

ETSI TS 129 509 V15.5.0 (2019-10)



**5G;
5G System;
Authentication Server Services;
Stage 3
(3GPP TS 29.509 version 15.5.0 Release 15)**



Reference

RTS/TSGC-0429509vf50

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 - NAF 742 C
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

The present document can be downloaded from:

<http://www.etsi.org/standards-search>

The present document may be made available in electronic versions and/or in print. The content of any electronic and/or print versions of the present document shall not be modified without the prior written authorization of ETSI. In case of any existing or perceived difference in contents between such versions and/or in print, the prevailing version of an ETSI deliverable is the one made publicly available in PDF format at www.etsi.org/deliver.

Users of the present document should be aware that the document may be subject to revision or change of status.

Information on the current status of this and other ETSI documents is available at

<https://portal.etsi.org/TB/ETSIDeliverableStatus.aspx>

If you find errors in the present document, please send your comment to one of the following services:

<https://portal.etsi.org/People/CommiteeSupportStaff.aspx>

Copyright Notification

No part may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm except as authorized by written permission of ETSI.

The content of the PDF version shall not be modified without the written authorization of ETSI.

The copyright and the foregoing restriction extend to reproduction in all media.

© ETSI 2019.

All rights reserved.

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.

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is 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 IPR Policy, no investigation, 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.

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 <http://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

| | |
|--|----|
| Intellectual Property Rights | 2 |
| Legal Notice | 2 |
| Modal verbs terminology..... | 2 |
| Foreword..... | 7 |
| 1 Scope | 8 |
| 2 References | 8 |
| 3 Definitions and abbreviations..... | 9 |
| 3.1 Definitions | 9 |
| 3.2 Abbreviations | 9 |
| 4 Overview | 9 |
| 4.1 Introduction | 9 |
| 5 Services offered by the AUSF..... | 10 |
| 5.1 Introduction | 10 |
| 5.2 Nausf_UEAuthentication Service | 10 |
| 5.2.1 Service Description..... | 10 |
| 5.2.2 Service Operations | 11 |
| 5.2.2.1 Introduction..... | 11 |
| 5.2.2.2 Authenticate | 11 |
| 5.2.2.2.1 General | 11 |
| 5.2.2.2.2 5G AKA | 11 |
| 5.2.2.2.3 EAP-based authentication method..... | 12 |
| 5.2.2.2.3.1 General..... | 12 |
| 5.2.2.2.3.2 EAP method: EAP-AKA'..... | 12 |
| 5.2.2.2.3.3 EAP method: EAP-TLS..... | 14 |
| 5.3 Nausf_SoRProtection Service | 14 |
| 5.3.1 Service Description..... | 14 |
| 5.3.2 Service Operations | 14 |
| 5.3.2.1 Introduction..... | 14 |
| 5.3.2.2 Protect | 14 |
| 5.3.2.2.1 General | 14 |
| 5.4 Nausf_UPUProtection Service | 15 |
| 5.4.1 Service Description..... | 15 |
| 5.4.2 Service Operations | 15 |
| 5.4.2.1 Introduction..... | 15 |
| 5.4.2.2 Protect | 15 |
| 5.4.2.2.1 General | 15 |
| 6 API Definitions | 16 |
| 6.1 Nausf_UEAuthentication Service API..... | 16 |
| 6.1.1 API URI..... | 16 |
| 6.1.2 Usage of HTTP | 16 |
| 6.1.2.1 General | 16 |
| 6.1.2.2 HTTP standard headers | 16 |
| 6.1.2.2.1 General | 16 |
| 6.1.2.2.2 Content type | 16 |
| 6.1.2.3 HTTP custom headers | 17 |
| 6.1.2.3.1 General | 17 |
| 6.1.3 Resources..... | 17 |
| 6.1.3.1 Overview..... | 17 |
| 6.1.3.2 Resource: List of ue-authentications | 18 |
| 6.1.3.2.1 Description | 18 |
| 6.1.3.2.2 Resource Definition..... | 18 |
| 6.1.3.2.3 Resource Standard Methods | 18 |

| | | |
|-------------|--|----|
| 6.1.3.2.3.1 | POST..... | 18 |
| 6.1.3.2.4 | Resource Custom Operations | 19 |
| 6.1.3.2.4.1 | Overview..... | 19 |
| 6.1.3.3 | Resource: 5g-aka-confirmation (Document)..... | 19 |
| 6.1.3.3.1 | Description | 19 |
| 6.1.3.3.2 | Resource Definition..... | 19 |
| 6.1.3.3.3 | Resource Standard Methods | 20 |
| 6.1.3.3.3.1 | PUT..... | 20 |
| 6.1.3.4 | Resource: eap-session (Document) | 20 |
| 6.1.3.4.1 | Description | 20 |
| 6.1.3.4.2 | Resource Definition..... | 20 |
| 6.1.3.4.3 | Resource Standard Methods | 21 |
| 6.1.3.4.3.1 | POST..... | 21 |
| 6.1.4 | Custom Operations without associated resources | 21 |
| 6.1.4.1 | Overview | 21 |
| 6.1.5 | Notifications | 21 |
| 6.1.5.1 | General | 21 |
| 6.1.6 | Data Model | 22 |
| 6.1.6.1 | General | 22 |
| 6.1.6.2 | Structured data types | 22 |
| 6.1.6.2.1 | Introduction | 22 |
| 6.1.6.2.2 | Type: AuthenticationInfo | 22 |
| 6.1.6.2.3 | Type: UEAuthenticationCtx | 23 |
| 6.1.6.2.4 | Type: 5gAuthData | 23 |
| 6.1.6.2.5 | Type: Av5gAka | 23 |
| 6.1.6.2.6 | Type: ConfirmationData..... | 23 |
| 6.1.6.2.7 | Type: EapSession | 24 |
| 6.1.6.2.8 | Type: ConfirmationDataResponse..... | 24 |
| 6.1.6.3 | Simple data types and enumerations | 24 |
| 6.1.6.3.1 | Introduction | 24 |
| 6.1.6.3.2 | Simple data types..... | 24 |
| 6.1.6.3.3 | Enumeration: AuthType | 24 |
| 6.1.6.3.4 | Enumeration: AuthResult | 25 |
| 6.1.6.3.5 | Relation Types..... | 25 |
| 6.1.6.3.5.1 | General..... | 25 |
| 6.1.6.3.5.2 | The "5g-aka" Link relation | 25 |
| 6.1.6.3.5.3 | The "eap-session" Link relation..... | 25 |
| 6.1.6.4 | Binary data | 25 |
| 6.1.6.4.1 | Introduction | 25 |
| 6.1.7 | Error Handling | 25 |
| 6.1.7.1 | General | 25 |
| 6.1.7.2 | Protocol Errors | 25 |
| 6.1.7.3 | Application Errors..... | 25 |
| 6.1.8 | Security | 26 |
| 6.2 | Nausf_SoRProtection Service API..... | 27 |
| 6.2.1 | API URI..... | 27 |
| 6.2.2 | Usage of HTTP | 27 |
| 6.2.2.1 | General | 27 |
| 6.2.2.2 | HTTP standard headers | 27 |
| 6.2.2.2.1 | General | 27 |
| 6.2.2.2.2 | Content type | 27 |
| 6.2.2.3 | HTTP custom headers | 27 |
| 6.2.2.3.1 | General | 27 |
| 6.2.3 | Resources..... | 27 |
| 6.2.3.1 | Overview..... | 27 |
| 6.2.3.2 | Resource: ue-sor..... | 28 |
| 6.2.3.2.1 | Description | 28 |
| 6.2.3.2.2 | Resource Definition..... | 28 |
| 6.2.3.2.3 | Resource Standard Methods | 28 |
| 6.2.3.2.4 | Resource Custom Operations | 29 |
| 6.2.3.2.4.1 | Overview..... | 29 |
| 6.2.3.2.4.2 | Operation: generate-sor-data..... | 29 |

| | | |
|---------------|--|----|
| 6.2.3.2.4.2.1 | Description | 29 |
| 6.2.3.2.4.2.2 | Operation Definition | 29 |
| 6.2.4 | Custom Operations without associated resources | 29 |
| 6.2.4.1 | Overview | 29 |
| 6.2.5 | Notifications | 30 |
| 6.2.5.1 | General | 30 |
| 6.2.6 | Data Model | 30 |
| 6.2.6.1 | General | 30 |
| 6.2.6.2 | Structured data types | 30 |
| 6.2.6.2.1 | Introduction | 30 |
| 6.2.6.2.2 | Type: SorInfo | 31 |
| 6.2.6.2.3 | Type: SorSecurityInfo | 31 |
| 6.2.6.2.4 | Type: SteeringInfo | 31 |
| 6.2.6.2.5 | Type: SteeringContainer | 31 |
| 6.2.6.3 | Simple data types and enumerations | 32 |
| 6.2.6.3.1 | Introduction | 32 |
| 6.2.6.3.2 | Simple data types | 32 |
| 6.2.6.3.3 | Enumeration: AccessTech | 32 |
| 6.2.7 | Error Handling | 32 |
| 6.2.7.1 | General | 32 |
| 6.2.7.2 | Protocol Errors | 32 |
| 6.2.7.3 | Application Errors | 32 |
| 6.2.8 | Security | 33 |
| 6.3 | Nausf_UPUProtection Service API | 33 |
| 6.3.1 | API URI | 33 |
| 6.3.2 | Usage of HTTP | 33 |
| 6.3.2.1 | General | 33 |
| 6.3.2.2 | HTTP standard headers | 33 |
| 6.3.2.2.1 | General | 33 |
| 6.3.2.2.2 | Content type | 33 |
| 6.3.2.3 | HTTP custom headers | 34 |
| 6.3.2.3.1 | General | 34 |
| 6.3.3 | Resources | 34 |
| 6.3.3.1 | Overview | 34 |
| 6.3.3.2 | Resource: ue-upu | 34 |
| 6.3.3.2.1 | Description | 34 |
| 6.3.3.2.2 | Resource Definition | 34 |
| 6.3.3.2.3 | Resource Standard Methods | 35 |
| 6.3.3.2.4 | Resource Custom Operations | 35 |
| 6.3.3.2.4.1 | Overview | 35 |
| 6.3.3.2.4.2 | Operation: generate-upu-data | 35 |
| 6.3.3.2.4.2.1 | Description | 35 |
| 6.3.3.2.4.2.2 | Operation Definition | 35 |
| 6.3.4 | Custom Operations without associated resources | 36 |
| 6.3.4.1 | Overview | 36 |
| 6.3.5 | Notifications | 36 |
| 6.3.5.1 | General | 36 |
| 6.3.6 | Data Model | 36 |
| 6.3.6.1 | General | 36 |
| 6.3.6.2 | Structured data types | 36 |
| 6.3.6.2.1 | Introduction | 36 |
| 6.3.6.2.2 | Type: UpuInfo | 37 |
| 6.3.6.2.3 | Type: UpuSecurityInfo | 37 |
| 6.3.6.2.4 | Type: UpuData | 37 |
| 6.3.6.3 | Simple data types and enumerations | 37 |
| 6.3.6.3.1 | Introduction | 37 |
| 6.3.6.3.2 | Simple data types | 37 |
| 6.3.6.3.3 | Enumeration: UpuDataType | 38 |
| 6.3.7 | Error Handling | 38 |
| 6.3.7.1 | General | 38 |
| 6.3.7.2 | Protocol Errors | 38 |
| 6.3.7.3 | Application Errors | 38 |

| | | |
|-------------------------------|------------------------------------|-----------|
| 6.3.8 | Security | 38 |
| Annex A (normative): | OpenAPI specification..... | 39 |
| A.1 | General | 39 |
| A.2 | Nausf_UEAuthentication API..... | 39 |
| A.3 | Nausf_SoRProtection API..... | 43 |
| A.4 | Nausf_UPUProtection API..... | 45 |
| Annex B (Informative): | Use of EAP-TLS | 47 |
| B.1 | General | 47 |
| B.2 | EAP method: EAP-TLS | 47 |
| Annex C (informative): | Withdrawn API versions..... | 49 |
| C.1 | General | 49 |
| C.2 | Nausf_SoRProtection API..... | 49 |
| Annex D (informative): | Change history | 50 |
| History | | 53 |

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.

1 Scope

The present document specifies the stage 3 protocol and data model for the Nausf Service Based Interface. It provides stage 3 protocol definitions and message flows, and specifies the API for each service offered by the AUSF.

The 5G System stage 2 architecture and procedures are specified in 3GPP TS 23.501 [2], 3GPP TS 23.502 [3] and 3GPP TS 33.501 [8].

The Technical Realization of the Service Based Architecture and the Principles and Guidelines for Services Definition are specified in 3GPP TS 29.500 [4] and 3GPP TS 29.501 [5].

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 23.501: "System Architecture for the 5G System; Stage 2".
- [3] 3GPP TS 23.502: "Procedures for the 5G System; Stage 2".
- [4] 3GPP TS 29.500: "5G System; Technical Realization of Service Based Architecture; Stage 3".
- [5] 3GPP TS 29.501: "5G System; Principles and Guidelines for Services Definition; Stage 3".
- [6] IETF RFC 7540: "Hypertext Transfer Protocol Version 2 (HTTP/2)".
- [7] IETF RFC 8259: "The JavaScript Object Notation (JSON) Data Interchange Format".
- [8] 3GPP TS 33.501: "Security Architecture and Procedures for 5G System".
- [9] IETF RFC 5448: "Improved Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA)".

Editor's Note: This reference may be removed and references to it updated when the IETF publishes the corresponding update version.

- [10] 3GPP TS 29.571: "5G System; Common Data Types for Service Based Interfaces; Stage 3".
- [11] IETF RFC 7807: "Problem Details for HTTP APIs".
- [12] 3GPP TS 29.503: "5G System; Unified Data Management Services; Stage 3".
- [13] IETF RFC 6749: "The OAuth 2.0 Authorization Framework".
- [14] 3GPP TS 29.510: "Network Function Repository Services; Stage 3".
- [15] 3GPP TS 31.102: "Characteristics of the Universal Subscriber Identity Module (USIM) application".
- [16] IETF RFC 5216: "The EAP-TLS Authentication Protocol".
- [17] Internet draft draft-ietf-emu-rfc5448bis: "Improved Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA)".

- [18] IETF RFC 3748: "Extensible Authentication Protocol (EAP)".
- [19] IETF RFC 4648: "The Base16, Base32 and Base64 Data Encodings".
- [20] 3GPP TS 24.501: "Non-Access-Stratum (NAS) protocol for 5G System (5GS); Stage 3".
- [21] 3GPP TR 21.900: "Technical Specification Group working methods".

3 Definitions and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions 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].

3.2 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].

| | |
|------|---|
| AMF | Access and Mobility Management Function |
| API | Application Programming Interface |
| AUSF | Authentication Server Function |
| MAC | Message Authentication Code |
| NF | Network Function |
| SEAF | SEcurity Anchor Function |
| SoR | Steering of Roaming |
| URI | Uniform Resource Identifier |
| UPU | UE Parameters Update |

4 Overview

4.1 Introduction

The Network Function (NF) Authentication Server Function (AUSF) is the network entity in the 5G Core Network (5GC) supporting the following functionalities:

- Authenticate the UE for the requester NF,
- Provide keying material to the requester NF,
- Protect the Steering Information List for the requester NF.

Figure 4-1 shows the reference architecture for the AUSF:

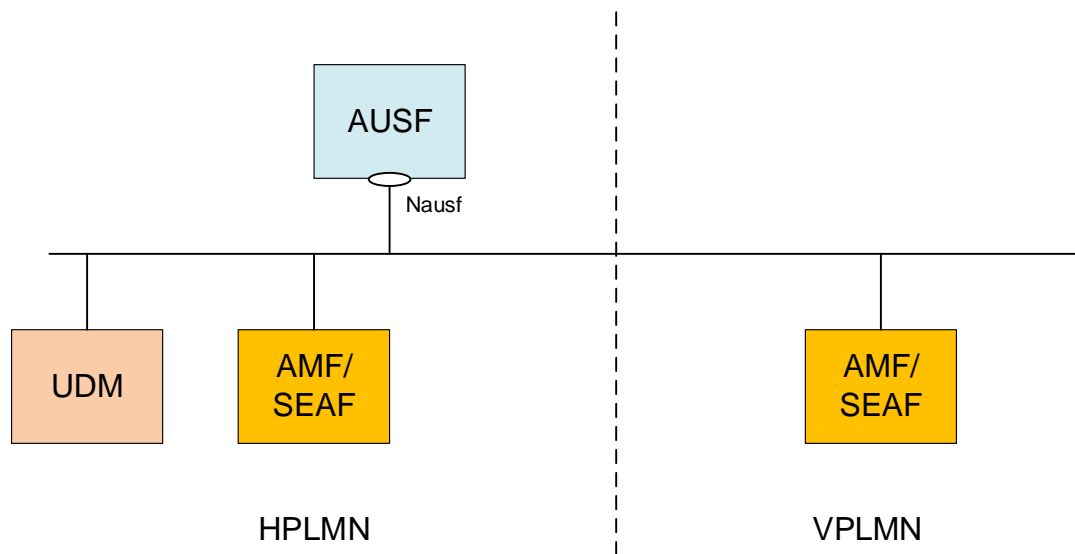


Figure 4-1: AUSF in 5G System architecture

This figure represents the AUSF architecture in the Service-based Architecture model. In the reference point model, the interface between the AMF and the AUSF is named N12. In this release, the SEAF function is collocated with the AMF. The AUSF may provide the service to the UDM.

5 Services offered by the AUSF

5.1 Introduction

The AUSF offers to NF Service Consumers (e.g. AMF) the following services:

- Nausf_UEAuthentication
- Nausf_SoRProtection
- Nausf_UPUProtection

5.2 Nausf_UEAuthentication Service

5.2.1 Service Description

The AUSF is acting as NF Service Producer. It provides UE authentication service to the requester NF. The NF Service Consumer is the AMF.

For this service, the following service operations are defined:

- Authenticate

This service permits to authenticate the UE and to provide one or more master keys which are used by the AMF to derive subsequent keys.

5.2.2 Service Operations

5.2.2.1 Introduction

The service operation defined for the Nausf_UEAuthentication is as follows:

- Authenticate: It allows the AMF to authenticate the UE.

5.2.2.2 Authenticate

5.2.2.2.1 General

The service operation "Authenticate" permits the requester NF to initiate the Authentication of the UE by providing the following information to the AUSF:

- UE id (e.g. SUPI)
- Serving Network Name

The AUSF retrieves the UE's subscribed authentication method from the UDM and depending on the information provided by the UDM, the AUSF enters in one of the following procedures:

- 5G-AKA
- EAP-based authentication'

For those two different procedures a new resource is generated by the AUSF. The content of the resource will depend on the procedure and will be returned to the AMF.

5.2.2.2.2 5G AKA

In this procedure, the NF Service Consumer (AMF) requests the authentication of the UE by providing UE related information and the serving network name and the 5G AKA is selected. The NF Service Consumer (AMF) shall then return to the AUSF the result received from the UE:

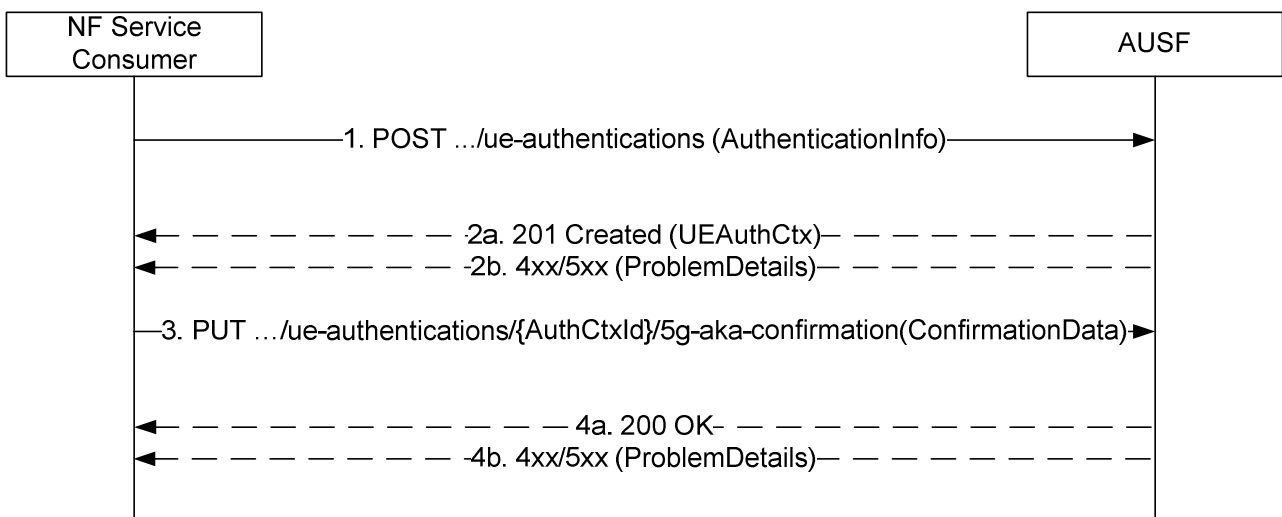


Figure 5.2.2.2.2-1: 5G AKA

1. The NF Service Consumer (AMF) shall send a POST request to the AUSF. The payload of the body shall contain at least the UE Id and the Serving Network Name.
- 2a. On success, "201 Created" shall be returned. The payload body shall contain the representation of the resource created and the "Location" header shall contain the URI of the created resource (e.g.

.../v1/ue_authentications/{authCtxId}). The AUSF generates a sub-resource "5g-aka-confirmation". The AUSF shall provide an hypermedia link towards this sub-resource in the payload to indicate to the AMF where it shall send a PUT for the confirmation.

- 2b. On failure, one of the HTTP status code listed in table 6.1.7.3-1 shall be returned with the message body containing a ProblemDetails structure with the "cause" attribute set to one of the application error listed in Table 6.1.7.3-1. If the serving network is not authorized, the AUSF shall use the SERVING_NETWORK_NOT_AUTHORIZED "cause".
3. Based on the relation type, the NF Service Consumer (AMF) deduces that it shall send a PUT containing the "RES*" provided by the UE to the URI provided by the AUSF or derived by itself. The NF Service Consumer (AMF) shall also send a PUT containing null value in the RES* to indicate the failure to the AUSF for the following cases:
 - if the UE is not reached, and the RES* is never received by the NF Service Consumer (AMF);
 - the comparison of the HRES* and HXRES* is unsuccessful in the NF Service Consumer (AMF);
 - the authentication failure is received from the UE, e.g. synchronization failure or MAC failure;
- 4a. On success, "200 OK" shall be returned. If the UE is not authenticated, e.g. the verification of the RES* was not successful in the AUSF, the AUSF shall set the value of AuthResult to AUTHENTICATION_FAILURE.
- 4b. On failure, one of the HTTP status code listed in table 6.1.7.3-1 shall be returned with the message body containing a ProblemDetails structure with the "cause" attribute set to one of the application error listed in Table 6.1.7.3-1.

5.2.2.2.3 EAP-based authentication method

5.2.2.2.3.1 General

In this procedure, the NF Service Consumer requests the authentication of the UE by providing UE related information and the serving network and the EAP-based authentication is selected (see IETF RFC 3748 [18]). EAP messages are exchanged between a UE acting as EAP peer, an NF Service Consumer (AMF/SEAF) acting as a pass-through authenticator and the AUSF acting as the EAP server.

5.2.2.2.3.2 EAP method: EAP-AKA'

EAP-AKA' is the EAP method used in this procedure

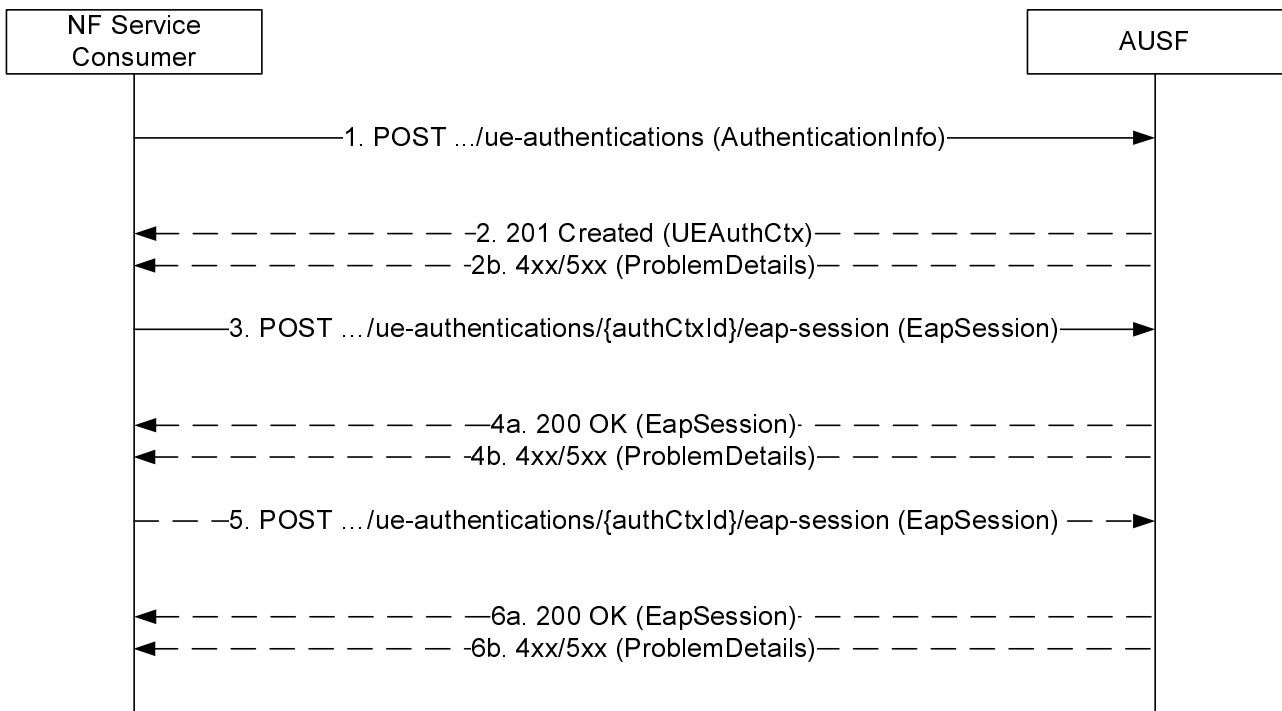


Figure 5.2.2.3-1: EAP-based authentication with EAP-AKA' method

1. The NF Service Consumer (AMF) shall send a POST request to the AUSF. The payload of the body shall contain at least the UE Id, Serving Network Name.
- 2a. On success, "201 Created" shall be returned. The payload body shall contain the representation of the resource generated and the "Location" header shall contain the URI of the generated resource (e.g. `.../v1/ue_authentications/{authCtxId}/eap-session`). The AUSF generates a sub-resource "eap-session". The AUSF shall provide an hypermedia link towards this sub-resource in the payload to indicate to the AMF where it shall send a POST containing the EAP packet response. The body payload shall also contain the EAP packet EAP-Request/AKA'-Challenge.
- 2b. On failure, one of the HTTP status code listed in table 6.1.7.3-1 shall be returned with the message body containing a ProblemDetails structure with the "cause" attribute set to one of the application error listed in Table 6.1.7.3-1. In particular, if the serving network is not authorized, the AUSF shall use the "Cause" `SERVING_NETWORK_NOT_AUTHORIZED`.
3. Based on the relation type, the NF Service Consumer (AMF) shall send a POST request including the EAP-Response/AKA' Challenge received from the UE. The POST request is sent to the URI provided by the AUSF or derived by the NF Service Consumer (AMF).
- 4a. On success, and if the AUSF and the UE have indicated the use of protected successful result indications as in IETF RFC 5448 [9] (to be superseded by draft-ietf-emu-rfc5448bis [17]), the AUSF shall reply with a "200 OK" HTTP message containing the EAP Request/AKA' Notification and an hypermedia link towards the sub-resource "eap-session".
- 4b. On failure, one of the HTTP status code listed in table 6.1.7.3-1 shall be returned with the message body containing a ProblemDetails structure with the "cause" attribute set to one of the application error listed in Table 6.1.7.3-1.

NOTE: Steps 4 to 5 are optional.

5. The NF Service Consumer (AMF) shall send a POST request including the EAP Response/AKA' Notification received from the UE. The POST request is sent to the URI provided by the AUSF or derived by the NF Service Consumer (AMF).

- 6a. If the EAP authentication exchange is successfully completed (with or without the optional Notification Request/Response messages exchange), "200 OK" shall be returned to the NF Service Consumer (AMF). The payload shall contain the result of the authentication, an EAP success/failure and the Kseaf if the authentication is successful. If the UE is not authenticated, the AUSF shall set the authResult to AUTHENTICATION_FAILURE.
- 6b. On failure, one of the HTTP status code listed in table 6.1.7.3-1 shall be returned with the message body containing a ProblemDetails structure with the "cause" attribute set to one of the application error listed in Table 6.1.7.3-1.

5.2.2.2.3.3 EAP method: EAP-TLS

The EAP-TLS method can be used in private networks as an EAP method (see 3GPP TS 33.501 [8] Annex B.1). The corresponding stage 3 implementation is described in Annex B.

5.3 Nausf_SoRProtection Service

5.3.1 Service Description

The AUSF is acting as NF Service Producer. It provides SoRProtection service to the NF Service Consumer.

This service permits to provide the NF Service Consumer (e.g. UDM) with the SoR-MAC-IAUSF and CounterSoR to protect the the Steering Information List from being tampered with or removed by the VPLMN.

NOTE: If the Steering Information List is not available or HPLMN determines that no steering of the UE is required, a SOR transparent container information element with an HPLMN indication that 'no change of the "Operator Controlled PLMN Selector with Access Technology" list stored in the UE' protected by SoR-MAC-IAUSF and CounterSoR is still sent to the UE during registration. The Steering Information List In such a case, the NF Service Consumer shall send an empty list to the AUSF when consuming the Nausf_SoRProtection Service.

In option this service also allows to provide the NF Service Consumer (e.g. UDM) with the SoR-XMAC-IUE that allows the NF Service Consumer (e.g. UDM) to verify that the UE received the Steering Information List.

5.3.2 Service Operations

5.3.2.1 Introduction

The service operation defined for the Nausf_SoRProtection is as follows:

- Protect

5.3.2.2 Protect

5.3.2.2.1 General

The Protect service operation is used in the following procedures:

- Procedure for steering of UE in VPLMN during registration (see clause 6.14.2.1 of 3GPP TS 33.501 [8]);
- Procedure for steering of UE in VPLMN after registration (see clause 6.14.2.2 of 3GPP TS 33.501 [8]).

The NF Service Consumer (e.g. UDM) uses this service operation to request the AUSF to compute the SoR-MAC-IAUSF and the CounterSoR by providing Steering Information List. The NF Service Consumer (e.g. UDM) may also request the AUSF to compute the SoR-XMAC-IUE by providing the indication that an acknowledgement is requested from the UE.

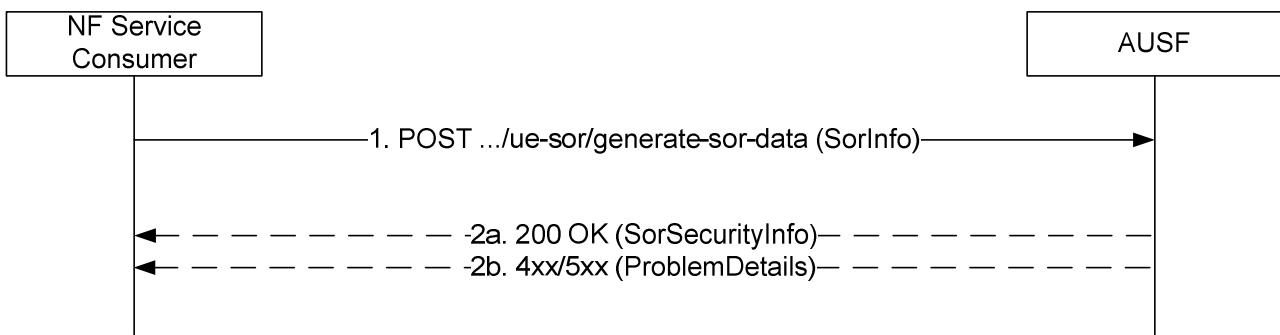


Figure 5.3.2.2.1-1: Steering of UE in VPLMN

1. The NF Service Consumer (e.g. UDM) shall send a POST request to the AUSF that was used to authenticate the UE. The payload of the body shall contain the Steering Information List and the acknowledge indication.
- 2a. On success, "200 OK" shall be returned. The payload body shall contain the requested security material necessary to protect the Steering of Roaming procedure.
- 2b. On failure, one of the HTTP status code listed in table 6.2.7.3-1 shall be returned with the message body containing a ProblemDetails structure with the "cause" attribute set to one of the application error listed in Table 6.2.7.3-1. If the Counter_{SOR} associated with the K_{AUSF} of the UE, is about to wrap around, the AUSF shall use the "COUNTER-WRAP" cause.

5.4 Nausf_UPUProtection Service

5.4.1 Service Description

The AUSF is acting as NF Service Producer. It provides UPUProtection service to the NF Service Consumer.

This service permits to provide the NF Service Consumer (e.g. UDM) with the UPU-MAC-I_{AUSF} and Counter_{UPU} to protect the UE Parameters Update Data from being tampered with or removed by the VPLMN.

In option this service also allows to provide the NF Service Consumer (e.g. UDM) with the UPU-XMAC-I_{UE} that allows the NF Service Consumer (e.g. UDM) to verify that the UE received UE Parameters Update Data correctly.

5.4.2 Service Operations

5.4.2.1 Introduction

The service operation defined for the Nausf_UPUProtection is as follows:

- Protect

5.4.2.2 Protect

5.4.2.2.1 General

The Protect service operation is used in the following procedures:

- Procedure for UE Parameters Update (see clause 6.15.2.1 of 3GPP TS 33.501 [8]).

The NF Service Consumer (e.g. UDM) uses this service operation to request the AUSF to compute the UPU-MAC-I_{AUSF} and Counter_{UPU} by providing the UE Parameters Update Data (UPU Data). The NF Service Consumer (e.g. UDM) may also request the AUSF to compute the UPU-XMAC-I_{UE} by providing the indication that an acknowledgement is requested from the UE.

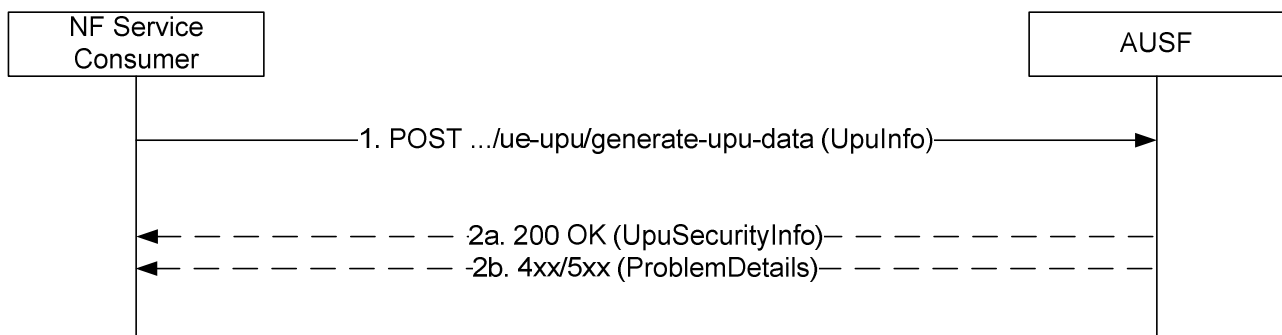


Figure 5.4.2.2-1: UE Parameters Update in VPLMN

1. The NF Service Consumer (e.g. UDM) shall send a POST request to the AUSF that was used to authenticate the UE. The payload of the body shall contain the UE Parameters Update Data (UPU Data) and the acknowledge indication.
- 2a. On success, "200 OK" shall be returned. The payload body shall contain the requested security material necessary to protect the UE Parameters Update procedure.
- 2b. On failure, one of the HTTP status code listed in table 6.4.7.3-1 shall be returned with the message body containing a ProblemDetails structure with the "cause" attribute set to one of the application error listed in Table 6.4.7.3-1. If the Counter_{UPU} associated with the K_{AUSF} of the UE, is about to wrap around, the AUSF shall use the "COUNTER-WRAP" cause.

6 API Definitions

6.1 Nausf_UEAuthentication Service API

6.1.1 API URI

URIs of this API shall have the following root:

```
{apiRoot}/{apiName}/{apiVersion}/
```

where "apiRoot" is defined in clause 4.4.1 of 3GPP TS 29.501 [5], the "apiName" shall be set to "nausf-auth" and the "apiVersion" shall be set to "v1" for the current version of this specification.

6.1.2 Usage of HTTP

6.1.2.1 General

HTTP/2, as defined in IETF RFC 7540 [6], shall be used as specified in clause 5 of 3GPP TS 29.500 [4].

6.1.2.2 HTTP standard headers

6.1.2.2.1 General

The usage of HTTP standard headers is specified in clause 5.2.2 of 3GPP TS 29.500 [4].

6.1.2.2.2 Content type

The following content types shall be supported:

- JSON, as defined in IETF RFC 8259 [7], shall be used as content type of the HTTP bodies specified in the present specification as indicated in clause 5.4 of 3GPP TS 29.500 [4].
- The Problem Details JSON Object (IETF RFC 7807 [11]). The use of the Problem Details JSON object in a HTTP response body shall be signalled by the content type "application/problem+json"
- The 3GPP hypermedia format as defined in 3GPP TS 29.501 [5]. The use of the 3GPP hypermedia format in a HTTP response body shall be signalled by the content type "application/3gppHal+json"

6.1.2.3 HTTP custom headers

6.1.2.3.1 General

The usage of HTTP custom headers shall be supported as specified in clause 5.2.3 of 3GPP TS 29.500 [4].

6.1.3 Resources

6.1.3.1 Overview

The structure of the Resource URIs of the "Authenticate" service is shown in Figure 6.1.3.1-1

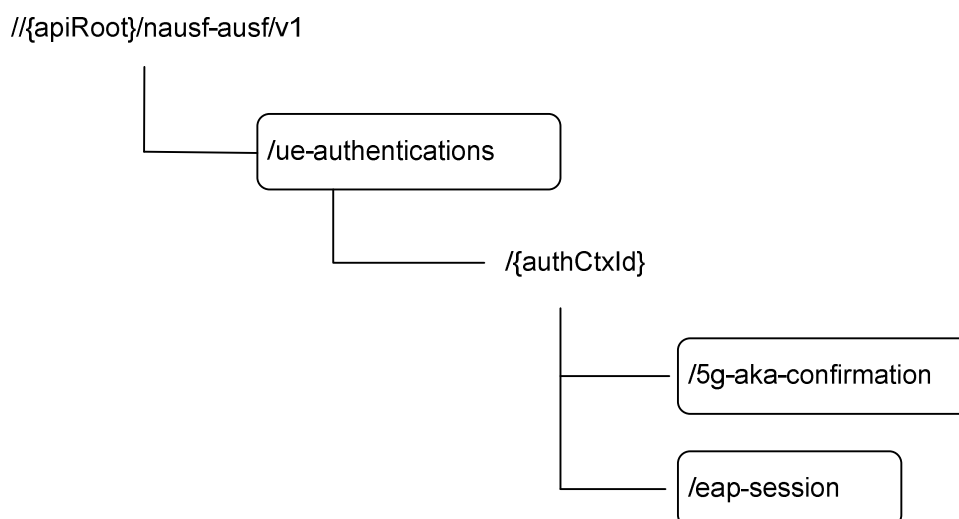


Figure 6.1.3.1-1: Resource URI structure of the AUSF API

Table 6.1.3.1-1 provides an overview of the resources and applicable HTTP methods.

Table 6.1.3.1-1: Resources and methods overview

| Resource name | Resource URI | HTTP method or custom operation | Description |
|--|--|---------------------------------|---|
| ue-authentications (Collection) | {apiRoot}/nausf-auth/v1/ue-authentications | POST | Initiate the authentication process by providing inputs related to the UE |
| 5g-aka-confirmation (Document) | {apiRoot}/nausf-auth/v1/ue-authentications/{authCtxId}/5g-aka-confirmation | PUT | Put the UE response from the 5G-AKA process. |
| eap-session (Document) | {apiRoot}/nausf-auth/v1/ue-authentications/{authCtxId}/eap-session | POST | Post the EAP response from the UE. See NOTE. |
| NOTE: This POST is used to provide EAP response to the AUSF in a sub-resource (Document) generated by the first POST operation. As this operation is not idempotent (it triggers subsequent EAP operations), a PUT was not adequate. | | | |

6.1.3.2 Resource: List of ue-authentications

6.1.3.2.1 Description

This resource represents a collection of the ue-authentication resources generated by the AUSF.

6.1.3.2.2 Resource Definition

Resource URI: **{apiRoot}/nausf-auth/v1/ue-authentications**

This resource shall support the resource URI variables defined in table 6.1.3.2.2-1.

Table 6.1.3.2.2-1: Resource URI variables for this resource

| Name | Definition |
|---------|------------------|
| apiRoot | See clause 6.1.1 |

6.1.3.2.3 Resource Standard Methods

6.1.3.2.3.1 POST

This method shall support the URI query parameters specified in table 6.1.3.2.3.1-1.

Table 6.1.3.2.3.1-1: URI query parameters supported by the POST method on this resource

| Name | Data type | P | Cardinality | Description |
|------|-----------|---|-------------|-------------|
| n/a | | | | |

This method shall support the request data structures specified in table 6.1.3.2.3.1-2 and the response data structures and response codes specified in table 6.1.3.2.3.1-3.

Table 6.1.3.2.3.1-2: Data structures supported by the POST Request Body on this resource

| Data type | P | Cardinality | Description |
|--------------------|---|-------------|---|
| AuthenticationInfo | M | 1 | Contains the UE id (i.e. SUCI or SUPI as specified in 3GPP TS 33.501 [8]) and the serving network name. It may also contain Trace Data as specified in 3GPP TS 23.501 [2]. |

Table 6.1.3.2.3.1-3: Data structures supported by the POST Response Body on this resource

| Data type | P | Cardinality | Response codes | Description |
|---|---|-------------|---------------------------|---|
| UEAuthentication Ctx | M | 1 | 201 Created | Upon success, if 5G AKA is selected, the response body will contain one AV and "link" for the AMF to PUT the confirmation. If an EAP-based method is selected, the response body will contain the EAP method selected, the corresponding EAP packet request and a "link" for the AMF to POST the EAP response. The HTTP response shall include a "Location" header that contains the resource URI of the created resource. |
| ProblemDetails | M | 1 | 400 Bad Request | This case represents the failure to start authentication service because of input parameter error. |
| ProblemDetails | M | 1 | 403 Forbidden | This case represents when the UE is not allowed to be authenticated. The "cause" attribute can be set to one of the following application errors: - AUTHENTICATION_REJECTED - SERVING_NETWORK_NOT_AUTHORIZED - INVALID_HN_PUBLIC_KEY_IDENTIFIER - INVALID_SCHEME_OUTPUT |
| ProblemDetails | M | 1 | 404 Not Found | The "cause" attribute can be set to the following application error: - USER_NOT_FOUND |
| ProblemDetails | M | 1 | 500 Internal Server Error | This case represents the failure in starting the authentication service because of a server internal error. If the error is due to a problem with UDM not able to generate the requested AV, the AUSF shall indicate the following application error: "AV_GENERATION_PROBLEM" |
| ProblemDetails | M | 1 | 501 Not Implemented | The "cause" attribute can be set to the following application error: - UNSUPPORTED_PROTECTION_SCHEME |
| NOTE: The mandatory HTTP error status codes for the POST method listed in Table 5.2.7.1-1 of 3GPP TS 29.500 [4] other than those specified in the table above also apply, with a ProblemDetails data type (see clause 5.2.7 of 3GPP TS 29.500 [4]). | | | | |

6.1.3.2.4 Resource Custom Operations

6.1.3.2.4.1 Overview

There is no Resource Custom Operations in the current version of this API.

6.1.3.3 Resource: 5g-aka-confirmation (Document)

6.1.3.3.1 Description

The subresource "5g-aka-confirmation" is generated by the AUSF. This subresource should not persist after the AUSF has read its content.

6.1.3.3.2 Resource Definition

Resource URI: **{apiRoot}/nausf-auth/v1/ue-authentications/{authCtxId}/5g-aka-confirmation**

This resource shall support the resource URI variables defined in table 6.1.3.2.2-1.

Table 6.1.3.3.2-1: Resource URI variables for this resource

| Name | Definition |
|-----------|---|
| apiRoot | See clause 6.1.1 |
| authCtxId | Represents a specific ue-authentication |

6.1.3.3.3 Resource Standard Methods

6.1.3.3.3.1 PUT

This method shall support the URI query parameters specified in table 6.1.3.2.3.1-1.

Table 6.1.3.2.3.1-1: URI query parameters supported by the PUT method on this resource

| Name | Data type | P | Cardinality | Description |
|------|-----------|---|-------------|-------------|
| n/a | | | | |

This method shall support the request data structures specified in table 6.1.3.2.3.1-2 and the response data structures and response codes specified in table 6.1.3.2.3.1-3.

Table 6.1.3.3.3.1-2: Data structures supported by the PUT Request Body on this resource

| Data type | P | Cardinality | Description |
|------------------|---|-------------|--|
| ConfirmationData | M | 1 | Contains the "RES*" generated by the UE and provided to the AMF. |

Table 6.1.3.3.3.1-3: Data structures supported by the PUT Response Body on this resource

| Data type | P | Cardinality | Response codes | Description |
|--|---|-------------|---------------------------|--|
| ConfirmationData Response | M | 1 | 200 OK | This case indicates that the AUSF has performed the verification of the 5G AKA confirmation. The response body shall contain the result of the authentication and the Kseaf if the authentication is successful. |
| ProblemDetails | M | 1 | 400 Bad Request | This case represents a 5G AKA confirmation failure because of input parameter error. This indicates that the AUSF was not able to confirm the authentication. |
| ProblemDetails | M | 1 | 500 Internal Server Error | This case represents a 5G AKA confirmation failure because of a server internal error. |
| NOTE: The mandatory HTTP error status codes for the PUT method listed in Table 5.2.7.1-1 of 3GPP TS 29.500 [4] other than those specified in the table above also apply, with a ProblemDetails data type (see clause 5.2.7 of 3GPP TS 29.500 [4]). | | | | |

6.1.3.4 Resource: eap-session (Document)

6.1.3.4.1 Description

The "eap-session" is generated by the AUSF if an EAP-based authentication method is selected. This resource is used to handle the EAP session. This subresource should not persist after the EAP exchanges.

6.1.3.4.2 Resource Definition

Resource URI: {apiRoot}/nausf-auth/v1/ue-authentications/{authCtxId}/eap-session

This resource shall support the resource URI variables defined in table 6.1.3.4.2-1.

Table 6.1.3.4.2-1: Resource URI variables for this resource

| Name | Definition |
|-----------|---|
| apiRoot | See clause 6.1.1 |
| authCtxId | Represents a specific ue-authentication |

6.1.3.4.3 Resource Standard Methods

6.1.3.4.3.1 POST

This method shall support the URI query parameters specified in table 6.1.3.4.3.1-1.

Table 6.1.3.4.3.1-1: URI query parameters supported by the POST method on this resource

| Name | Data type | P | Cardinality | Description |
|------|-----------|---|-------------|-------------|
| n/a | | | | |

This method shall support the request data structures specified in table 6.1.3.4.3.1-2 and the response data structures and response codes specified in table 6.1.3.4.3.1-3.

Table 6.1.3.4.3.1-2: Data structures supported by the POST Request Body on this resource

| Data type | P | Cardinality | Description |
|------------|---|-------------|--|
| EapSession | M | 1 | Contains the EAP packet response (see IETF RFC 3748 [18]) from the UE and transferred by the AMF |

Table 6.1.3.4.3.1-3: Data structures supported by the POST Response Body on this resource

| Data type | P | Cardinality | Response codes | Description |
|---|---|-------------|---------------------------|---|
| EapSession | M | 1 | 200 OK | During an EAP session, the body response shall contain the EAP packet Response and an hypermedia link. At the end of the EAP session, the body response shall contain the EAP packet Success or Failure (see IETF RFC 3748 [18]) and the Kseaf if the authentication is successful |
| ProblemDetails | M | 1 | 400 Bad Request | This case represents an EAP session failure because of input parameter error. This indicates that the AUSF was not able to continue the EAP session. |
| ProblemDetails | M | 1 | 500 Internal Server Error | This case represents an EAP session failure failure because of a server internal error. |
| NOTE: The mandatory HTTP error status codes for the POST method listed in Table 5.2.7.1-1 of 3GPP TS 29.500 [4] other than those specified in the table above also apply, with a ProblemDetails data type (see clause 5.2.7 of 3GPP TS 29.500 [4]). | | | | |

6.1.4 Custom Operations without associated resources

6.1.4.1 Overview

There is no Custom Operation in the current version of this API.

6.1.5 Notifications

6.1.5.1 General

There is no use of notification in the current version of this API.

6.1.6 Data Model

6.1.6.1 General

This clause specifies the application data model supported by the API.

Table 6.1.6.1-1 specifies the data types defined for the Nausf service based interface protocol.

Table 6.1.6.1-1: Nausf specific Data Types

| Data type | Clause defined | Description |
|---------------------|----------------|---|
| AuthenticationInfo | 6.1.6.2.2 | Contains the UE id (i.e. SUCI or SUPI) and the Serving Network Name. |
| UEAuthenticationCtx | 6.1.6.2.3 | Contains the information related to the resource generated to handle the UE authentication. It contains at least the UE id, Serving Network, the Authentication Method and related EAP information or related 5G-AKA information. |
| 5gAuthData | 6.1.6.2.4 | Contains 5G authentication related information. |
| AV5gAka | 6.1.6.2.5 | Contains Authentication Vector for method 5G AKA. |
| ConfirmationData | 6.1.6.2.7 | Contains the "RES*" generated by the UE. |
| EapSession | 6.1.6.2.8 | Contains information related to the EAP session. |

Table 6.1.6.1-2 specifies data types re-used by the Nausf service based interface protocol from other specifications, including a reference to their respective specifications and when needed, a short description of their use within the Nausf service based interface.

Table 6.1.6.1-2: Nausf re-used Data Types

| Data type | Reference | Comments |
|-----------------------|---------------------|--|
| LinksValueSchema | 3GPP TS 29.571 [10] | 3GPP Hypermedia link |
| ProblemDetails | 3GPP TS 29.571 [10] | Common Data Type used in response bodies |
| Supi | 3GPP TS 29.571 [10] | |
| Uri | 3GPP TS 29.571 [10] | |
| ResynchronizationInfo | 3GPP TS 29.503 [12] | |
| SupiOrSuci | 3GPP TS 29.503 [12] | |
| ServingNetworkName | 3GPP TS 29.503 [12] | |
| Autn | 3GPP TS 29.503 [12] | |
| Pei | 3GPP TS 29.571 [10] | |
| TraceData | 3GPP TS 29.571 [10] | |

6.1.6.2 Structured data types

6.1.6.2.1 Introduction

The following clause defines the structures to be used in resource representations.

6.1.6.2.2 Type: AuthenticationInfo

Table 6.1.6.2.2-1: Definition of type AuthenticationInfo

| Attribute name | Data type | P | Cardinality | Description |
|-----------------------|-----------------------|---|-------------|--|
| supiOrSuci | SupiOrSuci | M | 1 | Contains the SUPI or SUCI of the UE. |
| servingNetworkName | ServingNetworkName | M | 1 | Contains the Serving Network Name. |
| resynchronizationInfo | ResynchronizationInfo | O | 0..1 | Contains RAND and AUTS; see 3GPP TS 29.501 [8] clause 9.4. |
| pei | Pei | O | 0..1 | Permanent Equipment Identifier |
| traceData | TraceData | O | 0..1 | Contains TraceData provided by the UDM to the AMF |

6.1.6.2.3 Type: UEAuthenticationCtx

Table 6.1.6.2.3-1: Definition of type UEAuthenticationCtx

| Attribute name | Data type | P | Cardinality | Description |
|--------------------|-----------------------|---|-------------|---|
| authType | AuthType | M | 1 | Indicates the authentication method used for this UE ie. "5G-AKA-Confirmation", "EAP-AKA" or "EAP-TLS". See clause 6.1.6.3.3 |
| _links | map(LinksValueSchema) | M | 1..N | If 5G-AKA has been selected, this IE shall contain a member whose name is set to "5g-aka" and the URI to perform the confirmation. If an EAP-based method has been selected, this IE shall contain a member whose name is set to "eap-session" and the URI to perform the EAP session. See NOTE |
| 5gAuthData | 5GAuthData | M | 1 | Contains either 5G-AKA or EAP related information. |
| servingNetworkName | ServingNetworkName | O | 0..1 | Contains the Serving Network Name. |

NOTE: In the current version of this API, only one hypermedia link is provided

6.1.6.2.4 Type: 5gAuthData

Table 6.1.6.2.4-1: Definition of type 5gAuthData as a list of mutually exclusive alternatives

| Data type | Cardinality | Description |
|------------|-------------|---|
| Av5gAka | 1 | Contains the 5G AV if 5G-AKA has been selected. |
| EapPayload | 1 | Contains the EAP packet request. |

6.1.6.2.5 Type: Av5gAka

Table 6.1.6.2.5-1: Definition of type Av5gAka

| Attribute name | Data type | P | Cardinality | Description |
|----------------|-----------|---|-------------|-------------|
| rand | Rand | M | 1 | |
| autn | Autn | M | 1 | |
| hxresStar | HxresStar | M | 1 | |

6.1.6.2.6 Type: ConfirmationData

Table 6.1.6.2.6-1: Definition of type ConfirmationData

| Attribute name | Data type | P | Cardinality | Description |
|----------------|-----------|---|-------------|---|
| resStar | ResStar | M | 1 | Contains the "RES*" provided by the UE to the AMF. If no RES* has been provided by the UE the null value is conveyed to the AUSF. |

6.1.6.2.7 Type: EapSession

Table 6.1.6.2.7-1: Definition of type EapSession

| Attribute name | Data type | P | Cardinality | Description |
|----------------|-----------------------|---|-------------|---|
| eapPayload | EapPayload | M | 1 | Contains the EAP packet (see IETF RFC 3748 [18]). |
| kSeaf | Kseaf | C | 0..1 | If the authentication is successful, the Kseaf shall be included |
| _links | map(LinksValueSchema) | C | 1..N | If the EAP session requires another exchange e.g. for EAP-AKA' notification, this IE shall contain a member whose name is "eap-session" and the URI to continue the EAP session. See NOTE. |
| authResult | AuthResult | C | 0..1 | Indicates the result of the authentication. |
| supi | Supi | C | 0..1 | If the authentication is successful and if the AMF had provided a SUCI, this IE shall contain the SUPI of the UE. |

NOTE: In the current version of this API, only 0 or 1 hypermedia link is provided.

6.1.6.2.8 Type: ConfirmationDataResponse

Table 6.1.6.2.8-1: Definition of type ConfirmationDataResponse

| Attribute name | Data type | P | Cardinality | Description |
|----------------|------------|---|-------------|--|
| authResult | AuthResult | M | 1 | Indicates the result of the authentication |
| supi | Supi | C | 0..1 | If the authentication is successful and if the AMF had provided a SUCI, this IE shall contain the SUPI of the UE |
| kseaf | Kseaf | C | 0..1 | Contains the Kseaf if authentication is successful. |

6.1.6.3 Simple data types and enumerations

6.1.6.3.1 Introduction

This clause defines simple data types and enumerations that can be referenced from data structures defined in the previous clauses.

6.1.6.3.2 Simple data types

Table 6.1.6.3.2-1: Simple data types

| Type Name | Type Definition | Description |
|------------|-----------------|--|
| EapPayload | string | The EAP packet is encoded using base64 (see IETF RFC 4648 [19]) and represented as a String. |
| ResStar | string | pattern: "[A-Fa-f0-9]{32}"; nullable |
| Kseaf | string | pattern: "[A-Fa-f0-9]{64}" |
| HxresStar | string | pattern: "[A-Fa-f0-9]{32}" |

6.1.6.3.3 Enumeration: AuthType

Table 6.1.6.3.3-1: Enumeration AuthType

| Enumeration value | Description |
|-------------------|--|
| 5G_AKA | 5G AKA |
| EAP_AKA_PRIME | EAP-AKA' |
| EAP_TLS | EAP-TLS is only used in the case where the Annex B is supported. |

6.1.6.3.4 Enumeration: AuthResult

Table 6.1.6.3.4-1: Enumeration AuthResult

| Enumeration value | Description |
|------------------------|---|
| AUTHENTICATION_SUCCESS | This value is used to indicate that the AUSF successfully authenticate the UE |
| AUTHENTICATION_FAILURE | This value is used to indicate that the AUSF fails to authenticate the UE. |
| AUTHENTICATION_ONGOING | This value is used during an EAP Session to indicate that the EAP session is still ongoing. |

6.1.6.3.5 Relation Types

6.1.6.3.5.1 General

This clause describes the possible relation types defined within AUSF API.

Table 6.1.6.3.5-1: supported registered relation types

| Relation Name |
|---------------|
| 5g-aka |
| eap-session |

6.1.6.3.5.2 The "5g-aka" Link relation

The value "5g-aka" specifies that the value of the href attribute is the URI where NF Service Consumer shall send a PUT containing the result "RES*" received from the UE.

6.1.6.3.5.3 The "eap-session" Link relation

The value "eap-session" specifies that the value of the href attribute is the URI that will be used by the NF Service Consumer to provide EAP packet response during an EAP exchange. The NF Service Consumer shall use a POST to provide the EAP Packet Response to the AUSF to the corresponding URI.

6.1.6.4 Binary data

6.1.6.4.1 Introduction

There is no binary data in the current version of this API.

6.1.7 Error Handling

6.1.7.1 General

HTTP error handling shall be supported as specified in clause 5.2.4 of 3GPP TS 29.500 [4].

6.1.7.2 Protocol Errors

Protocol errors shall be supported as specified in clause 5.2.7 of 3GPP TS 29.500 [4].

6.1.7.3 Application Errors

The common application errors defined in the Table 5.2.7.2-1 in 3GPP TS 29.500 [4] may also be used for the Nausf_UEauthentication service. The following application errors listed in Table 6.1.7.3-1 are specific for the Nausf_UEauthentication service.

Table 6.1.7.3-1: Application errors

| Application Error | HTTP status code | Description |
|----------------------------------|---------------------------|--|
| SERVING_NETWORK_NOT_AUTHORIZED | 403 Forbidden | The serving network is not authorized, e.g. serving PLMN. |
| AUTHENTICATION_REJECTED | 403 Forbidden | The user cannot be authenticated with this authentication method e.g. only SIM data available |
| INVALID_HN_PUBLIC_KEY_IDENTIFIER | 403 Forbidden | Invalid HN public key identifier received |
| INVALID_SCHEME_OUTPUT | 403 Forbidden | SUCI cannot be decrypted with received data |
| CONTEXT_NOT_FOUND | 404 Not Found | The AUSF cannot found the resource corresponding to the URI provided by the NF Service Consumer. |
| USER_NOT_FOUND | 404 Not Found | The user does not exist in the HPLMN |
| UPSTREAM_SERVER_ERROR | 504 Gateway Timeout | No response is received from a remote peer, e.g. from the UDM |
| NETWORK_FAILURE | 504 Gateway Timeout | The request is rejected due to a network problem. |
| AV_GENERATION_PROBLEM | 500 Internal Server Error | The UDM has indicated that it was not able to generate AV. |
| UNSUPPORTED_PROTECTION_SCHEME | 501 Not implemented | The received protection scheme is not supported by HPLMN |

6.1.8 Security

As indicated in 3GPP TS 33.501 [8], the access to the Nausf_UEAuthentication Service API may be authorized by means of the OAuth2 protocol (see IETF RFC 6749 [13]), using the "Client Credentials" authorization grant, where the NRF (see 3GPP TS 29.510 [14]) plays the role of the authorization server.

If OAuth2 is used, an NF Service Consumer, prior to consuming service offered by the Nausf_UEAuthentication Service API, shall obtain a "token" from the authorization server, by invoking the Access Token Request service, as described in 3GPP TS 29.510 [14], clause 5.4.2.2.

NOTE: When multiple NRFs are deployed in a network, the NRF used as authorization server is the same NRF that the NF Service Consumer used for discovering the Nausf_UEAuthentication service.

The Nausf_UEAuthentication Service API does not define any scopes for OAuth2 authorization as specified in 3GPP TS 33.501 [8]; it defines a single scope consisting on the name of the service (i.e., "nausf-auth"), and it does not define any additional scopes at resource or operation level.

6.2 Nausf_SoRProtection Service API

6.2.1 API URI

URIs of this API shall have the following root:

```
{apiRoot}/{apiName}/{apiVersion}/
```

where "apiRoot" is defined in clause 4.4.1 of 3GPP TS 29.501 [5], the "apiName" shall be set to "nausf-sorprotection" and the "apiVersion" shall be set to "v1" for the current version of this specification.

6.2.2 Usage of HTTP

6.2.2.1 General

HTTP/2, as defined in IETF RFC 7540 [6], shall be used as specified in clause 5 of 3GPP TS 29.500 [4].

6.2.2.2 HTTP standard headers

6.2.2.2.1 General

The usage of HTTP standard headers is specified in clause 5.2.2 of 3GPP TS 29.500 [4].

6.2.2.2.2 Content type

The following content types shall be supported:

- JSON, as defined in IETF RFC 8259 [7], shall be used as content type of the HTTP bodies specified in the present specification as indicated in clause 5.4 of 3GPP TS 29.500 [4].
- The Problem Details JSON Object (IETF RFC 7807 [11]). The use of the Problem Details JSON object in a HTTP response body shall be signalled by the content type "application/problem+json"

6.2.2.3 HTTP custom headers

6.2.2.3.1 General

In this version of the API, no specific custom headers are defined for the "Nausf_SoRProtection" service.

For 3GPP specific HTTP custom headers used across all service based interfaces, see clause 5.2.3 of 3GPP TS 29.500 [4].

6.2.3 Resources

6.2.3.1 Overview

The structure of the Resource URIs of the SoRProtection service is shown in Figure 6.2.3.1-1

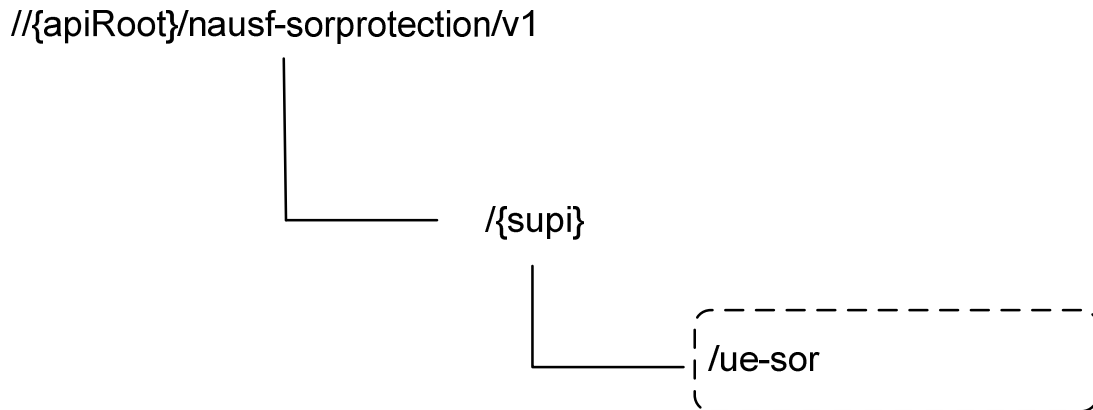


Figure 6.x.3.1-1: Resource URI structure of the SoRProtection API

Table 6.2.3.1-1 provides an overview of the resources and applicable HTTP methods.

Table 6.2.3.1-1: Resources and methods overview

| Resource name | Resource URI | HTTP method or custom operation | Description |
|------------------------------|--|---------------------------------|--|
| ue-sor (Custom operation) | {apiRoot}/nausf-sorprotection/v1/{supi}/ue-sor/generate-sor-data | generate-sor-data (POST) | Resource for SoR security material computation |

6.2.3.2 Resource: ue-sor

6.2.3.2.1 Description

It is the resource to which the custom operation used to generate the SoR security material is associated with.

6.2.3.2.2 Resource Definition

Resource URI: {apiRoot}/nausf-sorprotection/v1/supi/ue-sor

This resource shall support the resource URI variables defined in table 6.2.3.2.2-1.

Table 6.2.3.2.2-1: Resource URI variables for this resource

| Name | Definition |
|---------|---|
| apiRoot | See clause 6.2.1 |
| supi | Represents the Subscription Permanent Identifier (see 3GPP TS 23.501 [2] clause 5.9.2) pattern: '^(\imsi-[0-9]{5,15} nai-.+ .+)\$\$' |

6.2.3.2.3 Resource Standard Methods

No Standard Methods are supported for this resource.

6.2.3.2.4 Resource Custom Operations

6.2.3.2.4.1 Overview

Table 6.2.3.2.4.1-1: Custom operations

| Custom operation URI | Mapped HTTP method | Description |
|----------------------|--------------------|---|
| /generate-sor-data | POST | The AUSF calculates the SoR-MAC-IAUSF and the CounterSoR to protect the Steering Information List provided. It may also calculate the SoR-XMAC-IUE to verify that the UE received the Steering Information List if the indication that an acknowledgement is requested from the UE. |

6.2.3.2.4.2 Operation: generate-sor-data

6.2.3.2.4.2.1 Description

This custom operation is used by the NF service consumer (e.g. UDM) to request the AUSF to compute the security material (SoR-MAC-IAUSF, CounterSoR and SoR-XMAC-IUE) needed to ensure the protection of the SoR procedure (see 3GPP TS 33.501 [8]).

6.2.3.2.4.2.2 Operation Definition

This method shall support the request data structures specified in table 6.2.3.2.4.2.2-1 and the response data structures and response codes specified in table 6.2.3.2.4.2.2-2.

Table 6.2.3.2.4.2.2-1: Data structures supported by the POST Request Body on this resource

| Data type | P | Cardinality | Description |
|-----------|---|-------------|--|
| SorInfo | M | 1 | Contains the Steering Information List and may contain the indication that an acknowledgement is requested from the UE (as specified in 3GPP TS 33.501 [8]). |

Table 6.2.3.2.4.2.2-2: Data structures supported by the POST Response Body on this resource

| Data type | P | Cardinality | Response codes | Description |
|---|---|-------------|-------------------------|---|
| SorSecurityInfo | M | 1 | 200 OK | Upon success, the response body will contain SoR-MAC-IAUSF and CounterSoR and may contain the SoR-XMAC-IUE. |
| ProblemDetails | M | 1 | 503 Service Unavailable | The "cause" attribute shall be set to one of the following application error: - COUNTER_WRAP See table 6.2.7.3-1 for the description of these errors. |
| NOTE: The mandatory HTTP error status codes for the POST method listed in Table 5.2.7.1-1 of 3GPP TS 29.500 [4] other than those specified in the table above also apply, with a ProblemDetails data type (see clause 5.2.7 of 3GPP TS 29.500 [4]). | | | | |

6.2.4 Custom Operations without associated resources

6.2.4.1 Overview

There is no Custom Operation in the current version of this API.

6.2.5 Notifications

6.2.5.1 General

There is no use of notification in the current version of this API.

6.2.6 Data Model

6.2.6.1 General

This clause specifies the application data model supported by the API.

Table 6.2.6.1-1 specifies the data types defined for the Nausf-SORProtection service based interface protocol.

Table 6.x.6.1-1: Nausf specific Data Types

| Data type | Clause defined | Description |
|-----------------|----------------|---|
| SorInfo | 6.2.6.2.2 | Contains the Steering Information |
| SorSecurityInfo | 6.2.6.2.3 | Contains the material generated for securing of SoR. It contains at least the SoR-MAC-IAUSF and CounterSoR. |
| SteeringInfo | 6.2.6.2.4 | Contains a combination of one PLMN identity and zero or more access technologies. |
| SorMac | 6.2.6.3.2 | MAC value for protecting SOR procedure (SoR-MAC-IAUSF and SoR-XMAC-IUE) |
| Countersor | 6.2.6.3.2 | CounterSoR |
| AckInd | 6.2.6.3.2 | Contains indication whether the acknowledgement from UE is needed |
| AccessTech | 6.2.6.3.3 | Access Technology |

Table 6.2.6.1-2 specifies data types re-used by the Nausf-SORProtection service based interface protocol from other specifications, including a reference to their respective specifications and when needed, a short description of their use within the Nausf service based interface.

Table 6.2.6.1-2: Nausf re-used Data Types

| Data type | Reference | Comments |
|-----------|---------------------|----------|
| PlmnId | 3GPP TS 29.571 [10] | PLMN ID |

6.2.6.2 Structured data types

6.2.6.2.1 Introduction

The following clauses define the structures to be used in resource representations.

6.2.6.2.2 Type: SorInfo

Table 6.2.6.2.2-1: Definition of type SorInfo

| Attribute name | Data type | P | Cardinality | Description |
|-------------------|-------------------|---|-------------|---|
| ackInd | AckInd | M | 1 | Contains the indication whether the acknowledgement from UE is needed. |
| steeringContainer | SteeringContainer | C | 1 | When present, this information contains the information needed to update the "Operator Controlled PLMN Selector with Access Technology" list stored in the USIM. It may contain an array of preferred PLMN/AccessTechnologies combinations in priority order. The first entry in the array indicates the highest priority and the last entry indicates the lowest. Or it may contain a secured packet. If no change of the "Operator Controlled PLMN Selector with Access Technology" list stored in the USIM is needed then this attribute shall be absent. |

6.2.6.2.3 Type: SorSecurityInfo

Table 6..6.2.3-1: Definition of type SorSecurityInfo

| Attribute name | Data type | P | Cardinality | Description |
|----------------|------------|---|-------------|--|
| sorMaclausf | SorMac | M | 1 | Contains the SoR-MAC-IAUSF. |
| countersor | CounterSor | M | 1 | Contains the Counter _{SoR} . |
| sorXmaclue | SorMac | O | 0..1 | When present, contains the SoR-XMAC-I _{UE} . It shall be included, if the UDM requests the acknowledgement from the UE. |

6.2.6.2.4 Type: SteeringInfo

Table 6..6.2.4-1: Definition of type SteeringInfo

| Attribute name | Data type | P | Cardinality | Description |
|----------------|-------------------|---|-------------|---|
| plmnlid | Plmnlid | M | 1 | Contains a preferred PLMN identity. |
| accessTechList | Array(AccessTech) | C | 1..N | When present it contains the 31 referred access technologies as listed in clause 4.2.5 of 3GPP TS 31.102 [15]. If absent it means that all access technologies are equivalently preferred in this PLMN. |

6.2.6.2.5 Type: SteeringContainer

Table 6.2.6.2.5.-1: Definition of type SteeringContainer as a list of alternatives

| Data type | Cardinality | Description |
|---------------------|-------------|---|
| array(SteeringInfo) | 1..N | List of PLMN/AccessTechnologies combinations. |
| SecuredPacket | 1 | A secured packet containing one or more APDUs commands dedicated to Remote File Management. |

6.2.6.3 Simple data types and enumerations

6.2.6.3.1 Introduction

This clause defines simple data types and enumerations that can be referenced from data structures defined in the previous clauses.

6.2.6.3.2 Simple data types

Table 6.2.6.3.2-1: Simple data types

| Type Name | Type Definition | Description |
|---------------|-----------------|---|
| SorMac | string | pattern: "[A-Fa-f0-9]{32}\$" |
| CounterSor | string | pattern: "[A-Fa-f0-9]{4}\$" |
| AckInd | boolean | true indicates that the SoR-XMAC-IUE shall be computed and returned in the response |
| SecuredPacket | string | Contains a secure packet as specified in 3GPP TS 24.501 [20]. It is encoded using base64 and represented as a String. |

6.2.6.3.3 Enumeration: AccessTech

Table 6.2.6.3.3-1: Enumeration AccessTech

| Enumeration value | Description |
|-------------------------------------|-------------|
| "NR" | |
| "EUTRAN_IN_WBS1_MODE_AND_NBS1_MODE" | |
| "EUTRAN_IN_NBS1_MODE_ONLY" | |
| "EUTRAN_IN_WBS1_MODE_ONLY" | |
| "UTRAN" | |
| "GSM_AND_ECGSM_IoT" | |
| "GSM_WITHOUT_ECGSM_IoT" | |
| "ECGSM_IoT_ONLY" | |
| "CDMA_1xRTT" | |
| "CDMA_HRPD" | |
| "GSM_COMPACT" | |

6.2.7 Error Handling

6.2.7.1 General

HTTP error handling shall be supported as specified in clause 5.2.4 of 3GPP TS 29.500 [4].

6.2.7.2 Protocol Errors

Protocol Error Handling shall be supported as specified in clause 5.2.7.2 of 3GPP TS 29.500 [4].

6.2.7.3 Application Errors

The common application errors defined in the Table 5.2.7.2-1 in 3GPP TS 29.500 [4] may also be used for the Nausf_SoRProtection service. The following application errors listed in Table 6.7.3-1 are specific for the Nausf_SoRProtection service.

Table 6.2.7.3-1: Application errors

| Application Error | HTTP status code | Description |
|-------------------|-------------------------|---|
| COUNTER_WRAP | 503 Service Unavailable | The Counter _{SoR} associated with the KAUSF of the UE is about to wrap around. The AUSF suspends the SoR protection service for the UE until a new KAUSF is generated. |

6.2.8 Security

As indicated in 3GPP TS 33.501 [8], the access to the Nausf_SoRProtection API may be authorized by means of the OAuth2 protocol (see IETF RFC 6749 [13]), using the "Client Credentials" authorization grant, where the NRF (see 3GPP TS 29.510 [14]) plays the role of the authorization server.

If OAuth2 is used, an NF Service Consumer, prior to consuming services offered by the Nausf_SoRProtection API, shall obtain a "token" from the authorization server, by invoking the Access Token Request service, as described in 3GPP TS 29.510 [14], clause 5.4.2.2.

NOTE: When multiple NRFs are deployed in a network, the NRF used as authorization server is the same NRF that the NF Service Consumer used for discovering the Nausf_SoRProtection service.

The Nausf_SoRProtection Service API does not define any scopes for OAuth2 authorization as specified in 3GPP TS 33.501 [8]; it defines a single scope consisting on the name of the service (i.e., "nausf-sorprotection"), and it does not define any additional scopes at resource or operation level.

6.3 Nausf_UPUProtection Service API

6.3.1 API URI

URIs of this API shall have the following root:

```
{apiRoot}/{apiName}/{apiVersion}/
```

where "apiRoot" is defined in clause 4.4.1 of 3GPP TS 29.501 [5], the "apiName" shall be set to "nausf-upuprotection" and the "apiVersion" shall be set to "v1" for the current version of this specification.

6.3.2 Usage of HTTP

6.3.2.1 General

HTTP/2, as defined in IETF RFC 7540 [6], shall be used as specified in clause 5 of 3GPP TS 29.500 [4].

6.3.2.2 HTTP standard headers

6.3.2.2.1 General

The usage of HTTP standard headers is specified in clause 5.2.2 of 3GPP TS 29.500 [4].

6.3.2.2.2 Content type

The following content types shall be supported:

- JSON, as defined in IETF RFC 8259 [7], shall be used as content type of the HTTP bodies specified in the present specification as indicated in clause 5.4 of 3GPP TS 29.500 [4].

- The Problem Details JSON Object (IETF RFC 7807 [11]). The use of the Problem Details JSON object in a HTTP response body shall be signalled by the content type "application/problem+json"

6.3.2.3 HTTP custom headers

6.3.2.3.1 General

In this version of the API, no specific custom headers are defined for the "Nausf_UPUProtection" service.

For 3GPP specific HTTP custom headers used across all service based interfaces, see clause 5.2.3 of 3GPP TS 29.500 [4].

6.3.3 Resources

6.3.3.1 Overview

The structure of the Resource URIs of the UPUProtection service is shown in Figure 6.3.3.1-1

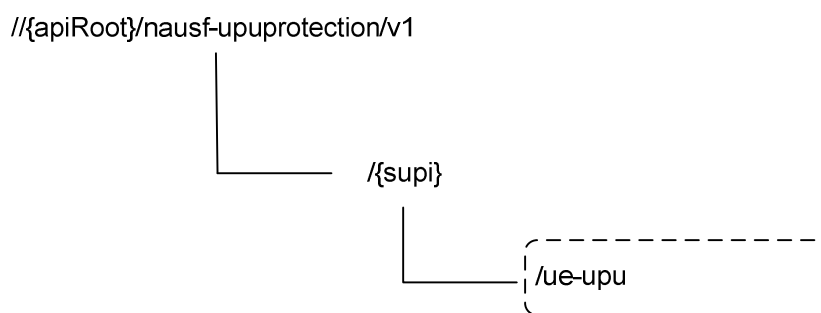


Figure 6.3.3.1-1: Resource URI structure of the UPUProtection API

Table 6.3.3.1-1 provides an overview of the resources and applicable HTTP methods.

Table 6.3.3.1-1: Resources and methods overview

| Resource name | Resource URI | HTTP method or custom operation | Description |
|---------------------------|--|---------------------------------|--|
| ue-upu (Custom operation) | {apiRoot}/nausf-upuprotection/v1/{{supi}}/ue-upu/generate-upu-data | generate-upu-data (POST) | Resource for UPU security material computation |

6.3.3.2 Resource: ue-upu

6.3.3.2.1 Description

It is the resource to which the custom operation used to generate the UPU security material is associated with.

6.3.3.2.2 Resource Definition

Resource URI: {apiRoot}/nausf-upuprotection/v1/supi/ue-upu

This resource shall support the resource URI variables defined in table 6.3.3.2.2-1.

Table 6.3.3.2.2-1: Resource URI variables for this resource

| Name | Definition |
|---------|---|
| apiRoot | See clause 6.3.1 |
| supi | Represents the Subscription Permanent Identifier (see 3GPP TS 23.501 [2] clause 5.9.2) pattern: '^(\imsi-[0-9]{5,15} nai-.+ .+)\$\$' |

6.3.3.2.3 Resource Standard Methods

No Standard Methods are supported for this resource.

6.3.3.2.4 Resource Custom Operations

6.3.3.2.4.1 Overview

Table 6.3.3.2.4.1-1: Custom operations

| Custom operation URI | Mapped HTTP method | Description |
|----------------------|--------------------|---|
| /generate-upu-data | POST | The AUSF calculates the UPU-MAC-I _{AUSF} and the Counter _{UPU} to protect the UE Parameters Update Data provided. It may also calculate the UPU-XMAC-I _{UE} to verify that the UE received the UE Parameters Update Data if the indication that an acknowledgement is requested from the UE is provided. |

6.3.3.2.4.2 Operation: generate-upu-data

6.3.3.2.4.2.1 Description

This custom operation is used by the NF service consumer (e.g. UDM) to request the AUSF to compute the security material (UPU-MAC-I_{AUSF}, Counter_{UPU} and UPU-XMAC-I_{UE}) needed to ensure the protection of the UPU procedure (see 3GPP TS 33.501 [8]).

6.3.3.2.4.2.2 Operation Definition

This method shall support the request data structures specified in table 6.3.3.2.4.2.2-1 and the response data structures and response codes specified in table 6.3.3.2.4.2.2-2.

Table 6.3.3.2.4.2.2-1: Data structures supported by the POST Request Body on this resource

| Data type | P | Cardinality | Description |
|-----------|---|-------------|--|
| UpuInfo | M | 1 | Contains the UE Parameters Update Data and may contain the indication that an acknowledgement is requested from the UE (as specified in 3GPP TS 33.501 [8]). |

Table 6.3.3.2.4.2.2-2: Data structures supported by the POST Response Body on this resource

| Data type | P | Cardinality | Response codes | Description |
|-----------------|---|-------------|-------------------------|---|
| UpuSecurityInfo | M | 1 | 200 OK | Upon success, the response body will contain UPU-MAC-I _{AUSF} and Counter _{UPU} and may contain the UPU-XMAC-I _{UE} . |
| ProblemDetails | M | 1 | 503 Service Unavailable | The "cause" attribute shall be set to one of the following application error: - COUNTER_WRAP See table 6.3.7.3-1 for the description of these errors. |

6.3.4 Custom Operations without associated resources

6.3.4.1 Overview

There is no Custom Operation in the current version of this API.

6.3.5 Notifications

6.3.5.1 General

There is no use of notification in the current version of this API.

6.3.6 Data Model

6.3.6.1 General

This clause specifies the application data model supported by the API.

Table 6.3.6.1-1 specifies the data types defined for the Nausf-UPUProtection service based interface protocol.

Table 6.3.6.1-1: Nausf specific Data Types

| Data type | Clause defined | Description |
|-----------------|----------------|--|
| UpuInfo | 6.3.6.2.2 | Contains the UE parameters update Information |
| UpuSecurityInfo | 6.3.6.2.3 | Contains the material generated for securing of UPU. It contains at least the UPU-MAC-I _{AUSF} and Counter _{UPU} . |
| UpuData | 6.3.6.2.4 | Contains UE parameters update data set (e.g., the updated Routing ID Data or the Default configured NSSAI). |
| UpuMac | 6.3.6.3.2 | MAC value for protecting UPU procedure (UPU-MAC-I _{AUSF} and UPU-MAC-I _{UE}) |
| CounterUpu | 6.3.6.3.2 | Counter _{UPU} |
| AckInd | 6.3.6.3.2 | Contains the indication of whether the acknowledgement from UE is needed |

Table 6.3.6.1-2 specifies data types re-used by the Nausf-UPUProtection service based interface protocol from other specifications, including a reference to their respective specifications and when needed, a short description of their use within the Nausf service based interface.

Table 6.3.6.1-2: Nausf re-used Data Types

| Data type | Reference | Comments |
|-----------|---------------------|--------------------------|
| Snssai | 3GPP TS 29.571 [10] | Default configured NSSAI |

6.3.6.2 Structured data types

6.3.6.2.1 Introduction

The following clauses define the structures to be used in resource representations.

6.3.6.2.2 Type: UpuInfo

Table 6.3.6.2.2-1: Definition of type UpuInfo

| Attribute name | Data type | P | Cardinality | Description |
|----------------|----------------|---|-------------|--|
| upuDataList | array(UpuData) | M | 1..N | This information defines the UE Parameters Update (UPU). A secured packed with the Routing indicator update data and/or the Default configured NSSAI update data are included. See clause 6.3.6.2.4. |
| upuAckInd | UpuAckInd | M | 1 | Contains the indication of whether the acknowledgement from UE is needed. |

6.3.6.2.3 Type: UpuSecurityInfo

Table 6.3.6.2.3-1: Definition of type UpuSecurityInfo

| Attribute name | Data type | P | Cardinality | Description |
|----------------|------------|---|-------------|--|
| upuMacIaUSF | UpuMac | M | 1 | Contains the UPU-MAC-IAUSF. |
| counterUpu | CounterUpu | M | 1 | Contains the Counter _{UPU} . |
| upuXmaclue | UpuMac | O | 0..1 | When present, contains the UPU-XMAC-I _{UE} . It shall be included, if the UDM requests the acknowledgement from the UE. |

6.3.6.2.4 Type: UpuData

Table 6.3.6.2.4-1: Definition of type UpuData

| Attribute name | Data type | P | Cardinality | Description |
|------------------|---------------|---|-------------|--|
| secPacket | SecuredPacket | C | 0..1 | Presents if the Routing indicator update data is required to be updated, and contains a secured packet with the Routing indicator to be updated. |
| defaultConfNssai | array(Snssai) | C | 0..N | Presents if the Default configured NSSAI is required to be updated, and contains the Default configured NSSAI to be updated. |

6.3.6.3 Simple data types and enumerations

6.3.6.3.1 Introduction

This clause defines simple data types and enumerations that can be referenced from data structures defined in the previous clauses.

6.3.6.3.2 Simple data types

Table 6.3.6.3.2-1: Simple data types

| Type Name | Type Definition | Description |
|------------|-----------------|---|
| UpuMac | string | pattern: "[A-Fa-f0-9]{32}\$" |
| CounterUpu | string | pattern: "[A-Fa-f0-9]{4}\$" |
| UpuAckInd | boolean | true indicates that the UPU-XMAC-I _{UE} shall be computed and returned in the response |

6.3.6.3.3 Enumeration: UpuDataType

Table 6.3.6.3.3-1: Enumeration UpuDataType

| Enumeration value | Description |
|----------------------|--------------------------------------|
| "ROUT_INDICATOR" | Routing indicator update data |
| "DEFAULT_CONF_NSSAI" | Default configured NSSAI update data |

6.3.7 Error Handling

6.3.7.1 General

HTTP error handling shall be supported as specified in clause 5.2.4 of 3GPP TS 29.500 [4].

6.3.7.2 Protocol Errors

Protocol Error Handling shall be supported as specified in clause 5.2.7.2 of 3GPP TS 29.500 [4].

6.3.7.3 Application Errors

The common application errors defined in the Table 5.2.7.2-1 in 3GPP TS 29.500 [4] may also be used for the Nausf_UPUProtection service. The following application errors listed in Table 6.3.7.3-1 are specific for the Nausf_UPUProtection service.

Table 6.3.7.3-1: Application errors

| Application Error | HTTP status code | Description |
|-------------------|-------------------------|---|
| COUNTER_WRAP | 503 Service Unavailable | The Counter _{UPU} associated with the K _{AUSF} of the UE is about to wrap around. The AUSF suspends the UPU protection service for the UE until a new K _{AUSF} is generated. |

6.3.8 Security

As indicated in 3GPP TS 33.501 [8], the access to the Nausf_UPUProtection API may be authorized by means of the OAuth2 protocol (see IETF RFC 6749 [13]), using the "Client Credentials" authorization grant, where the NRF (see 3GPP TS 29.510 [14]) plays the role of the authorization server.

If OAuth2 is used, an NF Service Consumer, prior to consuming services offered by the Nausf_UPUProtection API, shall obtain a "token" from the authorization server, by invoking the Access Token Request service, as described in 3GPP TS 29.510 [14], clause 5.4.2.2.

NOTE: When multiple NRFs are deployed in a network, the NRF used as authorization server is the same NRF that the NF Service Consumer used for discovering the Nausf_UPUProtection service.

The Nausf_UPUProtection Service API does not define any scopes for OAuth2 authorization as specified in 3GPP TS 33.501 [8]; it defines a single scope consisting on the name of the service (i.e., "nausf-upuprotection"), and it does not define any additional scopes at resource or operation level.

Annex A (normative): OpenAPI specification

A.1 General

This Annex specifies the formal definition of the Nausf Service API(s). It consists of OpenAPI 3.0.0 specifications in YAML format.

NOTE 1: OpenAPI 3.0 does not support description of API using HATEOAS. Indeed, only relative paths can be used and as a consequence the URI provided in the "href" cannot be reused as it is.

This Annex takes precedence when being discrepant to other parts of the specification with respect to the encoding of information elements and methods within the API(s).

NOTE 2: The semantics and procedures, as well as conditions, e.g. for the applicability and allowed combinations of attributes or values, not expressed in the OpenAPI definitions but defined in other parts of the specification also apply.

Informative copies of the OpenAPI specification files contained in this 3GPP Technical Specification are available on the public 3GPP file server in the following locations (see clause 5B of the 3GPP TR 21.900 [21] for further information):

- <https://www.3gpp.org/ftp/Specs/archive/OpenAPI/<Release>/>, and
- <https://www.3gpp.org/ftp/Specs/<Plenary>/<Release>/OpenAPI/>.

NOTE 3: To fetch the OpenAPI specification file after CT#83 plenary meeting for Release 15 in the above links <Plenary> must be replaced with the date the CT Plenary occurs, in the form of year-month (yyyy-mm), e.g. for CT#83 meeting <Plenary> must be replaced with value "2019-03" and <Release> must be replaced with value "Rel-15".

A.2 Nausf_UEAuthentication API

```

openapi: 3.0.0
info:
  version: 1.0.3
  title: AUSF API
  description: |
    AUSF UE Authentication Service.
    © 2019, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).
    All rights reserved.

servers:
  - url: '{apiRoot}/nausf-auth/v1'
    variables:
      apiRoot:
        default: https://example.com
        description: apiRoot as defined in clause clause 4.4 of 3GPP TS 29.501.

security:
  - {}
  - oAuth2ClientCredentials:
    - nausf-auth

paths:
  /ue-authentications:
    post:
      requestBody:
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/AuthenticationInfo'
            required: true

```



```

responses:
  '201':
    description: UEAuthenticationCtx
    content:
      application/3gppHal+json:
        schema:
          $ref: '#/components/schemas/UEAuthenticationCtx'
    headers:
      Location:
        description: 'Contains the URI of the newly created resource according to the
structure: {apiRoot}/nausf-auth/v1/ue-authentications/{authCtxId}'
        required: true
        schema:
          type: string

  '400':
    description: Bad Request from the AMF
    content:
      application/problem+json:
        schema:
          $ref: 'TS29571_CommonData.yaml#/components/schemas/ProblemDetails'

  '403':
    description: Forbidden due to serving network not authorized
    content:
      application/problem+json:
        schema:
          $ref: 'TS29571_CommonData.yaml#/components/schemas/ProblemDetails'

  '404':
    description: User does not exist in the HPLMN
    content:
      application/problem+json:
        schema:
          $ref: 'TS29571_CommonData.yaml#/components/schemas/ProblemDetails'

  '500':
    description: Internal Server Error
    content:
      application/problem+json:
        schema:
          $ref: 'TS29571_CommonData.yaml#/components/schemas/ProblemDetails'

  '501':
    description: Received protection scheme is not supported by HPLMN
    content:
      application/problem+json:
        schema:
          $ref: 'TS29571_CommonData.yaml#/components/schemas/ProblemDetails'

/ue-authentications/{authCtxId}/5g-aka-confirmation:
  put:
    parameters:
      - name: authCtxId
        in: path
        required: true
        schema:
          type: string
    requestBody:
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/ConfirmationData'
    responses:
      '200':
        description: Request processed (EAP success or Failure)
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/ConfirmationDataResponse'

      '400':
        description: Bad Request
        content:
          application/problem+json:
            schema:
              $ref: 'TS29571_CommonData.yaml#/components/schemas/ProblemDetails'

      '500':
        description: Internal Server Error
        content:
          application/problem+json:
            schema:
              $ref: 'TS29571_CommonData.yaml#/components/schemas/ProblemDetails'

```

```

/ue-authentications/{authCtxId}/eap-session:
  post:
    operationId: EapAuthMethod
    parameters:
      - name: authCtxId
        in: path
        required: true
        schema:
          type: string
    requestBody:
      content:
        application/json:
          schema:
            $ref: '#/components/schemas/EapSession'
    responses:
      '200':
        description: Use to handle or close the EAP session
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/EapSession'

          application/3gppHal+json:
            schema:
              type: object
              properties:
                eapPayload:
                  $ref: '#/components/schemas/EapPayload'
                _links:
                  type: object
                  description: 'URI : /{eapSessionUri}'
                  additionalProperties:
                    $ref: 'TS29571_CommonData.yaml#/components/schemas/LinksValueSchema'
                  minProperties: 1
              required:
                - eapPayload
                - _links
      '400':
        description: Bad Request
        content:
          application/problem+json:
            schema:
              $ref: 'TS29571_CommonData.yaml#/components/schemas/ProblemDetails'
      '500':
        description: Internal Server Error
        content:
          application/problem+json:
            schema:
              $ref: 'TS29571_CommonData.yaml#/components/schemas/ProblemDetails'
components:
  securitySchemes:
    oAuth2ClientCredentials:
      type: oauth2
      flows:
        clientCredentials:
          tokenUrl: '{nrfApiRoot}/oauth2/token'
          scopes:
            nausf-auth: Access to Nausf_UEAuthentication API
schemas:
  AuthenticationInfo:
    type: object
    properties:
      supiOrSuci:
        $ref: 'TS29503_Nudm_UEAU.yaml#/components/schemas/SupiOrSuci'
      servingNetworkName:
        $ref: 'TS29503_Nudm_UEAU.yaml#/components/schemas/ServingNetworkName'
      resynchronizationInfo:
        $ref: 'TS29503_Nudm_UEAU.yaml#/components/schemas/ResynchronizationInfo'
      pei:
        $ref: 'TS29571_CommonData.yaml#/components/schemas/Pei'
      traceData:
        $ref: 'TS29571_CommonData.yaml#/components/schemas/TraceData'
    required:
      - supiOrSuci
      - servingNetworkName
  UEAuthenticationCtx:
    type: object
    properties:

```

```

    authType:
      $ref: '#/components/schemas/AuthType'
    5gAuthData:
      oneOf:
        - $ref: '#/components/schemas/Av5gAka'
        - $ref: '#/components/schemas/EapPayload'
    _links:
      type: object
      additionalProperties:
        $ref: 'TS29571_CommonData.yaml#/components/schemas/LinksValueSchema'
    servingNetworkName:
      $ref: 'TS29503_Nudm_UEAU.yaml#/components/schemas/ServingNetworkName'
  required:
    - authType
    - 5gAuthData
    - _links

Av5gAka:
  type: object
  required:
    - rand
    - hxresStar
    - autn
  properties:
    rand:
      $ref: 'TS29503_Nudm_UEAU.yaml#/components/schemas/Rand'
    hxresStar:
      $ref: '#/components/schemas/HxresStar'
    autn:
      $ref: 'TS29503_Nudm_UEAU.yaml#/components/schemas/Autn'
ConfirmationData:
  type: object
  required:
    - resStar
  properties:
    resStar:
      $ref: '#/components/schemas/ResStar'
ConfirmationDataResponse:
  type: object
  properties:
    authResult:
      $ref: '#/components/schemas/AuthResult'
    supi:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/Supi'
    kseaf:
      $ref: '#/components/schemas/Kseaf'

  required:
    - authResult
EapSession:
  type: object
  properties:
    eapPayload:
      $ref: '#/components/schemas/EapPayload'
    kSeaf:
      $ref: '#/components/schemas/Kseaf'
    _links:
      type: object
      additionalProperties:
        $ref: 'TS29571_CommonData.yaml#/components/schemas/LinksValueSchema'
    authResult:
      $ref: '#/components/schemas/AuthResult'
    supi:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/Supi'
  required:
    - eapPayload

AuthResult:
  type: string
  enum:
    - AUTHENTICATION_SUCCESS
    - AUTHENTICATION_FAILURE
    - AUTHENTICATION_ONGOING
EapPayload:
  type: string
  format: base64
  description: contains an EAP packet
Kseaf:

```

```

    type: string
    pattern: '[A-Za-f0-9]{64}'
  ResStar:
    type: string
    pattern: '[A-Za-f0-9]{32}'
    nullable: true
  HxresStar:
    type: string
    pattern: "[A-Za-f0-9]{32}"
  AuthType:
    anyOf:
      - type: string
        enum:
          - 5G_AKA
          - EAP_AKA_PRIME
          - EAP_TLS
      - type: string
externalDocs:
  description: 3GPP TS 29.509 V15.5.0; 5G System; 3GPP TS Authentication Server services.
  url: http://www.3gpp.org/ftp/Specs/archive/29_series/29.509

```

A.3 Nausf_SoRProtection API

```

openapi: 3.0.0
info:
  version: 1.0.2
  title: Nausf_SoRProtection Service
  description: |
    AUSF SoR Protection Service.
    © 2019, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).
    All rights reserved.

servers:
  - url: '{apiRoot}/nausf-sorprotection/v1'
    variables:
      apiRoot:
        default: https://example.com
        description: apiRoot as defined in clause 4.4 of 3GPP TS 29.501

security:
  - {}
  - oAuth2ClientCredentials:
      - nausf-sorprotection

paths:
  /{supi}/ue-sor:
    post:
      parameters:
        - name: supi
          in: path
          description: Identifier of the UE
          required: true
          schema:
            $ref: 'TS29571_CommonData.yaml#/components/schemas/Supi'
      requestBody:
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/SorInfo'
            required: true
      responses:
        '200':
          description: SorSecurityInfo
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/SorSecurityInfo'
        '503':
          description: Service Unavailable
          content:
            application/problem+json:
              schema:
                $ref: 'TS29571_CommonData.yaml#/components/schemas/ProblemDetails'

components:
  securitySchemes:
    oAuth2ClientCredentials:
      type: oauth2

```

```

flows:
  clientCredentials:
    tokenUrl: '{nrfApiRoot}/oauth2/token'
    scopes:
      nausf-sorprotection: Access to the Nausf_SoRProtection API
schemas:
  SorInfo:
    type: object
    properties:
      steeringContainer:
        $ref: '#/components/schemas/SteeringContainer'
      ackInd:
        $ref: '#/components/schemas/AckInd'
    required:
      - ackInd
  SorSecurityInfo:
    type: object
    properties:
      sorMacIausf:
        $ref: '#/components/schemas/SorMac'
      counterSor:
        $ref: '#/components/schemas/CounterSor'
      sorXmacLue:
        $ref: '#/components/schemas/SorMac'
    required:
      - sorMacIausf
      - counterSor
  SteeringContainer:
    oneOf:
      - type: array
        items:
          $ref: 'TS29509_Nausf_SoRProtection.yaml#/components/schemas/SteeringInfo'
        minItems: 1
      - $ref: '#/components/schemas/SecuredPacket'
  SteeringInfo:
    type: object
    properties:
      plmnId:
        $ref: 'TS29571_CommonData.yaml#/components/schemas/PlmnId'
      accessTechList:
        type: array
        items:
          $ref: '#/components/schemas/AccessTech'
        minItems: 1
    required:
      - plmnId
  SorMac:
    type: string
    pattern: '^[A-Fa-f0-9]{32}$'
  CounterSor:
    type: string
    pattern: '^[A-Fa-f0-9]{4}$'
  AckInd:
    type: boolean
  SecuredPacket:
    type: string
    format: base64
  AccessTech:
    anyOf:
      - type: string
        enum:
          - NR
          - EUTRAN_IN_WBS1_MODE_AND_NBS1_MODE
          - EUTRAN_IN_NBS1_MODE_ONLY
          - EUTRAN_IN_WBS1_MODE_ONLY
          - UTRAN
          - GSM_AND_ECGSM_IoT
          - GSM_WITHOUT_ECGSM_IoT
          - ECGSM_IoT_ONLY
          - CDMA_1xRTT
          - CDMA_HRPD
          - GSM_COMPACT
      - type: string
externalDocs:
  description: 3GPP TS 29.509 V15.4.0; 5G System; Authentication Server Services
  url: 'http://www.3gpp.org/ftp/Specs/archive/29_series/29.509'

```

A.4 Nausf_UPUProtection API

```

openapi: 3.0.0
info:
  version: 1.0.1
  title: Nausf_UPUProtection Service
  description: |
    AUSF UPU Protection Service.
    © 2019, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).
    All rights reserved.

servers:
  - url: '{apiRoot}/nausf-upuprotection/v1'
    variables:
      apiRoot:
        default: https://example.com
        description: apiRoot as defined in clause 4.4 of 3GPP TS 29.501

security:
  - {}
  - oAuth2ClientCredentials:
      - nausf-upuprotection

paths:
  /{supi}/ue-upu:
    post:
      parameters:
        - name: supi
          in: path
          description: Identifier of the UE
          required: true
          schema:
            $ref: 'TS29571_CommonData.yaml#/components/schemas/Supi'
      requestBody:
        content:
          application/json:
            schema:
              $ref: '#/components/schemas/UpuInfo'
            required: true
      responses:
        '200':
          description: UpuSecurityInfo
          content:
            application/json:
              schema:
                $ref: '#/components/schemas/UpuSecurityInfo'
        '503':
          description: Service Unavailable
          content:
            application/problem+json:
              schema:
                $ref: 'TS29571_CommonData.yaml#/components/schemas/ProblemDetails'

components:
  securitySchemes:
    oAuth2ClientCredentials:
      type: oauth2
      flows:
        clientCredentials:
          tokenUrl: '{nrfApiRoot}/oauth2/token'
          scopes:
            nausf-upuprotection: Access to the Nausf_UPUProtection API

schemas:
  UpuInfo:
    type: object
    properties:
      upuDataList:
        type: array
        items:
          $ref: '#/components/schemas/UpuData'
        minItems: 1
      upuAckInd:
        $ref: '#/components/schemas/UpuAckInd'
    required:
      - upuDataList
      - upuAckInd
  UpuSecurityInfo:

```

```
type: object
properties:
  upuMacIausf:
    $ref: '#/components/schemas/UpuMac'
  counterUpu:
    $ref: '#/components/schemas/CounterUpu'
  upuXmacIue:
    $ref: '#/components/schemas/UpuMac'
required:
  - upuMacIausf
  - counterUpu
UpuData:
type: object
properties:
  secPacket:
    $ref: 'TS29509_Nausf_SoRProtection.yaml#/components/schemas/SecuredPacket'
  defaultConfNssai:
    type: array
    items:
      $ref: 'TS29571_CommonData.yaml#/components/schemas/Snssai'
    minItems: 1
  oneOf:
    - required: [secPacket]
    - required: [defaultConfNssai]
UpuMac:
type: string
pattern: '^[-Fa-f0-9]{32}$'
CounterUpu:
type: string
pattern: '^[-Fa-f0-9]{4}$'
UpuAckInd:
type: boolean

externalDocs:
description: 3GPP TS 29.509 V15.4.0; 5G System; Authentication Server Services
url: 'http://www.3gpp.org/ftp/Specs/archive/29_series/29.509'
```

Annex B (Informative): Use of EAP-TLS

B.1 General

The Annex B of 3GPP TS 33.501 [8] describes the use of EAP-TLS as an alternative authentication method in the case of private network. This annex describes corresponding stage 3.

B.2 EAP method: EAP-TLS

EAP-TLS as defined in IETF RFC 5216 [16] is the EAP method used in this procedure. This procedure is described in Annex B.2.1 of 3GPP TS 33.501 [8].



- 1 The NF Service Consumer (AMF) shall send a POST request to the AUSF. The payload of the body shall contain at least the UE Id and Serving Network Name.
- 2a. On success, "201 Created" shall be returned. The payload body shall contain the representation of the resource generated and the "Location" header shall contain the URI of the generated resource (e.g. `.../v1/ue_authentications/{authCtxId}/eap-session`). The AUSF generates a sub-resource "eap-session". The AUSF shall provide a hypermedia link towards this sub-resource in the payload to indicate to the AMF where it shall send a POST containing the EAP packet response. The body payload shall also contain the EAP packet `EAP-Request/EAP-Type=EAP-TLS (TLS Start)`
- 2b. On failure, one of the HTTP status code listed in table 6.1.7.3-1 shall be returned with the message body containing a ProblemDetails structure with the "cause" attribute set to one of the application error listed in Table

6.1.7.3-1. In particular, if the serving network is not authorized, the AUSF shall use the "Cause" SERVING_NETWORK_NOT_AUTHORIZED.

3. Based on the relation type, the NF Service Consumer (AMF) shall send a POST request including the EAP-Response/EAP-Type=EAP-TLS (TLS client_hello) received from the UE. The POST request is sent to the URI provided by the AUSF or derived by the NF Service Consumer (AMF).
- 4a. On success, the AUSF shall reply with a "200 OK" HTTP message containing the EAP Request as described in Annex B.2.1 of 3GPP TS 33.501 [8] and a hypermedia link towards the sub-resource "eap-session".
- 4b. On failure, one of the HTTP status code listed in table 6.1.7.3-1 shall be returned with the message body containing a ProblemDetails structure with the "cause" attribute set to one of the application error listed in Table 6.1.7.3-1.
5. The NF Service Consumer (AMF) shall send a POST request including the EAP Response received from the UE. The POST request is sent to the URI provided by the AUSF or derived by the NF Service Consumer (AMF).
- 6a. On success, the AUSF shall reply with a "200 OK" HTTP message containing the EAP Request as described in Annex B.2.1 of 3GPP TS 33.501 [8] and a hypermedia link towards the sub-resource "eap-session".
- 6b. On failure, one of the HTTP status code listed in table 6.1.7.3-1 shall be returned with the message body containing a ProblemDetails structure with the "cause" attribute set to one of the application error listed in Table 6.1.7.3-1.
7. The NF Service Consumer (AMF) shall send a POST request including the EAP Response received from the UE. The POST request is sent to the URI provided by the AUSF or derived by the NF Service Consumer (AMF).
- 8a. If the EAP authentication exchange is successfully completed (with or without the optional Notification Request/Response messages exchange), "200 OK" shall be returned to the NF Service Consumer (AMF). The payload shall contain the result of the authentication, an EAP success/failure and the Kseaf if the authentication is successful.
- 8b. On failure, one of the HTTP status code listed in table 6.1.7.3-1 shall be returned with the message body containing a ProblemDetails structure with the "cause" attribute set to one of the application error listed in Table 6.1.7.3-1.

Annex C (informative): Withdrawn API versions

C.1 General

This Annex lists withdrawn API versions of the APIs defined in the present specification. 3GPP TS 29.501 [5] clause 4.3.1.6 describes the withdrawal of API versions.

C.2 Nausf_SoRProtection API

The API versions listed in table C.2-1 are withdrawn for the Nausf_SoRProtection API.

Table C.2-1: Withdrawn API versions of the Nausf_SoRProtection service

| API version number | Reason for withdrawal |
|--------------------|--|
| 1.0.0 | The version 1.0.0 indicates that a 201 "Created" must be sent in response to the POST Custom Operations while it should be a "200 OK" as indicated in clause 6.2.3.2.4.2.2. The version 1.0.1 corrects this mistake. As a consequence, the version 1.0.0 is withdrawn in order to avoid interoperability problems with further version of the Nausf_SoRProtection service. |

Annex D (informative): Change history

| Change history | | | | | | | |
|----------------|---------|-----------|------|-----|-----|---|-------------|
| Date | Meeting | TDoc | CR | Rev | Cat | Subject/Comment | New version |
| 2017-10 | CT4#80 | C4-175268 | | | | Initial Draft.(Agreed Skeleton) | 0.1.0 |
| 2017-10 | CT4#80 | C4-175394 | | | | Inclusion of pCR agrees during CT4#80: C4-175269 and C4-175270 | 0.2.0 |
| 2017-12 | CT4#81 | C4-176437 | | | | Inclusion of pCR agrees during CT4#81: C4-176267, C4-176269, C4-176426, C4-17427 | 0.3.0 |
| 2018-01 | CT4#82 | C4-181391 | | | | Inclusion of pCR agrees during CT4#82: C4-181341, C4-181342, C4-181343, C4-181344, C4-181345, C4-181346, C4-181347, C4-181155 | 0.4.0 |
| 2018-03 | CT4#83 | C4-182434 | | | | Inclusion of pCRs agrees during CT4#83: C4-182283 and C4-182279 | 0.5.0 |
| 2018-03 | CT#79 | CP-180031 | | | | Presented for information | 1.0.0 |
| 2018-04 | CT4#84 | C4-183516 | | | | Inclusion of pCRs agreed during CT4#84: C4-183309, C4-183313, C4-183346, C4-183347 and C4-183448 | 1.1.0 |
| 2018-05 | CT4#85 | C4-184623 | | | | Inclusion of PCR agrees during CT4#83: C4-184219, C4-184220, C4-184224, C4-184227, C4-184227, C4-184362, C4-184363, C4-184367, C4-184368, C4-184370, C4-184376, C4-184380, C4-184584, C4-184624 | 1.2.0 |
| 2018-06 | CT#80 | CP-181104 | | | | Presented for approval | 2.0.0 |
| 2018-06 | CT#80 | | | | | Approved in CT#80. | 15.0.0 |
| 2018-09 | CT#81 | CP-182059 | 0002 | 2 | F | Requester ID in Authentication Info | 15.1.0 |
| 2018-09 | CT#81 | CP-182059 | 0003 | 1 | F | HTTP method in figure 5.2.2.2-1 (Note: clause 6.1.3.1 is not included, already covered) | 15.1.0 |
| 2018-09 | CT#81 | CP-182059 | 0004 | 4 | F | SoRProtection service operation | 15.1.0 |
| 2018-09 | CT#81 | CP-182059 | 0010 | 1 | F | Adding TS 33.501 reference | 15.1.0 |
| 2018-09 | CT#81 | CP-182059 | 0011 | - | F | HTTP Custom Header | 15.1.0 |
| 2018-09 | CT#81 | CP-182059 | 0013 | 1 | F | SUPI sends to AMF | 15.1.0 |
| 2018-09 | CT#81 | CP-182068 | 0014 | 2 | B | 5G Trace for AUSF | 15.1.0 |
| 2018-09 | CT#81 | CP-182013 | 0015 | 2 | F | Making Oauth 2.0 optional in OAS description | 15.1.0 |
| 2018-09 | CT#81 | CP-182059 | 0016 | 1 | F | Editorial Corrections | 15.1.0 |
| 2018-09 | CT#81 | CP-182059 | 0017 | 1 | F | Error code correction | 15.1.0 |
| 2018-09 | CT#81 | CP-182059 | 0018 | 1 | F | Add support to EAP-TLS (Optional) | 15.1.0 |
| 2018-09 | CT#81 | CP-182059 | 0019 | - | F | Correcting Presentation of resources for AUSF API | 15.1.0 |
| 2018-09 | CT#81 | CP-182059 | 0020 | 1 | F | Correcting confirmation message | 15.1.0 |
| 2018-09 | CT#81 | CP-182059 | 0021 | - | F | API version number update | 15.1.0 |
| 2018-12 | CT#82 | C4-187253 | 0026 | - | F | Remove the "supiOrSuci" in Confirmation Data | 15.2.0 |
| 2018-12 | CT#82 | C4-187254 | 0027 | - | F | Correcting Resource URI structure of the SoRProtection Service | 15.2.0 |
| 2018-12 | CT#82 | C4-187359 | 0030 | - | F | Cardinality | 15.2.0 |
| 2018-12 | CT#82 | C4-187370 | 0031 | - | F | Add supi and authResult to EapSession in OpenAPI definitions | 15.2.0 |
| 2018-12 | CT#82 | C4-187478 | 0022 | - | F | Requester ID not needed in initial request from AMF | 15.2.0 |
| 2018-12 | CT#82 | C4-187479 | 0023 | - | F | Delaying transmission of Kseaf | 15.2.0 |
| 2018-12 | CT#82 | C4-187480 | 0024 | 1 | F | Correcting the reference to EAP-AKA' | 15.2.0 |
| 2018-12 | CT#82 | C4-187481 | 0025 | 1 | F | Adding a reference to the Annex in the Specification | 15.2.0 |
| 2018-12 | CT#82 | C4-187483 | 0028 | 1 | F | Error handling in AUSF | 15.2.0 |
| 2018-12 | CT#82 | C4-187485 | 0029 | 1 | F | Add a reference to the IETF RFC 3748 on EAP Framework | 15.2.0 |
| 2018-12 | CT#82 | C4-188060 | 0032 | - | F | Base64 reference | 15.2.0 |
| 2018-12 | CT#82 | C4-188109 | 0033 | - | F | APIRoot Clarification | 15.2.0 |
| 2018-12 | CT#82 | C4-188150 | 0034 | - | F | Reference correction | 15.2.0 |
| 2018-12 | CT#82 | C4-188251 | 0036 | - | F | OpenAPI version number for Nausf_UEAuthentication service | 15.2.0 |
| 2018-12 | CT#82 | C4-188494 | 0037 | 1 | F | OpenAPI version number for Nausf_SoRProtection | 15.2.0 |
| 2018-12 | CT#82 | C4-188495 | 0038 | 1 | F | Correct "externalDocs" for Nausf_UEAuthentication OAS | 15.2.0 |
| 2018-12 | CT#82 | C4-188497 | 0039 | 1 | F | Clarification on the 200 OK returned by AUSF in case of authentication failure | 15.2.0 |
| 2018-12 | CT#82 | C4-188547 | 0040 | - | F | Secured packet in SorInfo | 15.2.0 |
| 2018-12 | CT#82 | CP-183170 | 0035 | 2 | | Location Header in OpenAPI | 15.2.0 |
| 2018-12 | CT#82 | CP-183172 | 0041 | - | F | Alignment for Oauth scopes - Nausf_UEAuthentication | 15.2.0 |
| 2018-12 | CT#82 | CP-183173 | 0042 | - | F | Alignment for Oauth scopes - Nausf_SoRProtection | 15.2.0 |
| 2018-12 | CT#82 | CP-183203 | 0043 | - | F | externalDocs for Nausf_SoRProtection OpenAPI Annex | 15.2.0 |
| 2019-03 | CT#83 | CP-190022 | 0047 | 1 | F | Mandatory HTTP status codes | 15.3.0 |
| 2019-03 | CT#83 | CP-190022 | 0049 | 1 | F | SoR Protection response code alignment | 15.3.0 |
| 2019-03 | CT#83 | CP-190022 | 0044 | 3 | F | Authentication Failure scenarios | 15.3.0 |
| 2019-03 | CT#83 | CP-190153 | 0046 | 7 | F | UE parameters update support (indicated as C4-190618 + C4-190618_rev71) | 15.3.0 |
| 2019-03 | CT#83 | CP-19205 | 0050 | 1 | F | 3GPP TS 29.509 API version update | 15.3.0 |
| 2019-06 | CT#84 | CP-191033 | 0051 | - | F | Correct the expression of URI variables in 5g-aka-confirmation resource | 15.4.0 |
| 2019-06 | CT#84 | CP-191033 | 0053 | 2 | F | Storage of OpenAPI specification files | 15.4.0 |
| 2019-06 | CT#84 | CP-191033 | 0056 | - | F | AUSF Tracing Targeting a PEI | 15.4.0 |
| 2019-06 | CT#84 | CP-191033 | 0055 | 1 | F | Determination of the Authentication Method | 15.4.0 |

| | | | | | | | |
|---------|-------|-----------|------|---|---|---|--------|
| 2019-06 | CT#84 | CP-191033 | 0058 | 1 | F | Essential correction to add Copyright on OpenAPI specifications | 15.4.0 |
| 2019-06 | CT#84 | CP-191218 | 0059 | 4 | F | Add withdrawn API version Annex | 15.4.0 |
| 2019-06 | CT#84 | CP-191033 | 0061 | - | F | 3GPP TS 29.509 API version update | 15.4.0 |
| 2019-09 | CT#85 | CP-192106 | 0062 | 1 | F | Missing status codes | 15.5.0 |
| 2019-09 | CT#85 | C4-192119 | 0065 | - | F | API version and ExternalDocs update | 15.5.0 |

History

| Document history | | |
|-------------------------|----------------|-------------|
| V15.0.0 | September 2018 | Publication |
| V15.1.0 | October 2018 | Publication |
| V15.2.0 | April 2019 | Publication |
| V15.3.0 | April 2019 | Publication |
| V15.4.0 | July 2019 | Publication |
| V15.5.0 | October 2019 | Publication |