**3GPP TSG-SA3 Meeting #119AdHoc-e S3-250089-r2**

**Online, Electronic meeting, 13 -16 January 2025**

**Source: Samsung**

**Title: Updates to solution#6**

**Document for: Approval**

**Agenda item: 5.2**

**Spec: 3GPP TR 33.790**

**Version: 0.6.0**

**Work Item: FS\_NG\_RTC\_SEC\_Ph2**

**Comments**

In SA2#166, there was agreement on the avatar communication procedures for network centric, sending UE centric, and receiving UE centric. The procedures are captured in TS 23.228 [1]. This contribution proposes to update solution #6 to be aligned with the procedures in TS 23.228 [1]. The following Editor's Note can be resolved:

Editor's Note: The alignment with SA2 conclusions from TR 23.700-77 [2] is FFS.

**Proposed Changes**

\* \* \* First Change \* \* \* \*

## 6.6 Solution #6: Solution for secure IMS based avatar communication

### 6.6.1 Introduction

This solution addresses key issue #2: Security of IMS based Avatar Communication.

According to TR 23.700-77 [2], rendering of avatar media can be performed by network, UE1, or UE2, which are called network centric IMS avatar call, sending UE centric IMS avatar call, and receiving UE centric avatar call, respectively.

In this solution, only authorized entity from UE1 can access and retrieve the UE1's avatar representation which is stored in a data storage called Base Avatar Repository (BAR). This solution proposes to use UE1's attestation for the security procedure of IMS based avatar communication.

In this solution, it is assumed that BAR has CA certificate to verify UE1's (UE2's) certificate, with UE1's (UE2's) avatar representation and avatar ID. The UE certificate for media plane security can be reused to generate the signature in the attestation. In this solution, only unidirectional avatar communication is described. When bidirectional avatar communication is used (i.e., when UE1's avatar and UE2's avatar are sent to each other), UE2 also performs the operation same as UE1 described in this solution.

NOTE: Which entity acts as certificate authority is out of scope of this solution.

### 6.6.2 Solution details

#### 6.6.2.1 Network centric IMS avatar call flow



Figure 6.6.2.1-1 Network centric IMS avatar call flow

1. The UE1 initiates an IMS session and establishes audio and video session connections with the UE2. The bootstrap channel is established for both the UE1 and UE2. UE1 obtains UE1's avatar id(s) through bootstrap data channel.

2. The UE1 decides to request network to perform avatar animation based on its status such as power, signal, computing power, internal storage, etc.

3. The UE1 performs the avatar animation negotiation with the DC AS. The UE1 generates UE1's attestation. The UE1's attestation consists of Avatar ID, rendering option (i.e., network centric), expiration time, signature generated by using UE1's private key, as described in clause 6.6.2.4 of this document. The UE1 sends UE1's attestation and UE1's certificate to the DC AS via DCSF. Before sending the attestation, DCSF may check whether the UE1 is allowed to use the avatar ID after reading the avatar ID in the UE1's attestation.

4. P2A2P Application data channel is established between UE1/UE2 and DC AS, and media re-negotiation between UE1, UE2, and MF may be performed. In this step, DC AS sends UE1's attestation and UE1's certificate to MF as described in step 7 in clause AC.11.3.3 of TS 23.228 [7].

5. MF requests avatar representation using UE1's attestation and UE1's certificate.

6. After BAR verifies the UE1's certificate, BAR verifies the signature in UE1's attestation using the UE1's public key. If the verification is successful, the BAR responds with the UE1's avatar representation.

NOTE 1: When BAR and DC AS are outside of IMS network, step 5-6 is up to implementation.

NOTE 2: Protection between BAR and DC AS is out of scope of this solution.

7. The UE1 sends avatar metadata for the rendering (e.g., facial feature points).

8. The MF/MRF performs the media transcoding and animates the UE1's avatar representation.

10. The animated avatar media is sent as regular video media to UE2.

11. The animated avatar media is sent back to the UE1 as feedback.

#### 6.6.2.2 Sending UE centric IMS avatar call flow

Figure 6.6.2.2-1 Sending UE centric IMS avatar call flow

1. The UE1 initiates an IMS session and establishes audio and video session connections with the UE2. The bootstrap channel is established for both the UE1 and UE2. UE1 obtains UE1's avatar id(s) through bootstrap data channel.

2. Application data channel is established between UE1 and DC AS, and media re-negotiation may be performed.

3. The UE1 generates UE1's attestation. The UE1's attestation consists of Avatar ID, rendering option (i.e., sending UE centric), expiration time, signature generated by using UE1's private key, as described in clause 6.6.2.4 of this document. The UE1 sends UE1's attestation and UE1's certificate to the DC AS to download UE1's avatar representation.

4. DC AS may check whether the UE1 is allowed to use the avatar ID after reading the avatar ID in the UE1's attestation. DC AS requests avatar representation using UE1's attestation and UE1's certificate.

5. After BAR verifies the UE1's certificate, BAR verifies the signature in UE1's attestation using the UE1's public key. If the verification is successful, the BAR responds with the UE1's avatar representation.

6. The DC AS sends the avatar representation to UE1.

NOTE: If UE1 is already storing its own avatar representation locally, step 3-6 may be skipped.

7. The UE1 performs avatar animation.

8. The UE1 transmits the animated video stream over RTP to UE2.

#### 6.6.2.3 Receiving UE centric IMS avatar call flow



Figure 6.6.2.3-1 Receiving UE centric IMS avatar call flow

1. The UE1 initiates an IMS session and establishes audio and video session connections with the UE2. The bootstrap channel is established for both the UE1 and UE2. UE1 obtains UE1's avatar id(s) through bootstrap data channel.

2. P2A2P Application data channel is established between UE1/UE2 and DC AS, and media re-negotiation may be performed.

3. UE1 decides to request UE2 to perform avatar animation based on its status such as power, signal, computing power, internal storage, etc.

4. UE1 performs avatar animation negotiation with the DC AS and UE2. The UE1 obtains UE2's ephemeral public key. The UE1 generates UE1's attestation. The UE1's attestation consists of Avatar ID, rendering option (i.e., receiving UE centric), ephermeral public key of UE2, expiration time, signature generated by using UE1's private key, as described in clause 6.6.2.4 of this document. The UE1 sends UE1's attestation and UE1's certificate to the UE2.

Editor's Note: Whether the information (e.g., ephemeral public key of UE2, UE1's attestation, UE1's certificate, etc.) can be exchanged in step 4 is FFS.

5. UE2 downloads UE1's avatar representation from DC AS using the UE1's attestation and UE1's certificate.

6. DC AS may check whether the UE1 is allowed to use the avatar ID after reading the avatar ID in the UE1's attestation. DC AS requests avatar representation using UE1's attestation and UE1's certificate.

Editor's Note: Whether DC AS needs to check whether UE2 is allowed to download UE1's avatar representation is FFS.

7. After BAR verifies the UE1's certificate, BAR verifies the signature in UE1's attestation using the UE1's public key. If the verification is successful, the BAR generates ephemeral public/private key pair and protects the UE1's avatar representation using the session key generated by ephemeral public key of UE2 included in the UE1's attestation and ephemeral private key of BAR. The BAR responds with the protected UE1's avatar representation and ephemeral public key of the BAR.

8. The MF sends the protected avatar representation and ephemeral public key of the BAR to UE2.

9. The UE1 sends avatar metadata to UE2.

10. The UE2 generates session key using the ephemeral public key of the BAR and the ephemeral private key of UE2. UE2 verifies the protected avatar representation using the session key and performs the UE1's avatar representation animation using the UE1's avatar representation and the avatar metadata received from UE1 in step 9.

Editor's Note: Whether it is necessary to protect the privacy of an avatar representation is FFS.

Editor's Note: The alignment with TS 23.228 [7] is FFS.

#### 6.6.2.4 UE1 attestation

UE1's attestation is generated by UE1 and it consists as follows:

Table 6.6.2.4-1 UE1 attestation

|  |  |
| --- | --- |
| Parameter | Description |
| Avatar ID | REQUIRED. ID used to retrieve UE1's avatar representation. |
| Rendering option | REQUIRED. One of the followings: network centric, sending UE centric, or receiving UE centric |
| Ephemeral public key of UE2 | OPTIONAL. This is included when the rendering option is receiving UE centric. BAR uses ephemeral public key of UE2 and ephemeral private key of BAR to protect the UE1's avatar representation. The protected avatar representation is sent to UE2 and it is end-to-end protected between BAR and UE2. |
| Expiration time | REQUIRED. The expiration time of the UE1 attestation. |
| Signature | REQUIRED. Signature generated by UE1 using UE1's public key and the parameters in the attestation. |

\* \* \* \* End of Changes \* \* \* \*