Analysis of UDM Originated and Terminated messages during 5G Registration process.

### *Referring 23.502 (5G Procedures)*

The following table illustrates the **UDM Services and Service Operations**.

Table 5.2.3.1-1: NF services provided by UDM

|  |  |  |  |
| --- | --- | --- | --- |
| NF service | Service Operations | Operation Semantics | Example Consumer(s) |
| Subscriber Data | Get | Request/Response | AMF, SMF, SMSF, NEF |
| Management | Subscribe | Subscribe/Notify | AMF, SMF, SMSF, NEF |
| (SDM) | Unsubscribe | Subscribe/Notify | AMF, SMF, SMSF, NEF |
|  | Notification | Subscribe/Notify | AMF, SMF, SMSF, NEF |
|  | Info | Request/Response | AMF, NEF |
|  | Update | Request/Response | NEF |
| UE Context | Registration | Request/Response | AMF, SMF, SMSF |
| Management | DeregistrationNotification | Subscribe/Notify | AMF |
| (UECM) | Deregistration | Request/Response | AMF, SMF, SMSF |
|  | Get | Request/Response | NEF, SMSF, GMLC, NWDAF |
|  | Update | Request/Response | AMF, SMF |
|  | PCscfRestoration | Subscribe/Notify | AMF, SMF |
| **UE** | **Get** | **Request/Response** | **AUSF** |
| **Authentication** | ResultConfirmation | Request/Response | AUSF |
| EventExposure | Subscribe | Subscribe/Notify | NEF (NOTE), NWDAF |
|  | Unsubscribe |  | NEF (NOTE), NWDAF |
|  | Notify |  | NEF (NOTE), NWDAF |
| Parameter | Update | Request/Response | NEF, AMF |
| Provision | Create | Request/Response | NEF |
|  | Delete | Request/Response | NEF |
|  | Get | Request/Response | NEF |
| NIDDAuthorisation | Get | Request/Response | NEF |
|  | UpdateNotify | Subscribe/Notify | NEF |
| NOTE: Other NFs are allowed to consume the service based on roaming agreement or operator policy. | | | |

*Referring 33.501 (Security architecture and procedures for 5G system)*

### 14.2.2 Nudm\_UEAuthentication\_Get service operation

**Service operation name:** Nudm\_UEAuthentication\_Get

**Description:** Requester NF(AUSF) gets the authentication data from UDM. For AKA (Authentication and key Agreement) based authentication, this operation can be also used to recover from synchronization failure situations. If SUCI is included, this service operation returns the SUPI.

**Inputs, Required:** *SUPI or SUCI, serving network name.*

**Inputs, Optional:** *Synchronization Failure indication and related information (i.e. RAND/AUTS).*

**Outputs, Required:** *Authentication method and corresponding authentication data for a certain UE as identified by SUPI or SUCI input*.

**Outputs, Optional:** *SUPI if SUCI was used as input.*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

*Referring 33.501 (Security architecture and procedures for 5G system)*

#### 6.1.3.2 Authentication procedure for 5G AKA

##### 6.1.3.2.0 5G AKA

5G AKA enhances EPS AKA [10] by providing the home network with proof of successful authentication of the UE from the visited network. The proof is sent by the visited network in an Authentication Confirmation message. The selection of using 5G AKA is described in sub-clause 6.1.2 of the present document.

**NOTE 1: 5G AKA does not support requesting multiple 5G AVs, neither the SEAF pre-fetching 5G AVs from the home network for future use.**

## 

Figure 6.1.3.2-1: Authentication procedure for 5G AKA

The **authentication procedure for 5G AKA works as follows**, cf. also Figure 6.1.3.2-1:

1. For each **Nudm\_UEAuthenticate\_Get Request**, the **UDM/ARPF shall create a 5G HE AV**. The UDM/ARPF does this by generating an AV with the Authentication Management Field (AMF) separation bit set to "1" as defined in TS 33.102 [9]. The UDM/ARPF shall then derive KAUSF (as per Annex A.2) and calculate XRES\* (as per Annex A.4). Finally, the UDM/ARPF shall create a **5G HE AV** from **(RAND, AUTN, XRES\*, and KAUSF).**

*From 33.102 (3G Security; Security architecture (Release 15)*

**Generation of authentication vector**



2. The UDM shall then return the **5G HE AV** to the AUSF together with an indication that the 5G HE AV is to be used for **5G-AKA** in a Nudm\_UEAuthentication\_Get Response. In case SUCI was included in the Nudm\_UEAuthentication\_Get Request, UDM will include the SUPI in the Nudm\_UEAuthentication\_Get Response.

3. The AUSF shall store the XRES\* temporarily together with the received SUCI or SUPI.

4. The AUSF shall then generate **the 5G AV from the 5G HE AV** received from the UDM/ARPF by **computing the HXRES\* from XRES\*** (according to Annex A.5) and **KSEAF from KAUSF**(according to Annex A.6), and replacing the XRES\* with the HXRES\* and KAUSF with KSEAF in the 5G HE AV.

5. The AUSF shall then remove the KSEAF return the **5G SE AV (RAND, AUTN, HXRES\*)** to the SEAF in a Nausf\_UEAuthentication\_Authenticate Response.

6. The SEAF shall send **RAND, AUTN to the UE in a NAS message Authentication -Request**. This message shall also include the **ngKSI** that will be used by the UE and AMF to **identify the KAMF** and the partial native security context that is created if the authentication is successful. This message shall also include the ABBA parameter. The SEAF shall set the ABBA parameter as defined in Annex A.7.1. The ME shall forward the RAND and AUTN received in NAS message Authentication Request to the USIM.

NOTE 2: The ABBA parameter is included to enable the bidding down protection of security features that may be introduced later.

7. At receipt of the RAND and AUTN, the USIM shall verify the freshness of the 5G AV by checking whether AUTN can be accepted as described in TS 33.102[9]. If so, the USIM computes a response RES. The USIM shall return RES, CK, IK to the ME. If the USIM computes a Kc (i.e. GPRS Kc) from CK and IK using conversion function c3 as described in TS 33.102 [9], and sends it to the ME, then the ME shall ignore such GPRS Kc and not store the GPRS Kc on USIM or in ME. The ME then shall compute RES\* from RES according to Annex A.4. The ME shall calculate KAUSF from CK||IK according to clause A.2. The ME shall calculate KSEAF from KAUSF according to clause A.6. An ME accessing 5G shall check during authentication that the "separation bit" in the AMF field of AUTN is set to 1. The "separation bit" is bit 0 of the AMF field of AUTN.

NOTE 3: This separation bit in the AMF field of AUTN cannot be used anymore for operator specific purposes as described by TS 33.102 [9], Annex F.

8. The UE shall return **RES\* to the SEAF** in a NAS message **Authentication Response**.

9. The SEAF (SEcurity Anchor Function) shall then compute HRES\* from RES\* according to Annex A.5, and the SEAF shall compare HRES\* and HXRES\*. If they coincide, the **SEAF shall consider the authentication successful from the serving network point of view**. If not, the SEAF proceed as described in sub-clause 6.1.3.2.2. If the UE is not reached, and the RES\* is never received by the SEAF, the SEAF shall consider authentication as failed, and indicate a failure to the AUSF.

10. The SEAF shall send RES\*, as received from the UE, in a **Nausf\_UEAuthentication\_Authenticate Request message to the AUSF.**

11. When the AUSF receives as authentication confirmation the **Nausf\_UEAuthentication\_Authenticate Request message including a RES\*** it may verify whether the AV has expired. If the AV has expired, the AUSF may consider the authentication as unsuccessful from the home network point of view. Upon successful authentication, the AUSF shall store the KAUSF. AUSF shall compare the received RES\* with the stored XRES\*. If the RES\* and XRES\* are equal, the AUSF shall consider the authentication as successful from the home network point of view. AUSF shall inform UDM about the authentication result (see sub-clause 6.1.4 of the present document for linking with the authentication confirmation).

12. The AUSF shall indicate to the SEAF in the Nausf\_UEAuthentication\_Authenticate Response whether the authentication was successful or not from the home network point of view. If the authentication was successful, the KSEAF shall be sent to the SEAF in the Nausf\_UEAuthentication\_Authenticate Response. In case the AUSF received a SUCI from the SEAF in the authentication request (see sub-clause 6.1.2 of the present document), and if the authentication was successful, then the AUSF shall also include the SUPI in the Nausf\_UEAuthentication\_Authenticate Response message.

If the authentication was successful, the key KSEAF received in the Nausf\_UEAuthentication\_Authenticate Response messageshall become the anchor key in the sense of the key hierarchy as specified in sub-clause 6.2 of the present document. Then the SEAF shall derive the KAMF from the KSEAF, the ABBA parameter and the SUPI according to Annex A.7. The SEAF shall provide the ngKSI and the KAMF to the AMF.

If a SUCI was used for this authentication, then the SEAF shall only provide ngKSI and KAMF to the AMF after it has received the Nausf\_UEAuthentication\_Authenticate Response message containing SUPI; no communication services will be provided to the UE until the SUPI is known to the serving network.

The further steps taken by the AUSF after the authentication procedure are described in sub-clause 6.1.4 of the present document.

*Referring 33.102 (3G Security architecture)*

### 6.3.7 Length of authentication parameters

1. The authentication key (K) shall have a length of 128 bits or 256 bits.

NOTE: Examples of algorithm set for 3GPP authentication and key agreement functions allow either an authentication key K with only a length of 128 bits, or an authentication key K with a length of 128 bits or 256 bits. Depending on the chosen algorithm set, the operator may have the choice of the length of the authentication key K (128 bits or 256 bits).

1. The random challenge (RAND) shall have a length of 128 bits.
2. Sequence numbers (SQN) shall have a length of 48 bits.
3. The anonymity key (AK) shall have a length of 48 bits.
4. The authentication management field (AMF) shall have a length of 16 bits.
5. The message authentication codes MAC in AUTN and MAC‑S in AUTS shall have a length of 64 bits.
6. The cipher key (CK) shall have a length of 128 bits.
7. The integrity key (IK) shall have a length of 128 bits.
8. The authentication response (RES) shall have a variable length of 4‑16 octets.

*Referring 33.102(3G Security architecture)*

Annex H (normative):   
Usage of the AMF

The 16 bits in the AMF (Authentication management field) are numbered from "0" to "15" where bit "0" is the most significant bit and bit "15" is the least significant bit (see subclause 3.4)

Bit "0" is called the "AMF separation bit". It is used for the purposes of EPS (Evolved Packet System) and is specified in

* TS 33.401 [28] for E-UTRAN access to EPS;
* TS 33.402 [29] for non-3GPP access to EPS;
* TS 33.501 [42] for 5G-RAN access to 5G System.

Bits "1" to "7" are reserved for future standardization use. Bits "1" to "7" shall be set to 0 while not yet specified for a particular use.

Bits "8" to "15" can be used for proprietary purposes. See Annex F for examples usages.

*Referring 33.220(Generic Authentication Architecture (GAA))*

**B.2 Generic key derivation function**

B.2.0 General

The input parameters and their lengths shall be concatenated into a string S as follows:

1. The length of each input parameter measured in octets shall be encoded into a two octet-long string:

a) express the number of octets in input parameter Pi as a number k in the range [0, 65535].

b) Li is then a 16-bit long encoding of the number k, encoded as described in clause B.2.1.

2. String S shall be constructed from n+1 input parameters as follows:

**S = FC || P0 || L0 || P1 || L1 || P2 || L2 || P3 || L3 ||... || Pn || Ln**

where

**FC is used to distinguish between different instances of the algorithm** and is either a single octet or consists of two octets of the form FC1|| FC2 where FC1 = 0xFF and FC2 is a single octet,

P0 ... Pn are the n+1 input parameter encodings, and

L0 ... Ln are the two-octet representations of the length of the corresponding input parameter encodings P0.. Pn.

In this specification the following restriction applies to P0: P0 is a static ASCII-encoded string.

This restriction is not part of the KDF definition and does not apply to the KDF when used by other 3GPP specifications unless explicitly stated so in those specifications.

3. The final output, i.e. the derived key is equal to the KDF computed on the string S using the key, denoted Key. The present document defines the following KDF:

**derived key = HMAC-SHA-256 (Key, S)**

as specified in [22] and [23].

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_*Referring 33.501 (Security architecture and procedures for 5G system)*

# A.1 KDF interface and input parameter construction

## A.1.1 General

All key derivations (including input parameter encoding) for 5GC shall be performed using the key derivation function (KDF) specified in Annex B.2.0 of TS 33.220 [28].

This clause specifies how to construct the input string, S, and the input key, KEY, for each distinct use of the KDF. Note that "KEY" is denoted "Key" in TS 33.220 [28].

A.1.2 FC value allocations

The FC number space used is controlled by TS 33.220 [28], FC values allocated for the present document are in range of 0x69 – 0x76.

# **A.2 KAUSF derivation function**

This clause applies to 5G AKA only. When deriving a KAUSF from CK, IK and the serving network name when producing authentication vectors, and when the UE computes KAUSF during 5G AKA,

The following parameters shall be used to form the input S to the KDF:

* **FC = 0x6A;**
* **P0 = serving network name;**
* **L0 = length of the serving network name (variable length as specified in 24.501 [35]);**
* **P1 = SQN ⊕ AK,**
* **L1 = length of SQN ⊕ AK (i.e. 0x00 0x06).**

The XOR of the Sequence Number (SQN) and the Anonymity Key (AK) is sent to the UE as a part of the Authentication Token (AUTN), see TS 33.102. If AK is not used, AK shall be treated in accordance with TS 33.102, i.e. as 000…0.

The serving network name shall be constructed as specified in clause 6.1.1.4.

**The input key KEYs shall be equal to the concatenation CK || IK of CK and IK*.***

# **A.4 RES\* and XRES\* derivation function**

When deriving RES\* from RES, RAND, and serving network name in the UE and when deriving XRES\* from XRES, RAND, and the serving network name in the UDM/ ARPF, the following parameters shall be used to form the input S to the KDF.

**- FC = 0x6B,**

**- P0 = serving network name,**

**- L0 = length of the serving network name (variable length as specified in 24.501 [35]),**

**- P1 = RAND,**

**- L1 = length of RAND (i.e. 0x00 0x10),**

**- P2 = RES or XRES,**

**- L2 = length RES or XRES (i.e. variable length between 0x00 0x04 and 0x00 0x10).**

**The input key KEY shall be equal to the concatenation CK || IK of CK and IK.**

The serving network name shall be constructed as specified in clause 6.1.1.4.

The (X)RES\* is identified with the 128 least significant bits of the output of the KDF

# A.5 HRES\* and HXRES\* derivation function

When deriving HRES\* from RES\* in the SEAF and when deriving HXRES\* from XRES\* in the AUSF the following parameters shall be used to form the input S to the SHA-256 hashing algorithm:

- P0 = RAND,

- P1 = RES\* or XRES\*,

The input S shall be equal to the concatenation P0||P1 of the P0 and P1.

The H(X)RES\* is identified with the 128 least significant bits of the output of the SHA-256 function.

# A.6 KSEAF derivation function

When deriving a KSEAF from KAUSF, the following parameters shall be used to form the input S to the KDF:

- FC = 0x6C,

- P0 = <serving network name>,

- L0 = length of <serving network name>.

The input key KEY shall be KAUSF.

The serving network name shall be constructed as specified in clause 6.1.1.4.

# A.7 KAMF derivation function

# A.7.0 Parameters for the input S to the KDF

When deriving a KAMF from KSEAF the following parameters shall be used to form the input S to the KDF.

- FC = 0x6D

- P0 = SUPI

- L0 = P0 length - number of octets in P0

- P1 = ABBA parameter

- L1 = P1 length - number of octets in P1

The input key KEY shall be the 256-bit KSEAF.

For P0, when the SUPI type is IMSI, SUPI shall be set to IMSI as defined in clause 2.2 of TS 23.003 [19]. For P0, when the SUPI type is network specific identifier, the SUPI shall be set to Network Access Identifier (NAI) as defined in clause 28.7.2 of TS 23.003 [19]. SUPI shall be represented as a character string as specified in B.2.1.2 of TS 33.220 [28], for both IMSI based SUPI as well as in NAI based SUPI.

For ABBA parameter values please refer to clause A.7.1.

From 33.501 Section 6.1.1.4

#### 6.1.1.4 Construction of the serving network name

##### 6.1.1.4.1 Serving network name

The serving network name is used in the derivation of the anchor key. It serves a dual purpose, namely:

- It binds the anchor key to the serving network by including the serving network identifier (SN Id).

- It makes sure that the anchor key is specific for authentication between a 5G core network and a UE by including a service code set to "5G".

In 5G AKA, the serving network name has a similar purpose of binding the RES\* and XRES\* to the serving network.

The serving network name is the concatenation of a service code and the SN Id with a separation character ":" such that the service code prepends the SN Id.

NOTE: No parameter like 'access network type' is used for serving network name as it relates to a 5G core procedure that is access network agnostic.

The SN Id identifies the serving PLMN and, except for standalone non-public networks, is defined as SNN-network-identifier in TS 24.501[35].

NOTE 1: For standalone non-public networks, the definition of SN Id is given in Annex I.3.

##### 6.1.1.4.2 Construction of the serving network name by the UE

The UE shall construct the serving network name as follows:

- It shall set the service code to "5G".

- It shall set the network identifier to the SN Id of the network that it is authenticating to.

- Concatenate the service code and the SN Id with the separation character ":".

##### 6.1.1.4.3 Construction of the serving network name by the SEAF

The SEAF shall construct the serving network name as follows:

- It shall set the service code to "5G".

- It shall set the network identifier to the SN Id of the serving network to which the authentication data is sent by the AUSF.

- It shall concatenate the service code and the SN Id with the separation character ":".

NOTE: AUSF gets the serving network name from the SEAF. Before using the serving network name, AUSF checks that the SEAF is authorized to use it, as specified in clause 6.1.2.

Referring 24.501

### 9.12.1 Serving network name (SNN)

The serving network name (SNN) is used:

- in the Network name field of the AT\_KDF\_INPUT attribute defined in IETF RFC 5448 [40];

- in KAUSF derivation function as specified in 3GPP TS 33.501 [24] annex A; and

- in RES\* and XRES\* derivation function as specified in 3GPP TS 33.501 [24] annex A.

SNN shall contain a UTF-8 string without terminating null characters.

SNN is of maximum length of 1020 octets.

SNN consists of SNN-service-code and SNN-network-identifier, delimited by a colon.

SNN-network-identifier identifies the serving PLMN or the serving SNPN.

MCC and MNC in the SNN-PLMN-ID are MCC and MNC of the serving PLMN. If the MNC of the serving PLMN has two digits, then a zero is added at the beginning.

MCC and MNC in the SNN-SNPN-ID are MCC and MNC of the serving SNPN. If the MNC of the serving SNPN has two digits, then a zero is added at the beginning.

SNN-NID contains an NID in hexadecimal digits.

ABNF syntax of SNN is specified in table 9.12.1.1

Table 9.12.1.1: ABNF syntax of SNN

SNN = SNN-service-code ":" SNN-network-identifier

SNN-service-code = %x35.47 ; "5G"

SNN-network-identifier = SNN-PLMN-ID / SNN-SNPN-ID

SNN-PLMN-ID = SNN-mnc-string SNN-mnc-digits "." SNN-mcc-string SNN-mcc-digits "." SNN-3gppnetwork-string "." SNN-org-string ; applicable when not operating in SNPN access mode.

SNN-SNPN-ID = SNN-mnc-string SNN-mnc-digits "." SNN-mcc-string SNN-mcc-digits "." SNN-3gppnetwork-string "." SNN-org-string ":" SNN-NID ; applicable when operating in SNPN access mode.

SNN-mnc-digits = DIGIT DIGIT DIGIT ; MNC of the PLMN ID

SNN-mcc-digits = DIGIT DIGIT DIGIT ; MCC of the PLMN ID

SNN-mnc-string = %x6d.6e.63 ; "mnc" in lower case

SNN-mcc-string = %x6d.63.63 ; "mcc" in lower case

SNN-3gppnetwork-string = %x33.67.70.70.6e.65.74.77.6f.72.6b ; "3gppnetwork" in lower case

SNN-org-string = %x6f.72.67 ; "org" in lower case

SNN-NID = x SNN-hexadecimal-digit ; NID in hexadecimal digits

SNN-hexadecimal-digit = DIGIT / %x41 / %x42 / %x43 / %x44 / %x45 / %x46

NOTE: SNN-service-code allows for distinguishing of ANID specified in 3GPP TS 24.302 [16] and SNN as either of SNN or ANID can be carried in the AT\_KDF\_INPUT attribute.

EXAMPLE 1: In case of a PLMN, if PLMN ID contains MCC = 234 and MNC = 15, SNN is 5G:mnc015.mcc234.3gppnetwork.org.

EXAMPLE 2: In case of an SNPN, if SNPN ID contains a PLMN ID of MCC = 234 and MNC = 15 and an NID of 123ABH, SNN is 5G:mnc015.mcc234.3gppnetwork.org:123AB.

Editor's note (WID: Vertical\_LAN, CR#1511): size of NID is FFS

Referring 3GPP TS 23.502(5G\_Procedures)

### 5.2.10 AUSF Services

#### 5.2.10.1 General

The following table illustrates the AUSF Services.

Table 5.2.10.1-1: List of AUSF Services

|  |  |  |  |
| --- | --- | --- | --- |
| Service Name | Service Operations | Operation  Semantics | Example Consumer(s) |
| Nausf\_UEAuthentication | Authenticate | Request/Response | AMF |
| Nausf\_SoRProtection | Protect | Request/Response | UDM |
| Nausf\_UPUProtection | Protect | Request/Response | UDM |
| Nausf\_NSSAA | Authenticate | Request/Response | AMF |
|  | Notify | Subscribe/Notify | AMF |