**3GPP TSG-SA3 Meeting #114e *ad-hoc S3-240025***

Electronic meeting, online, 22 - 26 January 2024

**Source: Federal Office for Information Security (BSI)**

**Title: Discussion of the protection mechanism of the permanent key leaving the UDR environment.**

**Document for: Discussion**

**Agenda Item: 4.2**

# 1 Decision/action requested

***This document is intended to serve as a basis for improving and protecting the permanent key leaving the UDR environment and related SCAS test cases.***

# 2 References

[1] 3GPP TS 33.501 Security architecture and procedures for 5G system

[2] 3GPP TS 33.117 Catalogue of general security assurance requirements

[3] OpenAPI Specification v3.1.0: <https://spec.openapis.org/oas/v3.1.0#optional-oauth2-security>.

[4] 3GPP TS 29.505 Usage of the Unified Data Repository services for Subscription Data

# 3 Rationale

The Unified Data Repository (UDR) provides an API that delivers subscriber authentication data to the Unified Data Management (UDM). This data includes the UE’s permanent key, encPermanentKey, which serves as the anchor of trust in the entire 5G system. This discussion paper analyzes the current state of protection mechanisms for sensitive data and suggests ways to improve them. Our rationale and proposal is split into transfer and authorization security.

## Transfer Security

According to clause 5.8.1 in TS33.501 [1], it is the responsibility of the implementation to secure the transfer of data between the UDR and the UDM, as seen in the screenshot below. This can result in a situation where the transfer is completely unencrypted or proprietary protocols, which have not been security-analyzed, are used to transfer the user's permanent key. Our proposal focuses solely on the transfer, not on the storage.

*A white paper with black text

Description automatically generated*

## Authorization Security

TS 29.505 [4] defines the use of the API to access the UE's long-term key (also called encPermanentKey in this context). The authorisation is done by means of OAuth2.0 according to TS 33.501 [1]. This is important because no network function other than the UDM shall have access to the UE's long-term key. This is also specified in clause 5.2.2.1 of TS 29.505 [4] as shown below. A close-up of a document

Description automatically generated

A screen shot of a computer code

Description automatically generatedAccording to the OpenAPI definition of the API endpoint (which can be seen below), the security section has three different security definitions which are “OR” connected. The first security definition ({}) indicates that security is optional (see OpenAPI Specification [3]). The next two security definitions (oAuth2ClientCredentials) require the OAuth2 Client Credentials Grant type for authentication: one with the scope nudr-dr, the other with the scope nudr-dr AND nudr-dr:subscription-data:authentication-subscription:read.

With the current specification, there is a risk that any network function can access the permanent key with the optional, hence non-security, mechanism ({}), which contradicts the initial requirement of having limited access to the permanent key stored in the UDR.

We can use two routes to improve the security which are discussed below.

# 4 Detailed proposal

## Transfer Security

Clause 13.1.0 of TS 33.501 [1] points out that all network functions shall support mutually authenticated TLS and HTTPS and it is also recommended for usage. On the other hand, the same clause also states that “TLS shall be used for transport protection within a PLMN unless network security is provided by other means.” So in fact, it is possible to not encrypt the traffic between two network functions at all which is not sufficient for highly sensitive information exchange like the permanent key transfer. Therefore, we see the uttermost importance of enforcing the usage of TLS for the transfer of authentication subscription data from the UDR to the UDM/ARPF. Therefore we suggest the following changes to clause 5.8.1 “Generic requirements” in TS 33.501 [1].

*NOTE 2: Security mechanisms for storage of subscription credentials in the UDR ~~and for the transfer of authentication subscription data (as specified in 3GPP TS 29.505 [70]) between UDR and ARPF~~ are left to implementation.*

Further, we add a note to point out the secure transport for this specific API endpoint.

*NOTE 3: The UDR should support the access and transport of authentication subscription data only through the security mechanisms defined in section 13.1.0.*

## Authorization Security

To limit the access to the permanent key stored in the UDR, we propose two directions:

1. **Deleting the nonsecurity mechanism:** We suggest to delete the nonsecurity mechanism ({}). From a security perspective, we also suggest to limit the access to those NFs who have access to the end-point nudr-dr:subscription-data:authentication-subscription:read. Also having the nudr-dr scope can have a wide audience gaining access to the security-sensitive key material. Therefore we would only see this one security definition for the endpoint /subscription-data/{ueId}/authentication-data/authentication-subscription: (get, patch, put). The final API specifcation could look like the following:

A close-up of a white background

Description automatically generated

1. **Extra visibility document:** The second approach involves defining the visibility of data when using the non-security parameter. The data returned depends on the authorization level provided to the API endpoint by the consumer. This could work in a similar way as the GitHub API, for example, curl https://api.github.com/users/3gpp will only return public information. If you provide an OAuth2.0 token, you can also access private information. However, the current OpenAPI specification does not allow you to define which data is returned for each authorization level. Therefore, the suggestion is to map the response parameter to the authorization level **outside** of the OpenAPI specification, in a supplementary document, such as the TS 29.505 [4].

## Implications for SCAS Test Cases for the UDR

In relation to the SCAS\_5G\_UDR work item, the decisions on how we secure the UDR API will have some impact on the UDR-specific test cases. For example, if we add specific requirements to the UDR, then they should be tested in the UDR-specific SCAS document. On the other hand, if an additional scope definition is proposed in the extra visibility document and if this scope definition is rather general, then the application of the scope definition could be tested via the TS 33.117 [2]. Nevertheless, it is important to discuss the requirements before deciding on the required SCAS actions.