**3GPP TSG-SA3 Meeting #109AdHoc-e *draft\_S3-230224-r1***

**Electronic meeting, 16 - 20 January 2023**

**Source: Ericsson**

**Title: A new solution for EEC authentication utilizing tokens**

**Document for: Approval**

**Agenda Item: 5.9**

# 1 Decision/action requested

***It is proposed to approve the pCR to TR 33.739.***

# 2 References

[1] 3GPP TS 33.558: "Security aspects of enhancement of support for enabling edge applications"

[2] 3GPP TR 33.739: "Study on security enhancement of support for edge computing phase 2"

[3] GSMA TS.43: "Service Entitlement Configuration"

# 3 Rationale

The methods for authentication of EEC by the ECS/EES has been left as out of scope for Rel-17 in TS 33.558 [1]. In Rel-18 possible methods and negotiation mechanisms have been studied under key issue #2.1 and #2.2 in TR 33.739 [2], respectively.

Possible options proposed so far include AKMA, GBA and TLS client certificate. Also, how to handle negotiation failure cases are under discussion. To prevent negotiation failures, there can be two alternatives 1) the UE side or PLMN side have to support all the methods 2) at least one mandatory method should be specified. If none of the alternatives cannot be aggregable by SA3 working group, then it is possible to have negotiation failures in run time.

To handle the negotiation failures, two approaches have been proposed so far in the discussions. 1) no client authentication 2) TLS client certificates usage. It should be noted that TLS client certificates are not feasible in practice because it is not appropriate to use Internet PKI for many instances of applications running different UEs and to handle private PKIs for the setting that there are different players and providers. Thus, the only alternative becomes no client authentication. However, if no client authentication is specified as one option, then having other methods will not bring any security value because the malicious client will find a way to use no client authentication method.

To look to the problem from a different perspective, SA3 working group should also consider commonly used methods in the ecosystem for client authentications in UEs. One of the methods commonly used in the ecosystem is the methods based on token usage. There can be different methods how and where the client can receive a token. One example can be the utilization of the Service Entitlement Configuration mechanisms documented by GSMA in [3] where the Entitlement Configuration Server provides a token to the client to be used towards to the servers for related services.

This contribution proposes a high-level solution for EEC authentication using token-based mechanisms.

# 4 Detailed proposal

Approve the following changes to TR 33.739 [2].

\*\*\* Start of 1st Change \*\*\*

# 2 References

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

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

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

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

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

[2] 3GPP TR 23.700-48: "5G System Enhancements for Edge Computing; Phase 2".

[3] 3GPP TR 23.700-98: "Study on Enhanced architecture for enabling Edge Applications ".

[4] 3GPP TS 33.558: "Security aspects of enhancement of support for enabling edge applications".

[5] 3GPP TS 33.839: "Study on security aspects of enhancement of support for edge computing in the 5G Core (5GC)".

[6] 3GPP TS 33.220: "Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (GBA)"

[7] 3GPP TS 33.501: "Security architecture and procedures for 5G System"

[8] 3GPP TS 33.535: Authentication and Key Management for Applications (AKMA) based on 3GPP credentials in the 5G System (5GS)

[9] 3GPP TS 23.502: "Procedures for the 5G System (5GS)"

[10] 3GPP TS 33.222: "Generic Authentication Architecture (GAA); Access to network application functions using Hypertext Transfer Protocol over Transport Layer Security (HTTPS)".

[11] 3GPP TS 23.558: "Architecture for enabling Edge Applications."

[12] IETF RFC 8446: "The Transport Layer Security (TLS) Protocol Version 1.3".

[13] 3GPP TS 33.210: "3G security; Network Domain Security (NDS); IP network layer security".

[XX] GSMA TS.43: "Service Entitlement Configuration"

\*\*\* End of 1st Change \*\*\*

\*\*\* Start of 2nd Change \*\*\*

## 6.X Solution #X: Utilizing Token-Based Solutions for EEC authentication

### 6.X.1 Solution overview

The methods for authentication of EEC by the ECS/EES has been left as out of scope for Rel-17 in TS 33.558 [4]. In the present document, possible methods and negotiation mechanisms have been studied under key issue #2.1 and #2.2, respectively.

Possible options proposed so far include AKMA, GBA and TLS client certificate. Also, how to handle negotiation failure cases are under discussion. To prevent negotiation failures, there can be two alternatives 1) the UE side or PLMN side has to support all the methods 2) at least one mandatory method should be specified. If none of the alternatives cannot be aggregable by SA3 working group, then it is possible to have negotiation failures in run time.

To handle the negotiation failures, two approaches have been proposed so far in the discussions. 1) no client authentication 2) TLS client certificates usage. It should be noted that TLS client certificates are not feasible in practice because it is not appropriate to use Internet PKI for many instances of applications running different UEs and to handle private PKIs for the setting that there are different players and providers. Thus, the only alternative becomes no client authentication. However, if no client authentication is specified as one option, then having other methods will not bring any security value because the malicious client will find a way to use no client authentication method.

To look to the problem from a different perspective, commonly used methods in the ecosystem should be considered for client authentications in UEs. One of the methods commonly used in the ecosystem is the methods based on token usage. There can be different methods how and where the client can receive a token. One example can be the utilization of Service Entitlement Configuration mechanisms documented by GSMA in [XX] where the Entitlement Configuration Server provides a token to the client to be used towards to the servers for related services.

This contribution proposes a high-level solution for EEC authentication using token-based mechanisms.

### 6.X.2 Solution details

The EEC authentication is executed by a mechanism out of scope of this solution. Such a mechanism can be Service Entitlement Configuration specified in [XX] where the Entitlement Configuration Server issues a token for the client. After the authentication, the EEC receives a token to be used towards to the Edge Configuration Server (ECS) / Edge Enabler Server (EES).

Utilization of such a token-based mechanism can be embedded to the authentication between the EEC and the ECS/EES mechanism as follows.

1. The EEC gets a token after authenticating itself with the Entitlement Configuration Server.
2. For the ECS/EES authentication by the EEC, TLS server certificate is used. For EEC authentication by the ECS/EES, the EEC provides the token to the ECS/EES so that the ECS/EES accepts the request coming from the EEC.

Editor’s Note: Details on token retrieval, token content, token verification, and EEC configuration to support this solution are FFS.

### 6.X.3 Solution evaluation

This solution introduces another authentication method option, token-based approach, which is a method widely used in the ecosystem. The details of the token-based solution are not specified but an example is presented to skect how this method can be used.

\*\*\* End of 2nd Change \*\*\*