|  |
| --- |
|  |

|  |  |
| --- | --- |
| CHANGE REQUEST | |
| Meeting:\* | ARC#25 |
| Source:\* | C-DOT |
| Date:\* | 2016-10-14 |
| Contact:\* | Poornima ([poornima@cdot.in](mailto:poornima@cdot.in)), Chaitan([chaitan.yadav@cdot.in](mailto:chaitan.yadav@cdot.in)),  Anupama([anupama@cdot.in](mailto:anupama@cdot.in)) |
| Reason for Change/s:\* | See the introduction |
| CR against: Release\* | Release 2 |
| CR against: WI\* | Active <Work Item number>  MNT maintenace / < Work Item number(optional)>  STE Small Technical Enhancements / < Work Item number (optional)>  Only ONE of the above shall be ticked |
| CR against: TS/TR\* | TS-0001 v2.10.0 |
| Clauses/Sub Clauses\* | Section 9.6.5,7.1.17 |
| 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 |
| 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  This CR is a mirror CR? YES  if YES, please indicate the document number of the original CR: : NO |
| Template Version:27 May 2015 (Dot 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 separated “mirror CR” should be posted at the same time of this CR

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 sections need not show surrounding clauses as long as the proposed section number clearly shows where the new section 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

In TS-0001, accessControlPolicyIDs description, it is mentioned that for updating accessControlPolicyIDs attribute, it checks in selfPrivileges as highlighted below:

|  |  |
| --- | --- |
| *accessControlPolicyIDs* | The attribute contains a list of identifiers of an *<accessControlPolicy>* resource. The privileges defined in the *<accessControlPolicy>* resource that are referenced determine who is allowed to access the resource containing this attribute for a specific purpose (e.g. Retrieve, Update, Delete, etc.).  If a resource type does not have an *accessControlPolicyIDs* attribute definition, then the *accessControlPolicyIDs* for that resource is governed in a different way, for example, the *accessControlPolicy* associated with the parent may apply to a child resource that does not have an *accessControlPolicyIDs* attribute definition, or the privileges for access are fixed by the system. Refer to the corresponding resource type definitions and procedures to see how access control is handled in such cases.  If a resource type does have an *accessControlPolicyIDs* attribute definition, but the (optional) *accessControlPolicyIDs* attribute is not set, or it is set to a value that does not correspond to a valid, existing *<accessControlPolicy>* resource, or it refers to an *<accessControlPolicy>* resource that is not reachable (e.g. because it is located on a remote CSE that is offline or not reachable), then the system default access privileges shall apply.  All resources are accessible if and only if the privileges (i.e. shored as *privileges* or *selfPrivileges* attribute of <accessControlPolicy> resource) allow it, therefore all resources shall have an associated *accessControlPolicyIDs* attribute, either explicitly (setting the attribute in the resource itself) or implicitly (either by using the parent privileges or the system default policies). Which means that the system shall provide a default access privileges in case that the Originator does not provide a specific *accessControlPolicyIDs* during the creation of the resource.  To update this attribute, a Hosting CSE shall check whether an Originator has Update permission in any *selfPrivileges* of the *<accessControlPolicy>* resources which this attribute originally indicates. |

### But for *accessControlPolicyIDs* attribute*,* it should be *privileges* not *selfPrivileges.* Because selfPrivileges are used for <accessControlPolicy> resource, which doesn’t have this attribute.So for resource other than <accessControlPolicy>, when *accessControlPolicyIDs* attribute is to be updated or any other attribute is to be updated then ***privileges*** are checked corresponding to existing *accessControlPolicyIDs* attribute in that resource

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

##### 9.6.1.3.2 Common attributes

The following attributes are commonly used in multiple, but not all, resource types which are normal, not virtual or announced. Common attributes for announced resource types are independently defined in claused 9.6.26.3.

NOTE: The list of attributes in table 9.6.1.3.2-1 is not exhaustive.

Table 9.6.1.3.2-1: Common Attributes

| Attribute Name | Description |
| --- | --- |
| *accessControlPolicyIDs* | The attribute contains a list of identifiers of an *<accessControlPolicy>* resource. The privileges defined in the *<accessControlPolicy>* resource that are referenced determine who is allowed to access the resource containing this attribute for a specific purpose (e.g. Retrieve, Update, Delete, etc.).  If a resource type does not have an *accessControlPolicyIDs* attribute definition, then the *accessControlPolicyIDs* for that resource is governed in a different way, for example, the *accessControlPolicy* associated with the parent may apply to a child resource that does not have an *accessControlPolicyIDs* attribute definition, or the privileges for access are fixed by the system. Refer to the corresponding resource type definitions and procedures to see how access control is handled in such cases.  If a resource type does have an *accessControlPolicyIDs* attribute definition, but the (optional) *accessControlPolicyIDs* attribute is not set, or it is set to a value that does not correspond to a valid, existing *<accessControlPolicy>* resource, or it refers to an *<accessControlPolicy>* resource that is not reachable (e.g. because it is located on a remote CSE that is offline or not reachable), then the system default access privileges shall apply.  All resources are accessible if and only if the privileges (i.e. shored as *privileges* or *selfPrivileges* attribute of <accessControlPolicy> resource) allow it, therefore all resources shall have an associated *accessControlPolicyIDs* attribute, either explicitly (setting the attribute in the resource itself) or implicitly (either by using the parent privileges or the system default policies). Which means that the system shall provide a default access privileges in case that the Originator does not provide a specific *accessControlPolicyIDs* during the creation of the resource.  To update this attribute or any other attribute in a resource, a Hosting CSE shall check whether an Originator has Update permission in any *privileges* of the *<accessControlPolicy>* resources which this attribute originally indicates. |
| *stateTag* | An incremental counter of modification on the resource. When a resource is created, this counter is set to 0, and it will be incremented on every modification of the resource (see notes 1 and 2). |
| *announceTo* | This attribute may be included in a CREATE or UPDATE Request in which case it contains a list of addresses/CSE-IDs where the resource is to be announced. For the case that CSE-IDs are provided, the announced-to CSE shall decide the location of the announced resources based on the rules described in clause 9.6.26.  For the original resource, this attribute shall only be present if it has been successfully announced to other CSEs. This attribute maintains the list of the resource addresses to the successfully announced resources. Updates on this attribute will trigger new resource announcement or de-announcement.  If *announceTo* attribute includes resource address(s), the present document does not provide any means for validating these address(s) for announcement purposes. It is the responsibility of the Hosting-CSE referenced by the resource address(s) to validate the access privileges of the originator of the Request that triggers the announcement. |
| *announcedAttribute* | This attributes shall only be present at the original resource if some Optional Announced **(OA)** type attributes have been announced to other CSEs. This attribute maintains the list of the announced Optional Attributes (**OA** type attributes) in the original resource. Updates to this attribute will trigger new attribute announcement if a new attribute is added or de-announcement if the existing attribute is removed. |
| *labels* | Tokens used to add meta-information to resources.  This attribute is optional.  The value of the *labels* attribute is a list of individual labels, each of them being:   * Either a standalone label-key, used as a simple “tag”, that can be used for example for discovery purposes when looking for particular resources that one can “tag” using that label-key * Or a composite element made of a label-key and a label-value, separated by a special character defined in [3]. A label-key itself can consist of several sub-elements, separated by a special character also defined in [3].   The list of allowed characters in a label (and in label-keys and label-values) and separator characters is defined in [3], clause 6.3.3. |
| *e2eSecInfo* | Present in a resource representing an AE or CSE. Indicates the end-to-end security capabilities supported by the AE or CSE. May indicate supported end-to-end security frameworks. May also contains a certificate or credential identifier used by the AE or CSE. May include random values for use in end-to-end security protocols. The details of this attributes are described in oneM2M TS-0003 [2].  This attribute is optional and if not present it means that the represented entity does not support oneM2M end-to-end security procedures. |
| *dynamicAuthorizationConsultationIDs* | This attribute contains a list of identifiers of *<dynamicAuthorizationConsultation>* resources. The information defined in a *<dynamicAuthorizationConsultation>* resource is used by a CSE for initiating consultation-based dynamic authorization requests.  Consultation-based dynamic authorization is only performed for a targeted resource if and only if it is linked to an enabled *<dynamicAuthorizationConsultation>*  resource.  If the attribute is not set or has a value that does not correspond to a valid *<dynamicAuthorizationConsultation>* resource(s), or it refers to an *<dynamicAuthorizationConsultation>* resource(s) that is not reachable, then the *dynamicAuthorizationConsultationIDs* associated with the parent may apply to the child resource if present, or a system default *<dynamicAuthorizationConsultation>* may apply if present. |
| *creator* | The AE-ID or CSE-ID of the entity which created the resource containing this attribute. |
| NOTE 1: In order to enable detection of overflow, the counter needs to be capable of expressing sufficiently long numbers.  NOTE 2: This attribute has the scope to allow identifying changes in resources within a time interval that is lower than the one supported by the attribute *lastModifiedTime* (e.g. less than a second or millisecond). This attribute can also be used to avoid race conditions in case of competing modifications. | |

### -----------------------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 only include a proposal to change only 3 tables. Includes any changes to references, definitions, and acronyms in the same deliverable?
* 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 sections need not show surrounding clauses as long as the proposed section number clearly shows where the new section 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.?