue #5 and proposes to reuse the solution we adopted in the study of FS\_ACM during Release 18. The solution depends on the preconfigured policy and internal implementation of the NF/CA.

## 6.5 Solution #5: Using ACME protocol for certificate enrolment

### 6.5.1 Introduction

This solution proposes to use the ACME protocol to address the requirements in key issue KI#4 (Certificate enrolment).

### 6.5.2 Solution details

#### 6.5.2.1 Initial Trust

This solution can assume that the initial trust has already been established via the initial trust schema defined in TS 33.310 [3], which is briefly described as follows.



Figure 6.5.2.1.1 Initial trust schema

As depicted in Figure 6.5.2.1.1, Operation, administration and maintenance (OAM) system has a preestablished trust with Operator CA/RA. An operator CA/RA can be a trusted third-party CA/RA, with which the 5GS has a pre-established trust. The OAM can configure the 5G Core NF with a list of trust anchors and with a private/public key pair to be used for ACME account creation. Alternatively, the 5G Core NF can generate its own key pair.

Note that the Operator CA/RA behaves as an ACME server and the 5G Core NF acts as an ACME client.

####  6.5.2.2 Certificate enrolment

Figure 6.5.2.2.1 describes the ACME certificate enrolment procedure for a 5G NF. Note that 5G Core NF can also be referred to as 5G NF.



Figure 6.5.2.2.1 – ACME certificate enrolment

1. The ACME client requests a certificate by sending a new order request for 5G SBA ACME Identifier to the CA’s newOrder resource using a POST request. 5G SBA ACME Identifier can be any ACME identifier shown to work with 5G SBA, e.g., Solution #1, Solution #2, Solution #3.

2. The ACME server responds with a 201 (Created) response that includes authorization objects with challenges to be satisfied as described in RFC 8555[2].

3. The ACME client checks the authorization objects within the response and completes the listed challenges before requesting the ACME server to sign the certificate as described in RFC 8555 [2]. Any challenge validation methods shown to work for 5G SBA can be included in this list.

4. After the ACME client successfully completes the challenge validation procedure, it sends a Certificate Signing Request (CSR) to the ACME server.

5. The ACME server issues the certificate and publishes it in the corresponding resource directory to the URL provided in the order object.

6. The ACME client downloads the certificate by sending a POST-as-GET request to the certificate URL provided.

NOTE: The 5G NF/client may proactively obtain authorization and may not have to perform challenge-response procedure in Steps 2 – Step 4 based on pre-authorization procedure described in clause 7.4.1 of RFC 8555 [2].

### 6.5.3 Evaluation

This solution addresses KI#4.

This solution impacts core network function, OAM and service protocols in the 5G core network.

The solution outlines how certificate enrolment in 5G SBA may be performed using the ACME protocol [2] with any ACME identifier type and corresponding ACME challenge validation types that are suited for use in 5G SBA deployments.

Editor’s Note: Further evaluation is FFS.

## 6.6 Solution #6: ACME automated revocation of certificates

### 6.6.1 Introduction

This solution addresses key issue #6 on certificate revocation.

The ACME protocol [2] defines automated revocation procedures 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. The 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 from the CA (i.e., ACME server).

NOTE: This client-side certificate revocation procedure does not impact existing CA 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 the revocation status messages (i.e., via CRL or OCSP) of the revoked certificates are solely dependent on CA operator’s implementation.

### 6.6.2 Solution Details

This solution proposes certificate revocation procedure specified in RFC 8555 [2] to revoke valid certificates before expiration.

The solution assumes:

- CRL and OCSP certificate revocation status checking profiles defined in TS 33.310 clause 6.1a and 6.1b, respectively, are reused [3].

- The certificate being requested for revocation has not expired.

- ACME client maintains the valid account key pair for the NF identifier 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.

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

- This solution does not impact the end entity certificate revocation procedure defined in TS 33.310 [3] in clause 10.5.

Figure 6.6.2.1 provides an overview of the ACME certificate revocation procedure, as summarized below:

1. To initiate the certificate revocation request, the ACME client generates a JWS object, in which the JSON payload contains the certificate to be revoked. The revocation request is signed using the account private key or the certificate private key.

2. The ACME client sends the revocation request to the ACME server. The reason for revocation is optional to include with valid reasonCode defined in RFC 5280 [18].

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

NOTE 2: RFC 8555 includes optional revocation reason codes, such as keyCompromise. 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.

3. The ACME server validates the revocation request by verifying that the private key used to sign the request is authorized to revoke the certificate. If the account private key was used, the request must come from the account to which the certificate was issued or the account that holds the authorization for all the identifiers in the certificate.

4. If the revocation request is deemed valid during Step 3, the ACME server sends status is OK message. If revocation fails, the ACME server returns an error. If the certificate was already revoked, the ACME server returns status that it has been already revoked.



Figure 6.6.2.1: Overview of ACME-based automated certificate revocation

### 6.6.3 Evaluation

This solution addresses key issue #6 and utilizes an automated certificate revocation procedure based on the ACME protocol.

Ability to revoke certificates is limited to the original enrolling NF ACME client or if the ACME client has knowledge of the certificate private key.

In scenarios where the NF has been compromised and 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.

An ACME client’s (5G core NF) ability to request revocation of its own certificate is a potential risk of DoS in a scenario where an adversary has gained control of the ACME client and uses this control to request revocation of the certificate used by the 5G core NF, making it unable to render its 5G SBA services. However, the CA is not required to honor the request, and unexpected revocation requests outside of the operator’s certificate management practices (e.g., the superseded reason code if no new certificate has been issued) can be used to detect abnormal NF behavior.

Use of end entity certificate revocation allows efficient automated management of NF certificate lifecycle.

## 6.7 Solution #7: Using ACME protocol for secure transport of messages

### 6.7.1 Introduction

This contribution addresses key issue #2.

### 6.7.2 Solution details

The solution assumes that the 5G NF is issued with the operator CA’s root certificate, which is used to validate the ACME server’s TLS certificate.

This solution is based on RFC 8555 [2] wherein the communication between ACME client and the ACME server are done over HTTPS for authentication and confidentiality.

When an ACME client fetches a resource from an ACME server, the server authenticates the requester and verify any access control as described in RFC 8555 [2].

ACME for 5G SBA uses JWS based integrity protection as described in RFC 8555 [2].

ACME for 5G SBA uses nonces to protect messages against replay-attacks. An ACME server maintains a list of nonces that it has issued and requires any signed request from the client to carry such a nonce as described in RFC 8555 [2].

### 6.7.3 Evaluation

This solution addresses KI#2.

This solution impacts 5G core network function and 5G OAM system.

All exchanges initiated by the ACME client meet the requirement for confidentiality, integrity protection and replay protection. Once the client has established initial trust, messages can be considered mutually authenticated. Depending on the method of initial trust establishment, all messages could be considered mutually authenticated.

The server is always authenticated to the client prior to sending any data from the client, and no certificates are issued until mutual authentication is established.

The ACME server may need to initiate exchanges with the client, e.g. for http-01 challenge, or the DNS server, for example.

Editor's Note: When these server-initiated exchanges require protection and how that would be achieved is FFS.

## 6.Y Solution #Y: <Title>

### 6.Y.1 Introduction

Editor’s Note: Each solution should list the key issues being addressed.

### 6.Y.2 Solution details

### 6.Y.3 Evaluation

Editor’s Note: Each solution should motivate how the potential security requirements of the key issues being addressed are fulfilled.

# 7 Conclusions

Editor’s Note: This clause contains the agreed conclusions that will form the basis for any normative work.

Annex <X> :
Change history

|  |
| --- |
| **Change history** |
| **Date** | **Meeting** | **TDoc** | **CR** | **Rev** | **Cat** | **Subject/Comment** | **New version** |
| 2024-02 | SA3#115 | S3-240207 |  |  |  | Skeleton | 0.0.0 |
| 2024-02 | SA3#115 | S3-240982 |  |  |  | Incorporate pCRs that add introduction (S3-240983), scope (S3-240987), and five key issues (S3-240998, S3-240997, S3-240984, S3-240985, S3-240986). | 0.1.0 |
| 2024-04 | S3#115-adhoc-e | S3-241536 |  |  |  | Incorporate pCRs that add assumptions (S3-241600), add two new key issues (S3-241133 and S3-241650 and), update one previous key issue (S3-241382), and add three new solutions (S3-241383, S3-241534, and S3-241539). | 0.2.0 |
| 2024-05 | S3#116 | S3-242440 |  |  |  | Incorporates pCRs that add one new solution (S3-242439) and update three existing solutions (S3-241950, S3-242445, and S3-242446). | 0.3.0 |
| 2024-08 | S3#117 | S3-243722 |  |  |  | Incorporates pCRs that update one existing solution (S3-243488) add three new solutions (S3-243486, S3-243487, and S3-243662) | 0.4.0 |