Lightweight M2M – Portfolio Object (LwM2M Object – PortfolioObj)
Approved Version 1.0 – 14 Nov 2017

Open Mobile Alliance
OMA-TS-LWM2M_PortfolioObj-V1_0-20171114-A
Contents

1. SCOPE ...........................................................................................................................................4
2. REFERENCES .....................................................................................................................................5
   2.1 NORMATIVE REFERENCES ........................................................................................................5
   2.2 INFORMATIVE REFERENCES .....................................................................................................5
3. TERMINOLOGY AND CONVENTIONS ..............................................................................................6
   3.1 CONVENTIONS ..........................................................................................................................6
   3.2 DEFINITIONS ..............................................................................................................................6
   3.3 ABBREVIATIONS ........................................................................................................................6
4. INTRODUCTION ................................................................................................................................7
5. PORTFOLIO OBJECT FUNCTIONALITY ...........................................................................................8
   5.1 PORTFOLIO DATA STORAGE EXTENSION ...............................................................................8
   5.2 PORTFOLIO SERVICES ..............................................................................................................8
      5.2.1 Resources description .............................................................................................................8
      5.2.2 Services Description ..............................................................................................................9
6. LWM2M OBJECT: PORTFOLIO .......................................................................................................10

APPENDIX A. CHANGE HISTORY (INFORMATIVE) .........................................................................13
   A.1 APPROVED VERSION HISTORY ..............................................................................................13

APPENDIX B. PORTFOLIO OBJECT USAGE EXAMPLES ................................................................14
   B.1 ILLUSTRATION I: CURRENT GSMA CLP.03 REQUIREMENT [GSMA] ......................................14
   B.2 ILLUSTRATION II: EXTENDED USAGE ...............................................................................15

Figures

Figure 1: GSMA CLP.03 Requirement (IoT Device Connection Efficiency Guidelines) .........14
Figure 2: LWM2M Object Device (ID:3) extended in using a link to a dedicated GSMA Portfolio Object Instance ....15
Figure 3: End-to-end security for application ...........................................................................16
Figure 4: Process Illustration for Host Device Authentication ..................................................17

Tables

Table 1: Portfolio Object Resources Definition ............................................................................11
Table 2: GetAuthData Executable Resource Arguments Definition ...........................................12
1. Scope

This document defines the technical specification for an Object to be used in conjunction with the Lightweight M2M enabler in order to extend the data storage capability of other Object Instances in the system.

Moreover, such data extension MAY come with the support of new functionalities as strong data authentication (which confidence can be granted to a data likely to represent a personal ID).
2. References

2.1 Normative References

[LwM2M] “OMA LightweightM2M”, Version 1.0, Open Mobile Alliance™,
URL: http://www.openmobilealliance.org/

[RFC2119] “Key words for use in RFCs to Indicate Requirement Levels”, S. Bradner, March 1997,
URL: http://www.ietf.org/rfc/rfc2119.txt

2.2 Informative References


[GP_SE] “Global Platform SE configuration v1.0”, URL:https://globalplatform.org/specifications.asp


[OMADICT] “Dictionary for OMA Specifications”, Version 2.9, Open Mobile Alliance™,
OMA-ORG-Dictionary-V2_9, URL: http://www.openmobilealliance.org/

3. Terminology and Conventions

3.1 Conventions
The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119].

All sections and appendixes, except “Scope” and “Introduction”, are normative, unless they are explicitly indicated to be informative.

3.2 Definitions

| Secure Component               | A generic term used to refer to either a Secure Element (SE) - i.e. a Tamper-resistant component - or a Trusted Execution Environment (TEE) |
| Secure Element                 | A hardware component used to store and process private or sensitive information, such as private keys. It’s a tamper-resistant component meaning this device is designed in such a way the information it contains is prevented against hardware and software attacks. |
| Trusted Execution Environment (TEE) | An isolated area of a processor insuring that code and data loaded inside are protected with respect to confidentiality and integrity. |

3.3 Abbreviations

| OMA                        | Open Mobile Alliance |
4. Introduction

Some LwM2M Applications may require extended data storage capacity over what is available with existing LwM2M Object Instances. This storage capacity extension can be used to manage information from a device that is closely related to a LwM2M Client even if this information is not directly used by the LwM2M Client itself (e.g. the identity of a Host Device having an embedded module containing a LwM2M Client).

Moreover, these extended data may be associated with specific constraints. For example, some data could be used to univocally identify a person or a device; in that case appropriate authentication services MUST be proposed to fulfil that need.

The Portfolio Object specification defines a mechanism for extending the data storage capability of other Object Instances in the LwM2M system, as well as the services which may be used to authenticate and to protect privacy of data contained in those extensions. In addition, a data encryption service is also defined.
5. Portfolio Object Functionality

The Portfolio Object offers the possibility to extend the data storage capability of another LwM2M Object instance by attaching one of the Portfolio Object Instances to the set of resources of the original LwM2M Object Instance.

Moreover, a second capability of the Portfolio Object is to provide security services for applications associated with this data extension. These services can be used to authenticate identities and to encrypt data.

5.1 Portfolio Data Storage Extension

A Portfolio Object Instance and its Resources MUST be fully dedicated to a given LwM2M Object Instance.

Identity Resource (Resource 0 of the Portfolio Object definition) is a mandatory resource that maybe instantiated multiple times.

Appendix B provides one example targeting the GSMA CLmP0.3 requirement: a Portfolio Object Instance (e.g. Instance 0) is attached to the LwM2M Device Object Instance (LwM2M Object ‘3’). This Portfolio Object Instance defines four Identity resource instances fitting with the GSMA requirements related to the Host Device information.

5.2 Portfolio Services

5.2.1 Resources description

The Portfolio Object defines three Resources to support Services that can be used in conjunction with the Identity resource: GetAuthData, AuthData, AuthStatus.

1. **GetAuthData**: defines an executable Resource which is used to trigger the process of generating data according to the set of arguments provided with the LwM2M Server request.
   - **Argument 0**: ‘Service’: the type of Service required. Two Services are specified in section 5.2.2: Data Authentication and Data encryption. Services
     - 0: SRV_AUTHENT: Data Authentication Service
     - 1: SRV_ENCRYPT: Data Encryption Service
   - **Argument 1**: Identity Instance ID: determines which Identity Resource Instance is concerned by the GetAuthData request
   - **Argument 2**: Challenge: a random value generated by the LwM2M server to increase the response entropy.
   - **Argument 3**: Key Index: when needed, used to select the key which has to be used for the cryptographic operation

2. **AuthData**: defines a String Resource which receives the data generated according to the GetAuthData request. This resource is a Read-Only resource.
   - when AuthData resource receives valid information, the AuthStatus MUST be turned to DATA_AVAIL_STATE
   - when AuthData resource is accessed through a LwM2M READ request or a LwM2M NOTIFY, the AuthStatus MUST be reset to IDLE_STATE

3. **AuthStatus**: defines a Resource which contains the state related to the process triggered by GetAuthData request.
   - 0: IDLE_STATE : the buffer AuthData doesn’t contain any valid data
   - 1: DATA_AVAIL_STATE : the buffer AuthData contains valid data
   - 2: ERROR_STATE : the process triggered by GetAuthData request incurred an error
     - the AuthStatus MUST be set to IDLE_STATE when a GetAuthData request is triggered
     - the AuthStatus MUST be set to DATA_AVAIL_STATE, when AuthData is filled up with valid data
     - the AuthStatus MUST be set to ERROR_STATE when the process triggered by a GetAuthData request incurred an error

For supporting such Services, the Portfolio Objects resources of this sub-section MUST be bound to the interface of a Secure Component. This binding is implementation specific.
The Secure Component MUST be provisioned with a dedicated cryptographic key set for each Portfolio Object Instance Service. The cryptographic key sets MUST NOT be shared by the different Portfolio Object Instances.

### 5.2.2 Services Description

1. **Data Authentication**: this Service will be used when an Identity Resource Instance of a Portfolio Object Instance must be certainly authenticated and must be guaranteed against cloning threats.

   Strong authentication requires the cryptographic capability provided by a Secure Component which MUST be provisioned according to the security policy rules of the Application Provider.

   At least two data items MUST be provisioned in the Secure Component:
   - a Hash of the identity to authenticate
   - a cryptographic keyset shared with the Server intended to authenticate the identity above.

   **Cryptographic Process**: when a GetAuthData request is forwarded to the Secure Component, this cryptographic element will process the request to generate a Signature of the specified Identity Resource Instance value which MUST be of the following form:

   \[
   \text{Signature} = \text{SIG}_{Kn} (\text{Hash}(\text{Identity}) + \text{Challenge})
   \]

   Where:
   - SIG is a signature function shared between the Secure Component and the Server qualified to authenticate the Identity
   - Kn is one cryptographic key of the Data Authentication Service keyset provisioned in the Secure Component and selected by the Argument 3 of the GetAuthData request.
   - \(\text{Hash}(\text{Identity})\) is the data provisioned in the Secure Component and selected by the Argument 1 of the request
   - Challenge is a random value provided by the Server in Argument 2 of the GetAuthData request
   - Signature is the result of this cryptographic process which will be stored in the AuthData resource, and will be transported by the LwM2M protocol for authentication by the qualified Server.

   **Note**: the properties of the Hash and Signature functions as well as the cryptographic key set, only have to be known between the Secure Component and the Server intended to authenticate the provided Identity; that’s why these elements are out of the scope of the Portfolio Object Specification (e.g. Hash function can be based on SHA-2 algorithm, and the Signature function can use the AES-CMAC algorithm specified in [RFC4493]).

2. **Data Encryption**: this Service will be used when a data contained in an Identity Resource Instance of a Portfolio Object Instance needs to be encrypted.

   At least one cryptographic key must be provisioned in the Secure Component

   **Cryptographic Process**: when a GetAuthData request is forwarded to the Secure Component, this cryptographic element will process the request to cipher the specified Identity Resource Instance value. The cipher function MUST be of the form:

   \[
   \text{Encryption} = \text{ENCRYPT}_{Kn} (\text{data} + \text{Challenge})
   \]

   Where:
   - ENCRYPT is a cryptographic function (e.g. AES-128 function) shared between the Secure Component and the Server intended to decrypt the data
   - Kn is one cryptographic key of the Data Encryption Service key set provisioned in the Secure Component and selected by the Argument 3 of the GetAuthData request.
   - data is the content of the Identity Resource Instance as selected by the Argument 1 of the GetAuthData request
   - Challenge is a random value provided by the Server in Argument 3 of the GetAuthData request
   - Encryption is the result of this cryptographic process which will be stored in the AuthData resource.
6. LwM2M Object: Portfolio

Description
This section formalizes the Resources definitions of the Portfolio Object described in section 5 of this document.

Object definition

<table>
<thead>
<tr>
<th>Name</th>
<th>Object ID</th>
<th>Instances</th>
<th>Mandatory</th>
<th>Object URN</th>
</tr>
</thead>
<tbody>
<tr>
<td>Portfolio</td>
<td>16</td>
<td>Multiple</td>
<td>Optional</td>
<td>urn:oma:lwm2m:oma:16</td>
</tr>
</tbody>
</table>
Resource definition

<table>
<thead>
<tr>
<th>ID</th>
<th>Name</th>
<th>Operations</th>
<th>Instances</th>
<th>Mandatory</th>
<th>Type</th>
<th>Range or Enumeration</th>
<th>Units</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Identity</td>
<td>RW</td>
<td>Multiple</td>
<td>Mandatory</td>
<td>String</td>
<td></td>
<td></td>
<td>Data Storage extension for other Object Instances. e.g for [GSMA]: 0: Host Device ID, 1: Host Device Manufacturer 2: Host Device Model 3: Host Device Software Version, This Resource contains data that the GetAuthData executable Resource can work with.</td>
</tr>
<tr>
<td>1</td>
<td>GetAuthData</td>
<td>E</td>
<td>Single</td>
<td>Optional</td>
<td>none</td>
<td></td>
<td></td>
<td>Executable resource to trigger Services described in Section 5.2.2 Arguments definitions are described in Section 5.2.1 as well as in table 2 of this document</td>
</tr>
<tr>
<td>2</td>
<td>AuthData</td>
<td>R</td>
<td>Multiple</td>
<td>Optional</td>
<td>Opaque</td>
<td></td>
<td></td>
<td>Buffer which contains the data generated by the process triggered by a GetAuthData request</td>
</tr>
<tr>
<td>3</td>
<td>AuthStatus</td>
<td>R</td>
<td>Single</td>
<td>Optional</td>
<td>Integer</td>
<td>[0-2]</td>
<td></td>
<td>This Resource contains the state related to the process triggered by GetAuthData request. 0: IDLE_STATE: AuthData doesn’t contain any valid data 1: DATA_AVAIL_STATE: AuthData contains valid data 2: ERROR_STATE: an error occurred This state is reset to IDLE_STATE, when the executable resource “GetAuthData” is triggered or when the AuthData resource has been returned to the LwM2M Server (READ / NOTIFY).</td>
</tr>
</tbody>
</table>

Table 1 : Portfolio Object Resources Definition

Execution Resource Arguments definition

<table>
<thead>
<tr>
<th>ID</th>
<th>Resource Name</th>
<th>Order</th>
<th>Name</th>
<th>Type</th>
<th>Range or Enum</th>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>GetAuthData</td>
<td>0</td>
<td>Service</td>
<td>Integer</td>
<td>[0-1]</td>
<td></td>
<td>Data Security Services 0: SRV_AUTHENT: Authentication 1: SRV_ENCRYPT: Encryption</td>
</tr>
<tr>
<td></td>
<td>Description</td>
<td>Type</td>
<td>Optional</td>
<td>Description</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-------------</td>
<td>------</td>
<td>----------</td>
<td>-------------</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>Identity Instance Id</td>
<td>Integer</td>
<td>-</td>
<td>Identity Resource Instance ID of the current Portfolio Object Instance</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Challenge</td>
<td>Integer</td>
<td>-</td>
<td>A random provided by the Server</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>Key Index</td>
<td>Integer</td>
<td>-</td>
<td>To select the cryptographic key in the Service dedicated key set provisioned in the Secure Component</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4..9</td>
<td>-</td>
<td>String</td>
<td>-</td>
<td>Application dependent arguments</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 2: GetAuthData Executable Resource Arguments Definition
Appendix A. Change History (Informative)

A.1 Approved Version History

<table>
<thead>
<tr>
<th>Reference</th>
<th>Date</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>OMA-TS-LWM2M_PortfolioObj-V1_0-20171114-A</td>
<td>14 Nov 2017</td>
<td>Status changed to Approved by TP TP Ref # OMA-TP-2017-0047-INP_LWM2M_Portfolio-V1_0_RRP_for_Final_Approval</td>
</tr>
</tbody>
</table>
Appendix B. Portfolio Object Usage Examples

B.1 Illustration I: Current GSMA CLP.03 requirement [GSMA]

In the GSMA CLP.03 document, there are specific requirements (DID4-7) related to the mandatory capability of the Device Management Technology. LwM2M TS 1.0 using Portfolio Object functionality is able to cover such a requirement as illustrated below:

| DID4 | The following LwM2M resource has been defined to specify information related to the manufacturer of the IoT Host Device, this field will need to match the IoT Device Host manufacturer name that is referenced in the Mobile Network Operator lab certification of the IoT Device.  
Type: Host Device Manufacturer  
Occurrence: One  
Format: String  
Name: /16/x/0/1 (HostMan)  
Access Type: READ  
The IoT Device Host manufacturer will be maintained in this resource by the Communications Module LwM2M client. |
| DID5 | The following LwM2M resource has been defined to specify the Model name/number of the IoT Device Host. This shall match the model name/number used in the certification of the IoT Device.  
Type: Host Device Model  
Occurrence: One  
Format: String  
Name: /16/x/0/2 (HostMod)  
Access Type: READ  
The IoT Host Device model will be maintained in the node by the Communication Module LwM2M client. |
The following LwM2M resource has been defined to specify the software version of the IoT Device Host, this information shall be populated by the IoT Device Host manufacturer, shall match the version of SW certified by PTCRB and must be updated whenever the SW is updated on the device.

**Type:** Host Device Software Version  
**Occurrence:** One  
**Format:** String  
**Name:** /16/x/0/3 (HostSwV)  
**Access Type:** READ

The IoT Host Device software version will be maintained in this resource by the Communication Module LwM2M client.

The following OMA-DM node has been defined to specify the unique ID allocated to the IoT Device Host by the certifying Mobile Network Operator. Mobile Network Operators’ may decide to include this field if they need a way to monitor for uncertified devices used on the network.

**Type:** Host Device Unique ID  
**Occurrence:** One  
**Format:** Alphanumeric String  
**Name:** /16/x/0/0 (HostUniqueID)  
**Access Type:** READ

The IoT Device Host Unique ID is assigned by the Mobile Network Operator and will be stored in this resource.

In a given LwM2M Client implementation, the graphical data representation of the GSMA-compatible Connection Module can be illustrated as follows (e.g. the GSMA Portfolio Object Instance ID is assigned to /16/1).

![Figure 2: LwM2M Object Device (ID:3) extended in using a link to a dedicated GSMA Portfolio Object Instance](image)

**B.2 Illustration II: Extended usage**

The use cases below provide only few examples on how the association of the LwM2M Client, the Portfolio Object and a Secure Component can work together. It is left to implementation to further enhance the capabilities of such an association.
Coupled with a Secure Component, the LwM2M Client and the Portfolio Object may support Authentication and end-to-end data encryption services exposed to applications.

**Use Case 1: Vehicle (Host Device) Authentication**

A Car Manufacturer Server decides to contact a vehicle (Host Device) and would like to rely on the Host Device ID managed by the LwM2M Client of the vehicle to authenticate that car.

The LwM2M Client and its association with a Secure Component have been configured according to an out-of-band process specific to the Car Manufacturer security policy. Typically, the Secure Component has to be provisioned with the Hash of the Host Device ID and a key shared with the LwM2M Server of the Car Manufacturer.

Following the LwM2M process with the GSMA-based Portfolio functionality **guarantees** that the Host Device ID is really unique on the network.
Host Device Authentication Process

Initial Step: Provisioning: in a secure environment according to the Car Manufacturer security policy, the Secure Component is provisioned with:

- a key- $K_0$ - shared with the Car Manufacturer LwM2M Server
- a hash of the Host Device ID: Hash(Host Device ID)

Step 1) Host Device Secured Identification Request

a) An initial request is initiated by the LwM2M Server in addressing the Executable resource GetAuthData of the Portfolio Object Instance with the proper parameters (the AuthStatus resource is reset to IDLE_STATE):

e.g. LwM2M command: EXECUTE /16/0/1 0=SRV_AUTHENT, 1=3, 2=0x2345178, 3=0

- argument 0: An Authentication process is requested
- argument 1: the Identity Resource Instance 3 is the target (e.g. Host Device Software Version)
- argument 2: 0x2345178: a Challenge
- argument 3: the cryptographic key 0 of the Authentication key set must be used

b) Information is exchanged between the Secure Component and the LwM2M Client:

In the Secure Component a signature is generated in using the pre-registered key 0 and the hash of the Host Device ID. The challenge is used for introducing a random in the signature.

Step 2) the LwM2M Server is polling the AuthStatus Resource or is waiting for a notification related to the availability of the AuthData Resource

Step 3) when available, a Signature of the requested Identity Resource Instance [Sig (Hash(Host Device ID)+ Challenge)] is delivered by the Secure Component into the AuthData Resource of the Portfolio Object Instance.

Step 4) when the Signature is available in the Portfolio Instance (AuthStatus Resource is set to DATA_AVAIL_STATE), either a notification is sent to the LwM2M Server with such data or the LwM2M Server will perform a READ request on the AuthData resource(s) of the Portfolio Instance. In both cases the AuthStatus Resource is reset to IDLE-STATE.

When receiving this Signature, the LwM2M Server which initiated the request can authenticate the identity of the vehicle with the highest degree of confidence.

Use Case 2: Encryption of sensitive data from the Portfolio Object Instance
Very similar to the “Use Case 1” process, this new use case is addressing the need to answer to the Server request for returning sensitive data in encrypted form. The initial data is present in a Portfolio Object Instance (i.e. in one Instance of the Identity multi-instance Resource).

**LwM2M Server request:** EXECUTE /16/0/GetAuthData 0=SRV_ENCRYPT, 1=2, 2=0x2345178, 3=1

- argument 0: the data encryption service is requested (SRV_ENCRYPT)
- argument 1: the Identity Instance of ID=2 is the target (e.g. Host Device Model)
- argument 2: 0x2345178: the Challenge
- argument 3: the cryptographic key 1 of the Authentication key set must be used.

When generated by the Secure Component, the AuthData Resource will contain an encrypted value for e.g. the Host Device Model data.