**3GPP TSG-SA3 Meeting #103-e *S3-211713r1***

**e-meeting, 17- 28 May 2021**

**Source: Apple**

**Title: Kedge ID generation in solution#2**

**Document for: Approval**

**Agenda Item: 5.8**

1 Decision/action requested

***It is proposed to address the EN in solution#2 in MEC TR 33.839.***

2 References

[1] 3GPP TS 23.558: "Architecture for enabling Edge Applications (EA)"

3 Rationale

This pCR proposes to address the EN in solution#2

“Editor’s note: How to generate the Kedge ID in AUSF and UE is FFS.”

4 Detailed proposal

**\*\*\*\*START OF CHANGES \*\*\***

6.2 Solution #2: Authentication between EEC and ECS based on primary authentication

6.2.1 Introduction

This solution is addressing key issue#2-Authentication and Authorization between EEC and ECS.

This solution proposes the authentication between EEC (Edge Enabler Client) and ECS (Edge Configuration Server). To be more specific, it is proposed to use the Kausf derived from the primary authentication as the trust root to perform the authentication between EEC and ECS.

It is assumed in this solution that ECS is located outside of the MNO’s network.

6.2.2 Solution details

6.2.2.1 Procedure

Timeline

Description automatically generated

Figure-6.2.2.1-1. Authentication between the EEC and ECS based on primary authentication

The authentication procedure details are as following:

Step 0: UE performs primary authentication with the network. Then KAUSF is shared between UE and AUSF in Home network. UE performs PDU session establishment procedure as defined in TS 23.502.

Step 1: UE generates a credential Kedge and Kedge ID using KAUSF and SUPI, and stored securely. The method to derive generate Kedge and Kedge ID is in 6.2.2.2.

Step 2: AUSF generates a credential Kedge and Kedge ID using KAUSF and SUPI, and stored securely.

Step 3: UE computes MACEEC using the Kedge and EEC ID (defined in TS 23.558 [2]). The method to generate MACEEC is in 6.2.2.3.

Step 4: UE sends Application Registration request (EEC ID, MACEEC, Kedge ID) to ECS. Whether this message is send using NAS or user plane is based on SA2’s decision.

Step 5: ECS sends Authentication verification (EEC ID, MACEEC, Kedge ID) to NEF for verification.

Step 6: NEF discovers the AUSF based on Kedge ID, and sends Authentication verification (EEC ID, MACEEC, Kedge ID) to AUSF for MACEEC verification.

Editor’s Note: How to discover the AUSF is FFS.

Step 7: AUSF retrieves Kedge using Kedge ID, and verify MACEEC using the (Kedge and EEC ID).

Step 8: If verification in AUSF succeed, then AUSF sends Authentication verification response(success) back to NEF, otherwise, AUSF sends Authentication verification response(fail) to NEF.

Step 9: NEF sends Authentication verification response(success/fail) from AUSF to ECS.

Step 10: Based on the verification results, ECS decides whether to accept or reject the authentication request, and sends Authentication Request accept/rejection to EEC in the UE.

Editor’s Note: How the AUSF can be aware of each specific Kedge per UE is FFS

6.2.2.2 Derivation of Kedge and Kedge ID

Kedge is generated using KDF defined in Annex B.2.0 of TS 33.220 [8]. When deriving a Kedge from KAUSF, the following parameters should be used to form the input S to the KDF:

- FC = xxxx(to be allocated by 3GPP)

- P0 = <SUPI>,

- L0 = length of <SUPI>.

The input key KEY should be KAUSF.

Kedge ID is generated by AUSF and UE, and uniquely identify only one Kedge.

Kedge ID should contain the “routing indicator|| service identifier part || uniqueness part”.

* The routing indicator part is used by ECS to find the correct NEF,
* The service identifier part is used to represent different service, and it could be the string of the service name or the EAS ID defined in TS 23.558.
* The uniqueness part is used to make sure there is no collision among different Kedge IDs for different EECs.

Editor’s note: It is FFS how to generate the uniqueness part.

**\*\*\*\*END OF CHANGES \*\*\***