**3GPP TSG SA WG3 Security — S3#31**
**18 - 21 November 2003**
**Munich, Germany**

**S3-030730**

**Source:** **Nokia**

**Title:** **Subscriber Certificate Enrollment Protocol**

**Document for:** **Discussion/Decision**

**Agenda Item:** **6.9 (SCC)**

## 1. INTRODUCTION

Current draft TS for Support for Subscriber Certificates (TS SSC) [TS SSC] describes two potential solutions for enrollment protocol for subscriber certificates:

1. PKCS#10 with HTTP Digest Authentication and
2. Certificate Management Protocols (CMP).

These two alternatives were investigated in Alcatel [S3-030036] and Nokia [S3-030073] contributions. Also other solutions were investigated in Nokia contribution, but only these two solutions were agreed by SA3 as potential solutions. SSH contribution to San Francisco meeting [S3-030347] further refined the usage of CMPv2[1] for 3GPP subscriber certificate enrollment.

This contribution also considers two additional possibilities for enrollment which haven't been fully discussed in SA3:

3. Certificate enrollment specified by OMA, and
4. PKCS#10 with shared key TLS.

The certificate enrollment specified by OMA was discussed in some detail in a contribution by SchlumbergerSema to San Francisco meeting [S3-030355].

The shared key TLS approach is similar to HTTP Digest approach where the HTTP Digest authentication and integrity protection is replaced by shared secret TLS, which in addition to authentication and integrity protection also provides confidentiality.

A selection needs to be made about which enrollment protocol is best for subscriber certificates. This contribution discusses these solutions and proposes a selection.

## 2. DISCUSSION

### 2.1 Requirements for enrollment protocol

The following requirements must be fulfilled by the enrollment protocol [TS SSC] (note that the revision marks below denote the changes made to the existing requirements):

1. UE ~~is~~shall be able to request for subscriber's certification from the PKI portal that plays the role of the NAF over a network connection.
2. NAF ~~is~~shall be able to authenticate UE's certificate request.
3. UE ~~is~~shall be able to acquire an operator's CA certificate over the network connection.

[1] CMPv2 updates original CMP specification and is in internet draft status in IETF.

4. UE ~~is~~shall be able to authenticate the NAF response (i.e., operator CA certificate delivery).

5. The procedure ~~is~~shall be independent of the access network used.

6. The NAF shall have access to the subscriber profile to check the certification policies. This means that the Zn interface shall support for retrieving a subset of the subscriber profile.

7. The response and delivery of certificate to UE ~~must~~shall be within a few seconds after the initial certification request.

8. Certification request format shall be PKCS#10.

9. Certification response format shall be one of the following: a certificate, a pointer to the certificate, or a full certificate chain.

SA3 has discussed that enrollment protocol should be able to deliver a pointer to the issued subscriber certificate (e.g., an URL) since it potentially saves bandwidth. This requirement should be reflected in the [TS SSC].

## 2.2 Enrollment protocols

CMPv2 subset has been considered as one of the possible solutions for the enrollment protocol for subscriber certificate issuing. However, CMPv2 has the disadvantage of not supporting the delivery of a pointer to the issued certificate. This conflicts with requirement 9. Also, CMPv2 has not yet achieved RFC status and its progress in IETF is somewhat unclear; IESG has evaluated the draft and concluded that a revision is needed. However, the authors have not produced a new version as of writing this contribution. The latest version of CMPv2 internet draft is over 6 months old, and has therefore expired. Because of these reasons, CMPv2 subset cannot be selected as the solution for the subscriber certificate enrollment.

The other solutions are all based on PKCS#10 certificate request syntax. The suggested solutions are:

- PKCS#10 with HTTP Digest,
- PKCS#10 with shared key TLS, and
- OMA's enrollment.

They all are very similar. As said, all of them use PKCS#10 certificate request syntax encoded using base64 and which is sent to the PKI portal using an HTTP request. The difference is how the certificate is delivered to UE, and how the authentication of the certificate request is done.

3GPP plans [TS SSC] to deliver the subscriber certificate to the UE as base64 encoded certificate in the HTTP response. OMA supports this method as well, but also supports the delivery of a pointer to the issued subscriber certificate. This is defined in section 7.3.5 in [WPKI].

In the 3GPP case, the authentication of the enrollment is based on a shared secret established using GBA. Currently, either using HTTP Digest (as described in Annex A of [TS GBA]) or shared key TLS (e.g., as described in Nokia contribution [S3-030555]) could be used to authenticate the enrollment request. In OMA enrollment, the authentication is done in the browser, e.g., using login-password pair or using normal TLS where client is authenticated using, e.g., a device certificate.

In SchlumbergerSema contribution [S3-030355], it was stated that WIM serial number could be used as the transaction identifier (TID) during subscriber certificate enrollment.

Although this is a viable solution, the authentication of the subscriber's enrollment request should be based on GBA (i.e., AKA).

## 2.3 Solution

**Subscriber certificate enrollment**

To combine and reuse existing (and emerging) specifications from 3GPP and OMA, we suggest having the following sequence for subscriber enrollment:

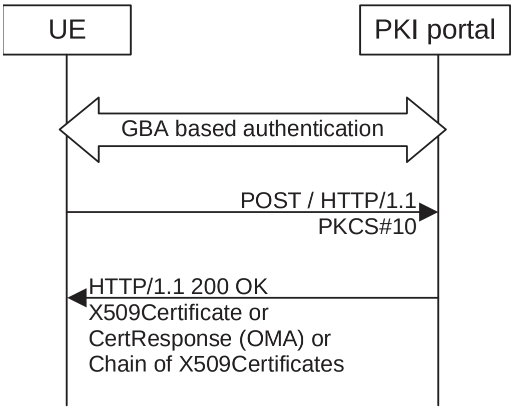


Figure 1. Subscriber certificate enrollment.

The authentication of the subscriber certificate enrollment is based on GBA.  This can be either based on HTTP Digest or shared key TLS. Whether GBA based authentication for subscriber certificate enrollment is based on HTTP Digest or shared key TLS is ffs.

The transport protocol for the PKCS#10 and certificate response messages is HTTP.

The subscriber certificate enrollment request is in the format of PKCS#10 and the content-type for the request is "application/x-pkcs10".

If the enrollment is successful, the response to the certificate enrollment contains one of the following:

- **X509Certificate**, the issued subscriber certificate base64 encoded, the content-type would be "application/x-x509-user-cert".
- **CertResponse structure** (as defined in [WPKI]) base64 encoded, the content type would be "application/vnd.wap.cert-response".
- **Chain of X509Certificates**, a certificate chain from the issued subscriber to the root CA certificate, the content-type is ffs.

UE indicates to the PKI portal the desired response type between single certificate, full certificate chain, or a pointer to the certificate by using the HTTP request line:

`http://<base` URL>?response=<indication>[other URL parameters]

where

`<base URL>` identifies a server/program.

`<indication>` indicates the UE's desired response type for the enrollment. Possible values are "single" for subscriber certificate only, "pointer" for pointer to the subscriber certificate, and "chain" for full certificate chain from the issued subscriber certificate to the root certificate.

`[other URL parameters]` are additional, optional, URL parameters.

However, PKI portal may reject the response type indicated by the UE, and use the one it desires.

The changes to the current TS SSC are in the way certificate is sent back from the PKI portal to the UE. In addition to pure certificate, response may also contain the full certificate chain from the issued subscriber certificate to the root certificate, or contain CertResponse structure, which enables the sending of a pointer to the issued certificate instead of certificate itself. The possibility to receive the full certificate chain reduces the number of roundtrips between UE and the PKI portal, because UE does not need to request the CA certificate separately (which is described below).

The certificate chain would be just a list of base64 encoded certificates which are separated by using "-----BEGIN CERTIFICATE-----" and "-----END CERTIFICATE-----" tags. This method of indicating the beginning and the end of a base64-encoded certificate is quite common in the Internet today.

**CA certificate delivery**

CA certificate delivery is done as it is described in currently TS SSC section 4.4.2.1. It is assumed that any root certificate delivered through a GBA authenticated channel is by default trusted, and hence the additional procedures to protect the CA certificate delivery defined in chapter 7.1 in [WPKI] are not needed.

CA certificate delivery can be further refined by using the Client Certificate URL format defined in chapter 7.4 of [WPKI]. In that section, the format of the client certificate URLs are defined which can be based on either HTTP scheme or LDAP scheme. Below, we concentrate on the HTTP scheme. According to [WPKI], the format of the HTTP based certificate should be as follows:

```
http://<base URL>?in=<issuer name>&sn=<serial number>[other URL
parameters]
```

where

`<base URL>` identifies a server/program;

`<issuer name>` identifies the certificate issuer. It is a base64 encoding of the DER encoded Issuer field in the X.509 certificate.

`<serial number>` identifies the serial number of the certificate. It is a base64 encoding of the DER encoded serialNumber in the X.509 certificate.

`[other URL parameters]` are additional, optional, URL parameters.

We propose to use this HTTP scheme to address a particular CA certificate (or any other certificate). From the above-mentioned parameters only <base URL> and <issuer name> are used to address a certain CA certificate, which can be identified by using the issuer name in a certificate. The <serial number> parameter is not needed, and thus is omitted.

For example, when UE has received a subscriber certificate, it can check whether if it already has the corresponding issuer certificate identified by the issuer field in the subscriber certificate. If it doesn't, it would use this method to retrieve the issuer certificate (i.e., CA certificate). This method is particularly useful, if the PKI portal is hosting more than one CAs or if the certificate chain consists of more than two certificates, i.e., there are intermediate CA certificates.

## 2.4 Conclusion

The enrollment of subscriber certificates as specified above, has several synergies with the OMA based enrollment:

- 3GPP specifications are in line with OMA specifications.
- OMA's PKI portal may be also used in subscriber certificate enrollment provided that PKI portal is able to do GBA base authentication, i.e., it assumes the role of a NAF.
- Code from OMA based enrollment may be reused on the UE for doing the subscriber certificate enrollment.

## 3. PROPOSAL

SA3 is asked to endorse the subscriber certificate enrollment procedures described in this contribution section 2.3 as the working assumption for TS SSC.

## REFERENCES

[TS GBA] Draft 3GPP TS 33.220: "Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture", Rel-6.

[TS SSC] Draft 3GPP TS 33.221: "Generic Authentication Architecture (GAA); Support for Subscriber Certificates", Rel-6.

[S3-030036] "Alternative proposals for subscriber certificate bootstrapping", Alcatel, S3#27. Link: S3-030036

[S3-030073] "Protocol B: Subscriber Certificate Enrollment based on Bootstrapping", Nokia, S3#27. Link: S3-030073

[S3-030239] "Notes for the use of CMPv2 as the subscriber certificate enrollment protocol (Protocol B)", SSH Communications Security, S3#28. Link: S3-030239

[S3-030347] "CMPv2 profile for 3GPP subscriber certificate enrollment", SSH Communications Security, S3#29. Link: S3-030347

[S3-030355] "Support for Subscriber Certificates", SchlumbergerSema, S3#29. Link: S3-030355

[S3-030555] "Using shared key TLS with GAA NAFs", Nokia, S3#30. Link: S3-030355

[CMP] Adams C., Farrell S., "Internet X.509 Public Key Infrastructure Certificate Management Protocols", RFC 2510, March 1999. Link: RFC 2510

[CRMF] Myers M., Adams C., Solo D., Kemp D., "Internet X.509 Certificate Request Message Format", RFC 2511, March 1999. Link: RFC 2511

[WPKI] WAP-217-WPKI: "Wireless Application Protocol Public Key Infrastructure Definition", April 2001. Link: WPKI

[PKCS#10] PKCS#10 v1.7: "Certification Request Syntax Standard", RSA Laboratories, May 2000.

[CMPv2] Adams C., Farrell S., "Internet X.509 Public Key Infrastructure Certificate Management Protocol", Internet-Draft, April 2003. Link: http://www.ietf.org/internet-drafts/draft-ietf-pkix-rfc2510bis-08.txt

[CRMFv2] Adams C., Farrell S., “Internet X.509 Public Key Infrastructure Certificate Request Message Format (CRMF)”, Internet-Draft, April 2003.
Link: http://www.ietf.org/internet-drafts/draft-ietf-pkix-rfc2511bis-06.txt

[SharedKey] Gutmann P., “Use of Shared Keys in the TLS Protocol”, Internet-Draft, October 2003.
Link: http://www.ietf.org/internet-drafts/draft-ietf-tls-sharedkeys-02.txt

***** BEGIN CHANGE *****

# 4 Support for Subscriber Certificates

## 4.1 Introduction

Digital signatures can be used, for instance, to secure mobile commerce, service authorization and accounting. But digital signature by itself is not enough; there is need of a global support for authorization and charging. Thus 3GPP shall use global and secure authorization and charging infrastructure of mobile networks to support local architecture for digital signatures.

Subscriber certificates provide a migration path towards global Public Key Infrastructure (PKI). Local architecture for digital signatures can be deployed incrementally; an operator can choose to deploy independently of the others. On the other hand, the existence of subscribers and service providers that use digital signatures makes it easier to build global PKI.

3GPP systems shall issue subscriber certificates in order to authorize and account for service usage both in home and in visited network. This requires specification of:

1. Procedures to issue temporary or long-term certificates to subscribers.

2. Standard format of certificates and digital signatures, e.g. re-using wireless PKI.

The mechanism shall allow a cost efficient implementation of the security support of the UE. It will also enable a user’s anonymity towards the service provider, whilst the user who invoking the service, can be identified by the network.

Subscriber certificates support services whose provision mobile operator assists, as well as services that mobile operator provides. There is no need to standardize those services. Also, the communication between service provider and the operator (in the role of certificate issuer) need not be standardized.

## 4.2 Requirements and principles for issuing subscriber certificates

The following prerequisites for issuing of subscriber certificates exits:

- The shared key material is available for the UE application, which does the certificate request and operator CA certificate retrieval.

- The issuing of requested certificate is allowed according to subscriber profile. NAF is responsible for performing this check before issuing the subscriber certificate.

### 4.2.1 Usage of Bootstrapping

Issuing procedures of the subscriber certificate and operator CA certificate shall be secured by using shared keys obtained from bootstrapping function.

### 4.2.2 Access independence

Subscriber certificate and operator CA certificate issuing procedures are access independent. Certificate issuing procedures require IP connectivity from UE.

### 4.2.3 Roaming and service network support

The roaming subscriber shall be able to request subscriber certificates and operator CA certificates from home network.

Editor’s note: Certificate requests to any than home network may be supported in later phase of the present specification.

### 4.2.4 Home operator control

Home operator shall be able to control the issuing of subscriber certificates. The control includes to whom the certificates are allowed to issue and the types of issued certificates.

Operator control is supported by information in the subscriber profile. For each type of subscriber certificate, i.e. for different keyUsage in WAP Certificate and CRL Profile, subscriber profile shall contain a flag that allows or disallows the issuing of that type of certificate to subscriber.

Editor's note: Currently two keyUsage values are envisioned: authentication and signing.

Delivery of operator CA certificates is always allowed.

Editor's note: For the first phase of standardisation, only the case is considered where bootstrapping server functionality and network application function are located in the same network as the HSS. Thus is the first phase the home network control does not require any communication between home and visited networks. In later phases, when also visited network may issue certificates, standardized way of transferring the control information from home network to visited network is needed.

### 4.2.5 Charging principles

The operator shall be capable to charge issuing of subscriber certificates or delivery of operator CA certificates.

Editor's note: The charging mechanism and whether it needs to be standardized in 3GPP is FFS.

### 4.2.6 Subscriber Certificate Profile

Subscriber certificate profile shall be based on WAP Certificate and CRL Profile [7], which in turn is based on profiles defined in [6] and [10]. A certificate profile defines the format and semantics of certificates in a specific context. WAP Certificate and CRL profiles specification defines four certificate profiles: two user certificate profiles – one for authentication and the other for non-repudiation purposes, server certificate profile for authentication, and authorization certificate profile (i.e., CA certificate). Since subscriber certificates are issued to users, and since services need CA certificate to validate subscriber certificates, the relevant WAP certificate profiles to be used with subscriber certificate profiles are the user certificate profiles, and CA certificate profile.

Editor's note: Applicability of other certificate profile specifications, e.g. RFC 3281, ETSI QC profile is FFS.

The following certificate extensions shall be filled with the information given by the UE in the certification request:

- Intended certificate usage (i.e., using keyUsage and/or extKeyUsage extensions [7]).
- Subscriber identities (i.e., subject name field, and possible additional identities defined in the subjectAltName extension [7]). Operator CA shall authorize each suggested subscriber identity.
- Proof of key origin (i.e., keyGenAssertion). Operator CA shall verify the proof of key origin if it is presented.

NOTE: It is not mandatory for Operator CA to insert these suggested extensions by UE to the certificate. Rather, Operator CA shall issue certificates based on its certification policies. It may write a certification practice statement (CPS) [4], where it describes the general requirements and steps taken during the certificate issuing.

### 4.2.7 Service Discovery

The addresses of bootstrapping server and PKI portal may be pre-configured to the UE or UICC. The possible service discovery or over-the-air configuration mechanism are FFS.

Editor's note: For the first phase of standardisation, when bootstrapping server functionality and network application function are always located in home network, therefore pre-configuration of addresses is sufficient. In later phases, however, when UE needs to address of PKI Portal in the visited network, more flexible is needed in the solution.

### 4.2.8 Requirements on Ua interface

The requirements for Ua interface are:

- UE ~~is~~shall be able to request for subscriber’s certification from the PKI portal that plays the role of the NAF over a network connection.
- NAF ~~is~~shall be able to authenticate UE’s certificate request.
- UE ~~is~~shall be able to acquire an operator’s CA certificate over the network connection.
- UE ~~is~~shall be able to authenticate the NAF response (i.e., operator CA certificate delivery).
- The procedure ~~is~~shall be independent of the access network used.
- The NAF shall have access to the subscriber profile to check the certification policies. This means that the Zn interface [11] shall support for retrieving a subset of the subscriber profile.
- The response and delivery of certificate to UE ~~must~~shall be within a few seconds after the initial certification request.
- Certification request format shall be PKCS#10.
- Certification response format shall be one of the following: a certificate, a pointer to the certificate, or a full certificate chain.

## 4.3 Certificate issuing architecture

### 4.3.1 Reference model

Figure 1 below shows a simple network model of the entities involved in the certificate issuing, and the protocols used between the network entities.

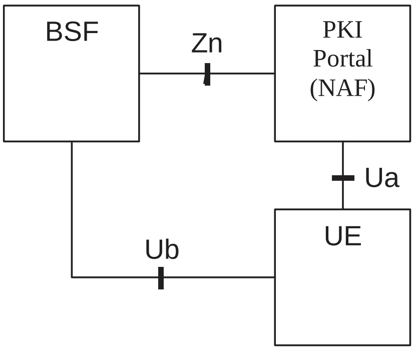


**Figure 1: Simple network model for certificate issuing.**

### 4.3.2 Network elements

#### 4.3.2.1 PKI Portal

A PKI Portal shall issue a certificate for UE and deliver an operator CA certificate. In both cases, requests and responses are protected by shared key material that has been previously established between UE and a BSF

In PKI terms, the PKI portal is a Registration Authority (RA) who authenticates the certification request based on cellular subscription (over Ub interface). PKI Portal may also function as a Certification Authority (CA) who issues certificates. However, this task may also be done in an existing PKI infrastructure towards which the PKI Portal would function as a RA only, and the CA would be in the PKI infrastructure.

#### 4.3.2.2 Bootstrapping Server Function

The bootstrapping server function (BSF) shall support the PKI portal by providing the authentication (c.f. subclause 4.2.2.1) and subscriber profile information (i.e., whether subscriber is able to enrol a certain types of subscriber certificate).

#### 4.3.2.3 UE

The required new functionality from UE is the support of the Ua interface (i.e. certification enrolment protocol) that is protected using the shared keys established during bootstrapping function.

In addition UE may have the capability to generate public and private key pairs, store the private key part to a non-volatile memory (e.g., in UICC), and protect the usage of the private key part (e.g., with a PIN).

### 4.3.3 Reference points

#### 4.3.3.1 Ua interface

##### 4.3.3.1.1 General description

In the certificate issuing, Ua interface is used to for:

- The operator CA certifying subscriber's public keys in format of certificates, and
- The delivery of the Operator CA certificate to the UE.

During subscriber certificate issuing, UE may request a certification of a public key. The supported request formats shall include PKCS#10. It is used to encapsulate the public key and other attributes (i.e., subject name, intended key usage, etc.). The request is transported from the UE to the PKI Portal over Ua interface. Upon receiving the certification request, PKI portal will certify the public key according to its own certification practice policies and subscriber profile which is fetched through BSF from HSS. If PKI Portal decides to certify the public key, it will digitally sign it, and generate the corresponding certificate, which is returned from PKI Portal to the UE, over Ua interface.

During operator CA certificate delivery, the UE may request the PKI Portal to deliver operator CA's certificate. In the corresponding response, the PKI Portal will deliver the CA's certificate to the UE. Since the operator's CA certificate is typically a self-signed certificate and the validation of certificates signed by this CA is based on this particular CA certificate, it needs to be delivered over authenticated and secured channel.

Authentication, integrity protection, and possibly encryption of the messages sent over Ua interface are based on the BSF generated shared secret.

##### 4.3.3.1.2 Functionality and protocols

~~Editor's note: From five alternatives investigated in S3-030073 and S3-030036, only the following two have been agreed to add to the present document as potential solutions.~~

###### 4.3.3.1.2.1 PKCS#10 with HTTP Digest Authentication

Editor's note: This section uses HTTP Digest authentication to authenticate and integrity protect the certificate request and response. Shared key TLS is another solution to authenticate and protect the certificate enrolment, and whether it should be used instead of HTTP Digest is ffs.

HTTP Digest Authentication scheme [5] may be done with BSF shared key material the following way.

- UE makes a blank HTTP request to the NAF
- NAF returns a HTTP response with "WWW-Authenticate" header indicating that HTTP Digest Authentication is needed. Quality of protection (qop) attribute is set to "auth-int" meaning that the content in following HTTP requests and responses are integrity protected.
- UE calculates the correct response to the "WWW-Authenticate" header using the *identifier* (base64 encoded) as the username and the session key K (base64 encoded) as the password. The session key K is has been previously

derived from the key material Ks that resulted from using Ub interface. HTTP Digest Authentication parameters are returned in the “Authorization” header of HTTP Response.

- NAF validates the “Authorization” header and upon successful validation, performs the requested task. In the corresponding HTTP response, NAF calculates the relevant values for “Authentication-Info” header, which is used to authenticate and integrity protect the NAF response.
- UE validates the “Authentication-Info” header and upon successful validation, accepts the payload in the HTTP response.

A PKCS#10 [1] based certification request is sent to the CA NAF using a HTTP POST request, which MUST be authenticated and integrity protected by HTTP Digest Authentication.

Certificate is delivered using the HTTP response, which MAY be authenticated and integrity protected by HTTP Digest Authentication. The content-type of the HTTP response depends on the response format. If a certificate is returned then it is ~~either~~ “application/x-x509-user-cert”. ~~Or~~If a pointer to the certificate is returned then it is “application/vnd.wap.cert-response” as specified in [9]. The content-type and the format of the certificate chain is ffs.

The UE requests a CA certificate delivery by sending a plain HTTP GET request with specific parameters in the request URI . The request MAY be authenticated and integrity protected by HTTP Digest Authentication.

CA certificate is delivered using the HTTP response, which MUST be authenticated and integrity protected by HTTP Digest Authentication. The content-type of the HTTP response would be “application/x-x509-ca-cert”. Note that the user should always be notified when a new CA certificate is taken into use.

#### ~~4.3.3.1.2.2 Certificate Management Protocols (CMP)~~

~~Certificate Management Protocols (CMP) [2] describes a set of messages that can be used between different PKI components, e.g., between the CA and the end entity as well as between two CAs. The messages used in the specification have the following general message structure called PKIMessage. PKIMessage contains four fields: PKIHeader, PKIBody, optional PKIProtection, and optional certificate list. The PKIHeader contains information, which is common to many PKI messages. The PKIBody contains the message-specific information. The PKIProtection, when used, contains bits that protect the PKI message. The certificate list can contain certificates that may be useful to the recipient. [2]~~

~~In CMP, authentication is achieved by the PKI issuing the end entity with a secret value (initial authentication key) and reference value (used to identify the transaction) via some out-of-band means. The initial authentication key can then be used to protect relevant PKI messages (see chapters 2.2.1.2. and 3.1.3 of [2] for details). Also a replay prevention mechanism is specified.~~

~~The supported certificate request formats are PKCS#10 [1] and CRMF [2]. The certificate request is inserted in the PKIBody field of the PKIMessage. The response to the certificate request is a CertRespMessage that is inserted in the PKIBody field of the PKIMessage. The CertRespMessage contains the status of the response, and if certificate request was approved the certificate itself. CMP supports also a certification procedure where the key generation happens in the CA rather the in the UE. However, CMP states that this procedure is only optionally implemented by CAs. See more details in [2].~~

~~CMP defines data structures, which can support mechanism where the CA is able to publish its current public key using self-signed certificates that are distributed via some “out-of-band” means. Alternatively the self-signed CA certificate can be published on a directory server and a hash of the certificate can be distributed via some out-of-band means. The idea is that anyone who has securely received a hash value can verify the authenticity of the CA certificate. The structure of such a self-signed out-of-band certificate or hash is specified in the RFC. However, the way how the CA publishes the self-signed certificate and/or securely delivers the hash value is considered out-of-scope for CMP (see chapter 3.2.5 of [2]).~~

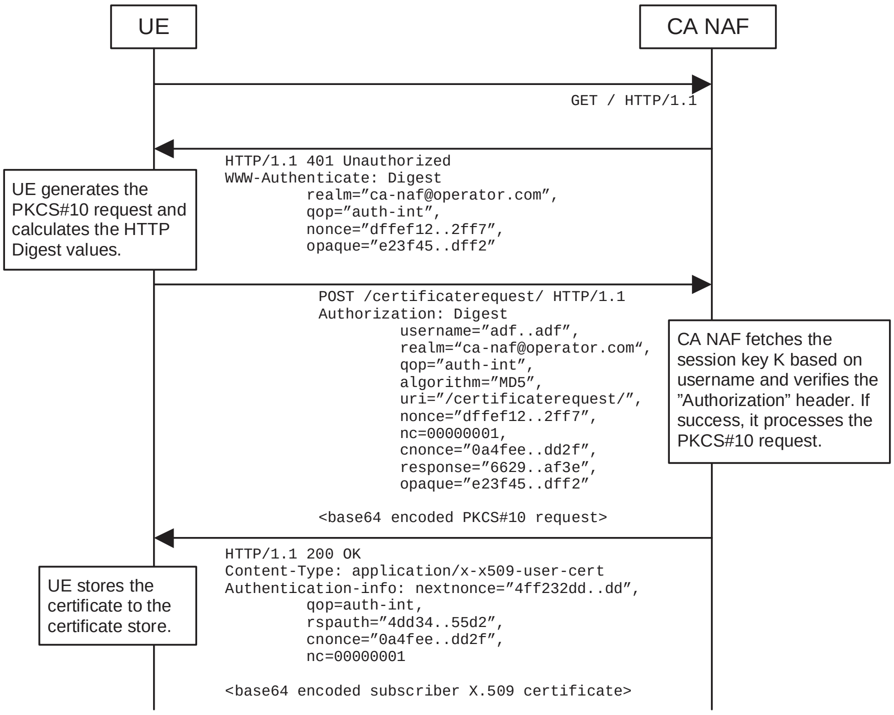
## 4.4 Certificate issuing procedure

Editor’s note: This section uses HTTP Digest authentication to authenticate and integrity protect the certificate request and response. Shared key TLS is another solution to authenticate and protect the certificate enrolment, and whether it should be used instead of HTTP Digest is ffs.

### 4.4.1 Certificate issuing

~~Editor's note: From five alternatives investigated in S3-030073 and S3-030036, only the following two have been agreed to add to the present document as potential solutions.~~

#### 4.4.1.1 Certificate issuing using PKCS#10 with HTTP Digest Authentication



```
GET / HTTP/1.1
```

```
HTTP/1.1 401 Unauthorized
WWW-Authenticate: Digest
         realm="ca-naf@operator.com",
         qop="auth-int",
         nonce="dffef12..2ff7",
         opaque="e23f45..dff2"
```

UE generates the PKCS#10 request and calculates the HTTP Digest values.

```
POST /certificaterequest/ HTTP/1.1
Authorization: Digest
         username="adf..adf",
         realm="ca-naf@operator.com",
         qop="auth-int",
         algorithm="MD5",
         uri="/certificaterequest/",
         nonce="dffef12..2ff7",
         nc=00000001,
         cnonce="0a4fee..dd2f",
         response="6629..af3e",
         opaque="e23f45..dff2"

<base64 encoded PKCS#10 request>
```

CA NAF fetches the session key K based on username and verifies the "Authorization" header. If success, it processes the PKCS#10 request.

```
HTTP/1.1 200 OK
Content-Type: application/x-x509-user-cert
Authentication-info: nextnonce="4ff232dd..dd",
         qop=auth-int,
         rspauth="4dd34..55d2",
         cnonce="0a4fee..dd2f",
         nc=00000001

<base64 encoded subscriber X.509 certificate>
```

UE stores the certificate to the certificate store.

**Figure 2: Certificate request using PKCS#10 with HTTP Digest Authentication.**

The sequence diagram above describes the certificate request when using PKCS#10 with HTTP Digest. The sequence starts with an empty HTTP request to CA NAF. The CA NAF responds with HTTP response code 401 "Unauthorized" which contains a WWW-Authenticate header. The header instructs the UE to use HTTP Digest authentication.

The UE generates a PKCS#10 request with the subject name, public key, additional attributes and extensions. Then it will generate the HTTP request by calculating the Authorization header values using the identifier it received from the BSF as username and the session key K. UE may indicate the desired format of the certification response: a certificate, a pointer to the certificate (e.g., URL), or a full certificate chain (i.e., from the issued certificate to the corresponding root certificate). The enrolment request shall be as follows:

POST <base URL>?response=<indication>[other URL parameters] HTTP/1.1
Content-Type: application/x-pkcs10

<base64 encoded PKCS#10 blob>

where

<base URL> identifies a server/program.

<indication> used to indicate to the CA NAF what is desired response type for the UE. The possible values are: "single" for subscriber certificate only, "pointer" for pointer to the subscriber certificate, or "chain" for full certificate chain.

[other URL parameters] are additional, optional, URL parameters

When CA NAF receives the request, it will verify the Authorization header by fetching the session key K from the bootstrapping server using the identifier, then calculating the corresponding digest values using K, and finally comparing the calculated values with the received values in the Authorization header. If the verification succeeds, the incoming PKCS#10 request is taken in for further processing. If the CA NAF is actually a registration authority (RA NAF), the PKCS#10 request is forwarded to CA using any protocol available (e.g., CMC or CMP). After the PKCS#10 request has been processed and a certificate has been created, the new certificate is returned to the CA NAF. It will generate a HTTP response containing the certificate, or the pointer to the certificate as defined subclause 7.4 of [9], or a full certificate chain from issued certificate to the root certificate.

If the HTTP response contains the subscriber certificate itself, it shall be base64 encoded, and it may be demarcated as follows:

```
HTTP/1.1 200 OK
Content-Type: application/x-x509-user-cert

-----BEGIN CERTIFICATE-----
<base64 encoded X.509 certificate blob>
-----END CERTIFICATE-----
```

If the HTTP response contains the pointer to the certificate itself, the CertResponse structure defined in subclause 7.3.5 of [9] shall be used, and it may be demarcated as follows:

```
HTTP/1.1 200 OK
Content-Type: application/vnd.wap.cert-response

-----BEGIN CERTIFICATE RESPONSE-----
<base64 encoded CertResponse structure blob>
-----END CERTIFICATE RESPONSE-----
```

If the HTTP response contains a full certificate chain, each certificate shall be base64 encoded and shall be demarcated as follows:

```
HTTP/1.1 200 OK
Content-Type:  ffs

-----BEGIN CERTIFICATE-----
<base64 encoded X.509 certificate blob>
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
<base64 encoded X.509 certificate blob>
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
<base64 encoded X.509 certificate blob>
-----END CERTIFICATE-----
```

The certificates in the response are not needed to be in any particular order.  The content-type header value for the certificate chain is ffs.~~.~~

The CA NAF may use session key K to integrity protect and authenticate the response, if a certificate or a pointer to the certificate is sent to the UE. The CA NAF shall use integrity protection and authenticate the response if full certificate chain is sent to the UE.

When UE receives the subscriber certificate, it is stored to local certificate management system.

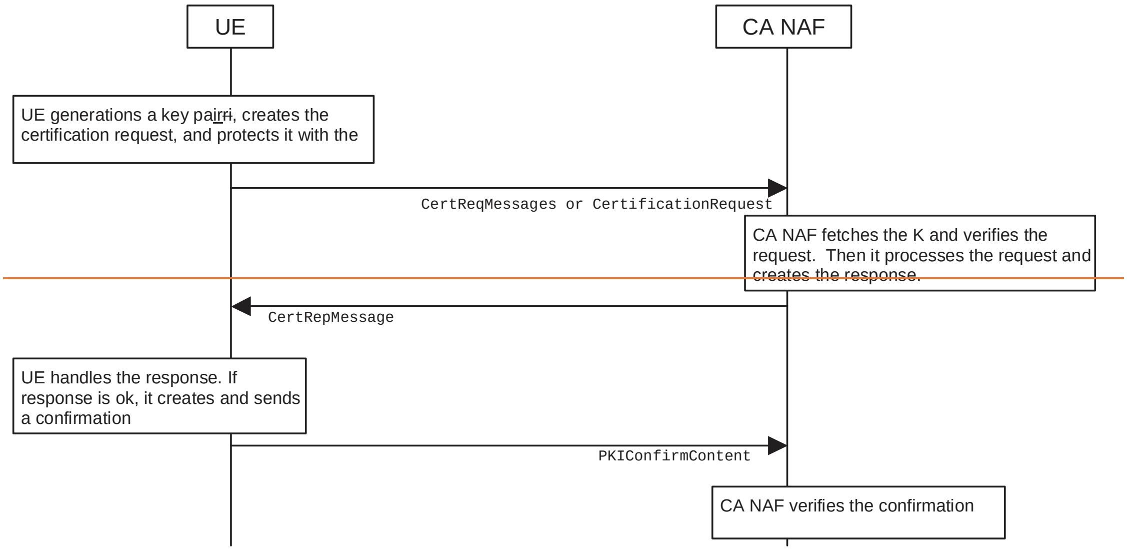
NOTE: On board key generation is already defined in the WIM specification [8] issued by Open Mobile Alliance (OMA) group."

### ~~4.4.1.2 Certificate issuing with CMP~~

~~CMP defines two methods to do the certificate issuing: basic authenticated scheme and centralized scheme. In the basic authenticated scheme the key generation happens in the UE while in the centralized scheme the key generation is done in the CA (or RA). CMP states that the support for the basic authenticated scheme for certificate issuing is mandatory for CAs while the support for the centralized scheme is optional. See more details in chapters 2.2 and B8 of [2].~~

~~The messages can be transported using various methods such as file based protocol, (such files can be used to transport PKI messages e.g. using FTP, HTTP, email etc.), direct TCP-based management protocol, management protocol via e-mail, and management protocol via HTTP mentioned in section 5 of [2].~~

### ~~4.4.1.2.1 Basic authenticated scheme~~


UE
CA NAF
UE generations a key pair~~ri~~, creates the certification request, and protects it with the
CertReqMessages or CertificationRequest
CA NAF fetches the K and verifies the request. Then it processes the request and creates the response.
CertRepMessage
UE handles the response. If response is ok, it creates and sends a confirmation
PKIConfirmContent
CA NAF verifies the confirmation

~~**Figure 3: Certificate request using basic authentication scheme of CMP.**~~

~~The sequence diagram above describes the certificate request and delivery procedure when using CMP and basic authenticated scheme [2]. The sequence starts with UE generating a key pair, creating the certificate request message format (CRMF) message, inserting it to CertReqMessages message, and integrity protecting this message with the initial authentication key (IAK). The session key K, which has been derived earlier using Ub interface, can be used as IAK.~~

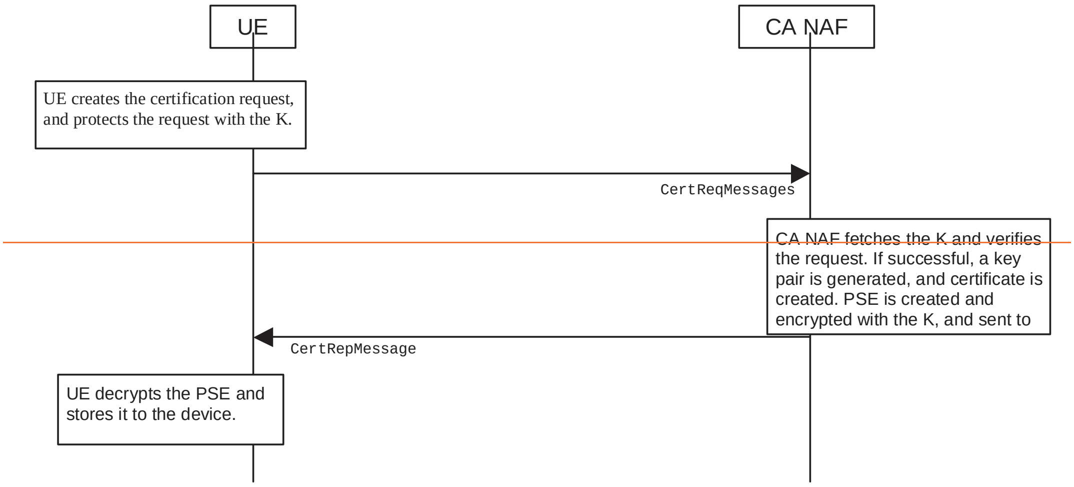
~~The certificate request message is sent to CA NAF who fetches the corresponding K based on the identifier received in the request. CA NAF verifies the request with the K. If the verification succeeds, the CA NAF processes the request, i.e. generates and signs the certificate and sends the certification response to the UE.~~

~~UE verifies the certificate response message with the K. If the message verification is successful, the issued certificate is stored to the device, and UE sends a confirmation message to the CA NAF.~~

~~CA NAF verifies the confirmation message. If the verification fails or CA NAF never receives the confirmation message, CA NAF must revoke the newly issued certificate if it has been already published.~~

### ~~4.4.1.2.2 Centralized scheme initiated by the UE~~

~~The centralized scheme provides a mechanism where the public/private key pair is generated outside the UE, e.g. by the CA.~~


UE
CA NAF
UE creates the certification request, and protects the request with the K.
CertReqMessages
CA NAF fetches the K and verifies the request. If successful, a key pair is generated, and certificate is created. PSE is created and encrypted with the K, and sent to
CertRepMessage
UE decrypts the PSE and stores it to the device.

~~**Figure 4: Certificate request using centralized scheme of CMP.**~~

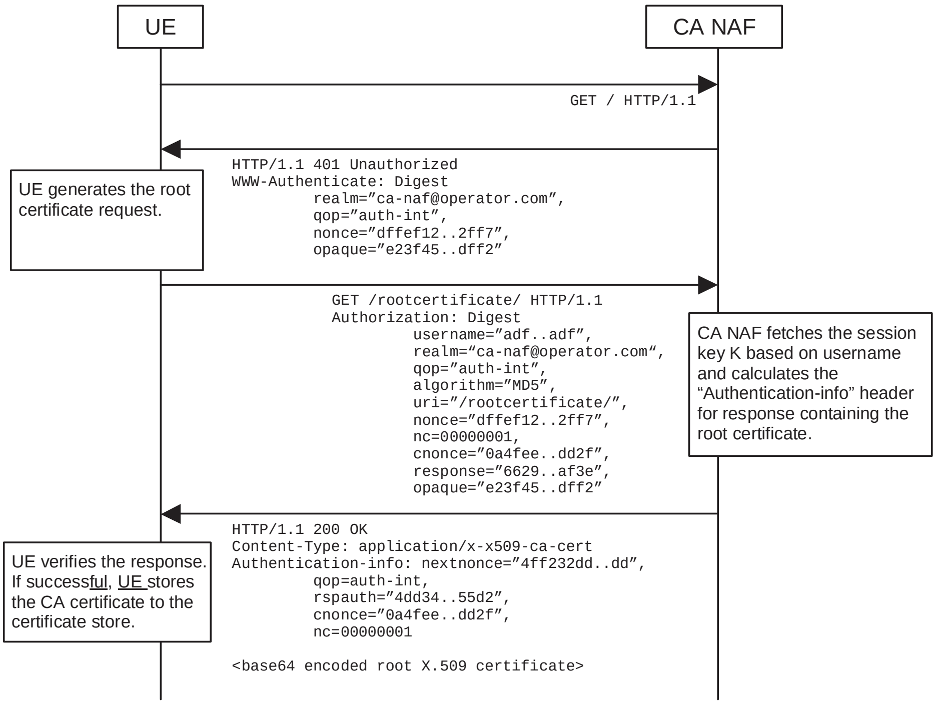
~~The sequence diagram above describes the delivery mechanism initiated by the UE using CMP in centralized scheme. This scheme is optional in CMP [2]. The sequence starts with the UE by creating CertReqMessages message with certain parameters, and protecting this message with initial authentication key (IAK). The session key K, which has been derived earlier using Ub interface, can be used as IAK.~~

~~The certificate request message is sent to CA NAF who fetches the corresponding K based on the identifier received in the request. CA NAF verifies the request with the K. If the verification succeeds, CA NAF processes the request, i.e. generates a key pair, generates and signs the certificate, and sends the certification response containing the Personal Security Environment (PSE) encrypted to the UE. PSE typically contains the generated private key and newly issued certificate with corresponding public key.~~

~~UE verifies the certificate response message with the K. If the message verification is successful, the issued PSE is decrypted and stored to the device. A confirmation message is not sent in the centralized scheme.~~

## 4.4.2 CA Certificate delivery

### 4.4.2.1 CA Certificate delivery with HTTP Digest Authentication



```
GET / HTTP/1.1
```

```
HTTP/1.1 401 Unauthorized
WWW-Authenticate: Digest
          realm="ca-naf@operator.com",
          qop="auth-int",
          nonce="dffef12..2ff7",
          opaque="e23f45..dff2"
```

UE generates the root certificate request.

```
GET /rootcertificate/ HTTP/1.1
Authorization: Digest
          username="adf..adf",
          realm="ca-naf@operator.com",
          qop="auth-int",
          algorithm="MD5",
          uri="/rootcertificate/",
          nonce="dffef12..2ff7",
          nc=00000001,
          cnonce="0a4fee..dd2f",
          response="6629..af3e",
          opaque="e23f45..dff2"
```

CA NAF fetches the session key K based on username and calculates the "Authentication-info" header for response containing the root certificate.

```
HTTP/1.1 200 OK
Content-Type: application/x-x509-ca-cert
Authentication-info: nextnonce="4ff232dd..dd",
          qop=auth-int,
          rspauth="4dd34..55d2",
          cnonce="0a4fee..dd2f",
          nc=00000001

<base64 encoded root X.509 certificate>
```

UE verifies the response. If successful, UE stores the CA certificate to the certificate store.

**Figure 5: CA certificate delivery with HTTP Digest authentication.**

The sequence diagram above describes the CA certificate delivery when using HTTP Digest authentication. The sequence starts with an empty HTTP request to CA NAF. The CA NAF responds with HTTP response code 401 "Unauthorized" which contains a WWW-Authenticate header. The header instructs the UE to use HTTP Digest for authentication.

The UE generates another ~~empty~~ HTTP request for requesting the CA certificate. UE shall indicate the CA issuer name in the request URL as specified in subclause 7.4.1 of [9]. The serial number field shall be omitted. The Authorization header values are calculated using the identifier and the session key K. The authentication of this HTTP request is not necessary, but it is done in order to follow HTTP Digest authentication specification. Also, the identifier needs to be transported to the CA, i.e. the NAF. A request of subscriber's certificate is specified in subclause 4.4.1.1. The CA certificate delivery request shall be as follows:

GET <base URL>?in=<issuer name>[other URL parameters] HTTP/1.1

where

<base URL> identifies a server/program.

<issuer name> identifies the certificate issuer. It is a base64 encoding of the DER encoded Issuer field in the X.509 certificate.

[other URL parameters] are additional, optional, URL parameters

When CA NAF receives the request, it may verify the Authorization header by fetching the session key K from the bootstrapping server using the identifier. CA NAF will generate a HTTP response containing the CA certificate and use the session key K to authenticate and integrity protect the HTTP response using the Authentication-info header. Essentially, the response could also be other delivery protocol in HTTP format, e.g. PKCS#7 cryptographic message with content type signedData.

HTTP response contains the CA certificate.  The CA certificate shall be base64 encoded, and it may be demarcated as follows:

```
HTTP/1.1 200 OK
Content-Type: application/x-x509-ca-cert

-----BEGIN CERTIFICATE-----
<base64 encoded X.509 certificate blob>
-----END CERTIFICATE-----
```

When UE receives the new CA certificate, it must validate the Authentication-info header. If validation succeeds, the user is notified that a new CA certificate is taken into use. If user accepts the new CA certificate, it is stored to the local certificate management system and marked as “trusted” CA certificate.

#### ~~4.4.2.2 CA Certificate delivery with CMP~~

~~CMP defines only out-of-band method for delivering CA certificates. CA certificate may be delivered as part of the certificate request, where the response could contain certificates that may be useful to the recipient. It can contain the whole certificate chain (including the CA certificate). The root CA produces a “self-certificate” and also produces a fingerprint of its public key. End entities that acquire this fingerprint securely via some out-of-band means can then verify the CA’s self-certificate and hence the other attributes contained therein.~~

********** END CHANGE *****