|  |
| --- |
| Input Contribution |
| Meeting ID\* | TST#33 |
| Title:\* | Input on “Registration upon successful SAE” procedure for the Security Developer’s Guide TR-0038 |
| Source:\* | Wolfgang Granzow, Qualcomm, wgranzow@qti.qualcomm.comPhil Hawkes, Qualcomm, phawkes@qti.qualcomm.com |
| Uploaded Date:\* | 2018-01-15 |
| Document(s) Impacted\* | TR-0038v0\_3\_0 |
| Intended purpose ofdocument:\* | [x]  Decision[x]  Discussion[ ]  Information[ ]  Other <specify> |
| Decision requested or recommendation:\* | Include the text proposed in this contribution into TR-0038v0.3.0 |
| Template Version:23 February 2015 (Do not modify) |

**oneM2M Notice**

The document to which this cover statement is attached is submitted to oneM2M. Participation in, or attendance at, any activity of oneM2M, constitutes acceptance of and agreement to be bound by terms of the Working Procedures and the Partnership Agreement, including the Intellectual Property Rights (IPR) Principles Governing oneM2M Work found in Annex 1 of the Partnership Agreement.

**Introduction**

This contribution proposes initial text for section 7.1.5 of TR-0038.

*======== start of change 1 =============================*

### 7.1.5 Registration upon successful SAE

An AE or CSE which has not registered to its registrar CSE yet, should be pre-configured such that it attempts to perform a registration procedure right after the device is powered on and after it has established network connectivity. In this case the first request primitive sent by an AE or CSE entity via Mca or Mcc interfaces is either a “Create <*AE*>” or “Create <*remoteCSE*>” request primitive, respectively.

When an AE registers, the registrar CSE needs to retrieve and check the service subscription information which is defined in a <*m2mServiceSubscriptionProfile*> instance on the IN-CSE (see clause 10.2.2.2 of TS-0001 [i.2]).

NOTE: In the present Release, <serviceSubscribedAppRule>does not allow to validate the association between CSE registrees and their applicable credential identifiers when registering to their registrar CSE.

For the use case example illustrated in figure 6.1-1, the overall structure of service subscription information may look as shown in figure 7.1.5-1. It is assumed that these resources exist on the IN-CSE prior to the registration procedure. Their creation is out of scope of the present document.

The instance of a <*m2mServiceSubscriptionProfile*> with its children and linked resources as shown in figure 7.1.5-1 includes all information exposed on the Mcc interface related to a service subscription of the subscriber who owns and operates the considered example home network in figure 6.1-1. An <*m2mServiceSubscriptionProfile*> resource does not include any resource-specific attributes itself. It acts as parent of all <*serviceSubscribedNode*> resources related to a specific subscriber. Every node shown in figure 6.1-1, i.e. Front Door Lock, Back Door Lock, Smartphone, Home Gateway and Cloud Infrastructure, may have an associated instance of a <*serviceSubscribedNode*> child resource configured. However, the <*serviceSubscribedNode*> resource of an ADN only includes the *nodeID* attribute, which is relevant for Device Management procedures but irrelevant in the context of the registration procedure. Therefore figure 7.1.5-1 shows <*serviceSubscribedNode*> resources related to the MN and IN only. These include in addition to the *nodeID* attribute a *CSE-ID* and a *ruleLinks* attribute. The *CSE-ID* relates to the CSE of the node identified by the *nodeID* attribute.

The *ruleLinks* attribute assign <*serviceSubscribedAppRule*> resources to a <*serviceSubscribedNode*> resourcs. (in terms of a list of their *resourceID* values). In the specific example considered here, it is assumed that there is one <*serviceSubscribedAppRule*> resource instance configures for each AE which is allowed to register to a given CSE.

In the example considered in figure 6.1-1, the Home Gateway (MN) hosts three registree AEs: ADN-AE1, ADN-AE2 and MN-AE. Therefore, the <*serviceSubscribedNode*> resource associated with the Home Gateway may have 3 different <*serviceSubscribedAppRule*> resources assigned, one for each AE shown in figure 6.1-1. The service subscriber employs ADN-AE3 as door lock controller which registers to the IN-CSE directly. The resource tree in figure 7.1.5-1 therefore also includes a <*serviceSubscribedNode*> resource associated with the IN. This <*serviceSubscribedNode*> reveals *nodeID* and *CSE-ID* of the IN-CSE and it is assumed to have a *ruleLink* attribute which includes the resource identifier of a <*serviceSubscribedAppRule*> resource which includes information related to ADN-AE3.

The <*serviceSubscribedAppRule*> resource can have 3 specific attributes: *allowedCredIDs*, *allowedAppIDs* and *allowedAEs*. Each of these attributes generally can include a list of elements. If a <*serviceSubscribedAppRule*> relates to a single AE only, the *allowedAppIDs* and *allowedAEs* attributes contain a single element only.

Table 7.1.5-1 shows a suitable setting of these attributes for each of the three <*serviceSubscribedAppRule*> resources.

For instance the column with heading ADN-AE1, shows the attributes of the <*serviceSubscribedAppRule*> resource which relates to ADN-AE1. In this case, the *allowedAEs* attribute includes the AE-ID stem (to be) assigned to ADN-AE1, which is Clock-AE1, as used in the example in clause 7.1.2. The *allowedAppIDs* attribute includes the App-ID and the *allowedCredIDs* attribute includes security credential identifiers applicable for ADN-AE1. The columns with headings ADN-AE2 and ADN-AE3 of Table 7.1.5-1 shows the set of applicable parameters for those respective AEs.

At <AE> registration, information included in applicable <*serviceSubscribedAppRule*> resources is examined by the registrar CSE and compared if it matches with security credentials employed for Security Association Establishment (SAE), and App-ID and AE-ID indicated in the registration request message.

In case the information used by the registree does not match with the information given in applicable <*serviceSubscribedAppRule*> resources, the registration request needs to be rejected by the registrar CSE.

Note that if no applicable <*serviceSubscribedAppRule*> resources are configured, all registration requests passing Security Association Establishment successfully may be granted by the registrar.

Figure 7.1.5-2 outlines the message and processing flow related to Security Association Establishment as described in clauses 7.1.2, 7.1.3 and 7.1.4 for ADN-AE1 and ADN-AE2, MN-CSE and ADN-AE3, respectively, and the subsequent registration procedures, where service subscription information is evaluated. The description under the figure describes each step of the message and processing sequence.



Figure 7.1.5-1: Service subscription information stored on the IN-CSE for the use case in Fig. 6.1-1

**Table 7.1.5-1: Value setting of <serviceSubscribedAppRule> attributes**

|  |  |  |  |
| --- | --- | --- | --- |
| **Attribute** | **ADN-AE1** | **ADN-AE2** | **ADN-AE3** |
| *allowedCredIDs* | 12-AE123456789012-Lock@in.provider.com | 12-AE123456789015-Lock@in.provider.com | 32-\*@myMAF.provider.org |
| *allowedAppIDs* | doorlock-123 | doorlock-123 | lockControl**-**ABC12 |
| *allowedAEs* | Clock-AE\* | Clock-AE\* | C-lockControl-AE\* |



Figure 7.1.5-2: Message sequence of Security Association Establishment and registration procedures

1) A Security Association between MN-CSE and IN-CSE is established as described in clause 7.1.3 using public key certificates.

2) The MN-CSE sends a registration request message to the IN-CSE. Note that the MN-CSE must register to the IN-CSE before any AEs can register to the MN-CSE.

3) The IN-CSE checks the content of the registration request message (i.e. create <remoteCSE> request) as specified in clause 10.2.2.6 of TS-0001 (Rel-3) and clause 7.4.4 of TS-0004 (Rel-3). At CSE registration, there is no check of <*serviceSubscribedAppRule*> resources required.

 *Editor’s Note: TS-0004 states: “The Hosting CSE shall check if the credential provided by the Originator is valid. If the check is failed, then the Hosting CSE shall return response primitive with a Response Status Code indicating "SECURITY\_ASSOCIATION\_REQUIRED" error.“*

4) The registrar IN-CSE replies with a Registration response message. For the following steps it assumed that the MN-CSE registration was successful (Response Status Code 2001 “CREATED”).

5) ADN-AE1 (and ADN-AE2) establish a security association with the MN-CSE using the procedure described in clause 7.1.2, using symmetric key credentials.

6) The AE sends a registration request message to the MN-CSE. The registration request may or may not include an AE-ID in the ***From*** parameter and in the <AE> resource representation included in the ***Content***.

7) The AE registration procedure in clause 10.2.2.2 of TS-0001 (Rel-3) defines different processing cases depending in what information about AE-ID is provided with the request message. Here, it is assumed that the registering node already has an AE-ID preconfigured. Also the App-ID of the <AE> resource is indicated in the ***Content*** of the request message. In this step, the MN-CSE needs to check if there is a <*serviceSubscribedNode*> resource configured on the IN-CSE applicable to the MN-CSE, i.e. a resource instance which includes the CSE-ID assigned to the MN-CSE. This information can be obtained with a filtered retrieve request sent by the MN-CSE to its registrar IN-CSE, as described in clause 10.2.2.2 of TS-0001 (Rel-3). Once retrieved, the MN-CSE needs to retrieve any applicable <*serviceSubscribedAppRule*> resources as indicated in the *ruleLinks* attribute of the <*serviceSubscribedNode*> resource. In the example considered here, the MN-AE retrieves the <*serviceSubscribedAppRule*> with the setting for ADN-AE1 (or ADN-AE2) as shown in table 7.1.5-1. It then compares whether or not:

(i) the AE-ID given in the *allowedAEs* attribute matches the AE-ID given in the registration request,

(ii) the App-ID given in the *allowedAppIDs* attribute matches the App-ID given in the ***Content*** of the request,

(iii) the credential-ID given in the *allowedCredIDs* attribute matches the security credential which has been used in the SAE procedure (in this example the symmetric key credential derived from KpsaID as shown in table 7.1.2-1). A credential-ID included in the *allowedCredIDs* attribute is comprised of two parts:

* a credential-ID type identifier (CredIDTypeID, defined in Table 12.3.2.1-1 of TS-0001 [x]. In this example CredIDTypeID = 12 indicates that PSK-based SAE is used.
* a specific identifier of the allowed security credential, which is Kpsa for the given CredIDTypeID. The format of Kpsa is defined in clause xx of TS-0001 [x].

8) If any of the above checks fails, the registration request should be rejected with a respective error response. If the AE indicates the AE-ID and App-ID as given in the applicable <*serviceSubscribedAppRule*> the registration request can be granted with a successful response (Response Status Code 2001 “CREATED”).

9) ADN-AE3 establishes a security association with the MN-CSE using the procedure described in clause 7.1.4, using MAF-assigned symmetric key credentials. Note that this step and the subsequent registration procedure is independent of the previous steps and can occur at any time within the message sequence.

10) The ADN-AE3 sends a registration request message to the IN-CSE which is assumed to include preassigned AE-ID and App-ID.

11) Similarly as in step 7), the IN-CSE evaluates the registration requests. Since the IN-CSE is the host of any service subscription related resource, if configured, it is locally available and does not need to be retrieved via the Mcc interface. The IN-CSE performs the same checks between AE-ID and *allowedAEs*, AppID and *allowedAppIDs* and credential-IDs used in the SAE procedure and *allowedCredIDs* and as described in step 7)

12) If any of the above checks fails, the registration request should be rejected with a respective error response. If the AE indicates AE-ID and App-ID as given in the applicable <*serviceSubscribedAppRule*> the registration request can be granted with a successful response (Response Status Code 2001 “CREATED”).

***open issues:***

* *how is the mandatory nodeID attribute of <serviceSubscribedNode> derived at resource creation?*
* *How can credential-IDs be defined for certificates? credIDTypeID also defines types for certificates: 3x – 4x?*
* *Using an AE-certificate requires knowledge of the AE-ID stem. can we allow certificates which use wildcard character “\*” ?*
* *service subscription not applicable to check CSE registration credentials?*
* *We currently don’t have certificates defined that can include the App-ID. Text in TS-0001:*

*In case the Security Association Establishment in Step 001 was performed using security credentials in form of a Certificate that included an App-ID and an AE-ID-Stem attribute, check if they match with the App-ID attribute in the* ***Content*** *parameter of the request and the AE-ID-Stem in the* ***From*** *parameter of the request.*

Subsequent to successful registration, an AE can send any other request primitives. In such transactions, the receiver of any request message should perform a procedure denoted *AE impersonation prevention* (see clause 7.2 of TS-0003). For each received request message, the receiver shall check if the AE-ID in the ***From*** parameter is associated with the credentials used for security association establishment.

*======== End of change 1 =============================*