|  |  |
| --- | --- |
| 3GPP TR 33.737 V0.5.0 (2023-01) | |
| Technical Report | |
| 3rd Generation Partnership Project;  Technical Specification Group Services and System Aspects;  Study on Authentication and Key Management for Applications (AKMA) phase 2;  (Release 18) | |
|  | |
| *5G-logo_175px* | 3GPP-logo_web |
|  | |
| The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP. The present document has not been subject to any approval process by the 3GPPOrganizational Partners and shall not be implemented. This Specification is provided for future development work within 3GPPonly. The Organizational Partners accept no liability for any use of this Specification. Specifications and Reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices. | |

|  |
| --- |
|  |
| ***3GPP***  Postal address  3GPP support office address  650 Route des Lucioles - Sophia Antipolis  Valbonne - FRANCE  Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16  Internet  http://www.3gpp.org |
| ***Copyright Notification***  No part may be reproduced except as authorized by written permission. The copyright and the foregoing restriction extend to reproduction in all media.  © 2022, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).  All rights reserved.  UMTS™ is a Trade Mark of ETSI registered for the benefit of its members  3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners LTE™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners  GSM® and the GSM logo are registered and owned by the GSM Association |

Contents

Foreword 5

Introduction 6

1 Scope 7

2 References 7

3 Definitions of terms, symbols and abbreviations 7

3.1 Terms 7

3.2 Symbols 7

3.3 Abbreviations 8

4 Architectural assumptions 8

4.1 General 8

4.2 AKMA Non-Roaming network model 8

4.3 AKMA Roaming network model 8

5 Key issues 9

5.1 Key Issue #1: Support for AKMA roaming scenario 9

5.1.1 Issue details 9

5.1.2 Security Threats 10

5.1.3 Potential security requirements 10

5.2 Key Issue #2: Introducing the Authentication proxy into AKMA 10

5.2.1 Key issue details 10

5.2.2 Security threats 10

5.2.3 Potential architectural and security requirements 10

6 Solutions 11

6.1 Solution #1: AKMA roaming solution for Ua\* encryption key 11

6.1.1 Introduction 11

6.1.2 Solution details 12

6.1.2.1 Internal AF in HPLMN 12

6.1.2.2 External AF 13

6.1.2.3 Internal AF in VPLMN 13

6.1.3 Evaluation 14

6.2 Solution #2: New solution for AKMA roaming when both UE and AF are in VPLMN 15

6.2.1 Introduction 15

6.2.2 Solution details 15

6.2.2.1 Architecture 15

6.2.2.2 Solution detail 15

6.2.2.2.1 AKMA Application Key request via NEF 15

6.2.3 Evaluation 15

6.3 Solution #3: Roaming AKMA architecture of the AF in the HPLMN 16

6.3.1 Introduction 16

6.3.2 Solution details 16

6.3.2.1 Roaming AKMA architecture of the AF in the HPLMN 16

6.3.2.2 Roaming AKMA procedure of the AF in the HPLMN 17

6.3.2.3 New service: Nudm\_Get\_Roaming\_NFid service operation 18

6.3.3 Evaluation 18

6.4 Solution #4: Roaming AKMA architecture of the AF in the VPLMN 18

6.4.1 Introduction 18

6.4.2 Solution details 18

6.4.3 Evaluation 19

6.5 Solution #5: AKMA anchor key registration to the AAnF in VPLMN after primary authentication 19

6.5.1 Introduction 19

6.5.2 Solution details 20

6.5.2.1 AKMA anchor key registration in roaming scenario 20

6.5.2.2 UE in VPLMN accessing internal VPLMN AF 21

6.5.2.3 UE in VPLMN accessing internal HPLMN AF 22

6.5.3 Evaluation 22

6.6 Solution #6: AKMA roaming with VAAnF for LI 23

6.6.1 Introduction 23

6.6.2 Solution details 23

6.6.3 Evaluation 25

6.7 Solution #7: Introducing AP into AKMA 25

6.7.1 Introduction 25

6.7.2 Solution details 25

6.7.2.1 Architecture of using AP 25

6.7.2.2 AP-AS reference point 26

6.7.2.3 Example of using AP for TLS tunnels 26

6.7.3 Evaluation 27

6.8 Solution#8: AAnF discovery and selection for internal AF in AKMA roaming 28

6.8.1 Introduction 28

6.8.2 Solution details 28

6.8.2.1 AAnF discovery and selection for internal AF 28

6.8.3 Evaluation 29

6.9 Solution #9: Roaming AKMA architecture of the AF in Data Network (Internet) 29

6.9.1 Introduction 29

6.9.2 Solution details 29

6.9.2.1 Roaming AKMA architecture of the AF in Data Network (Internet) 29

6.9.2.2 Roaming AKMA procedure of the AF in Data Network (Internet) 30

6.9.2.3 New service: Nudm\_Get\_Roaming\_NFid service operation 30

6.9.3 Evaluation 31

6.10 Solution #10: Support of AKMA roaming with K\_SEAF 31

6.10.1 Introduction 31

6.10.2 Solution details 31

6.10.3 Evaluation 32

6.11 Solution #11: AKMA Authentication in roaming scenario 33

6.11.1 Introduction 33

6.11.2 Solution details 33

6.11.2.1 Option#1 details 33

6.11.2.2 Option#2 details 34

6.11.3 Evaluation 35

6.12 Solution #12: AKMA anchor key forwarding to the VPLMN during primary authentication procedure 35

6.12.1 Introduction 35

6.12.2 Solution details 35

6.12.2.1 AKMA anchor key registration in roaming scenario 35

6.12.2.2 UE in VPLMN accessing internal VPLMN AF 36

6.12.2.3 UE in VPLMN accessing internal HPLMN AF 37

6.12.3 Evaluation 37

6.13 Solution #13: AKMA support in roaming 38

6.13.1 Introduction 38

6.13.2 Solution details 39

6.13.3 Evaluation 40

6.14 Solution #14: AKMA roaming with AF outside VPLMN 40

6.14.1 Introduction 40

6.14.2 Solution details 40

6.14.3 Evaluation 42

6.15 Solution #15: AKMA roaming for external AF in Data Network 42

6.15.1 Introduction 42

6.15.2 Solution details 42

6.15.3 Solution Evaluation 43

6.16 Solution #16: AKMA roaming with VPLMN AKMA Support NF for inbound roamers 43

6.16.1 Introduction 43

6.16.2 Solution details 43

6.16.2 Evaluation 45

7 Conclusions 46

7.1 Conclusion to Key Issue#1 46

7.2 Conclusion to Key Issue#2 46

Annex A (informative): Change history 47

# Foreword

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

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

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

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

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

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

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

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

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

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

**should** indicates a recommendation to do something

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

**may** indicates permission to do something

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

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

**can** indicates that something is possible

**cannot** indicates that something is impossible

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

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

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

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

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

In addition:

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

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

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

# Introduction

Editor’s Note: This clause contains some background information for the study.

# 1 Scope

The present document studies key issues and potential solutions to support roaming aspects and the Authentication Proxy in AKMA, which is specified in TS 33.535[2]. Specifically, the present document:

- Investigates AKMA roaming architecture and requirements by taking regulatory compliance into account;

- Studies the architecture impact and procedures of introducing the Authentication Proxy (similar as the AP specified in GBA) into AKMA.

- Investigate the use cases and need for KAF refresh procedures and afterwards investigate the architecture impacts and procedures of KAF refresh in AKMA.

# 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 TR 33.535: " Authentication and key management for applications based on 3GPP credential in 5G （AKMA）".

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

[4] 3GPP TS 33.501: "Security architecture and procedures for 5G system".

[5] 3GPP TS 33.210: "Network Domain Security (NDS); IP network layer security ".

[6] 3GPP TS 23.501: "System architecture for the 5G System (5GS)".

[7] 3GPP TS 23.501: "5G System; Unified Data Management Services; Stage 3".

# 3 Definitions of terms, symbols and abbreviations

## 3.1 Terms

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

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

## 3.2 Symbols

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

<symbol> <Explanation>

## 3.3 Abbreviations

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

<ABBREVIATION> <Expansion>

# 4 Architectural assumptions

## 4.1 General

The present document is based on the AKMA architecture and procedures specified in TS 33.535 [2]. This clause captures the architectural assumptions for the non-roaming and roaming models. These architectural models do not include the AKMA functions which enable LI in the VPLMN as these functions are the objects of the study.

## 4.2 AKMA Non-Roaming network model

The following figure shows the fundamental non-roaming network model of AKMA, as well as the interfaces between them. Details of the AKMA requirements and procedures are documented in TS 33.535 [2].



Figure 4.2-1: Fundamental Non-Roaming Network Model for AKMA

## 4.3 AKMA Roaming network model

When the UE is roaming there are a few cases to consider for the AF, whether the AF is internal to the HPLMN, internal to VPLMN or external in a DN e.g. hosted on the Internet.

Figure 4.2-1 shows the fundamental roaming network model for the case that the UE is roaming in the VPLMN and accesses an AF internal to the HPLMN. This network model is a simplified version of the Roaming 5G System architecture - home routed scenario from TS 23.501 [6] since the UE accesses an AF internal to the HPLMN. It is therefore assumed that the UE is configured to setup a home routed PDU session for the Ua\* protocol between itself and the AF. The HPLMN AF accesses the AAnF following the internal AF model from TS 33.535 [2].



Figure 4.3-1: Fundamental Roaming Network Model for AKMA when the AF is internal to the HPLMN

Figure 4.2-2 shows the fundamental roaming network model for the case that the UE is roaming in the VPLMN and accesses an AF internal to the VPLMN or external in the DN. This network model is a combination and simplification of the Roaming 5G System architecture - local breakout and home-routed scenarios from TS 23.501 [6]. In the case that the UE accesses an internal AF to the VPLMN a local-breakout scenario is used between the UE and AF. In the case that a UE accesses an external AF which is in the DN, it does not matter if the UE uses the local breakout or the home-routed scenario. A VPLMN AF and an external AF accesses the AAnF via the HPLMN NEF following the external AF model from TS 33.535 [2]. In other words, the N32 interface is not used in the case of the AF internal to the VPLMN.

 **Figure 4.3-2: Fundamental Roaming Network Model for AKMA when the AF is internal to the VPLMN or AF is external**

# 5 Key issues

## 5.1 Key Issue #1: Support for AKMA roaming scenario

### 5.1.1 Issue details

AKMA roaming scenarios depend on UE and AF locations. There are different scenarios for AKMA roaming that need to be addressed:

Case 1: UE is in VPLMN and accessing an internal HPLMN AF

Case 2: UE is in VPLMN and accessing an internal VPLMN AF

Case 3: UE is in VPLMN and accessing an external AF in the Data Network (Internet)

The AKMA roaming solutions should comply with LI requirements. It’s required either decrypted traffic or the means (e.g. providing keys) for VPLMN or law enforcement to decrypt the traffic should be provided to VPLMN. The LI requirements for access to keys are only for encryption, and in the AKMA case applies when the Ua\* protocol is encrypted. Regarding the means (e.g. providing keys) for VPLMN or law enforcement to decrypt the traffic, for Case 1, the encryption key (not necessarily the KAF unless KAF is used directly as the encryption key) and related information to decrypt the user traffic need to be provided from the AF to the VPLMN. For Case 3, only the keys and corresponding information which are known to HPLMN and used to establish the encryption key need be provided to VPLMN.

### 5.1.2 Security Threats

N/A

### 5.1.3 Potential security requirements

The AKMA architecture shall support the above mentioned AKMA roaming cases.

AKMA service shall be made to comply with the LI requirements.

## 5.2 Key Issue #2: Introducing the Authentication proxy into AKMA

### 5.2.1 Key issue details

TS 33.222 specifies the use of Authentication Proxy in GBA [2], where an Authentication Proxy (AP) is a proxy resides between the UE and ASs. It helps to reduce the consumption of authentication vectors and/or to minimize SQN synchronization failures, and relieves the AS of security tasks. Similarly, introducing such an authentication proxy in AKMA is beneficial where different application servers (or Application Functions in AKMA) reside in the same trust domain or in the same edge node. With the AP, these application servers can rely on the AP to execute AKMA procedures, which is more cost efficient than the case where each application servers execute AKMA procedures separately. AKMA is a potential solution in MEC, and it is possible that different application servers reside in the same edge cloud or belong to the same service vendor, it is beneficial to consider the feasibility of introducing a similar proxy in AKMA.

### 5.2.2 Security threats

Different Application servers reside in the same trust domain may execute AKMA procedures separately, leading to consumptions of AAnF generating AKMA keys and signalling resources.

### 5.2.3 Potential architectural and security requirements

The AKMA architecture may support an authentication proxy to perform the AF functionality residing between UE and AS(s), in order to save the consumption of signalling resources and AAnF computing resources.

The connection between AP and AS(s) should be secured.

# 6 Solutions

## 6.1 Solution #1: AKMA roaming solution for Ua\* encryption key

### 6.1.1 Introduction

AKMA roaming scenarios depend on UE and AF locations. Therefore, there are different scenarios for AKMA roaming that need to be addressed:

Case 1: UE is in VPLMN and accessing an internal HPLMN AF.

Case 2: UE is in VPLMN and accessing an VPLMN AF (same VPLMN).

Case 3: UE is in VPLMN and accessing an external AF in the Data Network (Internet).

The LI requirements for access to keys are only for the encryption, and in the AKMA case applies when the UA\* protocol is encrypted. The encryption keys could be derived from KAF (Example TLS profile defined in 33.535) or could be KAF itself. The encryption key could also be a completely different key, derived according to an application layer protocol, not defined by 3GPP. So LI should be compliant by an internal AF in all cases.

Therefore, 3GPP should facilitate internal AF so that internal AF should be able to determine the UE roaming status and, accordingly, the AF shall:

* provide the encryption key to VPLMN and/or
* stop the encryption and still provide the service to UE and may continue with only integrity protection and/or
* stop providing service to UE

For external AF, 5GC cannot force external AF to provide the keys to VPLMN. Therefore, AAnF can provide whatever accurate information AAnF has to VPLMN. i.e., KAF and indication providing information either the KAF is used for encryption or KAF derived keys are used for encryption.

### 6.1.2 Solution details

#### 6.1.2.1 Internal AF in HPLMN



Figure 6.1.2.1: Roaming solution for Ua\* encryption key

Step 1-3 is the same as defined in TS 33.501[4] clause 6.1.3 and TS 33.535[2] Clause 6.1.

4. The AUSF shall send the Naanf\_AKMA\_AnchorKey\_Register Request with SUPI, A-KID, KAKMA, and Registered SN ID to the home AAnF. The AKMA context is updated with this received Registered SN ID.

Note: AUSF will get the SN-name from the request of 'Nausf\_UEAuthentication\_Authenticate' from AMF.

5a. The UE requests an application session establishment request with A-KID towards the AF. The AF shall discover the hAAnF and shall send the Naanf\_AKMA\_ApplicationKey\_Get request with A-KID and AF\_ID to the hAAnF .

Note: If AF is not in the operator domain, then AF requests the AAnF via NEF. If AF is the HPLMN, then the AF request directly land at HPLMN AAnF.

5b. The home PLMN AAnF shall respond with Naanf\_AKMA\_ApplicationKey\_Get response containing Registered SN ID.

6. As the Registered SN ID is different from the PLMN in which the AF is located/connected, the AF shall decide:

- supporting pushing encryption keys to VPLMN#1, therefore AF shall discover the VPLMN#1 AAnF or a new NF, (enable the LI) via Registered SN-ID and shall provide the encryption keys material to VPLMN#1. VPLMN#1 AAnF shall store the encryption keys related to UE which may be used by LI if required (step 6a1 and step 6a2). If AKMA is not supported in the VPLMN, then AAnF discovery in the VPLMN will fail or AF cannot push encryption keys to VPLMN.

NOTE: If VPLMN support AKMA, the vAAnF stores key materials. If VPLMN does not support AKMA, then a new NF stores the key material. The VPLMN AKMA capabilities and policies may be configured in the hAAnF and may be based on SLA. Based on the AKMA support in the VPLMN policies or SLAs, the hAAnF selects either the new NF storing the LI context in the VPLMN or the vAAnF.

- shall not enable the encryption and may continue with the UE session with only integrity protection (step 6b).

- shall not provide the service to UE attached to other PLMN and shall reject the request (step 6c).

NOTE: The LI interception point could be any NF in VPLMN (e.g. UPF) that can retrieve the keys from the AAnF in the same VPLMN to perform encryptions.

NOTE: vAAnF supports a new service or an existing service will be enhanced to allow hAAnF or AF to register the key materials.

#### 6.1.2.2 External AF

For external AF, 5GC cannot force external AF to provide the keys to VPLMN to fulfill the LI requirement. However, when the external AF retrieves the keys from AAnF via NEF, the external AF shall provide the indication that 'KAF is used for encryption' or 'KAF-derived keys are used for encryption' or 'keys independent of KAF are used for encryption'. When the indication value is 'Kaf is used for encryption' or 'KAF-derived keys are used for encryption', the hAAnF provides KAF along with the same indication to VPLMN (new AF or vAAnF).

If the indication value received at hAAnF is 'keys independent of KAF are used for encryption', the hAAnF shall provide only the indication to VPLMN without any keys.

NOTE: Depending on agreement between HPLMN and AF, indications could be extended to provide more complete information on security parameters used by AF.



NOTE: vAAnF supports a new service or an existing service will be enhanced to allow hAAnF or AF to register the key materials

#### 6.1.2.3 Internal AF in VPLMN

Solution is similar to 6.1.2.1 where interna AF in VPLMN can get the keys from hAAnF. The home PLMN AAnF shall respond with Naanf\_AKMA\_ApplicationKey\_Get response containing Registered SN ID which is the same PLMN where AF is located. Based on the SN ID, the AF shall decide:

- supporting pushing encryption keys to vAAnF or new NF (enable for LI)

- shall not enable the encryption and may continue with the UE session with only integrity protection.

- shall not provide the service to UE attached to other PLMN and shall reject the request.

### 6.1.3 Evaluation

The solution fulfils the requirements and all 3 use-cases defined in key issue#1.

|  |  |
| --- | --- |
| Use case | solution |
| Case 1: UE is in VPLMN and accessing an internal HPLMN AF. | Internal AF in HPLMN get the SN name from the hAAnF and   * pushes the key material (LI context) to vAAnF/new NF in VPLMN or * stop the encryption or * stop the UE service (don't provide the service) |
| Case 2: UE is in VPLMN and accessing an VPLMN AF (same VPLMN). | Internal AF in VPLMN get the SN name from the hAAnF and   * pushes the key material (LI context) to vAAnF/new NF in VPLMN or * stop the encryption or * stop the UE service (don't provide the service) |
| Case 3: UE is in VPLMN and accessing an external AF in the Data Network (Internet). | External AF provides an indication of encryption to hAAnF. Based on this indication, the hAAnF decides to send the Key material (what is known to hAAnF) to vAAnF/new NF in VPLMN.  If the indication value received at hAAnF is 'keys independent of KAF are used for encryption', the hAAnF shall provide only the indication to VPLMN without any keys. |

This solution brings following changes on the node/NFs:

|  |  |
| --- | --- |
| Node/NF/UE | Impact |
| UE impact | NO |
| UDM | NO |
| AAnF | hAAnF receives SN name from AUSF and provide the same SN name to AFs and pushes key material to VPLMN  vAAnF support new API to receive the key material from the AF and hAAnF. There will be a dedicated vAAnF for storage of the keys from other HPLMNs. HPLMN discovers the vAAnF based on the NRF discovery. |
| New NF | Register operation: If AKMA is not supported in the VPLMN, then new NF can be defined to receive the key material from the HPLMN for LI. There will be a dedicated new NF for storage of the keys from other HPLMNs. HPLMN discovers the new NF based on the NRF discovery. |
| AUSF | Provide SN name to hAAnF along with KAKMA after the primary authentication is completed. |
| Internal AF | get SN name from the hAAnF  pushes the key material (LI context) to VPLMN or stop the encryption or stop the UE service |
| External AF | Provide indication of encryption to hAAnF  Get SN name from the hAAnF |

## 6.2 Solution #2: New solution for AKMA roaming when both UE and AF are in VPLMN

### 6.2.1 Introduction

This solution addresses the requirement in KI#1 about the AKMA roaming. This solution is designed to case2 and case3.

This solution provides a method that when both UE and the AF in the same visited network. When the AF is inside 5GC, it can communicate with the UE’s AAnF in HPLMN directly. Otherwise, when AF is outside 5GC, it first connects to the NEF in HPLMN, and then the NEF will receive and send the message for the UE and the AAnF.

### 6.2.2 Solution details

#### 6.2.2.1 Architecture

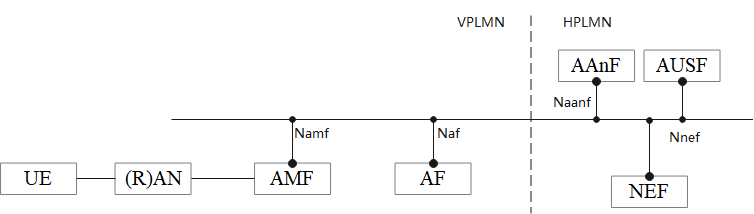


Figure 6.2.2.1-1: Reference Model when UE and the AF are in the VPLMN

The Figure 6.2.2.1-1 shows an architecture model when UE and the AF are in the VPLMN.

The AF can be an internal AF in VPLMN or an AF in the data network.

#### 6.2.2.2 Solution detail

##### 6.2.2.2.1 AKMA Application Key request via NEF

As described in clause 6.3 in TS 33.535[2] applies with additional descriptions:

- replace the NEF with the NEF in home network.

- replace the AF with the AF in visited network.

- replace the UE with the UE in visited network.

NOTE: the visited AF is an external AF to UE’s HPLMN

NOTE: for an AF in the data network the AF may not be allowed to use SBA to access internal VPLMN NFs.

### 6.2.3 Evaluation

This solution addresses the requirement of KI#1 when the AF is in the visited network, in particular, it addresses case 2 and case3.

This solution reuses the existing mechanisms and hence has no standard impact.

## 6.3 Solution #3: Roaming AKMA architecture of the AF in the HPLMN

### 6.3.1 Introduction

This solution addresses KI#1(case 1). This solution is proposed for the cases of function that enables LI deployment when AF located in the HPLMN.

### 6.3.2 Solution details

#### 6.3.2.1 Roaming AKMA architecture of the AF in the HPLMN

Figure 6.3.2.1-1 shows the SBA architecture that UE is roaming in a VPLMN and accesses the application deployed in the HPLMN.



Figure 6.3.2.1-1: The SBA architecture that UE is roaming in a VPLMN and accesses the application deployed in the HPLMN

Figure 6.3.2.1-2 shows the architecture that UE is roaming in a VPLMN and accesses the application deployed in the HPLMN. The UE accesses the home network deployment application. Like the third party deployment service, if the data between the UE and application had not been encrypted, the VPLMN can detect the unencrypted data flow through UPF in the VPLMN. If the data between the UE and application have been encrypted by using an encrypted Ua\* protocol, the AMF in the VPLMN has to obtain related decryption keys from the AF in the HPLMN based on the agreements between the HPLMN and the VPLMN. In this case, the function that enables LI could be AMF in the VPLMN.



Figure 6.3.2.1-2: The architecture that UE is roaming in a VPLMN and accesses the application deployed in the HPLMN (\*function that enables LI)

#### 6.3.2.2 Roaming AKMA procedure of the AF in the HPLMN

Figure 6.3.2.2-1 shows the procedure that the VPLMN NF obtains the encrypt key from AF in HPLMN. Because only the AF knows the encryption/decryption key used between the UE and the AF, the AF can send the corresponding encryption/decryption key to the VPLMN. The AF obtains the VPLMN NF (e.g. AMF) id that can store the key from the UDM base on the UE ID. An indication is needed to indicate whether the key sent by the AF is KAF or the encryption key derived from KAF.



Figure 6.3.2.2-1: VPLMN NF obtains the encrypt key from AF in the HPLMN

Step 1-4, and 9 are same as the step 1-4, and 5 in clause 6.2.1 in TS 33.535[2]. The differences to the normal procedures of TS 33.535[2] are described below.

Step 5, the AF sends the Nudm\_Get\_Roaming\_NFid Request message to the UDM with the UE ID. The UE ID can be the SUPI of the UE or another user identifier, e.g. GPSI.

Step 6, the UDM sends the Nudm\_Get\_Roaming\_NFid Response message to the AF with the VPLMN NF id. The VPLMN NF can be the AMF which the UE is currently attached to in VPLMN network.

Step 7, based on the agreements between HPLMN and VPLMN, the AF sends the Push application Key Request message to the VPLMN NF, where the message carries the KAF, or the encryption key derived from the KAF that is used to encrypt the data flow between the UE and the AF, an indication and the UE ID. The indication indicates that the carried key is the KAF or the encryption key, and may further indicate whether the carried key is used as the data flow encryption/decryption key between the UE and the AF or not. The UE ID can be the SUPI of the UE or another user identifier, e.g. GPSI.

Step 8, if the VPLMN NF supports AKMA roaming, the VPLMN NF stores the key and the indication, and sends the Push application Key Response message to the AF. If not, the VPLMN NF ignores the Push application Key Request message.

#### 6.3.2.3 New service: Nudm\_Get\_Roaming\_NFid service operation

**Service operation name:** Nudm\_Get\_Roaming\_NFid.

**Description:** The NF consumer requests VPLMN NF id from the UDM.

**Input, Required:** SUPI or GPSI.

**Input, Optional:** None.

**Output, Required:** VPLMN NF id.

**Output, Optional:** None.

### 6.3.3 Evaluation

This solution addresses key issue#1 (case 1) and fulfils the requirements in case that the AF is located in the HPLMN.

The solution proposes to obtain related decryption keys from the AF in the HPLMN based on the agreements between the HPLMN and the VPLMN when the data between the UE and application have been encrypted.

The solution proposes that the AF obtains the VPLMN NF id that can store the keys from the UDM base on the UE ID.

The solution proposes that the AF pushes the encryption key to the VPLMN NF.

The solution needs one VPLMN NF (e.g. AMF) to store the key.

## 6.4 Solution #4: Roaming AKMA architecture of the AF in the VPLMN

### 6.4.1 Introduction

This solution addresses KI#1(case 2). This solution is proposed for the cases of regulatory control point deployment when AF located in the VPLMN.

### 6.4.2 Solution details

Figure 6.4.1-1 shows the SBA architecture that UE is roaming in a VPLMN and accesses the application deployed in the VPLMN. The home network (HPLMN) needs an agreement with the service network (VPLMN) to use the AKMA service deployed in the home network. In this scenario, the AF in the service network can sense the AKMA service used by the UE. For this scenario, AF is located inside the VPLMN and needs to find the AAnF of the home network through the NEF of the home network.



Figure 6.4.2-1: The SBA architecture that UE is roaming in a VPLMN and accesses the application deployed in the VPLMN

Figure 6.4.1-2 shows the architecture that UE is roaming in a VPLMN and accesses the application deployed in the VPLMN. As described in clause 6.3 of the 3GPP TS 33.535[2], the AF needs to obtain the KAF from the AAnF through the NEF. Therefore, the function that enables LI in Figure 6.4.1-2 can be AF in the VPLMN and the AF knows the encryption key between the UE and the AF.



Figure 6.4.2-2: The architecture that UE is roaming in a VPLMN and accesses the application deployed in the VPLMN (\*function that enables LI)

### 6.4.3 Evaluation

This solution addresses key issue#1 (case 2) and fulfils the requirements in case that the AF is located in the VPLMN.

## 6.5 Solution #5: AKMA anchor key registration to the AAnF in VPLMN after primary authentication

### 6.5.1 Introduction

This solution addresses the KI #1. The proposed solution supports registration of AKMA anchor key (KAKMA) and A-KID to the AAnF in VPLMN after primary authentication for UE in the same manner of KAKMA and A-KID registration to the AAnF in HPLMN. Once the KAKMA and the A-KID are registered in the AAnFs in VPLMN and HPLMN, the UE in VPLMN is able to access to both VPLMN AF and HPLMN AF. Moreover, even if KAUSF which is a root key of the KAKMA is changed by new primary authentication, there is no need additional key update procedure for the VPLMN AAnF since the new KAKMA and A-KID will replace the old keys whenever primary authentication is performed.

### 6.5.2 Solution details

#### 6.5.2.1 AKMA anchor key registration in roaming scenario



Figure 6.5.2.1-1: AKMA anchor key registration to the AAnF in VPLMN after primary authentication

1. The AUSF requests authentication information to the UDM to acquire subscription information and authentication method in primary authentication procedure.

2. The UDM responses with the Authentication Vector. The AKMA indication and the RID may be included in the response if the UE needs AKMA anchor key generation.

3. If the AUSF receives the AKMA indication from the UDM, the AUSF shall store SN-name, KAUSF and generate AKMA Anchor Key (KAKMA) and the A-KID. The UE shall generate the KAKMA and the A-KID if the primary authentication procedure is completed.

4. If the keys generation is completed, the AUSF shall request to the HPLMN AAnF to register A-KID and KAKMA. The selection of the AAnF is described in TS 33.535 clause 6.7 [2].

5. The AAnF in HPLMN responses to the AUSF after key registration completed.

6. If the AUSF recognizes the UE is roaming based on SN-name which is received previously in the primary authentication procedure, the AUSF shall request to the VPLMN AAnF to register A-KID and KAKMA.

To select the AAnF in VPLMN to register KAKMA and A-KID, local configuration or NRF can be utilized. When NRF is used to discover and select the AAnF in VPLMN, serving PLMN ID shall be used in the discovery and selection by NRF in different PLMNs.

7. The AAnF in VPLMN responses to the AUSF after key registration completed.

#### 6.5.2.2 UE in VPLMN accessing internal VPLMN AF



Figure 6.5.2.2-1: Application session establishment between roaming UE and AF in VPLMN

0. The UE is roaming in VPLMN and AKMA anchor key is registred in the AAnF in VPLMN after the procedure in clause 6.5.2.1.

1. When the UE initiates communication with the AF in VPLMN, it shall include the derived A-KID in the Application Session Establishment Request message.

2. The AF in VPLMN identifies based on the realm part from the received A-KID whether the UE is from other serving network. If the AF decides to support the UE from other serving network, the AF requests application key to the AAnF in VPLMN.

To selects the AAnF in VPLMN for the AF to provide the KAF, local configuration or NRF can be utilized. When NRF is used to discover and select the AAnF in VPLMN, both RID and home network identifier from the received A-KID are used to select the AAnF.

3. The AAnF in VPLMN derives the KAF from KAKMA if it does not already have KAF.

4. The AAnF in VPLMN sends Naanf\_AKMA\_ApplicationKey\_Get response to the AF in VPLMN with SUPI, KAF and the KAF expiration time.

5. The AF in VPLMN sends the Application Session Establishment Response to the UE.

NOTE: NFs in VPLMN could provide LI context (A-KID, KAKMA, KAF, etc.) when LI enabled.

#### 6.5.2.3 UE in VPLMN accessing internal HPLMN AF



Figure 6.5.2.3-1: Application session establishment between roaming UE and AF in HPLMN

0. The UE is roaming in VPLMN and AKMA anchor key is registered in the AAnF in HPLMN after the procedure in clause 6.5.2.1.

1. When the UE initiates communication with the AF in HPLMN, it shall include the derived A-KID in the Application Session Establishment Request message.

2. The AF in HPLMN requests application key to the AAnF in HPLMN.

3. The AAnF in HPLMN derives the KAF from KAKMA if it does not already have KAF.

4. The AAnF in HPLMN identifies based on the realm part from the received A-KID whether the UE is from other serving network. If the AAnF in HPLMN decides to support the UE from other serving network, the AAnF in HPLMN provides AKMA context to the AAnF in VPLMN to support LI.

To selects the AAnF in VPLMN for the AF to provide the KAF, local configuration or NRF can be utilized. When NRF is used to discover and select the AAnF in VPLMN, both RID and home network identifier from the received A-KID are used to select the AAnF.

5. The AAnF in VPLMN stores the delivered AKMA related information from HPLMN to support LI.

6. The AAnF in VPLMN responses to the AAnF in HPLMN.

7. The AAnF in HPLMN sends Naanf\_AKMA\_ApplicationKey\_Get response to the AF in VPLMN with SUPI, KAF and the KAF expiration time.

8. The AF in HPLMN sends the Application Session Establishment Response to the UE.

NOTE: NFs in VPLMN could provide LI context (A-KID, KAKMA, KAF, etc.) when LI enabled.

### 6.5.3 Evaluation

This solution addresses case 1 of key issue#1 and fulfils the requirements in the case that the AF is located in the HPLMN.

The solution fulfils LI requirement that LI contexts (e.g. encrypted traffic data and keys related to AKMA) are ready before the application session established.

KAKMA in HPLMN is shared to VPLMN, and same KAKMA will be AKMA anchor key in HPLMN and VPLMN.

Editor’s Note: When same KAKMA is shared to VPLMN and HPLMN, how to make different the KAF in VPLMN and HPLMN is FFS.

Editor’s Note: How the AUSF discovers the AAnF in the VPLMN is FFS.

Editor’s Note: How the solution is aligned with clause 4 is FFS.

## 6.6 Solution #6: AKMA roaming with VAAnF for LI

### 6.6.1 Introduction

This solution is addressing Key Issue #1: Support for AKMA roaming scenario, especially the scanrio when the UE is in VPLMN and trying to access the HPLMN AF.

The issue of LI for AKMA roaming was described as follows: if the UE is roaming in a VPLMN, then the UE builds up a secure tunnel to an AF in the HPLMN and since the credentials used for the encryption are based on the 3GPP derived keys, the VPLMN must be able to perform LI. This is not possible compared to GBA, where the NAF and tunnel endpoint is located in the VPLMN.

Further it cannot be implied that the AF is always in the VPLMN for roaming scenarios, for typical deployments it can be a 3rd party AF in a data network.

If the VPLMN needs to perform LI, then the VPLMN needs to be enhanced to store the SUPI and the encryption key e.g. with a local AAnF. It is recommended to further also only provide the KAF to the VPLMN for the service the UE is currently requesting from the AF. In case the VPLMN is not enhanced but has a strong LI requirement for AKMA, the AF should not get the KAF and should get an indication that NULL encryption has to be used.

### 6.6.2 Solution details

The solution proposes to introduce a VAAnF in the VPLMN in order to store the connection details of the UE roaming in that VPLMN to the AF outside that VPLMN. The differences to the normal procedures of TS 33.535 are described below.



Figure 6.6.2-1: Deriving KAKMA after primary authentication

Additionally to the other AKMA related parameters, the AUSF provides also the SN name to the AAnF in the HPLMN in step 4. The SN name is later used to determine whether the Ue is roaming and to select an appropriate AAnF in the VPLMN (VAAnF) for storing the AKMA connection details.

**Figure 6.6.2-2: KAF generation from KAKMA and provisioning to VPLMN**

In step 3, the HAAnF derives the AKMA Application Key (KAF) from KAKMA if it does not already have KAF.

In step 4, the HAAnF provides the KAF and the KAF expiration time together with the SUPI of the UE to AF according to the AKMA procedure.

In step 5, the AF sends an Application Session Establishment Response to the UE according to the AKMA procedure.

In step 6, the UE and the AF may perform an additional key derivation from KAF in order to generate a session key KSession which is used to protect the application session between UE and AF. The key derivation is depending on the protocol used on the Ua\* interface between UE and AF.

In step 7, after the session establishment, the AF provides the KSession key to the HAAnF in an Naanf\_AKMA\_SessionKey\_Push\_Request. This request may be sent with each refresh of the KAF or KSession of the Ua\* protocol.

In step 8, the HAAnF acknowledges the request with an Naanf\_AKMA\_SessionKey\_Push\_Response.

In step 9, the HAAnF detects based on the SN name that the UE is roaming and if the VPLMN has AKMA LI enhancements. The VPLMN AKMA capabilities and policies may be configured in the HAAnF and may be based on SLAs. Based on the AKMA support in the VPLMN, policies or SLAs, the HAAnF selects either the NF which enables LI in the VPLMN (continues with step 10a) or the VAAnF (continues with step 10b).

VPLMN does not support AKMA:

In step 10a, The HAAnF sends a Nnf\_AKMA\_ApplicationKey\_Provisioning\_Request to the NF which enables LI in the VPLMN. The request contains the full security context for LI of the UE for this AKMA session, i.e. A-KID, AF\_ID, SUPI, KAF, KAF expiration time, KSession.

In step 11a, the NF which enables LI acknowledges the request with a Nnf\_AKMA\_ApplicationKey\_Provisioning\_Response.

In step 12a, the NF stores the LI security context for potential LI request in the VPLMN. The NF may delete the LI security context after expiration of KAF. In case of KAF or KSession key refresh, the NF needs to be informed about the new key with the same procedure.

VPLMN supports AKMA:

In step 10b, the HAAnF provides the KAF and the KAF expiration time together with the SUPI of the UE and the session key KSession to the VAAnF in the VPLMN for storing the AKMA LI context. The VAAnF acknowledges the request in step 11b.

### 6.6.3 Evaluation

The HAAnF retrieves the serving name of the UE and can check whether the UE is roaming in a VPLMN or not. If the AF further derives a session key from KAF, then this is also provided to the HAAnF. The HAAnF needs to be enhanced with SLAs, operator policies or corresponding contact NF (NEF or VAAnF) in the VPLMNs. Depending on the AKMA support in the VPLMN, the HAAnF provides the AKMA security context to the corresponding contact NF.

## 6.7 Solution #7: Introducing AP into AKMA

### 6.7.1 Introduction

TS 33.222 specifies the use of Authentication Proxy in GBA [3], where an Authentication Proxy (AP) is a proxy resides between the UE and ASs. It helps to reduce the consumption of authentication vectors and/or to minimize SQN synchronization failures, and relieves the AS of security tasks. Similarly, introducing such an authentication proxy in AKMA is beneficial where different application servers reside in the same trust domain or in the same edge node. With the AP, these application servers can rely on the AP to execute AKMA procedures, which is more cost efficient than the case where each application servers execute AKMA procedures separately. AKMA is a potential solution in MEC, and it is possible that different application servers reside in the same edge cloud or belong to the same service vendor, it is beneficial to consider the feasibility of introducing a similar proxy in AKMA.

### 6.7.2 Solution details

#### 6.7.2.1 Architecture of using AP

An Authentication Proxy (AP) is a proxy which takes the role of an AF and delegates a group of 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 TS 33.535 [2].

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



Figure 6.7.2.1-1: Use of AP when AP is internal



Figure 6.7.2.2-1: Use 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. KAF), 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’s 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.

#### 6.7.2.2 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 [5]. 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.

#### 6.7.2.3 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 application server (AS) of this task. When an HTTPS request is destined towards an application server (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 6.7.2.3-1: Use of AP for TLS tunnels when AP is internal



Figure 6.7.2.3-2: Use of AP for TLS tunnels when AP is external

### 6.7.3 Evaluation

The solution fulfills the requirements in key issue#2. The solution enables the application servers relying on the authentication proxy to execute AKMA procedures.

## 6.8 Solution#8: AAnF discovery and selection for internal AF in AKMA roaming

### 6.8.1 Introduction

This solution specified the AAnF discovery and selection for internal AF which supports case 1 and case 2 in KI#1 of TR 33.737[1] in AKMA roaming scenario.

### 6.8.2 Solution details

#### 6.8.2.1 AAnF discovery and selection for internal AF



Figure 6.8.2.1-1: AAnF discovery and selection for internal AF in AKMA roaming

Pre-requisite: The AKMA security context ((including the AKMA anchor key, A-KID, and SUPI) has been provided to the VPLMN before, e.g., the AUSF has registered the AKMA security context to the vAAnF after KAKMA derivation.

1. When the internal 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 of TS 33.535[2]. The internal AF preforms the AAnF selection, discovers an AAnF utilizing the NRF.

* If the internal AF resides in the HPLMN, the AAnF discovery and selection is performed as specified in the clause 6.7 of TS 33.535[2].
* If the internal AF resides in the VPLMN, the AF invokes the Nnrf\_NFDiscovery\_Request service from the vNRF to find the vAAnF, the input includes AF\_LOCATION\_IND which identifies the network where the AF resides, A-KID, and another parameter as specified in clause 5.2.7.3.2 of TS 23.502[3].

2. NRF performs the following operation:

* If the internal AF resides in the HPLMN, the AAnF discovery and selection is performed as specified in the clause 6.7 of TS 33.535[2].
* Or if the internal AF resides in the VPLMN, the vNRF provides the IP address or FQDN of the vAAnF instance or another parameter as specified in clause 5.2.7.3.2 of TS 23.502[3].

3. Then the internal AF sends a Naanf\_AKMA\_ApplicationKey\_Get request to hAAnF/vAAnF with the A-KID to request the KAF for the UE. The internal AF also includes its identity (AF\_ID) in the request.

4. The hAAnF/vAAnF generates the KAF as specified in clause 6.2.1 of TS 33.535[2]

5. The hAAnF/vAAnF sends the response to the AF with the KAF, the KAF expiration time (KAF exptime), SUPI.

### 6.8.3 Evaluation

TBD.

## 6.9 Solution #9: Roaming AKMA architecture of the AF in Data Network (Internet)

### 6.9.1 Introduction

This solution addresses KI#1 (Case 3). This solution is proposed for the cases of function that enables LI deployment when AF located in Data Network (Internet).

### 6.9.2 Solution details

#### 6.9.2.1 Roaming AKMA architecture of the AF in Data Network (Internet)

Figure 6.9.2.1-1 shows the architecture that UE is roaming in a VPLMN and accesses the application deployed in Data Network(internet). The UE accesses the third party deployment application. If the data between the UE and application had not been encrypted, the VPLMN can detect the unencrypted data flow through UPF in the VPLMN. If the data between the UE and application had been encrypted by using an encrypted Ua\* protocol, the AMF in the VPLMN has to obtain related the KAF from the AAnF in the HPLMN base on the agreements between the HPLMN and the VPLMN. When UE is roaming to a target AMF, the target AMF can obtain related the KAF from the old AMF with the UE context. In this case, the function that enables LI could be AMF in the VPLMN.



**Figure 6.9.2.1-1**: **The architecture that UE is roaming in a VPLMN and accesses the application deployed Data Network (\*function that enables LI)**

#### 6.9.2.2 Roaming AKMA procedure of the AF in Data Network (Internet)

Figure 6.9.2.2-1 shows the procedure that the VPLMN NF obtains the KAF from AF in Data Network (Internet). Because the AAnF knows KAF but does not know the exact encryption/decryption key used between the UE and the AF, the AAnF can only send the KAF to the VPLMN. The AAnF obtains the VPLMN NF (e.g. AMF) id that can store the key from the UDM base on the UE ID. An indication is needed to indicate whether the KAF is sent by the AAnF, and may further indicate that the AAnF does not know whether the KAF is used as the encryption/decryption key between the UE and the AF.



Figure 6.9.2.2-1: VPLMN NF obtains the KAF from AF in Data Network (Internet)

Step 1-3, 8 and 9 are same as the step 1-3, 4 and 5 in clause 6.3 in TS 33.535[2]. The differences to the normal procedures of TS 33.535[2] are described below.

Step 4, the AAnF sends the Nudm\_Get\_Roaming\_NFid Request message to the UDM with the UE ID. The UE ID is the SUPI of the UE.

Step 5, the UDM sends the Nudm\_Get\_Roaming\_NFid Response message to the AAnF with the VPLMN NF id. The VPLMN NF can be the AMF which the UE is currently attached to in VPLMN network.

Step 6, based on the agreements between HPLMN and VPLMN, the AAnF sends the Push application Key Request message to the VPLMN NF, where the message carries a KAF, an indication, and a UE ID. The indication indicates that the carried key is the KAF, and may further indicate that the HPLMN does not know whether the KAF is used as the data flow encryption/decryption key between the UE and the AF. The UE ID is the SUPI of the UE.

Step 7, if the VPLMN NF supports AKMA roaming, the VPLMN NF stores the key and the indication, and sends the Push application Key Response message to the AAnF. If not, the VPLMN NF ignores the Push application Key Request message.

#### 6.9.2.3 New service: Nudm\_Get\_Roaming\_NFid service operation

**Service operation name:** Nudm\_Get\_Roaming\_NFid.

**Description:** The NF consumer requests VPLMN NF id from the UDM.

**Input, Required:** SUPI .

**Input, Optional:** None.

**Output, Required:** VPLMN NF id.

**Output, Optional:** None.

### 6.9.3 Evaluation

This solution addresses key issue#1 (case 3) and fulfils the requirements in case that the AF is located in Data Network (Internet).

The solution proposes to obtain KAF from the AAnF in the HPLMN based on the agreements between the HPLMN and the VPLMN when the data between the UE and application have been encrypted.

The solution proposes that the AAnF obtains the VPLMN NF id that can store the key from the UDM base on the UE ID.

The solution proposes that the AAnF pushes the encryption key to the VPLMN NF.

The solution needs one VPLMN NF (e.g. AMF) to store the key.

## 6.10 Solution #10: Support of AKMA roaming with K\_SEAF

### 6.10.1 Introduction

This solution addresses the KI #1.

The proposed solution supports AKMA roaming by using K\_SEAF to derive K\_AKMA when the UE connects through the VPLMN. By doing this, the VPLMN does not require any further interactions with the HPLMN to provide AKMA services when the AF is in or connects to the VPLMN. Similarly, the HPLMN can also generate the same AKMA keys derived from K\_SEAF since the AUSF has K\_AUSF.

The proposed solution facilitates LI when UE is in the VPLMN since the VPLMN has all required AKMA keys independently of the location of the AF. In case that the AF is in/connects to the HPLMN and the Ua\* protocol involves additional keys, this solution includes interactions to let the VPLMN request / HPLMN deliver those additional keys.

### 6.10.2 Solution details



**Figure 6.10.2.1-1: AKMA Roaming solution**

Step 1 is as defined in TS 33.501[4] clause 6.1.3 and TS 33.535[2] Clause 6.1.

Step 2 is as defined in TS 33.535[2] Clause 6.1 with the exceptions that the UE roaming in the VPLMN computes:

* KvAKMA as in TS 33.535[2] Annex A.2 using KSEAF instead of KAUSF.
* A-vKID is as A-KID in TS 33.535[2] Clause 6.1 with the exception that the realm part of the A-vKID shall include Serving Network Identifier and A-vTID is used.
* A-vTID is as the A-TID in TS 33.535[2] Clause 6.1 with the exception that it shall be derived as specified in Annex A.3 using KSEAF instead of KAUSF.

Step 3 is as defined in TS 33.535[2] Clause 6.1 having the SEAF in the VPLMN to compute KvAKMA, A-vKID, and A-vTID with the same exceptions as in Step 2 and providing parameters to the vAAnF.

Step 4 is as defined in TS 33.535[2] Clause 6.1 having the AUSF in the HPLMN to compute KvAKMA, A-vKID, and A-vTID with the same exceptions as in Step 2 and providing parameters to the hAAnF.

In Step 5 the UE requests an application session establishment request with A-vKID towards the vAF.

In Step 6 the vAF shall discover the vAAnF and shall send the Naanf\_AKMA\_ApplicationKey\_Get response with A-vKID and AF\_ID to the vAAnF. The vAAnF derives then the AF key from K\_vAKMA.

In Step 7 the vAAnF shall respond with Naanf\_AKMA\_ApplicationKey\_Get response containing Registered SN ID.

NOTE: The VPLMN has all AKMA keys to support LI when the AF is in the VPLMN.

Steps 9-12 are as Steps 5-8, with the difference that the AF is in the hPLMN.

NOTE: The VPLMN has all AKMA keys to support LI when the AF is in the HPLMN.

In Steps 13 and 14, the AMF/SEAF in the VPLMN might – if required – request other security information for LI purposes to the vAF and/or hAF, e.g., Ua\* protocol, security algorithms, etc. This request might be done to a local vAF or to the hAF.

In Steps 15 and 16, the vAF and/or hAF deliver any additional security information for LI purposes when available or requested.

NOTE: As HPLMN provides K\_SEAF to VPLMN after primary authentication, it may also provide AKMA related parameters e.g., AKMA indication to check whether UE supports AKMA services, and if so, how to handle them. The HPLMN can provide the AKMA indication to the VPLMN (SEAF/AMF) as part of the subscription data. The routing indicator might be communicated or the VPLMN may use a default value so that the AAnF NF consumer can select any AAnF instance within the VPLMN.

### 6.10.3 Evaluation

This solution addresses KI#1.

The main benefit of this solution is that the VPLMN has direct access to AKMA keys as soon as the UE has finished the primary authentication procedure since AKMA keys are derived from K\_SEAF. This approach removes the need of interacting between VPLMN and HPLMN to retrieve those keys. This approach facilitates LI during roaming since AKMA keys are directly available in the VPLMN.

These benefits are achieved by changing the way K\_AKMA is derived when the UE is in the VPLMN, and this can lead to two issues:

1. R17 UEs cannot access AKMA services when roaming. This can be addressed by (1) having a different solution for R17 UEs in roaming; or (2) upgrading the way R17 UEs act in roaming as in this solution; or (3) not supporting AKMA roaming with R17 UEs. If this is not done, this solution has backwards compatibility issues due to the changes in the UE behaviour.
2. The proposed architecture in this solution deviates from the architectural assumptions in Clause 4 since, e.g., a VPLMN AF is served by the AAnF of the VPLMN and not the AAnF of the HPMLN through the hSEPP.

## 6.11 Solution #11: AKMA Authentication in roaming scenario

### 6.11.1 Introduction

This solution addresses the key issue #1: Support for AKMA roaming scenario.

### 6.11.2 Solution details

In this solution, it is assumed the VPLMN deployed AAnF (shown as VAAnF). There are 2 options for VAAnF to request KAF :

Option#1: VAAnF fetch KAF from HAAnF.

Option#2: VAAnF fetch KAKMA from HAUSF, and VAAnF generates KAF based on KAKMA.

#### 6.11.2.1 Option#1 details

Table

Description automatically generated with medium confidence

**Fig 6.11.2.1-1 Authentication in roaming scenario – Option#1**

Step 0: UE performs primary authentication with the network. Then KAUSF is shared between UE and AUSF in Home network.

Step 1.1: UE generates KAKMA and A-KID following AKMA procedure in TS 33.535 and stores them securely.

Step 1.2: AAnF generates KAKMA and A-KID following AKMA procedure in TS 33.535 and stores them securely.

Step 2: UE derives KAF following AKMA procedure in TS 33.535.

Step 3: AUSF selects the AAnF as defined in clause 6.7 in TS 33.535 and send the generated A-KID and KAKMA to the AAnF together with the SUPI of the UE using the Naanf\_AKMA\_KeyRegistration Request service operation.

Step 4: UE send Application session establishment request (A-KID) to AF.

Step 5: AF determines on whether to communicate with VAAnF or HAAnF.

NOTE: AF could contact HAAnF for fetch KAF, which is exactly the AKMA feature defined in R17 in TS 33.535. no need to repeat here. But AF may have local policy to decide which AAnF should it contact.

Editor’s Note: How the AF selects the HAAnF when the UE is roaming is FFS

Step 6: AF sends Naanf\_AKMA\_ApplicationKey\_Get request(A-KID, AF\_ID) to VAAnF

Step 7: Based on the information provided in A-KID, VAAnF checked this is a roaming UE, so VAAnF sent Naanf\_AKMA\_ApplicationKey\_Get request (A-KID) to HAAnF.

Step 8: HAAnF derives KAF from KAKMA following TS 33.535.

Step 9: HAAnF sent Naanf\_AKMA\_ApplicationKey\_Get response (KAF , KAF expTime, SUPI) to VAAnF.

Step 10: VAAnF send Naanf\_AKMA\_ApplicationKey\_Get response (KAF , KAF expTime, SUPI) to AF.

#### 6.11.2.2 Option#2 details

**Diagram

Description automatically generated**

**Fig 6.11.2.2-1 Authentication in roaming scenario – Option#2**

Step 0: UE performs primary authentication with the network. Then KAUSF is shared between UE and AUSF in Home network.

Step 1.1: UE generates KAKMA and A-KID following AKMA procedure in TS 33.535 and stores them securely.

Step 1.2: AAnF generates KAKMA and A-KID following AKMA procedure in TS 33.535 and stores them securely.

Step 2: UE derives KAF following AKMA procedure in TS 33.535.

Step 3: AUSF selects the AAnF as defined in clause 6.7 in TS 33.535 and send the generated A-KID and KAKMA to the AAnF together with the SUPI of the UE using the Naanf\_AKMA\_KeyRegistration Request service operation.

Step 4: UE send Application session establishment request (A-KID) to AF.

Step 5: AF determines on whether to communicate with VAAnF or HAAnF.

NOTE: AF could contact HAAnF for fetch KAF, which is exactly the AKMA feature defined in R17 in TS 33.535, thus no need to repeat here. But AF may have local policy to decide which AAnF should it contact.

Editor’s Note: How the AF selects the HAAnF or VAAnF when the UE is roaming is FFS

Step 6: if AF determines to contact VAAnF in step 5, AF sends Naanf\_AKMA\_ApplicationKey\_Get request(A-KID, AF\_ID) to VAAnF

Step 7: Based on the information provided in A-KID, VAAnF checked this is a roaming UE, so VAAnF sent Nausf\_AKMA\_Key\_Get request (A-KID) to AUSF in HPLMN.

Editor’s Note: How the VAAnF finds the AUSF registered by UEs through the A-KID is FFS.

Step 8: AUSF response with Nausf\_AKMA\_Key\_Get response (KAKMA) to VAAnF.

Step 9: VAAnF derives KAF , KAF expTime based on KAKMA and AF\_ID following the key derivation function defined in A.4 in TS 33.535.

Step 10: VAAnF send Naanf\_AKMA\_ApplicationKey\_Get response (KAF , KAF expTime, SUPI) to AF.

6.11.2.2.1 New service: Nausf\_AKMA\_Key\_Get service operation

**Service operation name:** Nausf\_AKMA\_Key\_Get.

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

**Input, Required:** A-KID, AF\_ID

**Input, Optional:** None.

**Output, Required:** KAKMA

**Output, Optional:** None.

### 6.11.3 Evaluation

TBA.

## 6.12 Solution #12: AKMA anchor key forwarding to the VPLMN during primary authentication procedure

### 6.12.1 Introduction

This solution addresses the KI #1. The proposed solution supports registration of AKMA anchor key (KAKMA) and A-KID to the AAnF in VPLMN after primary authentication for UE in the same manner of KAKMA and A-KID registration to the AAnF in HPLMN. The KAKMA and A-KID are forwarded to the VPLMN during primary authentication procedure. After the primary authentication procedure, the AMF in VPLMN registers the AKMA keys to the AAnF in VPLMN.

Once the KAKMA and the A-KID are registered in AAnFs in VPLMN and HPLMN, the UE in VPLMN is able to access to both VPLMN AF and HPLMN AF. Moreover, even if KAUSF which is a root key of the KAKMA is changed by new primary authentication, there is no need additional key update procedure for the VPLMN AAnF since the new KAKMA and A-KID will replace the old keys whenever primary authentication is performed.

### 6.12.2 Solution details

#### 6.12.2.1 AKMA anchor key registration in roaming scenario



Figure 6.12.2.1-1: AKMA anchor key registration to the AAnF in VPLMN after primary authentication

1-5. Step 1 to step 5 are defined in clause 6.1 of TS 33.535 [2].

6. Rest of primary authentication procedure is performed as defined in clause 6.1.3 of TS 33.501 [4].

7. During the primary authentication, the AUSF responds to the AMF in VPLMN with information of authentication result via Nausf\_UE Authentication\_Authentication Response. If the UE is from other serving network based on SN name, the AUSF includes A-KID and KAKMA to the AMF in VPLMN in the response.

8. The AUSF shall request to the HPLMN AAnF to register the A-KID and the KAKMA. The selection of the AAnF is described in clause 6.7 of TS 33.535 [2].

9. The AAnF in HPLMN responses to the AUSF after key registration completed.

10. If the A-KID and the KAKMA is forwarded to the AMF in VPLMN, the AMF registers the A-KID and the KAKMA to the AAnF in VPLMN.

11. The AAnF in VPLMN responses to the AMF after key registration completion.

#### 6.12.2.2 UE in VPLMN accessing internal VPLMN AF



Figure 6.12.2.2-1: Application session establishment between roaming UE and AF in VPLMN

0. The UE is roaming in VPLMN and AKMA anchor key is registred in the AAnF in VPLMN after the procedure in clause 6.12.2.1.1. When the UE initiates communication with the AF in VPLMN, it shall include the derived A-KID in the Application Session Establishment Request message.

2. The AF in VPLMN identifies based on the realm part from the received A-KID whether the UE is from other serving network. If the AF decides to support the UE from other serving network, the AF requests application key to the AAnF in VPLMN.

To selects the AAnF in VPLMN for the AF to provide the KAF, local configuration or NRF can be utilized. When NRF is used to discover and select the AAnF in VPLMN, both RID and home network identifier from the received A-KID are used to select the AAnF.

3. The AAnF in VPLMN derives the KAF from KAKMA if it does not already have KAF.

4. The AAnF in VPLMN sends Naanf\_AKMA\_ApplicationKey\_Get response to the AF with SUPI, KAF and the KAF expiration time.

5. The AF in VPLMNsends the Application Session Establishment Response to the UE.

NOTE: NFs in VPLMN could provide LI context (A-KID, KAKMA, KAF, etc.) when LI enabled.

#### 6.12.2.3 UE in VPLMN accessing internal HPLMN AF



Figure 6.12.2.3-1: Application session establishment between roaming UE and AF in HPLMN

0. The UE is roaming in VPLMN and AKMA anchor key is registered in the AAnF in HPLMN after the procedure in clause 6.12.2.1.

1. When the UE initiates communication with the AF in HPLMN, it shall include the derived A-KID in the Application Session Establishment Request message.

2. The AF in HPLMN requests application key to the AAnF in HPLMN.

3. The AAnF in HPLMN derives the KAF from KAKMA if it does not already have KAF.

4. The AAnF in HPLMN identifies based on the realm part from the received A-KID whether the UE is from other serving network. If the AAnF in HPLMN decides to support the UE from other serving network, the AAnF in HPLMN provides AKMA context to the AAnF in VPLMN to support LI.

To selects the AAnF in VPLMN for the AF to provide the KAF, local configuration or NRF can be utilized. When NRF is used to discover and select the AAnF in VPLMN, both RID and home network identifier from the received A-KID are used to select the AAnF.

5. The AAnF in VPLMN stores delivered AKMA related information as LI context.

6. The AAnF in VPLMN responses to the AAnF in HPLMN.

7. The AAnF in HPLMN sends Naanf\_AKMA\_ApplicationKey\_Get response to the AF in VPLMN with SUPI, KAF and the KAF expiration time.

8. The AF in HPLMN sends the Application Session Establishment Response to the UE.

NOTE: NFs in VPLMN could provide LI context (A-KID, KAKMA, KAF, etc.) when LI enabled.

### 6.12.3 Evaluation

This solution addresses case 1 of key issue#1 and fulfils the requirements in the case that the AF is located in the HPLMN.

The solution fulfils LI requirement that LI contexts (e.g. encrypted traffic data and keys related to AKMA) are ready before the application session established.

KAKMA in HPLMN is shared to VPLMN, and same KAKMA will be AKMA anchor key in HPLMN and VPLMN.

Editor’s Note: When same KAKMA is shared to VPLMN and HPLMN, how to make different the KAF in VPLMN and HPLMN is FFS.

Editor’s Note: How the AUSF discovers the AAnF in the VPLMN is FFS.

Editor’s Note: How the solution is aligned with clause 4 is FFS.

## 6.13 Solution #13: AKMA support in roaming

### 6.13.1 Introduction

This solution addresses KI#1. This solution is proposed for the case in which the VPLMN supports AKMA. The proposed solution includes vAAnF as a new functional entity which is the AKMA Anchor Function in the VPLMN. The vAAnF relays the Naanf\_AKMA\_ApplicationKey\_Get request and response between the AF in the visited network and the AAnF in the UE’s home network. vAAnF functionality includes:

* vAAnF functions as a proxy between the visited AF and the AAnF in the home network of the UE.
* vAAnF locates the AAnF in the home network of the UE and communicate with it over secure channel.
* vAAnF validates that the visited AF is authorized to participate in AKMA.

N64 (AKMA reference point):

* Reference point between the home AAnF and the visited AAnF.





Figure 6.13.1-1: AKMA Architecture in reference point representation for (a) internal AFs, (b) external AFs, (c) internal AFs (via vAAnF) and (d) external AFs (via vAAnF)

### 6.13.2 Solution details



Figure 6.13.2-1: AKMA Application Key retrieval by AAnF in visited network, from the AAnF in home network

1. UE registers to the network and performs the primary authentication as specified in TS 33.501 [4]. The UDM determines the support of AKMA in VPLMN based on the configuration or based on the subscription data. During primary authentication, the UDM provides the AUSF with “[AKMA Ind]” specifying whether AKMA Keys need to be generated for the UE as defined in TS 33.535 [2]. UDM also provides the AUSF with, AKMA Roaming Indication specifying whether the generated AKMA Keys can be propagated to the VPLMN, depending on operator agreements and/or AKMA support in VPLMN.

2. AUSF generates KAKMA and A-KID. Correspondingly, KAKMA and A-KID are generated at the UE. AUSF selects an AAnF instance to serve the UE, and registers SUPI, KAKMA and A-KID in the hAAnF. AUSF provides the hAAnF with AKMA Roaming Indication. AUSF can decide this based on indication from the UDM in Step (1) and/or local policy. AUSF also provides Visited-Network information to the hAAnF.

3a-3b. UE triggers communication over Ua\* interface with the AF based on its application needs. UE can optionally provide visited-network information (e.g. Serving PLMN ID) to the AF along with A-KID. Based on local configuration, and/or presence of visited-network information in the UE request, AF sends a request to the vAAnF (directly or via NEF) to provide KAF, if there is no secure interface with the hAAnF. The request contains AF-Identity (e.g. FQDN) and A-KID.

The AF can determine the PLMN in which the UE is currently roaming using the SN ID provided in the session establishment request.

4. Based on information contained in A-KID (Routing ID, Home Network Information), vAAnF discovers the hAAnF by querying the NRF.

5. vAAnF requests the hAAnF to provide application key. The request can be sent utilizing a new or existing service exposed by the hAAnF, and discovered by the vAAnF via the NRF. The request contains following information:

- A-KID of the UE as provided by the AF

- AF-Identity (e.g. FQDN) is provided by the AF

6. Based on the received request or local policy or the UDM provided AKMA Roaming Indication, hAAnF generates KAF from KAKMA and provides KAF and Key lifetime to the vAAnF.

7. Based on the received information, if vAAnF receives KAF and Key lifetime, it provides the same to AF.

NOTE: The LI aspects on providing the encryption keys to the VPLMN is not considered as part of this solution and solution#5 can be reused to address LI.

Editor’s Note: How this solution is aligned with the architectural assumption in clause 4 is FFS.

### 6.13.3 Evaluation

This solution partially addresses key issue#5.1 (Support for AKMA roaming scenario) and fulfils the first requirement on the support of AKMA in roaming.

Impacts to UE:

- UE needs to send the SN ID in the session establishment request to the AF.

## 6.14 Solution #14: AKMA roaming with AF outside VPLMN

### 6.14.1 Introduction

This solution addresses Key Issue #1 "Support for AKMA roaming scenario", especially the scenario when the UE is roaming in VPLMN and trying to access AF located outside VPLMN, i.e. this AF could be an internal HPLMN AF or external AF in Data Network.

The solution proposes to introduce an AAnF in the VPLMN, named vAAnF, to enable the VPLMN to decrypt user traffic encrypted between the roaming UE and AF thanks to AKMA-related credentials. The vAAnF receives from the HPLMN AKMA Application key KAF and associated parameters enabling the VPLMN to decrypt the user content exchanged between the roaming UE and AF located outside the VPLM. The solution also ensures that the AKMA Application Key KAF provided by the HPLMN to VPLMN is correct.

### 6.14.2 Solution details

The solution addresses the scenario where AF is an internal HPLMN AF.

 **Figure 6.14.2-1: Sending of AKMA credentials to VPLMN**

Pre-requisite: there is establishment of the key KAKMA between the UE and hAAnF. And, during the establishment of the AKMA key (KAKMA), the AUSF also provides the SN name to the AAnF in the HPLMN (hAAnF). The SN name is used later to determine whether the UE is roaming and to select appropriate AAnF in VPLMN (vAAnF) in order to store AKMA details.

Editor’s Note: this solution assumes that there is only one vAAnF per VPLMN.

Steps 1, 2, 3 take place as specified in TS 33.535 [2].

4. The hAAnF determine whether the UE is roaming. If the UE is roaming then the hAAnF provides the AKMA Application key KAF and associated parameters A-KID, AF\_ID, SUPI, and KAF expiration time to the vAAnF.

NOTE 1: same mechanism applies in case that the encryption key provided by the AAnF in the HPLMN (hAAnF) is not the AKMA Applicaton key KAF. The key KAF would be replaced by the encryption key provided by hAAnF in all the steps describing the use of the key KAF.

5. The vAAnF stores the key KAF and associated parameters received from the hAAnF and sends back response to the hAAnF.

Steps 6 and 7 take place as specified in TS 33.535 [2].

8. The UE sends a message to the AMF in order to check that the vAAnF stores the correct AKMA Application Key KAF to address LI requirement.This message contains the current GUTI value encrypted by the UE with KAF, and associated A-KID and AF-ID. The GUTI value encrypted with KAF is named GUTIe.

NOTE 2: GUTIe could also be Message Authentication Code.

9. The AMF sends to the vAANF: GUTIe, A-KID, AF-ID received from the UE in step 8, and GUTI value in clear text.

10. The vAANF retrieves the AKMA Application Key KAF thanks to A-KID, AF-ID and decrypts the GUTIe with KAF. Then, the vAAnF verifies that the decrypted GUTI is the same than the GUTI in clear text sent by the AMF.

11. The vAANF sends to the AMF the result of the KAF check.

12. The AMF forwards the result of the KAF check to the UE.

13. The UE can use KAF if the KAF check was successful.

The steps 8 and 12 correspond to new NAS messages.

NOTE 3: To avoid race conditions, i.e. that the AF might start an encrypted Ua\* session with the UE before the LI provisioning has been completed in the VPLMN, it is important that message ordering as proposed is strictly followed. That is, AKMA keys are not available to the AF, until the response message 5 has been received/verified.This solution also covers the scenario where AF is an external AF in Data Network. The same steps apply.

### 6.14.3 Evaluation

This solution addresses the Key Issue # (cases 1 and 3) and fulfils the security requirements in case that the UE is roaming in VPLMN and accesses an AF outside this VPLMN.

The solution addresses the scenarios where the encryption key provided by the AAnF in the HPLMN (hAAnF) to AAnF of VPLMN (vAAnF) could be the AKMA Applicaton key KAF, or any other key.

Additionnaly, this solution ensures that the encryption key provided by the HPLMN to VPLMN is valid to decrypt the traffic. The mechanism to ensure that the encryption key provided by the HPLMN is valid relies on steps 8 to 13.

The solution impacts the serving network and the ME.

The steps 8 to 13 to ensure that the encryption key provided to VPLMN is valid to decrypt the traffic could apply to any solution addressing Key Issue#1 where a UE is roaming in VPLMN and accesses an AF outside this VPLMN.

## 6.15 Solution #15: AKMA roaming for external AF in Data Network

### 6.15.1 Introduction

This solution addresses the KI #1.

This solution considers external AF in a data network in roaming scenarios. Specifically, AKMA architecture in clause 6.3 TS 33.535 [1] is reused to support the AKMA roaming which involves external AF in a data network.

### 6.15.2 Solution details

To support AKMA roaming which involves external AF in a data network, this solution reuses clause 6.3 TS 33.535 with the following modifications.

- In step 3, the NEF checks whether the connected AF is an external AF in the data network based on the local policies. If the AF is an external AF in the data network, the NEF sends an AF indicator to the AAnF in the home network of UE. The AF indicator indicates that the AF is an external AF in a data network.

- If the AAnF recieves the AF indicator and the UE is roaming, the AAnF in the home network sends SUPI of the UE, AF\_ID, KAF, the KAF expiration time (KAF exptime) to the visited network (e.g., UPF in the visited network) of the UE. The hAAnF may obtain the visited network name of the UE from the AUSF/UDM.

### 6.15.3 Solution Evaluation

This solution address part of case 2(UE in VPLMN accessing an internal AF of the VPLMN) and case 3 (UE is in VPLMN and accessing an external AF in the Data Network)

This solution has impact to both hAAnF and NEF. In specific, NEF needs to send an AF indicator to the AAnF in the home network of UE based on local policies. The AF indicator indicates that the AF is an external AF in a data network.

The advantages of this solution is given as follows.

In specific, this solution enables hAAnF to differ the VPLMN AF and the external AF.

For external AF, hAAnF needs to send AKMA related information (e.g., KAF) to the VPLMN.

For VPLMN AF, hAAnF does not need to send AKMA related information (e.g., SUPI of the UE, AF\_ID, KAF, the KAF expiration time) to the VPLMN.

The disadvantages of this solution is given as follows.

This solution has impacts to both AAnF and NEF.

## 6.16 Solution #16: AKMA roaming with VPLMN AKMA Support NF for inbound roamers

### 6.16.1 Introduction

The solution addresses case #1 (UE roaming and accessing internal HPLMN AF) and case #3 (UE roaming and accessing an external AF in the Data Network of KI#1.

It uses the principle that the UDM knows the VPLMN(s) that the UE is roaming after the AMF which handles the UE registers to the UDM.

If the UE has registered to multiple PLMNs the UDM has this knowledge and provide this to the AKMA AF or AAnF. The AF and AAnF may decide to send one or multiple VPLMNs the AKMA related keys and parameter to enable LI in the VPLMNs.

This solution proposes that the AMF is configured with or dynamically discovers the AKMA Support NF (ASNF) in the VPLMN which enables LI, and sends this information to the UDM when it registers the UE to the UDM. Any other NF in the HPLMN can discover the UE VPLMN information as well as the address of the AKMA Support NF enabling LI in the VPLMN (if any).

This solution also proposes that the AAnF could be used as the proxy in the HPLMN to push any AKMA related keys from itself or the AF.

### 6.16.2 Solution details

Editor's Note: Whether the address of the ASNF or an indication that the VPLMN supports AKMAis sent to the HPLMN is FFS.

Figure 6.16.2-1 provides the sequence diagram of the solution. The ASNF is the AKMA Support Network Function in the VPLMN. The ASNF represents an AKMA NF enabling LI which stores AKMA related key material and parameters. The ASNF can be any existing NF (AMF, AAnF, UPF) or part of an existing NF or a new NF.

Figure 6.16.2-1: Solution sequence diagram

0. The UE registers to the network. If a primary authentication procedure is required, steps 0a - 1 are executed. The AKMA Anchor Key (KAKMA) is generated as defined in TS 33.535 [2].

1. The AAnF optionally sends a request to the UDM to obtain the UE roaming information. The message contains the SUPI of UE, possibly with an indication to retrieve the AKMA ASNF information. The AAnF subscribes to the UDM on updates of UE roaming information.

The UDM optionally sends the UE roaming information to AAnF. The message contains the UE SUPI, the UE roaming information e.g., PLMN ID of visited network, ASNF address (if available). If the UE is not registered in UDM yet or registered over more than one visiting PLMNs, there could be zero or multiple sets of such information in the response message.

NOTE 1: According to TS 23.501[7], 4.2.8.2.2, the UE can be registered in two VPLMNs.

2. In the general case (of the ANSF being a standalone NF or integrated to other NFs) the AMF obtains the address of ASNF via the NRF or local configuration. The AMF sends Nudm\_UECM\_Registration to UDM for the UE. The message contains the UE SUPI, serving PLMN ID, and ASNF address. The UE registration information is updated and stored in the UDM.

3. The UDM optionally sends the UE roaming information Update to the AAnF. The message contains the UE SUPI, PLMN ID(s) of visited network(s) and ASNF address(s).

4a. The UE initiates communication with the AF via the Application Session Establishment Request message.

4b. The AF requests the AF key (KAF) from the AAnF, possibly via NEF.

4c. The AAnF generates AF key (KAF) and returns the key to the AF as specified in TS 33.535 [2].

The AAnF now have AKMA related key material and parameters KAF and the information used for key derivation thereof.

4d. In the AF key response message, the AAnF may include possibly with an indication to request the AF to report any AF key material and parameters.

NOTE 2: If the AF is in the Data Network (Internet), whether the AF can process this request for key material or whether the AF will provide the key material to the AAnF depends upon the HPLMN and AF provider agreements.

5. The AKMA application session between UE and AF is established and protected. The communication could be protected by the KAF or key material derived from the KAF or other key material derived in the AF.

The AF has now KAF, any security material derived from KAF or other key material derived in the AF, the information used for keys derivation thereof including selected encryption algorithms, cipher suites, input for ciphering: key(s), nonce(s), counter(s) etc.

5b. The AAnF optionally requests from the latest roaming information (if any) from the UDM and subscribes to these changes as Step 1.

NOTE 3: Step 1 is optional, and it shows the proactive aspect of the solution i.e. every time there is change of information in the UDM the AAnF is notified. However, the roaming information in the AAnF for a specific UE will not be needed if UE does not setup any AKMA session. Therefore, it could be better for the AAnF to request roaming information and changes after the AKMA session establishment response as it is performed in this step.

6. The AAnF obtains the ASNF address for the UE (e.g. received from step 1 and/or step 3) and sends a message e.g. AKMA Cryptographic Information Report to the ASNF(s). The message contains the UE ID (e.g. SUPI), and cryptographic information such as KAF, derived keys from KAF or any other keys and parameters used to protect the communication between the UE and the AF.

7. The AF sends a message to the AAnF potentially via NEF, e.g. AF Cryptographic Information Report, to provide or update the keys and parameters to the AAnF. The message contains for example the UE ID, keys and parameters used to derive AKMA application session keys.

NOTE 4: Step 7 can take place before or after step 5 and it can happen in parallel or after step 6

8. If any cryptographic information (used for the protection of the AKMA session between the UE and AF) is updated in AAnF, the AAnF sends a message e.g., AKMA Cryptographic Information Update to ASNF(s). The message contains UE ID (e.g., SUPI), and cryptographic information such as selected encryption algorithms, cipher suites, input for ciphering: key(s), nonce(s), counter(s) etc.

Now the ASNF(s) has/have the AKMA cryptographic information for the UE and perform(s) necessary regulatory actions accordingly.

Editor's Note: It is FFS how the selection of the ASNF is performed if the there are multiple ASNF(s) in the network

### 6.16.2 Evaluation

The solution addresses Key Issue #1: Support for AKMA roaming scenario, case #1 (UE roaming and accessing internal HPLMN AF) and case #3 (UE roaming and accessing an external AF in the Data Network.

The solution proposes that the AMF discovers the AKMA related function in the VPLMN which enables LI (AKMA Support Network Function, ASNF) and provides the address of such function to the UDM in the HPLMN. The AAnF subscribes to the UDM for the address of the ASNF in the VPLMN. Whenever the UE changes VPLMN the UDM gets this information from the AMF via Nudm\_UECM\_Registration. The AF provides keys and related parameters for the protection of the AKMA session between a UE and AF to the AAnF and the AAnF forwards this information to the ASNF in the VPLMN. For the case of multiple registrations, the AAnF has the information from the UDM and provides the related key material and parameters to all the VPLMN(s).

The ASNF can be any existing NF (AMF, AAnF, UPF) or part of an existing NF or a new NF.

The solution has impact on the AMF in the VPLMN, the UDM, the AAnF.

# 7 Conclusions

## 7.1 Conclusion to Key Issue#1

Regarding AKMA roaming architecture, AKMA architecture defined in TS 33.535[2] can be reused.

For case 2 (UE in VPLMN accessing an internal AF of the VPLMN), since the AF knows the encryption key used between the UE and the AF, the LI requirements can be fulfilled by the AF, thus no normative work is needed.

Editor’s Note: Further conclusion on LI is FFS.

## 7.2 Conclusion to Key Issue#2

It’s proposed that solution #7 is used for normative work.

Annex A (informative):  
Change history

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Change history** | | | | | | | |
| **Date** | **Meeting** | **TDoc** | **CR** | **Rev** | **Cat** | **Subject/Comment** | **New version** |
| 2022-05 | SA3#107-e |  |  |  |  | TR Skeleton | 0.0.0 |
| 2022-05 | SA3#107-e | S3-221169 |  |  |  | S3-221288, S3-220812, S3-221289, S3-221218 | 0.1.0 |
| 2022-07 | SA3#107Adhoc-e | S3-221687 |  |  |  | S3-221634, S3-221635, S3-221662, S3-221651, S3-221652, S3-221596, S3-221592, S3-221688 | 0.2.0 |
| 2022-10 | SA3#108Adhoc-e | S3-223089 |  |  |  | S3-222521, S3-223096, S3-223097, S3-222960, S3-222569, S3-223126, S3-223082, S3-223083, S3-223084, S3-222985, S3-222950, S3-222701, S3-222938, S3-222718, S3-222719, S3-223057, S3-222933, S3-223087, S3-222472, S3-222524, S3-222634, S3-223096 | 0.3.0 |
| 2022-11 | SA3#109-e | S3-224109 |  |  |  | S3-223215, S3-223267, S3-223268, S3-224078, S3-224010, S3-224032, S3-223433, S3-223436, S3-224115, S3-224057, S3-224058, S3-224030, S3-224071, S3-224141, S3-224136, S3-224137 | 0.4.0 |
| 2023-01 | SA3#109Adhoc-e | S3-230584 |  |  |  | S3-230112, S3-230487, S3-230107, S3-230108, S3-230145, S3-230146, S3-230147, S3-230148, S3-230270, S3-230458, S3-230533, S3-230271 | 0.5.0 |