ETSI TS 133 535 V18.5.0 (2024-10)

5G;
Authentication and Key Management for Applications (AKMA) based on 3GPP credentials in the 5G System (5GS)
(3GPP TS 33.535 version 18.5.0 Release 18)

Reference
RTS/TSGS-0333535vi50

Keywords
5G

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (https://ipr.etsi.org/).

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™** and **LTE™** are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM**® and the GSM logo are trademarks registered and owned by the GSM Association.

Legal Notice

This Technical Specification (TS) has been produced by ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities. These shall be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between 3GPP and ETSI identities can be found under https://webapp.etsi.org/key/queryform.asp.

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

Contents

Foreword

This Technical Specification 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.

1 Scope

The present document specifies the security features and mechanisms to support authentication and key management aspects for applications based on subscription credential(s) in 5G system as defined in TS 33.501 [2].

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

[3] 3GPP TS 23.501: "System Architecture for the 5G System".

[4] 3GPP TS 33.220: "Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (GBA)".

[5] 3GPP TS 23.222: "Common API Framework for 3GPP Northbound APIs".

[6] IETF RFC 7542: "The Network Access Identifier".

[7] 3GPP TS 33.222: " Generic Authentication Architecture (GAA); Access to network application functions using HypertextTransfer Protocol over Transport Layer Security (HTTPS)".

[8] Void

[9] 3GPP TS 23.003: "Numbering, addressing and identification".

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

[11] 3GPP TS 29.503: "5G System; Unified Data Management Services ".

[12] IETF RFC 9147: "The Datagram Transport Layer Security (DTLS) Protocol Version 1.3"

[13] 3GPP TS 33.210: "3G Security; Network Domain Security; IP network layer security".

[14] IETF RFC 8613: "Object Security for Constrained RESTful Environments (OSCORE)".

[15] IETF RFC 8949: "Concise Binary Object Representation (CBOR)".

[16] IETF RFC 5869: "HMAC-based Extract-and-Expand Key Derivation Function (HKDF)".

[17] 3GPP TS 23.502: "Procedures for the 5G System".

3 Definitions of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in 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].

AKMA subscription data: The data in the home operator's network indicating whether or not the subscriber is allowed to use AKMA.

AKMA context: A set of parameters stored in AAnF, including SUPI, GPSI, K_{AKMA},A-KID and K_{AF} expiration time.

3.2 Symbols

Void.

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].

A-KID	AKMA Key IDentifier
A-TID	AKMA Temporary UE IDentifier
AAnF	AKMA Anchor Function
AF	Application Function
AF_ID	AF Identifier
AKMA	Authentication and Key Management for Applications
AMF	Access and Mobility Management Function
AUSF	AUthentication Server Function
CBOR	Concise Binary Object Representation
CoAP	Constrained Application Protocol
K_{AF}	AKMA Application Key
K_{AKMA}	AKMA Anchor Key
KDF	Key Derivation Function
NEF	Network Exposure Function
OSCORE	Object Security for Constrained RESTful Environments
RID	Routing InDicator
UDM	Unified Data Management

4 Architecture for AKMA

4.1 Reference model

Figure 4.1-1 shows a fundamental network model of AKMA, as well as the interfaces between them.

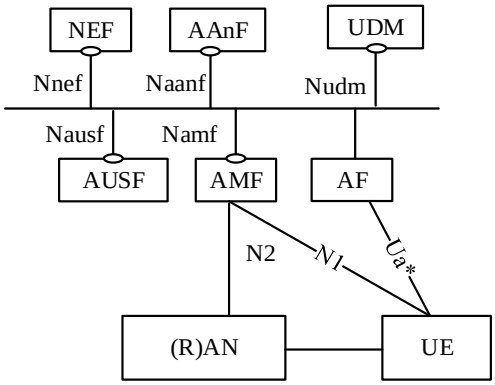

Figure 4.1-1: Fundamental Network Model for AKMA

NOTE: Figure 4.1-1 shows the case where AAnF is deployed as a standalone function. Deployments can choose to collocate AAnF with AUSF or with NEF according to operators' deployment scenarios.

Figure 4.1-2 shows the AKMA architecture using the reference point representation.

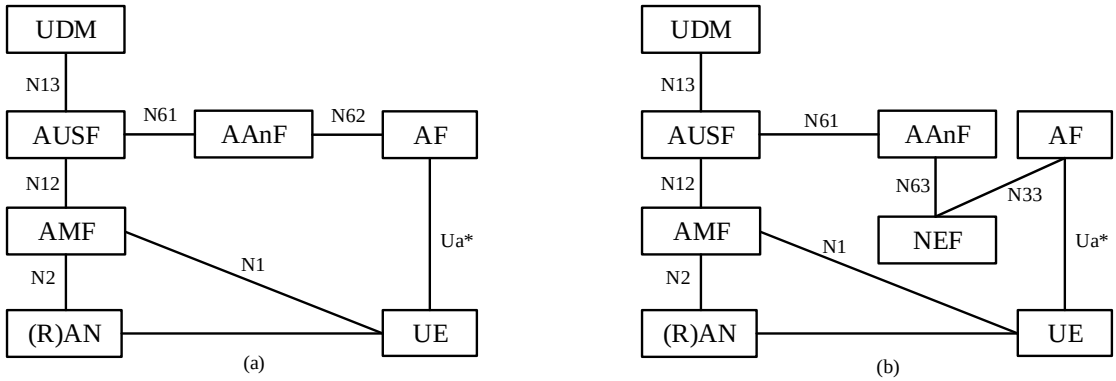

Figure 4.1-2: AKMA Architecture in reference point representation for (a) internal AFs of HPLMN and (b) external AFs

The AKMA service requires a new logical entity, called the AKMA Anchor Function (AAnF).

The AKMA Architecture in Figure 4.1-2 is applicable to both roaming scenario and non-roaming scenario:

- non-roaming: UE is in HPLMN and accessing an AF;
- roaming scenario#1: UE is in VPLMN and accessing an internal HPLMN AF;
- roaming scenario#2: UE is in VPLMN and accessing an internal VPLMN AF;
- roaming scenario#3: UE is in VPLMN and accessing an external AF in the Data Network.

4.2 Network elements

4.2.1 AAnF

The AAnF is the anchor function in the HPLMN. The AAnF stores the AKMA Anchor Key (K_{AKMA}) and SUPI/GPSI for AKMA service, which is received from the AUSF/UDM after the UE completes a successful 5G primary authentication. The AAnF also generates the key material to be used between the UE and the Application Function (AF) and maintains UE AKMA contexts. The AAnF sends SUPI/GPSI of the UE to AF located inside the operator's network according to the AF request or sends SUPI to NEF. If GPSI is required, the AAnF retrieves the GPSI from UDM based on available SUPI. The AAnF has the capability to trigger a primary authentication for K_{AKMA} refreshing purpose.

4.2.2 AF

The AF is defined in TS 23.501 [3] with additional functions:

- AF with the AKMA service enabling requests for AKMA Application Key, called $K_{AF,}$ from the AAnF using A-KID.
- AF shall be authenticated and authorized by the operator network before providing the K_{AF} to the AF.
- The AF located inside the operator's network performs the AAnF selection.

4.2.3 NEF

The NEF is defined in TS 23.501 [3] with additional functions:

- The NEF enables and authorizes the external AF assessing AKMA service and forwards the request towards the AAnF.
- The NEF performs the AAnF selection.

4.2.4 AUSF

The AUSF is defined in TS 23.501 [3] with additional functions:

- AUSF provides the SUPI and AKMA key material (A-KID, K_{AKMA}) of the UE to the AAnF.
- AUSF performs the AAnF selection.

4.2.5 UDM

The UDM is defined in TS 23.501 [3] with the additional functions:

- UDM stores AKMA subscription data of the subscriber and provides AKMA indication and RID to AUSF.
- UDM triggers primary authentication to refresh K_{AKMA}.

4.3 AKMA Service Based Interfaces(SBIs)

4.3.0 General

The following interfaces are involved in AKMA network architecture:

- **Nnef:** Service-based interface exhibited by NEF.
- **Nudm:** Service-based interface exhibited by UDM.

NOTE 1: UDM services related to AKMA service are defined in TS 33.501 [2] clauses 14.2.2, 14.2.6, TS 23.502 [17] clauses 5.2.3.3.2, 5.2.3.5.2.

- **Naanf:** Service-based interface exhibited by AAnF.

The AAnF interacts with the AUSF and the AF using Service-based Interfaces. When the AF is located in the operator's network, the AAnF shall use Service-Based Interface to communicate with the AF directly. When the AF is located outside the operator's network, the NEF shall be used to exchange the messages between the AF and the AAnF.

4.3.1 Void

4.4 Security requirements and principles for AKMA

4.4.0 General

The following security requirements are applicable to AKMA:

- AKMA shall reuse the same UE subscription and the same credentials used for 5G access.
- AKMA shall reuse the 5G primary authentication procedure and methods specified in TS 33.501 [2] for the sake of implicit authentication for AKMA services.
- The SBA interface between the AAnF and the AUSF shall be confidentiality, integrity and replay protected.
- The SBA interface between AAnF and AF/NEF shall be confidentiality, integrity and replay protected.
- The SBA interface between AAnF and UDM shall be confidentiality, integrity and replay protected.
- The AKMA Application Key (K_{AF}) shall be provided with a maximum lifetime based on the operator's local authentication policy.

NOTE: Void

4.4.1 Requirements on Ua* reference point

The Ua* reference point is application specific. The generic requirements for Ua* are:

- Ua* protocol shall be able to carry AKMA Key Identifier (A-KID) .
- The UE and the AKMA AF shall be able to secure the reference point Ua* using the AKMA Application Key derived from the AKMA Anchor Key.

NOTE 1: The exact method of securing the reference point Ua* depends on the application protocol used over reference point Ua*.

NOTE 2: Void

- The Ua* protocol shall be able to handle the expiration of $K_{AF.}$

4.4.2 Requirements on AKMA Key Identifier (A-KID)

Requirements for AKMA Key Identifier (A-KID) are:

- A-KID shall be globally unique.
- A-KID shall be usable as a key identifier in protocols used in the reference point Ua*.
- AKMA AF shall be able to identify the AAnF serving the UE from the A-KID.

4.4.3 Requirements on the UE

The requirements on the UE are:

- Applications on the UE shall not be able to get access to $K_{AKMA.}$
- An application on the UE shall only get the K_{AF} keys related to specific AF Identifiers (AF_IDs) that the application is authorized to get.
- An application on the UE shall not be able to get access to the K_{AF} keys that belong to other applications.

NOTE: How these requirements are satisfied is out of scope of 3GPP.

4.5 AKMA reference points

The AKMA architecture reuses the following reference point from the 5GC for the execution of the primary authentication procedure:

N1: Reference point between the UE and the AMF.

N2: Reference point between the (R)AN and the AMF.

N12: Reference point between AMF and AUSF.

N13: Reference point between the UDM and the AUSF.

N33: Reference point between NEF and an external AF.

The AKMA architecture defines the following reference points:

N61: Reference point between the AAnF and the AUSF.

N62: Reference point between the AAnF and an internal AF.

N63: Reference point between the AAnF and NEF.

Ua*: Reference point between the UE and an AF.

NOTE: The reference point Ua* carries the application protocol, which is secured using the key material agreed between UE and AAnF as a result of successful AKMA procedures.

4.6 Roaming

4.6.1 AKMA roaming requirements

Requirements for AKMA roaming are:

- The roaming subscriber shall be able to utilize the AKMA feature provided by the home network.
- The home network shall be able to control whether its subscriber is authorized to use the service in the visited network.

4.7 Use of Authentication Proxy (AP)

4.7.1 Architecture of using AP

An Authentication Proxy (AP) is a proxy which takes the role of an AF and delegates a group of Application Servers (ASs). It may reside between the UE and the AS as depicted in the figures below. The AP helps the ASs behind the AP to execute AKMA procedures to save the consumption of signalling resources and AAnF computing resources. It may also relieve the AS of security tasks. The use of an AP is fully compatible with the architecture specified in the present document.

The AP can assure the ASs that the request is coming from an authorized subscriber of the MNO.

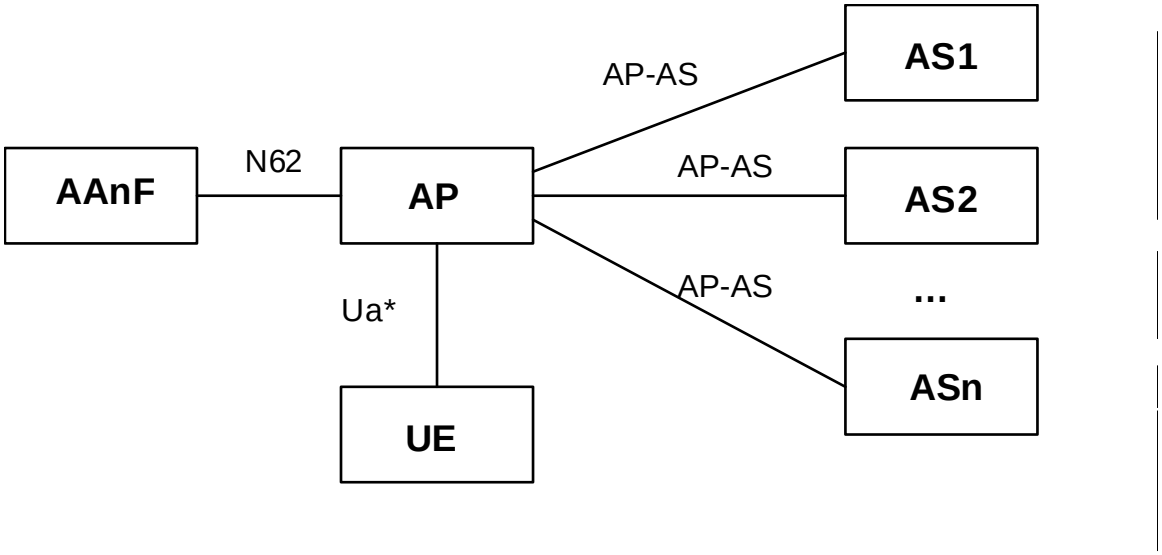

Figure 4.7.1-1: Environment and reference points of AP when AP is internal

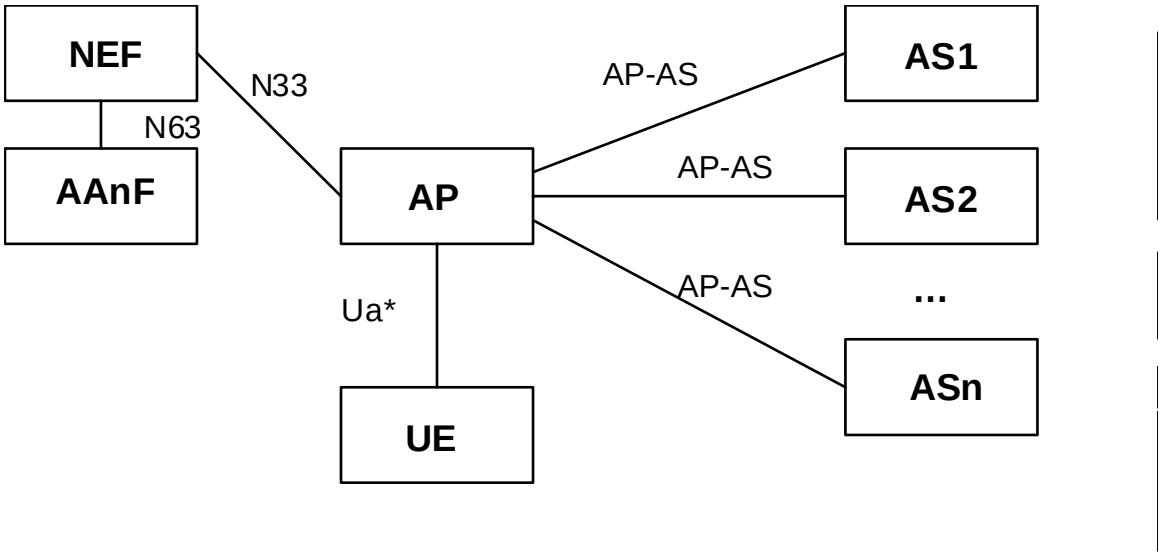

Figure 4.7.1-2: Environment and reference points of AP when AP is external

If the Ua* is HTTP based, the UE is configured with the FQDN of AS, and the AP is a reverse proxy to handle the communication between the UE and the AS. The AP takes the role of an AF. The AKMA Application Key (i.e. K_{AF}), which is utilized between the UE and the AP, is derived based on the FQDN of the AS.

If the Ua* is not HTTP based, it is left to implementation, e.g., how the AP identifies the traffic towards corresponding AS may be pre-configured in the AP by the operator who deploys the AP.

4.7.2 AP-AS reference point

The HTTP protocol is run over the AP-AS reference point.

Confidentiality and integrity protection can be provided for the reference point between the AP and the AS using NDS/IP mechanisms as specified in TS 33.210 [13]. For traffic between different security domains, the Za reference point shall be operated. For traffic inside a security domain, it is up to the operator to decide whether to deploy the Zb reference point.

4.7.3 Example of using AP for TLS tunnels

When the TLS based protocol is used as Ua* profile, the AP can be used to handle the TLS security relation with the UE and relieves the AS of this task. When an HTTPS request is destined towards an AS behind an AP, the AP terminates the TLS tunnel and performs UE authentication. The AP proxies the HTTP requests received from UE to one or many application servers. The AP may add an assertion of identity of the subscriber for use by the AS, when the AP forwards the request from the UE to the AS.

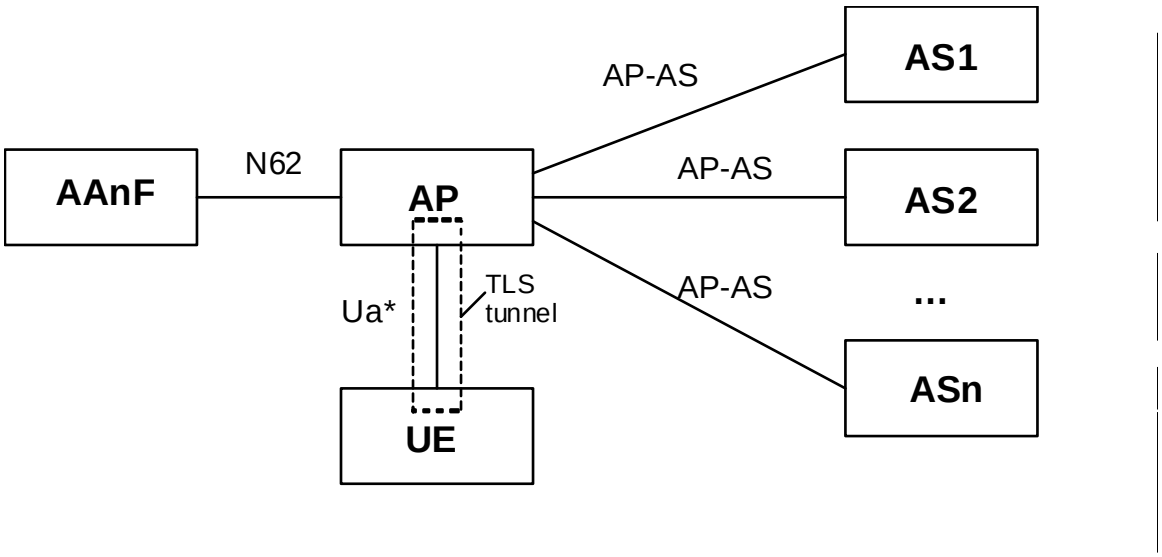

Figure 4.7.3-1: Environment and reference points of AP for TLS tunnels when AP is internal

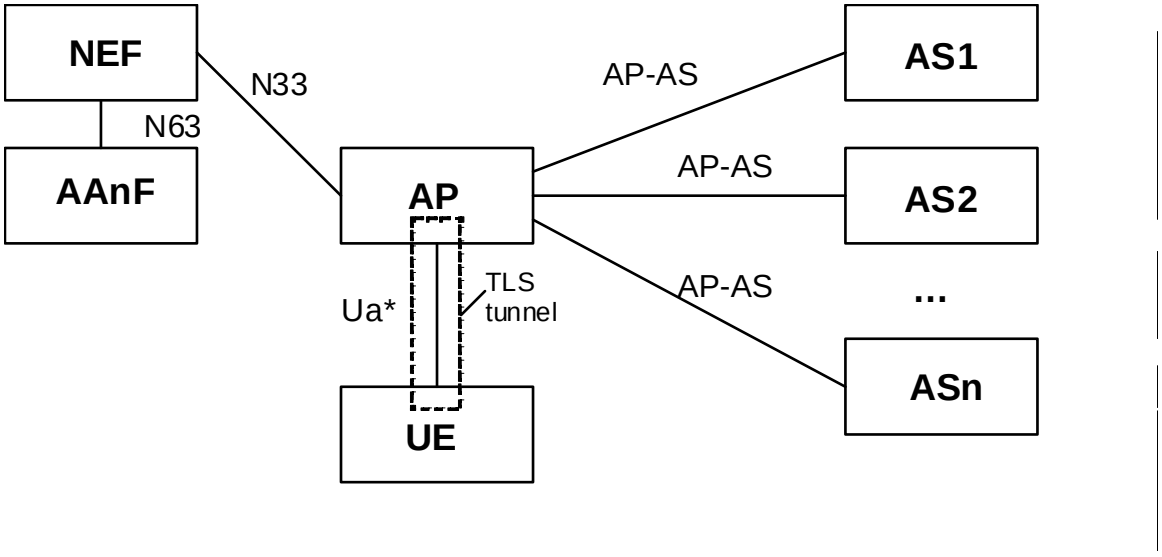

Figure 4.7.3-2: Environment and reference points of AP for TLS tunnels when AP is external

5 Key management

5.1 AKMA key hierarchy

The key hierarchy (see Figure 5.1-1) includes the following keys: K_{AUSF}, K_{AKMA}, $K_{AF.}$ K_{AUSF} is generated by AUSF as specified in clause 6.1 of TS 33.501 [2].

Keys for AAnF:

- K_{AKMA} is a key derived by ME and AUSF from K_{AUSF}.

Keys for AF:

- K_{AF} is a key derived by ME and AAnF from K_{AKMA}.

K_{AKMA} and K_{AF} are derived according to the procedures of clauses 6.1 and 6.2.

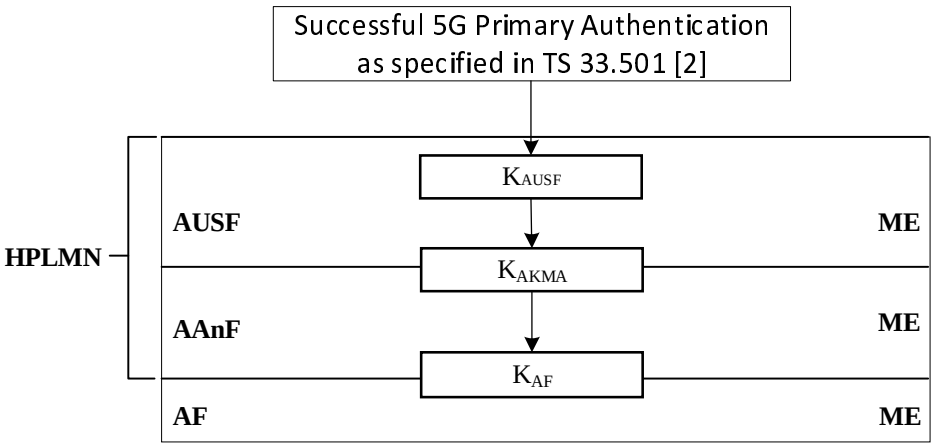

Figure 5.1-1: AKMA Key Hierarchy

5.2 AKMA key lifetimes

The K_{AKMA} and A-KID are valid until the next successful primary authentication is performed (implicit lifetime), in which case the K_{AKMA} and A-KID are replaced.

AKMA Application Keys K_{AF} shall use explicit lifetimes based on the operator's policy. The lifetime of K_{AF} shall be sent by the AAnF as described in clauses 6.2 and 6.3. In case that a new AKMA Anchor Key K_{AKMA} is established, the AKMA Application Key K_{AF} can continue to be used for the duration of the current application session or until its lifetime expires, whichever comes first. When the K_{AF} lifetime expires, a new AKMA Application Key is established based on the current AKMA Anchor Key K_{AKMA}.

NOTE: When the K_{AF} lifetime expires and the K_{AKMA} has not changed in AAnF, according to the Annex A.4, the AKMA Application Key which is established based on the current AKMA Anchor Key K_{AKMA} is not a new one.

6 AKMA Procedures

6.1 Deriving AKMA key after primary authentication

There is no separate authentication of the UE to support AKMA functionality. Instead, AKMA reuses the 5G primary authentication procedure executed e.g. during the UE Registration to authenticate the UE. A successful 5G primary authentication results in K_{AUSF} being stored at the AUSF and the UE. Figure 6.1-1 shows the procedure to derive K_{AKMA} after a successful primary authentication.

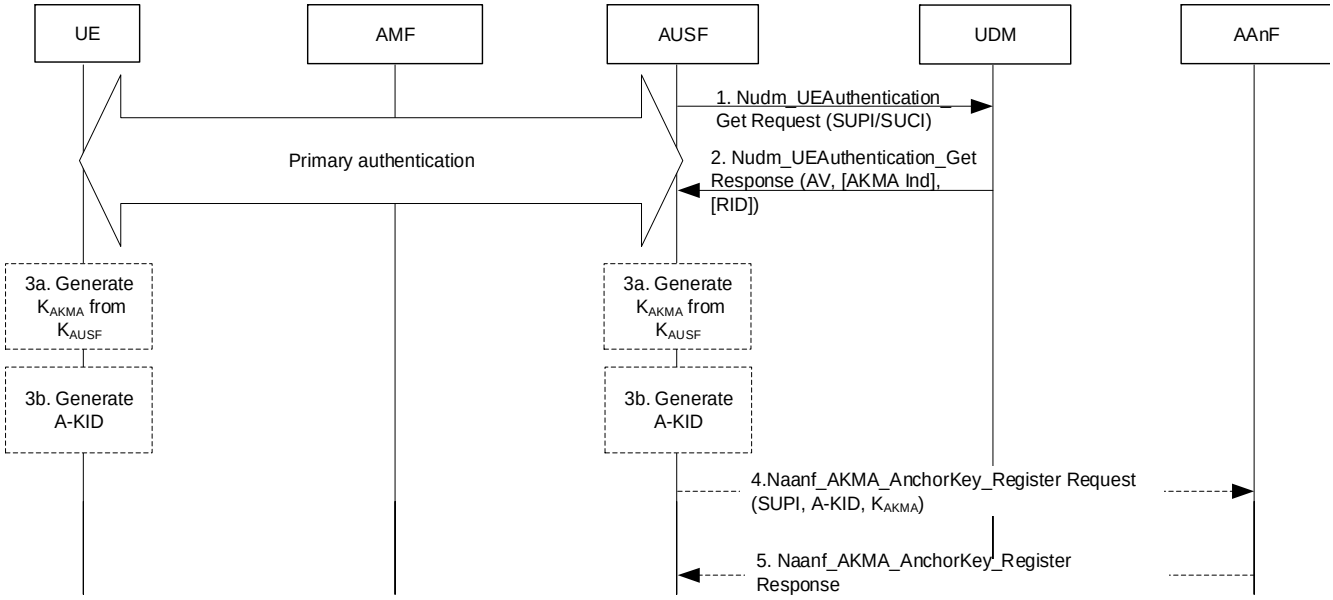

Figure 6.1-1: Deriving K_{AKMA} after primary authentication

1) During the primary authentication procedure, the AUSF interacts with the UDM in order to fetch authentication information such as subscription credentials (e.g. AKA Authentication vectors) and the authentication method using the Nudm_UEAuthentication_Get Request service operation.

2) In the response, the UDM may also indicate to the AUSF whether the AKMA Anchor key needs to be generated for the UE. If the AKMA indication is included, the UDM shall also include the RID of the UE.

3) If the AUSF receives the AKMA indication from the UDM, the AUSF shall store the K_{AUSF} and generate the AKMA Anchor Key (K_{AKMA}) and the A-KID from K_{AUSF} after the primary authentication procedure is successfully completed.

   The UE shall generate the AKMA Anchor Key (K_{AKMA}) and the A-KID from the K_{AUSF} before initiating communication with an AKMA Application Function.

4) After AKMA key material is generated, the AUSF selects the AAnF as defined in clause 6.7, and shall send the generated A-KID and K_{AKMA} to the AAnF together with the SUPI of the UE using the Naanf_AKMA_AnchorKey_Register Request service operation. The AAnF shall store the latest information sent by the AUSF.

NOTE 1: The AUSF need not store any AKMA key material after delivery to the AAnF.

NOTE 1a: When re-authentication runs, the AUSF generates a new A-KID, and a new K_{AKMA} and sends the new generated A-KID and K_{AKMA} to the AAnF. After receiving the new generated A-KID and K_{AKMA}, the AAnF deletes the old A-KID and K_{AKMA} and stores the new generated A-KID and K_{AKMA}.

5) The AAnF sends the response to the AUSF using the Naanf_AKMA_AnchorKey_Register Response service operation.

A-KID identifies the K_{AKMA} key of the UE.

A-KID shall be in NAI format as specified in clause 2.2 of IETF RFC 7542 [6], i.e. username@realm. The username part shall include the RID and the A-TID (AKMA Temporary UE Identifier), and the realm part shall include Home Network Identifier.

The A-TID shall be derived from K_{AUSF} as specified in Annex A.3.

The AUSF shall use the RID received from the UDM as described in step 2 to derive A-KID.

NOTE 2: The chance of A-TID collision is not zero but practically low as the A-TID derivation is based on KDF specified in Annex B of TS 33.220 [4]. The detection of A-TID collision as well as potential handling of collision is not addressed in the present document.

K_{AKMA} shall be derived from K_{AUSF} as specified in Annex A.2. Since K_{AKMA} and A-TID in A-KID are both derived from K_{AUSF} based on primary authentication run, the K_{AKMA} and A-KID can only be refreshed by a new successful primary authentication.

6.2 Deriving AKMA Application Key for a specific AF

6.2.1 AAnF response with UE Identity

Figure 6.2-1 shows the procedure used by the AF to request application function specific AKMA keys from the AAnF, when the AF is located inside the operator's network.

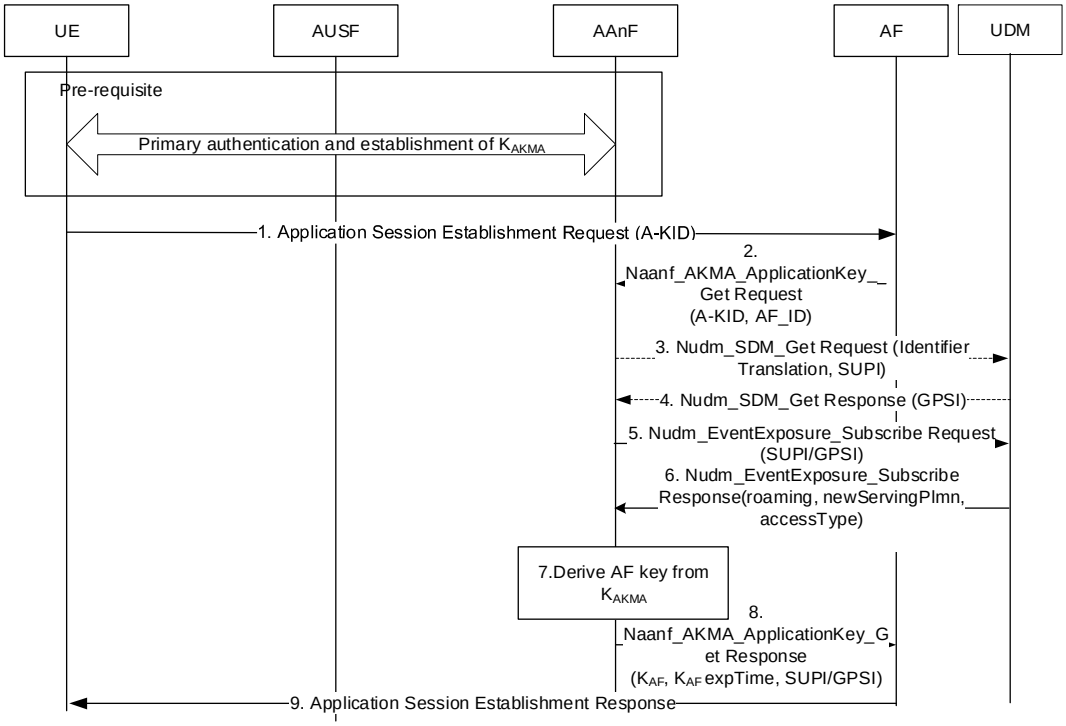

Figure 6.2-1: K_{AF} generation from K_{AKMA}

Before communication between the UE and the AKMA AF can start, the UE and the AKMA AF need to know whether to use AKMA. This knowledge is implicit to the specific application on the UE and the AKMA AF or indicated by the AKMA AF to the UE (see clause 6.5).

1. The UE shall generate the AKMA Anchor Key (K_{AKMA}) and the A-KID from the K_{AUSF} before initiating communication with an AKMA Application Function. When the UE initiates communication with the AKMA AF, it shall include the derived A-KID (see clause 6.1) in the Application Session Establishment Request message. The UE may derive K_{AF} before sending the message or afterwards.

2. If the AF does not have an active context associated with the A-KID, then the AF selects the AAnF as defined in clause 6.7, and sends a Naanf_AKMA_ApplicationKey_Get request to AAnF with the A-KID to request the K_{AF} for the UE. The AF also includes its identity (AF_ID) in the request. If AF wants to receive a notification for AKMA service disabling, the AF shall include AKMA service disable URI in the Naanf_AKMA_ApplicationKey_Get request. Based on the AKMA service disable URI, the AAnF shall create an implicit subscription for the AF for the AAnF to later notify the AF about AKMA service disable as defined in clause 6.8. Implicit subscription has an expiration time set by operator policy.

   AF_ID consists of the FQDN of the AF and the Ua* security protocol identifier (see Annex A.4). The latter parameter identifies the security protocol that the AF will use with the UE.

The AAnF shall check whether the AAnF can provide the service to the AF based on the configured local policy or based on the authorization information available in the signalling (i.e., Oauth2.0 token). If it succeeds, the following procedures are executed. Otherwise, the AAnF shall reject the procedure.

The AAnF shall verify whether the subscriber is authorized to use AKMA based on the presence of the UE specific K_{AKMA} key identified by the A-KID.

If K_{AKMA} is present in AAnF, the AAnF shall continue with step 3.

If K_{AKMA} is not present in the AAnF, the AAnF shall continue with step 6 with an error response.

3. Once receiving the request from the AF, if the AAnF determines this specific AF needs GPSI, according to its local policy, the AAnF sends a Nudm_SDM_Get Request to the UDM to fetch the GPSI of the UE. If the specific AF does not need GPSI, the AAnF shall continue with step 5.

4. The UDM responds with the GPSI of the UE. The AAnF shall store the received GPSI as part of UE's AKMA context.

5. Once receiving the request from the AF, the AAnF shall send a Nudm_EventExposure_Subscribe request to UDM with SUPI/GPSI to request the RoamingStatusReport from the UDM.

6. The UDM shall send the Nudm_EventExposure_Subscribe response to the AAnF with the information of roaming status.

NOTE: Later on, when the roaming status changes, the UDM also sends a notification to the AAnF about the updated roaming information.

7. Once the AAnF receives the roaming status from the UDM, it checks the local policy and determines whether to provide service to the UE. If yes, the AAnF derives the AKMA Application Key (K_{AF}) from K_{AKMA} if it does not already have K_{AF}. The AAnF shall store the K_{AF} expiration time as part of UE's AKMA context.

   When UE is dual registered, the UE is treated as roaming if at least one of the serving PLMNs indicates the UE is roaming.

   The key derivation of K_{AF} shall be performed as specified in Annex A.4.

8. If the AAnF determines to provide AKMA service to the UE, the AAnF sends Naanf_AKMA_ApplicationKey_Get response to the AF with SUPI/GPSI, K_{AF} and the K_{AF} expiration time. Whether to send SUPI or GPSI is determined by AAnF based on the local policy. If the AAnF finds that roaming is not allowed, it shall respond the AF containing a failure indication that roaming is not allowed. If AAnF has subscribed the event for RoamingStatusReport, then the AAnF is expected to keep track of the transmitted A-KIDs and the recipient AFs.

NOTE 1: When UE re-authentication occurs, a new A-KID is provided to AAnF for the same SUPI while the AF still maintains the originally transmitted A-KID. If the AAnF uses the new A-KID in the RoamingStatusReport the AF will not find the AF information asscociated with the new A-KID and the AF actions might fail.

9. The AF sends the Application Session Establishment Response to the UE. If the information in step 8 indicates failure of AKMA key request, the AF shall reject the Application Session Establishment by including a failure cause. Afterwards, UE may trigger a new Application Session Establishment request with the latest A-KID to the AKMA AF.

6.2.2 AAnF response without UE Identity

In some scenarios, anonymous user access to the AF is desirable (e.g., UE identification is not required at the AF). For allowing such anonymous user access to the AF, the procedure detailed in clause 6.2.1 of the present document is used with the following changes:

- in step 2, instead of Naanf_AKMA_ApplicationKey_Get request, Naanf_AKMA_ApplicationKey_AnonUser_Get request is used by the AF; and

- in step 6, the AAnF sends Naanf_AKMA_ApplicationKey_AnonUser_Get response to the AF with K_{AF} and the K_{AF} expiration time. The AAnF shall store the KAF expiration time as part of UE's AKMA context.

The A-KID functions as a temporary user identifier.

6.3 AKMA Application Key request via NEF

Figure 6.3-1 shows the procedure used by the AF to request K_{AF} from the AAnF via NEF, when the AF is located outside the operator's network.

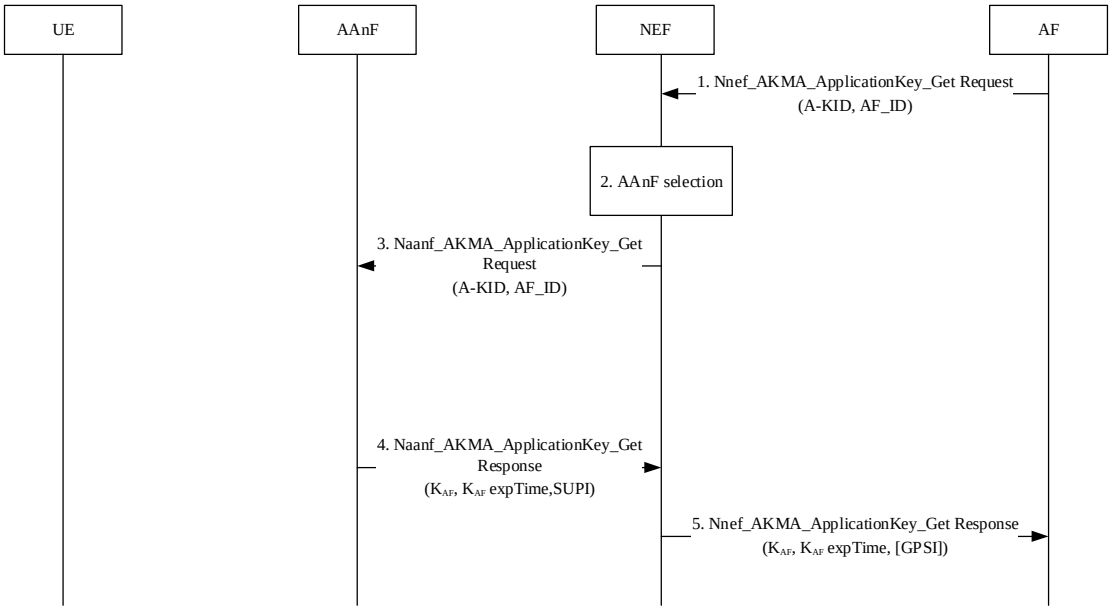

Figure 6.3-1: AKMA Application Key request via NEF

1. When the AF is about to request AKMA Application Key for the UE from the AAnF, e.g. when UE initiates application session establishment request as in clause 6.2.1, the AF discovers the HPLMN of the UE based on the A-KID and sends the request towards the AAnF via NEF service API. The request shall include the A-KID and the AF_ID and optionally UE Id not needed indication.

NOTE: In the case of architecture without CAPIF support, the AF is locally configured with the API termination points for the service. In the case of architecture with CAPIF support, the AF obtains the service API information from the CAPIF core function via the Availability of service APIs event notification or Service Discover Response as specified in TS 23.222 [5].

2. If the AF is authorized by the NEF to request K_{AF}, including the authorization after verification of the AF_ID in step 1, the NEF discovers and selects an AAnF as defined in clause 6.7.

3. The NEF sends a Naanf_AKMA_ApplicationKey_Get request to the selected AAnF with the A-KID to request the K_{AF} for the UE.

   The AAnF shall process the request in the same way as specified in clause 6.2.1 with following changes:

   If K_{AKMA} is present in AAnF, the AAnF shall continue with step 4 in this clause.

   If K_{AKMA} is not present in the AAnF, the AAnF shall continue with step 5 in this clause with an error response.

4. Once receiving the request from the AF, AAnF shall request the UE roaming status report from UDM as specified in clause 6.2.1, step 5-6. If the AAnF determines to provide AKMA service to the UE, the AAnF generates the K_{AF} as specified in clause 6.2.1 and sends the response to the NEF with the K_{AF}, the K_{AF} expiration time (K_{AF} exptime) and SUPI. The AAnF shall store the KAF expiration time as part of UE's AKMA context. If the AAnF finds that roaming is not allowed, it shall respond the AF containing a failure indication that roaming is not allowed.

5. The NEF forwards the response to the AF, the response contains the K_{AF}, the K_{AF} expiration time (K_{AF} exptime) and optionally GPSI (external ID) or the failure indication of roaming not allowed. Based on local policy, the NEF uses the Nudm_SubscriberDataManagement service which is specified in TS 29.503[11] to translate SUPI

to GPSI (external ID) and optionally include GPSI (external ID) in the response. If UE Id not needed indication is received in the incoming request, the NEF shall not provide the GPSI (external ID) to AF. The NEF shall not send the SUPI to the AF.

6.4 AKMA key change

6.4.1 K_{AKMA} re-keying

K_{AKMA} shall be re-keyed by running a successful primary authentication as described in clause 6.1.

6.4.2 K_{AF} re-keying

The K_{AF} re-keying depends on the lifetime of the K_{AF} and may be triggered by the AF, which means that when a new K_{AKMA} is derived, the K_{AF} will not be re-keyed automatically.

When the lifetime of K_{AF} expires, the AF may reject UE's access to the AF or refresh the K_{AF} as described in clause 6.4.3 based on its policy. If the AF chooses to reject UE's access, the AF may provide a cause indicating that the K_{AF} has expired via Ua* protocol specific means so that the UE can take appropriate action. If there has been a change of K_{AUSF} (e.g., due to a successful run of primary authentication), the UE may re-try accessing the AF by using the A-KID derived from the new K_{AUSF}.

6.4.3 K_{AF} refresh

There is no support for an explicit K_{AF} refresh procedure in this document. If a primary authentication does not take place, the K_{AUSF}, K_{AKMA} and K_{AF} remain unchanged since the latest primary authentication.

The K_{AF} may be refreshed by the K_{AKMA} refresh defined in clause 6.4.4 as decided by AAnF.

NOTE 1: The AAnF can decide K_{AKMA} refresh based on local policy.

Ua* protocol may support refresh of derived session keys from K_{AF}. If the Ua* protocol supports the refresh of derived session keys from K_{AF}, the AF may refresh the K_{AF} at any time using the Ua* protocol.

NOTE 2: How a fresh key is derived for AKMA is up to Ua* protocol implementation.

NOTE 3: A session key based on K_{AF} refreshed using the Ua* protocol is only known by UE and AF.

6.4.4 K_{AKMA} refresh

As defined in TS 33.501[2] clause 6.1.5, the AAnF may decide to refresh the K_{AKMA} based on the operator's local authentication policy by sending the Nudm_UECM_AuthTrigger Request message to the UDM. The UDM may further decide whether to trigger the primary authentication as defined in clause 6.1.5 of TS 33.501[2].

6.5 Initiation of AKMA

In case when the UE does not know to use AKMA for a service, then the following procedure shown in figure 6.5-1 applies.

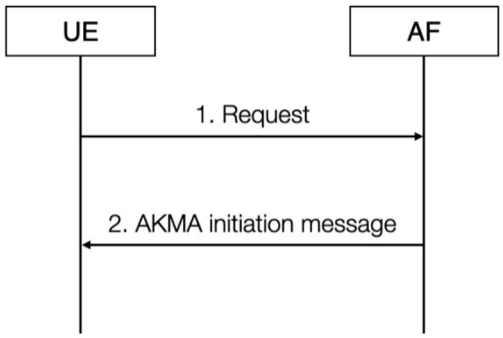

Figure 6.5-1: Initiation of AKMA

1. The UE may start communication over reference point Ua* with the AF with or without any AKMA-related parameters.
2. If the AF requires the use of shared keys obtained by means of the AKMA, but the request from UE does not include AKMA-related parameters, the AF replies with an AKMA initiation message. The form of this initiation message may depend on the particular reference point Ua*.

In case the UE knows to use AKMA for a service, then it directly initiates the procedure in clause 6.2.

6.6 AAnF AKMA context removal

6.6.1 General

This procedure is used to remove the AKMA context in the AAnF. NF consumers may initiate this procedure due to local policy.

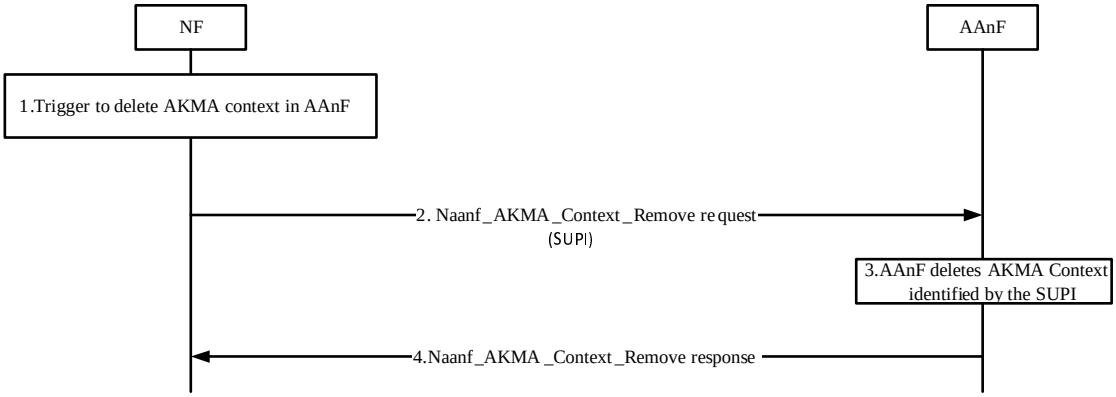

Figure 6.6.1-1: AAnF AKMA context removal procedure

1. NF initiates an AAnF AKMA context removal procedure to delete the AKMA context in AAnF.

2. NF discovers the AAnF of the UE, as specified in clause 6.7 and sends a Naanf_AKMA_Context_Remove request with SUPI to AAnF to remove AKMA context for the UE.

3. AAnF shall delete AKMA Context (e.g. SUPI, A-KID, K_{AKMA}, GPSI and K_{AF} expiration time) from its local database identified by SUPI.

4. AAnF sends a Naanf_AKMA_Context_Remove response to NF. This response is just an acknowledgement of the request received.

6.7 AAnF Discovery and Selection

The NF consumer or the SCP performs AAnF discovery to discover an AAnF instance.

In the case of NF consumer-based discovery and selection, the following applies:

- Internal AFs and the NEF performs AAnF instance selection that handles the AKMA request. The AF/NEF shall utilize the NRF to discover the AAnF instance(s) unless AAnF information is available by other means, e.g. locally configured on the AF/NEF.

- The AUSF performs AAnF selection to allocate an AAnF Instance to send the AKMA key material related to the UE. The AUSF shall utilize the NRF to discover the AAnF instance(s) unless AAnF information is available by other means, e.g. locally configured on the AUSF.

- The NF specified in clause 6.6 performs AAnF instance selection that handles the AKMA request. The NF shall utilize the NRF to discover the AAnF instance(s) unless AAnF information is available by other means, e.g. locally configured on the NF specified in clause 6.6.

The AAnF selection functionality in NF consumer or in SCP should consider the following factor:

- the UE's Routing Indicator.

NOTE 1: The AF/NEF obtains the Routing Indicator as part of the A-KID in the AKMA request. The AUSF obtains the Routing Indicator within the Nudm_UEAuthentication_Get Response from the UDM.

Internal AFs, the NEF and the AUSF shall select the same AAnF set based on the UE’s Routing Indicator.

When the UE's Routing Indicator is set to its default value as defined in TS 23.003 [9], the AAnF NF consumer can select any AAnF instance within the home network of the UE.

NOTE 2: In scenarios where multiple sets of AAnFs are deployed, it is left up to implementation how to ensure that the AAnF NF consumers select an AAnF instance within the AAnF set the UE belongs to when the UE's Routing Indicator is set to its default value.

In the case of delegated discovery and selection in SCP, the AAnF NF consumer shall send all available factors to the SCP.

6.8 Notification about AKMA service disabling

This procedure is used when the AKMA sessions have already been started (before roaming was detected), and as soon as PLMN change is detected at the AAnF, the AAnF may execute this procedure based on the roaming policy.

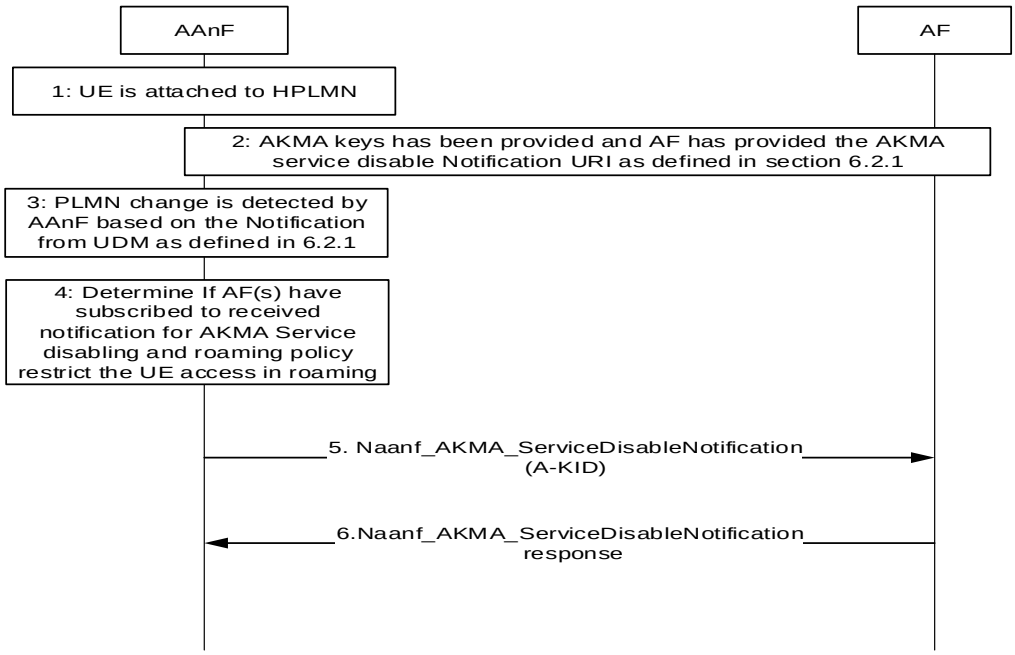

Figure 6.8.1-1: AAnF notification to AF about AKMA service disable

1. UE registers with a (H)PLMN.

2. UE is accessing the AF and key material is provided to AF as described in 6.2.1. While accessing the AAnF, AF may also provide the Notification URI.

3. UE is getting registered in a VPLMN and AAnF detects the PLMN change via the Nudm_EventExposure_Notification received from UDM.

4. AAnF determines if AF(s) have subscribed to receive notifications for AKMA service disabling and roaming policy is configured and restrict the AKMA access in the VPLMN; if yes, steps 5 and 6 are executed. Otherwise, steps 5 and 6 are skipped.

5. If AF(s) are determined at step 5, the AAnF shall send notifications to the subscribed AF(s) about AKMA roaming via Naanf_AKMA_ServiceDisableNotification. The A-KID is the transmitted A-KID for the corresponding AF, which is kept track of in step 8 in clause 6.2.1..

6. The AF shall send the response and based on the notification and internal policy, the AF may stop the UE service, may stop the encryption.

NOTE: By stopping the encryption (e.g., TLS 1.2 NULL cypher negotiation), LI interception could work in the VPLMN.

7 Security related services

7.1 Services provided by AAnF

7.1.1 General

The following table shows the AAnF Services and AAnF Service Operations.

Table 7.1.1-1: List of AAnF Services

Service Name	Service Operations	Operation Semantics	Example Consumer(s)
Naanf_AKMA	AnchorKey_Register	Request/Response	AUSF
	ApplicationKey_Get	Request/Response	AF, NEF
	Context_Remove	Request/Response	OAM
	ApplicationKey_ AnonUser_Get	Request/Response	AF
	ServiceDisableNotification	Notification/Response	AF

7.1.2 Naanf_AKMA_AnchorKey_Register service operation

Service operation name: Naanf_AKMA_AnchorKey_Register.

Description: The NF consumer requests the AAnF to store the AKMA related key material.

Input, Required: SUPI, A-KID, K_{AKMA}

Input, Optional: None.

Output, Required: None.

Output, Optional: None.

7.1.3 Naanf_AKMA_ApplicationKey_Get service operation

Service operation name: Naanf_AKMA_ApplicationKey_Get.

Description: The NF consumer requests AKMA Application Key and UE ID from the AAnF.

Input, Required: A-KID, AF_ID

Input, Optional: Service Disable URI.

Output, Required: .

Output, Optional: KAF, KAF expiration time and SUPI or GPSI or failure indication.

7.1.4 Naanf_AKMA_Context_Remove operation

Service operation name: Naanf_AKMA_Context_Remove.

Description: The NF consumer requests the AAnF to remove the AKMA related key material.

Input, Required: SUPI.

Input, Optional: None.

Output, Required: None.

Output, Optional: None.

7.1.5 Naanf_AKMA_ApplicationKey_AnonUser_Getservice operation

Service operation name: Naanf_AKMA_ApplicationKey_AnonUser_Get.

Description: The NF consumer requests only the AKMA Application Key from the AAnF. This service is for allowing anonymous user access to the AF based on A-KID (i.e., UE identification is not required at the AF). The A-KID functions as a temporary user identifier.

Input, Required: A-KID, AF_ID

Input, Optional: Service Disable URI.

Output, Required: K_{AF}, K_{AF} expiration time.

Output, Optional: None.

7.1.6 Naanf_AKMA_ServiceDisableNotification service operation

Service operation name: Naanf_AKMA_ServiceDisableNotification

Description: AAnF notifies the NF consumer about AKMA service disable

NOTE: The AF is implicitly subscribed to receive Naanf_AKMA_ServiceDisableNotification service operation.

Input, Required: A-KID

Input, Optional: None

Output, Required: None

Output, Optional: None

7.2 Void

7.3 Services provided by NEF

7.3.1 General

The NEF exposes AKMA Application Key derivation service to the requester NF.

The following table shows the NEF Services and NEF Service Operations related to AKMA service.

Table 7.3.1-1: List of NEF Services

Service Name	Service Operations	Operation Semantics	Example Consumer(s)
Nnef_AKMA	ApplicationKey_Get	Request/Response	AF

7.3.2 Nnef_AKMA_ApplicationKey_Get service operation

Service operation name: Nnef_AKMA_ApplicationKey_Get.

Description: The NF consumer requests the NEF to provide AF related key material.

Input, Required: A-KID, AF_ID

Input, Optional: UEID not needed indication.

Output, Required: K_{AF}, K_{AF} expiration time.

Output, Optional: GPSI (external ID).

7.3.3 Nnef_AKMA_ServiceDisableNotification service operation

Service operation name: Nnef_AKMA_ServiceDisableNotification

Description: NEF notifies the NF consumer about AKMA service is disabled.

Input, Required: A-KID

Input, Optional: None

Output, Required: None

Output, Optional: None

7.4 Services provided by UDM

UDM services related to AKMA service are defined in TS 33.501 [2] clauses 14.2.2, 14.2.6, TS 23.502 [17] clauses 5.2.3.3.2, 5.2.3.5.2.

Annex A (normative): Key derivation functions

A.1 KDF interface and input parameter construction

A.1.1 General

All key derivations for AKMA shall be performed using the key derivation function (KDF) specified in Annex B.2.2 of TS 33.220 [4].

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 [4].

A.1.2 FC value allocations

The FC number space used is controlled by TS 33.220 [4], FC values allocated for the present document are in the range of 0x80 – 0x82.

A.2 K_{AKMA} derivation function

When deriving a K_{AKMA} from K_{AUSF}, the following parameters shall be used to form the input S to the KDF:

- FC = 0x80;
- P0 = "AKMA";
- L0 = length of "AKMA"; (i.e. 0x00 0x04)
- P1 = SUPI;
- L1 = length of SUPI.

The input key KEY shall be the K_{AUSF}.

SUPI shall be the same value as parameter P0 in Annex A.7.0 of TS 33.501 [2].

A.3 A-TID derivation function

When deriving the A-TID from K_{AUSF}, the following parameters shall be used to form the input S to the KDF:

- FC = 0x81;
- P0 = "A-TID";
- L0 = length of "A-TID"; (i.e. 0x00 0x05)
- P1 = SUPI;
- L1 = length of SUPI.

The input key KEY shall be K_{AUSF}.

SUPI shall be the same value as parameter P0 in Annex A.7.0 of TS 33.501 [2].

A.4 K_{AF} derivation function

When deriving a K_{AF} from K_{AKMA}, the following parameters shall be used to form the input S to the KDF:

- FC = 0x82;
- P0 =AF_ID;
- L0 = length of AF_ID

The input key KEY shall be K_{AKMA}.

AF_ID is constructed as follows:

AF_ID = FQDN of the AF || Ua* security protocol identifier, where the Ua* security protocol identifier is specified as Ua security protocol identifier in Annex H of TS 33.220 [4].

Annex B (normative): AKMA profiles for Ua* protocols

B.1 TLS based protocols

B.1.1 General

This annex contains profiles of the share key-based UE authentication with certificate-based AF authentication and the shared key-based mutual authentication between UE and AF that are similar to the ones defined in 3GPP TS 33.222 [7].

B.1.2 Shared key-based UE authentication with certificate-based AF authentication

B.1.2.1 General

The following clause provides the changes needed to adapt the Ua protocol given in clause 5.3 of TS 33.222 [7] to work with a K_{AF} derived using the AKMA procedures.

B.1.2.2 Procedures

The procedures follow those given in clause 5.3.0 of TS 33.222 [7] with the AKMA AF taking the role of the NAF from GBA (see TS 33.220 [4]), with the following changes.

At step 2, if the clients supports AKMA with this protocol then the client shall add the constant string "3gpp-akma" to the "User-Agent" HTTP header as product tokens as specified in IETF RFC 9110 [10].

At step 3, if the AF selects AKMA for deriving the key, then the AF shall include the "3GPP-bootstrapping-akma" within the WWW-Authenticate header field. If the AF has choice between GBA_Digest (see TS 33.220 [4]) and AKMA keying, then the AF shall select AKMA over GBA_Digest (see TS 33.222 [7] for similar consideration between GBA methods).

NOTE 1: The choice between AKMA and AKA-based GBA is application dependent.

At step 4, on receiving the response from the AF, the client shall verify that the FQDN in the realm attribute corresponds to the FQDN of the AF it established the TLS connection with. If failure the client shall terminate the TLS connection with the AF.

At step 5 given AKMA has been selected for keying, the client shall send a response with an Authorization header field where Digest is inserted using the A-KID as username. K_{AF} shall be used as password in the Digest calculation.

At step 6 given AKMA has been selected for keying, the AF shall verify the value of the password attribute using K_{AF} retrieved from AAnF using the A-KID received as username attribute in the query. If the AF is not able to obtain the AF-specific key when using AKMA mode, the AF shall respond with an appropriate error message not containing the realm attributes from step 3.

B.1.3 Shared key-based mutual authentication between UE and AF

B.1.3.1 General

The following clause provides the changes needed to adapt the Ua protocol given in clause 5.4 of TS 33.222 [7] to work with a K_{AF} derived using the AKMA procedures.

B.1.3.2 Procedures

B.1.3.2.1 Procedures for TLS 1.2

The procedures follow those given in clause 5.4.0.1 of TS 33.222 [7] with the AKMA AF taking the role of the NAF from GBA (see TS 33.220 [4]), with the following changes.

At step 2, the AF shall include a constant string "3GPP-AKMA" is used as PSK-identity hint to indicate that AKMA based keying is supported.

At step 3, the UE may use an AKMA generated key if support was indicated by the AF (even if GBA-based keys were also indicated as supported by the AF). To use AKMA generated key, the UE shall derive the TLS premaster secret from K_{AF} and shall send a ClientKeyExchange message including a PSK identity consisting of "3GPP-AKMA" and the A-KID. If the UE has choice between GBA_Digest (see TS 33.220 [4]) and AKMA keying, then the UE shall select AKMA over GBA_Digest (see TS 33.222 [7] for similar consideration between GBA methods).

NOTE 1: The choice between AKMA and AKA-based GBA is application dependent.

At step 4, if the AF receives the "3GPP-AKMA" prefix and the A-KID in the ClientKeyExchange messages it fetches the AF specific shared secret (K_{AF}) from the AAnF using the A-KID. The AF shall derive the TLS premaster secret from the AF specific key (K_{AF}).

B.1.3.2.2 Procedures for TLS 1.3

The procedures follow those given in clause 5.4.0.2 of TS 33.222 [7] with the AKMA AF taking the role of the NAF from GBA (see TS 33.220 [4]), with the following changes.

In step 1, the PSK identities in the ClientHello shall include a prefix indicating the PSK-identity name space (i.e. "3GPP-AKMA") and the A-KID to indicate the UE supports keying with AKMA.

In step 2 if the AF is willing to establish a TLS tunnel using PSK authentication with AKMA keys, then the AF shall indicate the index of the AKMA psk identity in the ServerHello message. If the AF has choice between GBA_Digest (see TS 33.220 [4]) and AKMA keying, then the AF shall select AKMA over GBA_Digest (see TS 33.222 [7] for similar consideration between GBA methods).

NOTE 1: The choice between AKMA and AKA-based GBA is application dependent.

The UE and NAF shall derive the TLS external PSK from K_{AF}.

Annex C (normative): AKMA Ua* protocol based on DTLS

C.1 General

This Annex covers the aspects specific to the AKMA Ua* protocol based on DTLS. This feature is optional to be supported for the UE and AF. If the feature is supported, the following clauses apply.

C.1.1 Requirement on the UE

UE hosts the DTLS client. The UE should be able to send the AKMA PSK identity to the AF to indicate which key (K_{AF}) the UE intends to use to secure the Ua* reference point based on DTLS.

The PSK identity specified in B.1 for TLS is also applicable for DTLS.

C.1.2 Requirement on the AF

DTLS should be supported by the AF for the UE-AF reference point (Ua*).

The AF should be able to require that a certain key (i.e., K_{AF}) used to secure the Ua reference point based on DTLS.

C.2 Shared key-based mutual authentication between UE and AF

C.2.1 General

The TLS profile specified in TS 33.210 [13] clause 6.2 apply to DTLS 1.3[12].

C.2.2 Procedures for DTLS 1.3

The procedures given in B.1.3.2.2 for TLS 1.3 is also applicable for DTLS 1.3 [12].

AKMA PSK identity should be delivered via DTLS message.

Annex D (normative): Ua* security protocol: Object Security for Constrained RESTful Environments (OSCORE)

D.1 General

This annex describes how to secure access to an AF using Object Security for Constrained RESTful Environments (OSCORE) [14].

The specification of the OSCORE as an AKMA Ua* protocol follows the architecture of GBA OSCORE Ua protocol in TS 33.220 [4], Annex P with the AF taking the role of the NAF.

D.2 Requirements

D.2.1 General

This Annex covers the aspects specific to the AKMA Ua* protocol based on OSCORE. This feature is optional to be supported for the UE and AF. If the feature is supported, the following clauses apply.

D.2.2 Requirements on the UE

To utilise AKMA as described in this document the UE shall be equipped with an CoAP capable client implementing the particular features of AKMA as specified in this document.

D.2.3 Requirements on the AF

To utilise AKMA as described in this document the AF shall support the features of AKMA as specified in this document.

D.2.4 Requirements on the OSCORE

The same requirements outlined in TS 33.220 [4], clause P.2.4 apply in this clause.

D.3 IETF OSCORE as an AKMA Ua* protocol

D.3.1 General

The IETF OSCORE as an AKMA Ua* protocol is specified in this clause by providing the details about the procedures, the OSCORE security context and how it is related to the AKMA K_{AF} and the encoding of OSCORE messages using IETF CBOR specified in IETF RFC 8949 [15].

D.3.2 Procedures

The procedures for the AKMA OSCORE Ua* protocol are the same as the TS 33.220 [4], clause P.3.2 with the following changes.

1) In Step 1, the CoAP Client (UE) shall send a CoAP request to the AF. This is the Application Session Establishment Request in Step 1 in clause 6.2. The CoAP request shall consist of the following:

i) CoAP Method: POST.

ii) URI of the AKMA resource on the AF. The URI shall have the format of <AF_IP_or_FQDN>/akma, where AF_IP_or_FQDN indicates the IP address or the FQDN of the host that hosts the AF.

NOTE 1: It is assumed that the AF IP address or FQDN is already provisioned to the UE for AKMA purposes.

iii) Payload: CoAP Security protocol identifer, A-KID, N1, AF-SID, ?OSC-INP

The parameters "CoAP Security protocol identifier", N1, AF-SID, ?OSC-INP have the same semantics as the corresponding parameters in TS 33.220 [4], clause Y.2.3. Step 1.

2) Steps 2-4 follow clause 6.2 in the present document.

3) The CoAP Server (AF) shall respond to the CoAP Client (UE) with a CoAP response. This is the Application Session Establishment Response in Step 5 in clause 6.2. The response shall have the following content:

i) Response Code: "Created".

ii) Payload: N2, UE-SID.

The parameters N2, UE-SID have the same semantics as the corresponding parameters in TS 33.220 [4], clause P.2.3. Step 3.

D.3.3 OSCORE Security context

The OSCORE security context used in AKMA OSCORE Ua* protocol is similar to the GBA OSCORE security context specified in TS 33.220 [4], clause Y.3.3 with the following changes. The OSCORE security context for the OSCORE profile of Ua* shall have the following values:

- OMS = OSCORE Master Secret = HKDF(K_{AF}, "AKMA-OSCORE").
- Master Salt = Request Payload | Response Payload.
- UE Sender ID = UE-SID generated by CoAP Server and sent to the CoAP Client in the Application Session Establishment Response (Step 3 in clause D.3.2).
- AF Sender ID = AF-SID generated by CoAP Client and sent to the CoAP Server in the Application Session Establishment Request(Step 1 in clause D.3.2).

where HKDF shall be the HMAC-based Key Derivation Function specified in IETF RFC 5869 [16].

D.3.4 Refresh of OSCORE key material

OSCORE allows both the communication endpoints (UE or AF) to renegotiate the OSCORE security context after the OSCORE security context is established, according to Appendix B.2 in IETF RFC 8613 [14], which is shown in the figure D.3.4-1, Step 1.

Moreover even if K_{AF} remains constant upon a new application session establishment (Step 1 in clause D.3.2) or a renegotiation of the OSCORE key material, the nonces N1, N2, used in OSCORE security context shall be (stochastically) different from the previous OSCORE security context negotiation to ensure that the OSCORE security context is different.

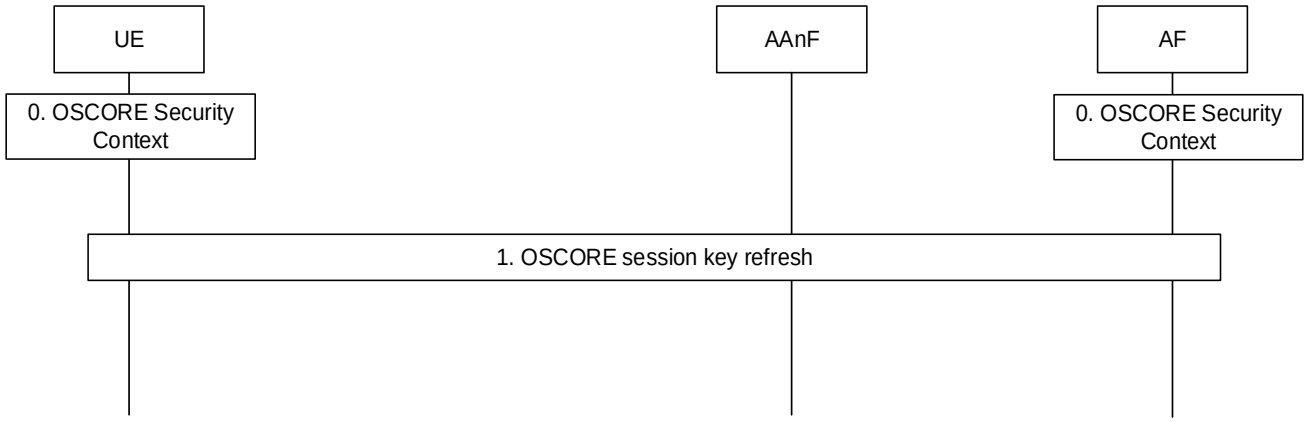

Figure D.3.4-1: OSCORE key refresh

D.3.5 OSCORE Ua* protocol payload encoding

IETF CoAP and OSCORE shall use the IETF Concise Binary Object Representation (CBOR) specified in the IETF RFC 8949 [15] for payload encoding for efficient information transfer between constrained IoT devices.

The CoAP media type for CBOR encoding shall be:

- Media Type: application/cbor
- CoAP Content-Format: 60

The Request Payload in the Application Session Request shall be formatted as a CBOR Array as follows:

```
Request Payload = [
A-KID : bstr,
    N1 : bstr,
    AF-SID : bstr,
    ? OSC-INP: bstr
]

A-KID = [
   RID : tstr,
   A-TID : bstr,
   HPLMN-ID : tstr
]

OSC-INP = {          ; CBOR Map
    ? 1 => int,   ; version
    ? 3 => int,   ; hkdf
    ? 4 => int,   ; alg
    ? 5 => bstr, ; salt
    ? 6 => bstr     ; contextId
}
```

The Response Payload in the Application Session Response shall be formatted as a CBOR Array as follows:

```
Reponse Payload = [        ; CBOR Array
    N2 : bstr,
    UE-SID : bstr
]
```

Annex E (informative): Change history

Change history							
Date	Meeting	TDoc	CR	Rev	Cat	Subject/Comment	New version
2020-06	SA#88-e	SP-200381				EditHelp review. Presented for information and approval	1.0.0
2020-07	SA#88-e					Upgrade to change control version	16.0.0
2020-09	SA#89-e	SP-200708	0001	-	D	Add Abbreviations to clause 3.3	16.1.0
2020-09	SA#89-e	SP-200708	0009	1	F	Clarifications on error response handling in AKMA process	16.1.0
2020-09	SA#89-e	SP-200708	0013	1	F	Re-authentication in AKMA	16.1.0
2020-09	SA#89-e	SP-200708	0020	-	F	Adding AKMA context description	16.1.0
2020-09	SA#89-e	SP-200708	0023	1	F	Corrections and clarifications to clause 4	16.1.0
2020-09	SA#89-e	SP-200708	0024	1	F	Corrections to AKMA key lifetimes	16.1.0
2020-09	SA#89-e	SP-200708	0025	1	F	Corrections and clarifications to AKMA procedures	16.1.0
2020-09	SA#89-e	SP-200708	0026	1	F	Assignment of FC values for key derivations	16.1.0
2020-09	SA#89-e	SP-200708	0027	-	F	Specification of value of SUPI for key derivations	16.1.0
2020-09	SA#89-e	SP-200708	0032	1	F	AKMA SBA interface clarifications	16.1.0
2020-09	SA#89-e	SP-200708	0034	1	F	Several clarifications and editorials	16.1.0
2020-12	SA#90e	SP-201006	0043	-	F	Lifetime of KAF expiration	16.2.0
2020-12	SA#90e	SP-201006	0045	-	F	Corrections of clause 6.1	16.2.0
2020-12	SA#90e	SP-201006	0046	-	F	Editorial modifications of AKMA	16.2.0
2020-12	SA#90e	SP-201006	0053	1	F	Update of the reference point interface names of AKMA	16.2.0
2020-12	SA#90e	SP-201006	0047	-	F	Adding details of AKMA application key generation in the UE	17.0.0
2021-03	SA#91e	SP-210118	0055	1	B	AAnF checks AKMA service for UE and AF in clause 6.3	17.1.0
2021-03	SA#91e	SP-210118	0056	1	B	Add AAnF selection function to AF	17.1.0
2021-03	SA#91e	SP-210118	0057	1	B	Add Application Key Get service in clause 7.1	17.1.0
2021-03	SA#91e	SP-210118	0060	1	F	KAF lifetime expiration in clause 5.2	17.1.0
2021-03	SA#91e	SP-210118	0062	1	F	Clarification on A-KID generation	17.1.0
2021-06	SA#92e	SP-210438	0066	2	B	Profiling the GBA TLS protocols for use with AKMA	17.2.0
2021-06	SA#92e	SP-210436	0072	1	F	AAnF AKMA context removal	17.2.0
2021-06	SA#92e	SP-210436	0075	1	D	Add an abbreviation to AKMA	17.2.0
2021-06	SA#92e	SP-210436	0076	1	F	Clarification on AAnF Selection	17.2.0
2021-06	SA#92e	SP-210436	0077	-	F	Editoral Change	17.2.0
2021-06	SA#92e	SP-210436	0079	1	F	AKMA Anchor Function selection clause	17.2.0
2021-06	SA#92e	SP-210436	0081	1	F	AKMA UE aspects	17.2.0
2021-06	SA#92e					Correcting implementation error for CR0076	17.2.1
2021-09	SA#93e	SP-210842	0088	-	F	Update clause 6.1 about Routing identifier	17.3.0
2021-09	SA#93e	SP-210841	0090	1	F	Add step 4 in annex B.1.2.2	17.3.0
2021-09	SA#93e	SP-210842	0093	1	F	Clarification on AAnF selection in clause 6.3	17.3.0
2021-12	SA#94e	SP-211374	0098	1	F	Corrections to the TLS with AKMA specification	17.4.0
2021-12	SA#94e	SP-211374	0099	1	B	Adding TLS 1.3 with AKMA keys	17.4.0
2021-12	SA#94e	SP-211373	0101	-	F	Clarification on Kaf lifetime in Clause 5.2	17.4.0
2021-12	SA#94e	SP-211374	0103	1	F	Delete the GBA_Digest in annex B.1.2.2	17.4.0
2021-12	SA#94e	SP-211373	0104	1	F	Clean up for clause 6.6.1	17.4.0
2021-12	SA#94e	SP-211373	0108	-	F	Sending UE ID to the AKMA AF	17.4.0
2022-03	SA#95e	SP-220207	0115	1	F	Add a Note about the Kaf refresh	17.5.0
2022-03	SA#95e	SP-220207	0116	-	F	Add function description about AAnF in 4.2.1	17.5.0
2022-03	SA#95e	SP-220207	0121	1	B	New AAnF application key get service without SUPI	17.5.0
2022-03	SA#95e	SP-220207	0122	1	B	Clarification on indication to UE when KAF is expired	17.5.0
2022-03	SA#95e	SP-220207	0123	-	D	Clean up for TS 33.535	17.5.0
2022-03	SA#95e	SP-220208	0124	1	F	Adding text on preferring AKMA keys to GBA Digest	17.5.0
2022-06	SA#95e	SP-220545	0125	-	F	Aligning text for AKMA procedure	17.6.0
2022-06	SA#95e	SP-220544	0126	1	F	Clarification on anonymization api	17.6.0
2022-06	SA#95e	SP-220545	0127	1	F	Correct AAnF service in clause 6.3	17.6.0
2022-06	SA#95e	SP-220545	0128	1	F	NF selects AAnF in clause 6.7	17.6.0
2022-06	SA#95e	SP-220545	0129	1	F	Clarification on the description about AAnF	17.6.0
2022-09	SA#97e	SP-220883	0132	1	F	Add ApplicationKey_ AnonUser_Get into table 7.1.1-1	17.7.0
2022-09	SA#97e	SP-220883	0137	-	F	A few clarifications to TS 33.535	17.7.0
2023-03	SA#99	SP-230147	0147	-	F	Clarification on NEF's authorization to AF	17.8.0
2023-03	SA#99	SP-230147	0148	1	F	AAnF sending GPSI to internal AKMA AF	17.8.0
2023-03	SA#99	SP-230147	0151	1	F	KAF lifetime and Ua protocol recommendations	17.8.0
2023-06	SA#100	SP-230602	0154		B	AKMA phase 2 security enhancement	18.0.0
2023-06	SA#100	SP-230605	0155	-	B	KAKMA re-keying relaed to HONTRA	18.0.0
2023-09	SA#101	SP-230881	0161	-	A	Correction of step numbers in clause 6.2 of TS 33.535	18.1.0
2023-09	SA#101	SP-230881	0163	-	A	Update the definition of AKMA context in TS 33.535	18.1.0
2023-09	SA#101	SP-230882	0164	-	B	Add AKMA Ua protocol based on DTLS to TS 33.535	18.1.0
2023-09	SA#101	SP-230896	0165	1	F	Link KAF refresh to KAKMA refresh	18.1.0
2023-09	SA#101	SP-230881	0168	1	A	Clarification on the description about AAnF	18.1.0
2023-09	SA#101	SP-230896	0170	1	F	Addition of AAnF functionality	18.1.0
2023-09	SA#101	SP-230896	0173	-	F	Update AKMA related UDM services	18.1.0
2023-09	SA#101	SP-230883	0175	-	B	IETF OSCORE as AKMA Ua protocol	18.1.0
2023-09	SA#101	SP-230884	0176	1	F	Clarification on limitation of session key based on Kaf using Ua	18.1.0

2023-12	SA#102	SP-231326	0181	1	A	Correction in UDM and GPSI related requirements	18.2.0
2023-12	SA#102	SP-231326	0185	1	A	Existing AKMA procedure alignment	18.2.0
2023-12	SA#102	SP-231326	0191	1	A	Editorial corrections to TS 33.535 in R18	18.2.0
2023-12	SA#102	SP-231332	0193	-	F	Update AKMA related UDM services	18.2.0
2023-12	SA#102	SP-231343	0195		F	HTTP RFC obsoleted by IETF RFC 9110	18.2.0
2024-03	SA#103	SP-240355	0201	1	F	KAF re-keying after expiration triggered by AAnF	18.3.0
2024-03	SA#103	SP-240355	0202	-	F	Adding UDM additional function to TS 33.535 in R18	18.3.0
2024-03	SA#103	SP-240371	0206	1	F	Update the reference to DTLS 1.3	18.3.0
2024-03	SA#103	SP-240347	0207	1	B	AKMA roaming policy control in AAnF	18.3.0
2024-07	SA#104	SP-240673	0210	1	F	AF disabling the encryption when roaming	18.4.0
2024-07	SA#104	SP-240672	0211	-	F	CR on editorial clear up	18.4.0
2024-07	SA#104	SP-240672	0212	-	F	CR to update AKMA related UDM services	18.4.0
2024-09	SA#105	SP-241102	0217	-	F	AKMA API Name correction along with other editorial corrections	18.5.0
2024-09	SA#105	SP-241102	0219	1	F	Update to AKMA service disabling	18.5.0
2024-09	SA#105	SP-241102	0220	-	F	Editorial correction for AKMA procedures	18.5.0

History

Document history		
V18.3.0	May 2024	Publication
V18.4.0	July 2024	Publication
V18.5.0	October 2024	Publication