|  |
| --- |
|  |

|  |  |
| --- | --- |
| CHANGE REQUEST | |
| Meeting ID:\* | SEC 32 |
| Source:\* | Dale Seed, Convida Wireless, [Seed.Dale@ConvidaWireless.com](mailto:Seed.Dale@ConvidaWireless.com) |
| Date:\* | 2017-11-07 |
| Reason for Change/s:\* | See Introduction |
| CR against: Release\* | Release 3 |
| CR against: WI\* | Active - WI-0073 – App-ID Registry Function  MNT maintenance / < Work Item number(optional)>  Is this a mirror CR? Yes  No  mirror CR number: (Note to Rapporteur - use latest agreed revision)  STE Small Technical Enhancements / < Work Item number (optional)>  Only ONE of the above shall be ticked |
| CR against: TS/TR\* | TR-0048 Version 0.0.2 |
| Clauses \* | Section 6.1 |
| Type of change: \* | Editorial change  Bug Fix or Correction  Change to existing feature or functionality  New feature or functionality  Only ONE of the above shall be ticked |
| Impacted other TS/TR(s) | <TS/TR number>, <Version Number>, and <Description on which aspect should be reflected in this TS/TR> |
| Post Freeze checking:\* | This CR contains only essential changes and corrections? YES  NO  This CR may break backwards compatibility with the last approved version of the TS? YES  NO |
| Template Version: January 2017 (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.

GUIDELINES for Change Requests:

Provide an informative introduction containing the problem(s) being solved, and a summary list of proposals.

Each CR should contain changes related to only one particular issue/problem.

In case of a correction, and the change apply to previous releases, a separate “mirror CR” should be posted at the same time of this CR

Mirror CR: applies only when the text, including clause numbering are exactly the same.

Companion CR: applies when the change means the same but the baselines differ in some way (e.g. clause number).

Follow the principle of completeness, where all changes related to the issue or problem within a deliverable are simultaneously proposed to be made E.g. A change impacting 5 tables should not only include a proposal to change only 3 tables. Includes any changes to references, definitions, and acronyms in the same deliverable.

Follow the drafting rules.

All pictures must be editable.

Check spelling and grammar to the extent practicable.

Use Change bars for modifications.

The change should include the current and surrounding clauses to clearly show where a change is located and to provide technical context of the proposed change. Additions of complete clauses need not show surrounding clauses as long as the proposed clause number clearly shows where the new clause is proposed to be located.

Multiple changes in a single CR shall be clearly separated by horizontal lines with embedded text such as, start of change 1, end of change 1, start of new clause, end of new clause.

When subsequent changes are made to content of a CR, then the accepted version should not show changes over changes. The accepted version of the CR should only show changes relative to the baseline approved text.

## Introduction

This contribution proposes candidate architectures for the App-ID Registry functionality.

### -----------------------Start of change 1-------------------------------------------

# 6 Candidate Architecture

## 6.1 Architecture Proposal For Registration Enrolment using the App-ID Metadata

### 6.1.1 Functionality Requirements

The following is a list of basic requirements to be considered in design and analysis of Registration Enrolment using the App-ID Metadata solutions:

* Allow a Registree AE to enroll and be provisioned with a unique App-ID from an App-ID Registry
* Allow a Registrar CSE to perform an App-ID Registry lookup to check whether an App-ID it receives within a registration request is valid and has been assigned to the Registree.

## 6.1.2 Architecture Option #1

A candidate architecture is shown below in Figure 6.1.1-1. In this architecture, an independent App-ID Registry Function (ARF) and an ARF Client are defined. The ARF Client can be hosted by Registree and Registrar nodes. A Registree node can use it’s ARF Client to enrol to an ARF and be provisioned with App-ID(s) for the AE(s) the node hosts. A CSE hosted on a Registrar node can use it’s ARF Client to perform a lookup to the ARF to check whether a Registree AE’s App-ID is valid and in turn whether the Registrar CSE should grant or deny the Registree AE’s registration request. Using this lookup capability and the information returned by the ARF, the Registrar CSE can initiate the dynamic and remote provisioning of AE service subscription information (i.e. <serviceSubscribedAppRule> resources).



Figure 6.1.1-1: Standalone App-ID Registry Function

**Step 1:** ADN/ASN/MN’s MEF Client enrols to MEF as defined in TS-0003 and TS-0032 (e.g. <mefClientReg> and <symmKeyReg> resources are created).

**Step 2:** ADN/ASN/MN’s App-ID Registry Function (ARF) Client enrols to ARF such that ADN is provisioned with an App-ID.

**Step 3:** Security Association Establishment is performed

* **Step 3a:** AE initiates a Security Association Establishment handshake with its Registrar CSE. Depending on the type of security used this handshake may contain security related information. E.g. in the case of Provisioned Symmetric Key (PSK) based security, an identifier (KpsaID) is included.
* **Step 3b:** Registrar CSE formulates a Credential-ID based on the type of security that is used. E.g. in the case of PSK, the Credential-ID is formulated by pre-pending KpsaID with a value of “12-“. Based on the Credential-ID, the Registrar checks whether it has been provisioned with the Registree’s credential.
* **Step 3c:** If the Registrar CSE has not been provisioned with the Registree’s credential, the Registrar can initiate a lookup to the MEF to retrieve it based on the Registree’s Credential-ID.
* **Step 3d:** If the Registrar CSE has authorization to access the Registree’s credential, the MEF can respond with the credential.
* **Step 3e:** Using the credential, the Registrar can authenticate the Registree and complete the Security Association Establishment handshake.

**Step 4:** If the Security Association Establishment is successful, the AE can initiate a registration request to the Registrar CSE. Within this request, the Registrar provides its App-ID.

* **Step 4a:** If the Security Association Establishment is successful, the AE can initiate a registration request to the Registrar CSE. Within this request, the Registrar provides its App-ID
* **Step 4b:** Upon receiving the registration request from the AE, the Registrar CSE will check whether it has a matching <serviceSubscribedAppRule> resource for the AE. To find a match, the Registrar checks whether the Credential-ID and App-ID of the Registree AE match the *applicableCredIDs* and *allowedAppIDs* attributes of any <serviceSubscribedAppRule> resources it hosts. Also, if the registration request includes an AE-ID in the ***From*** request parameter, the Registrar CSE will also check whether this AE-ID matches the *allowedAEIDs* attribute of the <serviceSubscribedAppRule> resource. If a match is found, then the Registrar CSE will allow the AE to register.
* **Step 4c:** If a matching <serviceSubscribedAppRule> resource is not found for the AE, the Registrar can initiate a lookup to the ARF to check whether the AE has enrolled and been assigned an App-ID. This lookup can include information such as App-ID, Credential-ID and/or AE-ID.
* **Step 4d:** Using information provided in the request, the ARF can check whether the AE has enrolled and been assigned the App-ID included in the registration request. Based on the outcome of this check, the ARF can respond with an indication whether the AE is allowed to register. In addition, the ARF can provide additional information to the Registrar such as a list of allowed roles for the Registree, other allowed AE-IDs and Credential-IDs that are permitted to register using the same App-ID and/or a lifetime which the App-ID is valid for.
* **Step 4e:** Upon receiving a response from the ARF, the Registrar CSE can create or update a <serviceSubscribedAppRule> resource to store the information it received from the ARF such that it can be used to process the current registration request as well as future registration requests. For example, this information can be stored in the *applicableCredIDs, allowedAppIDs, allowedAEs, allowedRoleIDs* and the *expirationTime* attributes.
* **Step 4f:** Based on the outcome of the <serviceSubscribedAppRule> resource checks, the Registrar CSE either allows or denies the registration request.
* **Step 4g:** The Registrar CSE returns a registration response to the Registree AE that includes a status indication of whether the registration was successful or not.

#### 6.1.2.1 Nodes

This candidate architecture does not propose any new nodes to the oneM2M architecture. However, the following new architectural entities are proposed. The proposed entities can be hosted on the existing set of oneM2M defined nodes.

* App-ID Registry Function (ARF)
* ARF Client

#### 6.1.2.2 Reference Points

This candidate architecture proposes to define a new reference point between an ARF and an ARF Client (Marf).

## 6.1.3 Architecture Option #2

Another candidate architecture is shown below in Figure 6.1.2-1. In this candidate architecture App-ID Registry functionality is integrated into the MEF. A Registree node can use it’s MEF Client to enrol to a MEF and be provisioned with App-ID(s) for the AE(s) the node hosts. A CSE hosted on a Registrar node can use it’s MEF Client to perform a lookup to the App-ID Registry functionality integrated within the MEF to check whether a Registree AE’s App-ID is valid and in turn whether the Registrar CSE should grant or deny the Registree AE’s registration request. Using this lookup capability and the information returned by the MEF, the Registrar CSE can initiate the dynamic and remote provisioning of AE service subscription information (i.e. <serviceSubscribedAppRule> resources).



Figure 6.1.2-1: App-ID Registry Function Integrated into MEF

**Step 1:** Same as Step 1 in clause 6.1.1.

**Step 2:** Same as Step 2 in clause 6.1.1 except MEF Client is used by Registree node to enrol to ARF functionality integrated within MEF.

**Step 3:** Same as Step 3 in clause 6.1.1.

**Step 4:** Same as Step 4 in clause 6.1.1 except MEF Client is by Registrar node to perform lookup to ARF functionality integrated within MEF.

#### 6.1.3.1 Nodes

This candidate architecture does not propose any new nodes nor any new functions to the oneM2M architecture. The App-ID Registry functionality is instead integrated into the existing MEF.

#### 6.1.3.2 Reference Points

This candidate architecture does not propose any new reference points. Instead App-ID Registry functionality is added to the existing Mmef reference point.

### -----------------------End of change 1 --------------------------------------------

CHECK LIST

* Does this Change Request include an informative introduction containing the problem(s) being solved, and a summary list of proposals.?
* Does this CR contain changes related to only one particular issue/problem?
* Have any mirror CRs been posted?
* Does this Change Request make **all** the changes necessary to address the issue or problem? E.g. A change impacting 5 tables should not include a proposal to change only 3 tables?Does this Change Request follow the drafting rules?
* Are all pictures editable?
* Have you checked the spelling and grammar?
* Have you used change bars for all modifications?
* Does the change include the current and surrounding clauses to clearly show where a change is located and to provide technical context of the proposed change? (Additions of complete clauses need not show surrounding clauses as long as the proposed clause number clearly shows where the new clause is proposed to be located.)
* Are multiple changes in this CR clearly separated by horizontal lines with embedded text such as, start of change 1, end of change 1, start of new clause, end of new clause.?