|  |  |
| --- | --- |
| 3GPP TS 33.535 V0.5.0 (2020-05) | |
| Technical Specification | |
| 3rd Generation Partnership Project;  Technical Specification Group Services and System Aspects;  Authentication and key management for applications;  based on 3GPP credential in 5G （AKMA）  (Release 16) | |
|  | |
| *5G-logo_175px* | 3GPP-logo_web |
|  | |
| The present document has been developed within the 3rd Generation Partnership Project (3GPP TM) and may be further elaborated for the purposes of 3GPP. The present document has not been subject to any approval process by the 3GPPOrganizational Partners and shall not be implemented. This Specification is provided for future development work within 3GPPonly. The Organizational Partners accept no liability for any use of this Specification. Specifications and Reports for implementation of the 3GPP TM system should be obtained via the 3GPP Organizational Partners' Publications Offices. | |

|  |
| --- |
|  |
| ***3GPP***  Postal address  3GPP support office address  650 Route des Lucioles - Sophia Antipolis  Valbonne - FRANCE  Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16  Internet  http://www.3gpp.org |
| ***Copyright Notification***  No part may be reproduced except as authorized by written permission. The copyright and the foregoing restriction extend to reproduction in all media.  © 2019, 3GPP Organizational Partners (ARIB, ATIS, CCSA, ETSI, TSDSI, TTA, TTC).  All rights reserved.  UMTS™ is a Trade Mark of ETSI registered for the benefit of its members  3GPP™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners LTE™ is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners  GSM® and the GSM logo are registered and owned by the GSM Association |

Contents

Foreword 4

Introduction 5

1 Scope 6

2 References 6

3 Definitions of terms, symbols and abbreviations 6

3.1 Terms 6

3.2 Symbols 6

3.3 Abbreviations 6

4 Architecture for Authentication and Key Management for Applications (AKMA) 7

4.1 Reference model 7

4.2 Network elements 7

4.2.1 AAnF 7

4.2.2 AF 7

4.2.3 NEF 7

4.2.4 AUSF 8

4.3 Interface description 8

4.3.1 Reference point Ua\* 8

4.4 Security requirements and principles for AKMA 8

4.4.1 Requirements on Ua\* Reference point 8

4.4.2 Requirements on AKMA Key Identifier (A-KID) 9

5 Key Management 9

5.1 AKMA key hierarchy 9

5.2 AKMA key lifetimes 9

6 AKMA Procedures 10

6.1 Deriving AKMA key during UE registration 10

6.2 Deriving AKMA Application key for a specific AF 10

6.3 AKMA Application key request via NEF 12

6.4 AKMA key change 12

6.4.1 KAKMA re-keying 12

6.4.2 KAF re-keying 13

7 Security related services 13

7.1 Services Provided by AAnF 13

7.1.1 General 13

7.1.2 Naanf\_KeyCreate Service 13

7.2 Services Provided by AUSF 13

7.2.1 General 13

7.1.2 Nausf\_AKMAKey\_Get Service 13

7.3 Services Provided by NEF 13

7.3.1 General 13

7.3.2 Nnef\_AKMA\_AFKeyCreate Service 13

Annex A (normative): Key derivation functions 14

A.1 KDF interface and input parameter construction 14

A.1.1 General 14

A.1.2 FC value allocations 14

A.2 KAKMA derivation function 14

Annex <X> (informative): Change history 15

# Foreword

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

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

Version x.y.z

where:

x the first digit:

1 presented to TSG for information;

2 presented to TSG for approval;

3 or greater indicates TSG approved document under change control.

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

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

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

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

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

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

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

**should** indicates a recommendation to do something

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

**may** indicates permission to do something

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

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

**can** indicates that something is possible

**cannot** indicates that something is impossible

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

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

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

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

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

In addition:

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

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

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

# Introduction

This clause is optional. If it exists, it shall be the second unnumbered clause.

# 1 Scope

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

# 2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non‑specific.

- For a specific reference, subsequent revisions do not apply.

- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TS 33.501: "Security architecture and procedures for 5G system".

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

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

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

# 3 Definitions of terms, symbols and abbreviations

## 3.1 Terms

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

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

## 3.2 Symbols

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

Symbol format (EW)

<symbol> <Explanation>

## 3.3 Abbreviations

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

AAnF AKMA Anchor Function

AF Application Function

A-KID AKMA Key IDentifier

AMFAccess and Mobility Management Function

AUSFAUthentication Server Function

KAF AKMA Application Key

KAKMA AKMA Anchor Key

NEFNetwork Exposure Function

UDM Unified Data Management

# 4 Architecture for Authentication and Key Management for Applications (AKMA)

## 4.1 Reference model

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



Figure 4.1-1: Fundamental Network Model for AKMA

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

The AKMA service requires a new logical entity: AKMA Anchor Function (AAnF).

AAnF is the anchor function in the HPLMN that generates the key material to be used between the UE and the AF and maintains UE AKMA contexts.

## 4.2 Network elements

### 4.2.1 AAnF

AAnF enables the AKMA Anchor Key (KAKMA) derivation for AKMA service. Before invoking AKMA service, UE shall have successfully registered to the 5G core, which results in KAUSF being stored at the AUSF and the UE after a successful 5G primary authentication.

### 4.2.2 AF

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

- AF with the AKMA service enabling requests for KAF from the AAnF using A-KID.

- AF shall be authenticated and authorized by the operator network before providing the AKMA Application Key to the AF.

### 4.2.3 NEF

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

- NEF finds the AAnF.

### 4.2.4 AUSF

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

* AUSF Provides the AKMA Anchor Key (KAKMA) to the AAnF.

### 4.2.5 UDM

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

- UDM stores AKMA subscription data of the subscriber.

## 4.3 Interface description

The following interfaces are involved in AKMA network architecture:

* **Nnef:** Service-based interface exhibited by NEF.
* **Nausf:** Service-based interface exhibited by AUSF.
* **Nudm:** Service-based interface exhibited by UDM.
* **Naanf:** Service-based interface exhibited by AAnF.
* **Naf:** Service-based interface exhibited by AF.

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

### 4.3.1 Reference point Ua\*

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

## 4.4 Security requirements and principles for AKMA

The following security requirements are applicable to AKMA:

- AKMA shall reuse the same UE subscription and the same credentials used for 5G access.

-    AKMA shall reuse the 5G primary authentication procedure and methods (both 5G AKA and EAP AKA’ shall be supported) for the sake of implicit authentication for AKMA services.

-    AAnF’s SBI interface to AUSF shall be confidentiality, integrity and replay protected.

-    The interface between AAnF and AF shall be confidentiality, integrity and replay protected.

-    The AKMA Application Key (KAF) shall be provided with a maximum lifetime. When the AKMA Application Key lifetime is expired, it shall be renegotiated.

Editor’s Note: Further security requirements regarding roaming and other aspects will be added.

### 4.4.1 Requirements on Ua\* Reference point

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

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

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

Editor’s Note: Further requirements (including the need to specify Ua\* protocol identifier) are FFS.

### 4.4.2 Requirements on AKMA Key Identifier (A-KID)

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

- A-KID shall be globally unique;

- A-KID shall be usable as a key identifier in protocols used in the reference point Ua\*;

- AKMA AF shall be able to identify the home network of the UE from the A-KID.

Editor’s Note: It is FFS which NF in the home network the AF request is routed to.

# 5 Key Management

## 5.1 AKMA key hierarchy

The key hierarchy (see Figure 5.1-1) includes the following keys: KAUSF, KAKMA, KAF. KAUSF is generated by AUSF as specified in clause 6 of TS 33.501 [2].

Keys for AAnF:

KAKMA is a key derived by ME and AUSF from KAUSF.

Keys for AF:

KAF is a key derived by ME and AAnF from KAKMA.

KAKMA and KAF are derived according to the procedures of clause 6.1 and 6.2.



Figure 5.1-1: AKMA Key Hierarchy

## 5.2 AKMA key lifetimes

The KAKMA and A-KID are valid until the next primary authentication is performed (implicit lifetime), in which case the KAKMA and A-KID might be replaced after a successful new authentication or removed after an unsuccessful one.

AKMA Application Keys KAF shall use explicit lifetimes based on the operator’s policy. The lifetime of KAF shall be sent by the AAnF as described in clause 6.2. In case that a new AKMA Anchor Key KAKMA is established, the AKMA Application Key KAF can continue to be used until its lifetime expires. When the KAF lifetime expires, a new AKMA Application Key is established based on the current AKMA Anchor Key KAKMA.

# 6 AKMA Procedures

## 6.1 Deriving AKMA key after primary authentication

There is no separate authentication of the UE to support AKMA functionality. Instead, it reuses the 5G primary authentication procedure executed e.g. during the UE Registration to authenticate the UE. A successful 5G primary authentication results in KAUSF being stored at the AUSF and the UE.

Figure 6.1-1 Deriving AKMA root key after primary authentication

During the primary authentication procedure, the AUSF interacts with the UDM in order to fetch authentication information such as subscription credentials (e.g. AKA Authentication vectors) and the authentication method using the Nudm\_UEAuthentication\_Get Request service operation. In the response, the UDM may also indicate to the AUSF whether AKMA keys need to be generated for the UE. If the AUSF receives the AKMA indication from the UDM, the AUSF shall store the KAUSF and generate the AKMA Anchor Key (KAKMA) and the A-KID from KAUSF after the primary authentication procedure is successfully completed.

After AKMA key material is generated, the AUSF shall send the generated A-KID, and KAKMA to the AAnF together with the UE SUPI using the Naanf\_AKMA\_KeyRegistration Request service operation. The AAnF shall store the latest information sent by the AUSF.NOTE: The AUSF need not store any AKMA key material after delivery to the AAnF.

The UE shall generate the AKMA Anchor Key (KAKMA) and the A-KID from the KAUSF before initiating communication with an AKMA Application Function.

A-KID identifies the KAKMA key of the UE from which other AKMA keys are derived.

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

The A-TID shall be derived from KAUSF as defined in Annex A.3.

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

The key derivation of KAKMA shall be performed using the key derivation function (KDF) specified in TS 33.220 [4]. KAKMA is computed (as per Annex A.2) as KAKMA=KDF (KAUSF, "AKMA", SUPI), where the key derivation parameters consist of a static string "AKMA", and SUPI.

Editor’s Note: A-KID and its association with UE identifier is FFS.

Since AKMA keys are based on KAUSF from primary authentication run, the AKMA keys can only be refreshed by running a fresh primary authentication.

## 6.2 Deriving AKMA Application Key for a specific AF

Figure 6.2-1 shows the procedure used by the AF to request application function specific AKMA keys from 5GC directly, when the AF is located in the operator’s network.



Figure 6.2-1 KAF generation from KAKMA

Before communication between the UE and the AKMA AF can start, the UE and the AKMA AF needs to know whether to use AKMA. This knowledge is implicit to the specific application on the UE and the AKMA AF.

1. When the UE initiates communication with the AKMA AF, it shall include the derived A-KID in the Application Session Establishment request message (cf. clause 6.1).

2. If the AF does not have an active context associated with the A-KID, then the AF sends a Naanf\_AKMA\_AFKey request to AAnF with the A-KID to request the AKMA Application Key for the UE. The AF also includes its identity (AF Id) in the request. The AAnF shall check whether the AAnF can provide the service to the AF by checking the AF Id. If succeeds, the following procedures are executed. Otherwise, the AAnF shall reject the procedure.

If the AAnF is in possession of the AKMA Application Key (KAF), it responds to the AF with the KAF. If not, the AAnF shall check if it has the UE specific KAKMA key identified by the A-KID.

If KAKMA is available in AAnF, the AAnF shall continue with step3 .

If KAKMA is not available, the AAnF shall continue with step 4 and send an error response.

3. The AAnF derives the AKMA Application Key (KAF) from KAKMA.

The key derivation of KAF shall be performed using the key derivation function (KDF) specified in TS 33.220 [4]. KAF is computed (as per Annex A.4) as KAF=KDF (KAKMA, AF\_ID), where the AF\_ID is constructed as follows: AF\_ID = FQDN of the AF || Ua\* security protocol identifier. The Ua\* security protocol identifier is specified as Ua security protocol identifirer in Annex H of TS 33.220 [4]. The key used for the derivation of KAF is KAKMA.

4. The AAnF sends Naanf\_AKMA\_AFKey response to the AF with KAF and lifetime.

5. The AF response the Application Session Establishment request to the UE.

## 6.3 AKMA Application Key request via NEF

Figure 6.3.1 shows the procedure used by the AF to request AKMA Application Key from 5GC via NEF, when the AF is located outside the operator’s network.



**Figure 6.3.1: AKMA Application** Key **request via NEF**

1. When the AF is about to requestAKMA Application Key for the UE from the 5GC, e.g. when UE initiates application session establishment request as in clause 6.2, the AF discovers the HPLMN of the UE based on the A-KID and sends the request towards the 5GC via NEF service API.

Editor’s Note: Format of A-KID and its association with a UE identifier is FFS.

NOTE 1: In the case of architecture without CAPIF support, the AF is locally configured with the API termination points for the service. In the case of architecture with CAPIF support, the AF obtains the service API information from the CAPIF core function via the Availability of service APIs event notification or Service Discover Response as specified in TS 23.222 [5].

2. If the AF is authorised by the NEF to request AKMA Application Key, the NEF discovers and selects an AAnF instance based on local configuration or via NRF in the same way as the AF selects the AAnF in clause 6.2.

3. The NEF forwards the AKMA Application Key request to the selected AAnF.

4. The AAnF generates the AKMA Application Key in clause 6.2 and sends the response to the NEF with the KAF, the KAF expiration time (KAF\_exptime) and potentially other parameters.

5. The NEF forwards the response to the AF.

Editor’s Note: Whether other parameters are to be returned to the AF via NEF is FFS.

## 6.4 AKMA key change

Editor's Note: It is FFS whether KAF refresh is needed.

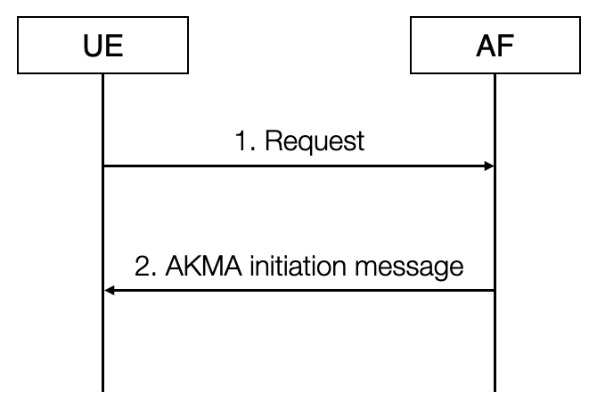
### 6.4.1 KAKMA re-keying

KAKMA shall be re-keyed by running a primary authentication as described in clause 6.1.

### 6.4.2 KAF re-keying

The KAF refresh depends on the lifetime of the KAF and may be trigged by the AF, which means when a new KAKMA is derived, the KAF will not be re-keyed automatically.

6.5 Initiation of AKMA

In case when the UE does not know to use AKMA for a service, then the following procedure applies.****

**Figure 6.5: Initiation of AKMA**

1. The UE may start communication over reference point Ua\* with the AF with or without any AKMA-related parameters.

2. If the AF requires the use of shared keys obtained by means of the AKMA, but the request from UE does not include AKMA-related parameters, the AF replies with an AKMA initiation message. The form of this initiation message may depend on the particular reference point Ua\*.

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

# 7 Security related services

## 7.1 Services Provided by AAnF

Editor’s Note: Further inputs and outputs as well SBI names are FFS.

### 7.1.1 General

The AAnF provides AKMA Application Key derivation service to the requester NF by Naanf\_Key\_Create.

### 7.1.2 Naanf\_KeyCreate Service

**Service operation name:** Naanf\_Key\_Create.

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

**Input, Required:** A-KID, AF ID

**Input, Optional:** None.

**Output, Required:** KAF, lifetime.

**Output, Optional:** None.

## 7.2 Services Provided by AUSF

### 7.2.1 General

The AUSF provides AKMA key provision service to the requester NF by Nausf\_AKMAkey\_Get.

### 7.1.2 Nausf\_AKMAKey\_Get Service

**Service operation name:** Nausf\_AKMAkey\_Get.

**Description:** The NF consumer requests the AUSF to get the KAKMA ofA-KID.

**Input, Required:** A-KID.

**Input, Optional:** None.

**Output, Required:** KAKMA.

**Output, Optional:** None.

## 7.3 Services Provided by NEF

### 7.3.1 General

The NEF exposes AKMA Application Key derivation service to the requester NF by Nnef\_AKMA\_AFKey.

### 7.3.2 Nnef\_AKMA\_AFKeyCreate Service

**Service operation name:** Nnef\_AKMA\_AFKey.

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

**Input, Required:** A-KID, AF ID

**Input, Optional:** None.

**Output, Required:** KAF, lifetime.

**Output, Optional:** None.

Editor’s Note: Further inputs and outputs as well SBI names are FFS.

Annex A (normative): Key derivation functions

# A.1 KDF interface and input parameter construction

## A.1.1 General

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

This clause specifies how to construct the input string, S, and the input key, KEY, for each distinct use of the KDF. Note that "KEY" is denoted "Key" in TS 33.220 [4].

## A.1.2 FC value allocations

The FC number space used is controlled by TS 33.220 [4], FC values allocated for the present document are in the range ofTBD1-TBDx.

# A.2 KAKMA derivation function

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

- FC = TBD1;

- P0 = "AKMA";

- L0 = length of "AKMA"; (i.e. 0x00 0x04)

- P1 = SUPI;

- L1 = length of SUPI.

The input key KEY shall be KAUSF.

A.3 A-TID derivation function

When deriving the A-TID from KAUSF, the following parameters shall be used to form the input S to the KDF:

- FC = TBD;

- P0 = "A-TID";

- L0 = length of "A-TID"; (i.e. 0x00 0x05)

- P1 = SUPI;

- L1 = length of SUPI.

The input key KEY shall be KAUSF.

A.4 KAF derivation function

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

- FC = TBD;

- P0 =AF\_ID;

- L0 = length of AF\_ID

The input key KEY shall be KAKMA.

Annex <X> (informative):  
Change history

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Change history** | | | | | | | |
| **Date** | **Meeting** | **TDoc** | **CR** | **Rev** | **Cat** | **Subject/Comment** | **New version** |
| 2019-10 | SA3 #96adhoc | S3-193817 |  |  |  | TS skeleton based on S3-193769; Scope is based on S3-193770; Other content including S3-193841, S3-193772 | 0.1.0 |
| 2019-11 | SA3 #97 | S3-194640 |  |  |  | Updates based on S3-194340, S3-194160, S3-194641, S3-194642, S3-194643, S3-194341, S3-194644, S3-194645, S3-194229, S3-194156 | 0.2.0 |
| 2020-03 | SA3 #98e | S3-200511 |  |  |  | Updates based on S3-200511, S3-200512, S3-200499, S3-200249, S3-200460, S3-200461, S3-200463, S3-200447, S3-200486, S3-200364, S3-200366, S3-200513 | 0.3.0 |
| 2020-04 | SA3 #98bis-e | S3-200831 |  |  |  | Updates based on S3-200640, S3-200661, S3-200669, S3-200826, S3-200714, S3-200814, S3-200815, S3-200816, S3-200817, S3-200803, S3-200830, S3-200773 | 0.4.0 |
| 2020-05 | SA3#99-e | S3-201xxx |  |  |  | Updates based on S3-201371, S3-201393, S3-2001051, S3-201188, S3-200968, S3-201343, S3-201387, S3-201370, S3-201394, S3-201395, S3-201028, S3-201145, S3-201168, S3-201169, S3-201287 |  |