**3GPP TSG-SA3 Meeting #120 S3-25xxxx**

Athens, Greece, 17 - 21 February 2025

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.1* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  |  | **CR** | ***draft*** | **rev** | **-** | **Current version:** |  |  |
|  | | | | | | | | |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* | | | | | | | | |
|  | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME | **X** | Radio Access Network |  | Core Network | **X** |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | Onboarding API Invoker Residing in the UE | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | Lenovo | | | | | | | | | |
| ***Source to TSG:*** | S3 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** |  | | | | |  | ***Date:*** | | | 2024-03-31 |
|  |  | | | |  | |  | | |  |
| ***Category:*** |  |  | | | | | ***Release:*** | | | Rel-19 |
|  | *Use one of the following categories:* ***F*** *(correction)* ***A*** *(mirror corresponding to a change in an earlier release)* ***B*** *(addition of feature),* ***C*** *(functional modification of feature)* ***D*** *(editorial modification)*  Detailed explanations of the above categories can be found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | | | | | | | | *Use one of the following releases: Rel-8 (Release 8) Rel-9 (Release 9) Rel-10 (Release 10) Rel-11 (Release 11) … Rel-15 (Release 15) Rel-16 (Release 16) Rel-17 (Release 17) Rel-18 (Release 18)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | As per TS 23.222 Clause 8.1.1, ‘The CAPIF enables a one time onboarding process that enrolls the API invoker as a recognized user of the CAPIF, which may be triggered by the API invoker via CAPIF-1 or CAPIF-1e, or may be based on provisioning’. Further Annex F clarifies, the API invoker can be Application servers or BSS/OSS system which invokes CAPIF APIs as per its business logic as there exits a business relationship. Existing Onboarding procedure specifed in TS 33.122 works for such case which depends on the pecondition that there exists an Onboarding enrollment information as in TS 33.122 Clause 6.1. Existing Onboarding performs only server side authentication and further relies upon the Onboarding enrollment information to verify and if succesful provides the API Invoker profile. Also existing Onboarding allows other credentials (e.g. message digest). But for the case where the API invoker resides as part of the UE, the only business relationship that exists between the UE and the Operator network is the subscription. Moreover the existing Onbaording procedure has no means to verify if the API invoker is the one that resides/belongs to the UE leading to the threats listed in TR 33.700-22 Clause 5.6 and it also has similar issues as described in TS 23.222 Annex G but while Onboarding of the API invoker of the UE. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | CCF verifies if the API invoker is part of the UE before onboarding the API invoker of the UE to the CAPIF, so that CAPIF can consider the onboarded API invoker as the recognozed user of the CAPIF. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | Any API invoker without being part of a UE will onboard to the CAPIF and can falsely claim as the API invoker residing in the UE causing the trheats | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 6.1, 6.8 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **Y** | **N** |  | | | |  | | |
| ***Other specs*** | |  | **X** | Other core specifications | | | | TS/TR ... CR ... | | |
| ***affected:*** | |  | **X** | Test specifications | | | | TS/TR ... CR ... | | |
| ***(show related CRs)*** | |  | **X** | O&M Specifications | | | | TS/TR ... CR ... | | |
|  | |  | | | | | | | | |
| ***Other comments:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | |  | | | | | | | | |

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

## 6.1 Security procedures for API invoker onboarding

The API invoker and the CAPIF core function shall follow the procedure in this subclause to secure and authenticate the onboarding of the API invoker to the CAPIF core function. The API invoker and the CAPIF core function shall establish a secure session using TLS. Security profiles for TLS implementation and usage shall follow the provisions given in TS 33.310 [2], Annex E .

With a secure session established, the API Invoker sends an Onboard API Invoker Request message to the CAPIF core function. The Onboard API Invoker Request message carries an onboard credential obtained during pre-provisioning of the onboard enrolment information, which may be an OAuth 2.0 [4] access token. When the OAuth 2.0 token based mechanism is used as the onboarding credential, the access token shall be encoded as JSON web token as specified in IETF RFC 7519 [6], shall include the JSON web signature as specified in IETF RFC 7515 [7], and shall be validated per OAuth 2.0 [4], IETF RFC 7519 [6] and IETF RFC 7515 [7]. Other credentials may also be used (e.g. message digest).

Figure 6.1-1 details the security information flow for the API invoker onboarding procedure. The OAuth 2.0 token based authentication credential is shown in this example.



Figure 6.1-1: Security procedure for API invoker onboarding

1. As a prerequisite to the onboarding procedure, the API invoker obtains onboarding enrolment information from the API provider domain. The onboarding enrolment information is used to authenticate and establish a secure TLS communication with the CAPIF core function during the onboarding process. The enrolment information includes details of the CAPIF core function (Address, and Root CA certificate) and includes an onboarding credential (the OAuth 2.0 [4] access token).

If the API invoker resides as part of the UE, the onboarding credential is bound to the UE ID i.e, GPSI.

NOTE 1: The procedure used to obtain the enrolment information by the API invoker is out of scope of the present document.

2. The API invoker and CAPIF core function shall establish a secure session based on TLS (Server side certificate authentication). The API invoker shall use the enrolment information obtained in step 1 to establish the TLS session with the CAPIF core function.

3. After successful establishment of the TLS session, the API invoker shall send an Onboard API invoker request message to the CAPIF core function along with the enrolment credential (OAuth 2.0 [4] access token). The API invoker generates the key pair {Private Key, Public key} and provides the public key along with the Onboard API invoker request.

If the API invoker resides as part of the UE, the Onboarding API invoker request additionally includes the UE ID i.e, GPSI, where the Onboarding credential includes either an OAuth 2.0 access token or message digest to be used as an authentication code to verify if the API invoker is part of the UE.

4. The CAPIF core function shall validate the enrolment credential (OAuth 2.0 [4] access token). If validation of the credential (the OAuth 2.0 [4] access token in this example) is successful, the CAPIF core function shall generate an API invoker's profile as specified in TS 23.222 [3] which may contain the selected method for AEF authentication and authorization between the API Invoker and the AEF (see subclause 6.5.2). The CAPIF core function may generate API invoker's certificate on its own, for the assigned API invoker identity and public key. This certificate shall be used by the API invoker for subsequent authentication procedures with the CAPIF core function and may be used for establishing a secure connection and authentication with the API Exposing Function. The CAPIF core function may optionally generate an Onboard\_Secret if the subscribed Service API uses Method 3 (as specified in clause 6.5.2.3 of the present document) for CAPIF-2e security. The Onboard\_Secret value remains the same during the lifetime of the onboarding, and shall be bound to the CAPIF core function specific API Invoker ID.

For the case of API invoker residing at UE, the CAPIF core function, on receiving the GPSI, validates the enrolment credential (i.e, OAuth 2.0 access token or message digest whichever is received) and if the validation is successful (e.g., if the GPSI bounded access token claims or message digest verification succeeds), the CAPIF core function generates the API invoker's profile and Onboard\_Secret as described above for step 4 and additionally bound to the UE ID i.e., GPSI.

NOTE: For the case of API invoker being part of the UE, the enrolment credential generation and validation by the CAPIF core function is upto the implementation.

NOTE 2: When API invoker's client certificate is issued by the third party, then in Step 3 the API invoker can additionally include the certificate in Onboard API Invoker request message. If the CAPIF core function trusts the issuer of the API invoker's client certificate, then the CAPIF Core Function includes the provided certificate in the API invoker's profile, in step 4. It is up to the CAPIF domain policy to accept the client certificates issued by third party.

5. The CAPIF core function shall respond with an Onboard API invoker response message. The response shall include the CAPIF core function assigned API invoker ID, AEF Authentication and authorization information (if generated in step 4), API invoker's certificate and the API invoker Onboard\_Secret (if generated by the CAPIF core function).

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

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

## 6.8 Security procedure for API invoker offboarding

Pre-conditions:

1. The API invoker has been onboarded successfully.
2. For the case API Invoker residing at the UE, during onboarding the CAPIF Core Function successfully verified that the API invoker is part of the UE.



Figure 6.8-1: Security procedure for API invoker offboarding

0. TLS session is established successfully between the CAPIF core function and the API invoker.

1. An event occurs within the API invoker to trigger the offboarding action.

NOTE: The definition of events that trigger offboarding is outside the scope of the present document.

2. The API invoker shall send Offboard API invoker request message to the CAPIF core function, including the CAPIF core function specific API invoker ID which was assigned by the CAPIF core function during the onboarding procedure.

3. The CAPIF core function shall verify the API invoker ID received in step 2 and check that the corresponding profile exists for this API invoker. With successful verification of the API invoker ID and its profile, the CAPIF core function shall cancel the enrolment of the API invoker and delete the API invoker profile. This includes deletion of API invoker certificate, service API authentication and authorization information, and onboard secret (if applicable). Depending on the operator policy, the CAPIF core function may retain the information of the offboarded API invoker.

4. The CAPIF core function sends Offboard API invoker response message, indicating the successful offboarding of the API invoker.

5. The API invoker shall delete the information, such as API invoker ID, Service API authentication / authorization information, API invoker certificate, Onboard\_Secret (if applicable).

6. The CAPIF core function shall tear down the TLS session with the API invoker.

7. The CAPIF core function shall send Event notification message to the API exposing function to indicate that this API invoker is no longer valid.

8. The API exposing function shall delete the security related information associated with this API invoker depending on the method that was used previously to authenticate the API invoker, e.g. AEFPSK (TLS-PSK method as described in subclause 6.5.2.1), root certificate to validate the API invoker certificate (PKI method as described in subclause 6.5.2.2), access token (OAuth 2.0 method as described in subclause 6.5.2.3 of the present document, respectively).

9. The API exposing function shall tear down the TLS connection with the API invoker.

10. The API exposing function shall return Event notification acknowledge message to indicate that the security related information associated with this API invoker is successfully deleted and thus the API invoker no longer an acknowledged user.

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