oneM2M
Technical Specification

Document Number	TS-0009-V1.0.1
Document Name:	HTTP Protocol Binding
Date:	2015-January-30
Abstract:	HTTP Protocol Binding TS

This Specification is provided for future development work within oneM2M only. The Partners accept no liability for any use of this Specification.

The present document has not been subject to any approval process by the oneM2M Partners Type 1. Published oneM2M specifications and reports for implementation should be obtained via the oneM2M Partners' Publications Offices.

About oneM2M

The purpose and goal of oneM2M is to develop technical specifications which address the need for a common M2M Service Layer that can be readily embedded within various hardware and software, and relied upon to connect the myriad of devices in the field with M2M application servers worldwide.

More information about oneM2M may be found at: http//www.oneM2M.org

Copyright Notification

Notice of Disclaimer & Limitation of Liability

The information provided in this document is directed solely to professionals who have the appropriate degree of experience to understand and interpret its contents in accordance with generally accepted engineering or other professional standards and applicable regulations. No recommendation as to products or vendors is made or should be implied.

NO REPRESENTATION OR WARRANTY IS MADE THAT THE INFORMATION IS TECHNICALLY ACCURATE OR SUFFICIENT OR CONFORMS TO ANY STATUTE, GOVERNMENTAL RULE OR REGULATION, AND FURTHER, NO REPRESENTATION OR WARRANTY IS MADE OF MERCHANTABILITY OR FITNESS FOR ANY PARTICULAR PURPOSE OR AGAINST INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS. NO oneM2M PARTNER TYPE 1 SHALL BE LIABLE, BEYOND THE AMOUNT OF ANY SUM RECEIVED IN PAYMENT BY THAT PARTNER FOR THIS DOCUMENT, WITH RESPECT TO ANY CLAIM, AND IN NO EVENT SHALL oneM2M BE LIABLE FOR LOST PROFITS OR OTHER INCIDENTAL OR CONSEQUENTIAL DAMAGES. oneM2M EXPRESSLY ADVISES ANY AND ALL USE OF OR RELIANCE UPON THIS INFORMATION PROVIDED IN THIS DOCUMENT IS AT THE RISK OF THE USER.

Contents

This is a draft oneM2M document and should not be relied upon; the final version, if any, will be made available by oneM2M Partners Type 1.

1 Scope

The present document will cover the protocol specific part of communication protocol used by oneM2M compliant systems as RESTful HTTP binding.

The scope of the present document is (not limited to as shown below):

- Binding oneM2M Protocol primitive types to HTTP method.
- Binding oneM2M response status codes (successful/unsuccessful) to HTTP response codes.
- Binding oneM2M RESTful resources to HTTP resources.

The present document is depending on Core Protocol specification (oneM2M TS-0004) for data types.

2 References

2.1 Normative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the reference document (including any amendments) applies.

The following referenced documents are necessary for the application of the present document.

[1] IETF RFC 7230 (June 2014): "Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing".

[2] oneM2M TS-0003: Security Solutions.

[3] oneM2M TS-0004: "Service Layer Core Protocol Specification".

[4] RFC7235: “Hypertext Transfer Protocol (HTTP/1.1): Authentication”, IETF, June 2014.

[5] RFC6750: “The OAuth 2.0 Authorization Framework: Bearer Token Usage”, October 2012.

[6] oneM2M TS-0011: Common Terminology.

[7] oneM2M TS-0001: Functional Architecture.

2.2 Informative references

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the reference document (including any amendments) applies.

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

[i.1] oneM2M Drafting Rules.

NOTE: Available at http://member.onem2m.org/Static_pages/Others/Rules_Pages/oneM2M-Drafting-Rules-V1_0.doc.

[i.2] IETF RFC 2617 (June 1999): "HTTP Authentication: Basic and Digest Access Authentication".

[i.3] IETF RFC 6750 (October 2012): "The OAuth 2.0 Authorization Framework: Bearer Token Usage".

[i.4] IETF RFC 6455 (December 2011):"The WebSocket Protocol".

[i.5]	oneM2M TS-0003: "Security Solutions".

3 Abbreviations

For the purposes of the present document, the following abbreviations those given in TS-0011-Common Terminology [6] apply:

CSE-ID	Common Service Entity Identifier
HTTP	Hyper Text Transfer Protocol
TLS	Trasport Layer Security
URI	Uniform Resource Identifier

4 Conventions

The keywords "Shall", "Shall not", "May", "Need not", "Should", "Should not" in this document are to be interpreted as described in the oneM2M Drafting Rules [i.1].

5 Overview of HTTP Binding

This clause describes what oneM2M primitive parameters can be mapped to HTTP request/response messages.

5.1 Introduction

The present document assumes AE has the capability of HTTP Client, and CSE has the capability of both HTTP Client and Server.

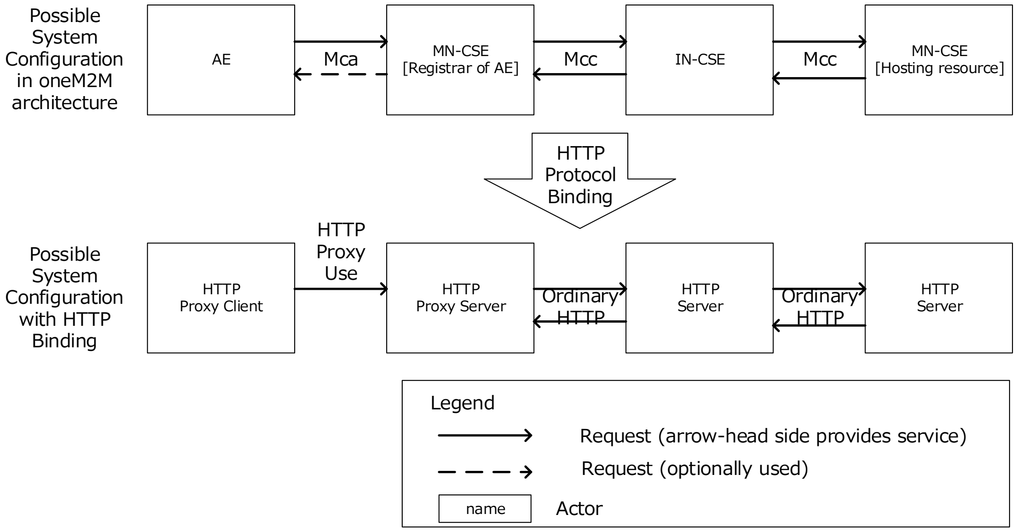

Possible System Configuration in oneM2M architecture
AE
Mca
MN-CSE [Registrar of AE]
Mcc
IN-CSE
Mcc
MN-CSE [Hosting resource]
HTTP Protocol Binding
Possible System Configuration with HTTP Binding
HTTP Proxy Client
HTTP Proxy Use
HTTP Proxy Server
Ordinary HTTP
HTTP Server
Ordinary HTTP
HTTP Server
Legend
Request (arrow-head side provides service)
Request (optionally used)
name
Actor

Figure 5.1-1 : Example of Mapping AE/CSE to HTTP Client and Server

Single request primitive will be mapped to single HTTP request message, and single response primitive will be mapped to single HTTP response message.

An HTTP request message consists of Request-Line, headers and message-body. An HTTP response message consists of Status-Line, headers and message-body [1]. This clause describes how oneM2M request/response primitives are mapped to HTTP messages at a high level. Corresponding details of each sub-clause are specified in clause 6.

The Registrar CSE shall behave as 'proxy server'(see [1]).

5.2 Request-Line

Method is mapped to the oneM2M ***Operation*** parameter.

Request-URI is derived from the oneM2M ***To*** parameter, including a query string which carries specific primitive parameters.

HTTP-Version is specified in clause 6.

5.3 Status-Line

HTTP Version is specified in clause 6.

Status-Code and Reason-Phrase are derived from the oneM2M ***Response Status Code*** parameter of the response primitive.

6 HTTP Message Mapping

6.1 Introduction

Mapping between HTTP message and oneM2M primitive shall be applied in the following cases:

- when the Originator sends a request primitive;
- when the Receiver receives a request primitive;
- when the Receiver sends a response primitive;
- when the Originator receives a response primitive.

The following clauses specify how to map each oneM2M primitive parameter to a corresponding HTTP message field to compose a HTTP request/response message.

6.2 Parameter Mappings on Request-Line

6.2.1 Method

The HTTP 'Method' shall be specified according to the oneM2M ***Operation*** parameter of the request primitive.

Table 6.2.1-1: HTTP Method Mapping

oneM2M Operation	HTTP Method
Create	POST
Retrieve	GET
Update	PUT (full update) or POST (partial update)
Delete	DELETE
Notify	POST

At the Receiver, an HTTP request message with POST method shall be mapped to a oneM2M Create or Notify request primitive in accordance with the value of the ***Operation*** parameter.

6.2.2 Request-Target

The path component of HTTP Request-Target shall be interpreted as the mapping of ***To*** parameter, and the query component (e.g. query-string) shall be interpreted as other primitive parameters (see [1]).

The ***To*** parameter shall be mapped to in Request-Target. Only SP-Relative-Resource-ID and Absolute-Resource-ID forms (see clause 7.2 [7]) shall be applicable for ***To*** parameter.

If ***To*** parameter is SP-Releative-Resource-ID form, then 'origin-form' shall be used for Request-Target. If ***To*** parameter is Absolute-Resource-ID form, then 'absolute-form' shall be used for Request-Target (see clause 6.2.3 [3]).

The prefix string "http:" shall be added to the 'absolute-form' when the address is used in the 'absolute-form' in the Request-Target.

The request parameters which are not specified as oneM2M extension headers in clause 6.4 shall be specified as pair of field-name and value in query-string as shown in table 6.2.2-1.

Table 6.2.2-1: oneM2M request parameters mapped as query-string field

Request Parameter	Field Name	Note
Response Type	***rt***	*responseType* element of the ***response type*** parameter
Result Persistence	***rp***	
Result Content	***rc***	
Delivery Aggregation	***da***	
createdBefore	***crb***	filterCriteria condition
createdAfter	***cra***	filterCriteria condition
modifiedSince	***ms***	filterCriteria condition
unmodifiedSince	***us***	filterCriteria condition
stateTagSmaller	***sts***	filterCriteria condition
stateTagBigger	***stb***	filterCriteria condition
expireBefore	***exb***	filterCriteria condition
expireAfter	***exa***	filterCriteria condition
labels	***lbl***	filterCriteria condition
resourceType	***rty***	filterCriteria condition
sizeAbove	***sza***	filterCriteria condition
sizeBelow	***szb***	filterCriteria condition
contentType	***cty***	filterCriteria condition
limit	***lim***	filterCriteria condition
attribute	***atr***	filterCriteria condition
filterUsage	***fu***	filterCriteria condition
Discovery Result Type	***drt***	

6.2.3 HTTP-Version

The present document supports binding to HTTP 1.1 [1], so the version field shall be set to "HTTP/1.1".

6.3 Status-Line

6.3.1 HTTP-Version

The present document supports binding to HTTP 1.1 [1], so the version field shall be set to "HTTP/1.1".

6.3.2 Status-Code

The ***Response Status Code*** parameter shall be mapped to HTTP Status-Code. Since the ***Response Status Code*** parameter have been defined more specifically than HTTP status codes, one or more ***Response Status Code*** may be mapped to one Status-Code. The original ***Response Status Code*** parameter shall be carried in X-M2M-RSC (see clause 6.4.14).

N:1 status code mapping from the oneM2M request primitive to HTTP request message shall be:

Table 6.3.2-1: Status Code Mapping

oneM2M Response Status Codes	HTTP Status Codes
1000 (ACCEPTED)	202 (Accepted)
2001 (CREATED)	201 (Created)
2101 (?) (CONFLICT)	409 (Conflict)
4000 (BAD_REQUEST)	400 (Bad Request)
4004 (NOT_FOUND)	404 (Not Found)
4005 (OPERATION_NOT_ALLOWED)	405 (Method Not Allowed)
4008 (REQUEST_TIMEOUT)	408 (Request Timeout)
4101 (SUBSCRIPTION_CREATOR_HAS_NO_PRIVILEGE)	403 (Forbidden)
4102 (CONTENTS_UNACCEPTABLE)	400 (Bad Request)
4103 (ACCESS_DENIED)	403 (Forbidden)
4104 (GROUP_REQUEST_IDENTIFIER_EXISTS)	409 (Conflict)
5000 (INTERNAL_SERVER_ERROR)	500 (Internal Server Error)
5001 (NOT_IMPLEMENTED)	501 (Not Implemented)
5103 (TARGET_NOT_REACHABLE)	404 (Not Found)
5105 (NO_PRIVILEGE)	403 (Forbidden)
5106 (ALREADY_EXISTS)	403 (Forbidden)
5203 (TARGET_NOT_SUBSCRIBABLE)	403 (Forbidden)
5204 (SUBSCRIPTION_VERIFICATION_INITIATION_FAILED)	500 (Internal Server Error)
5205 (SUBSCRIPTION_HOST_HAS_NO_PRIVILEGE)	403 (Forbidden)
5206 (NON_BLOCKING_REQUEST_NOT_SUPPORTED)	501 (Not Implemented)
6003 (EXTERNAL_OBJECT_NOT_REACHABLE)	404 (Not Found)
6005 (EXTERNAL_OBJECT_NOT_FOUND)	404 (Not Found)
6010 (MAX_NUMBER_OF_MEMBER_EXCEEDED)	400 (Bad Request)
6011 (MEMBER_TYPE_INCONSISTENT)	400 (Bad Request)
6020 (MANAGEMENT_SESSION_CANNOT_BE_ESTABLISHED)	500 (Internal Server Error)
6021 (MANAGEMENT_SESSION_ESTABLISHMENT_TIMEOUT)	500 (Internal Server Error)
6022 (INVALID_CMDTYPE)	400 (Bad Request)
6023 (INVALID_ARGUMENTS)	400 (Bad Request)
6024 (INSUFFICIENT_ARGUMENT)	400 (Bad Request)
6025 (MGMT_CONVERSION_ERROR)	500 (Internal Server Error)
6026 (CANCELLATION_FAILED)	500 (Internal Server Error)
6028 (ALREADY_COMPLETE)	400 (Bad Request)
6029 (COMMAND_NOT_CANCELLABLE)	400 (Bad Request)

6.3.3 Reason-Phrase

Reason-Phrase shall be omitted.

6.4 Header Fields

6.4.1 Host

The Host header shall be included in a HTTP request message.

While the Request-Target indicates a target resource on the Hosting CSE, the Host header indicates a Receiver CSE in multi-hop communication. Therefore, the the Request-Target is not changed but the Host header is changed each time when a request is forwarded.

The Host header shall be set as one of pointOfAccess attribute values of the Receiver(i.e. pointOfAccess attribute of the corresponding <remoteCSE> resource). Selection of the proper Receiver is described in oneM2M TS-0004 [3].

6.4.2 Accept

The Originator may use the Accept header to indicate which content-type is supported by the Originator. The Accept header shall be mapped to a set of media types among ”application/xml”, “application/json”, ”application/vnd.onem2m-prsp+xml”, “application/vnd.onem2m-prsp+json”.

6.4.3 Content-type

Any HTTP request or response containing message-body shall include the Content-type header set to one of ”application/xml”, “application/json”, ”application/vnd.onem2m-res+xml”, “application/vnd.onem2m-res+json”, “application/vnd.onem2m-ntfy+xml” , “application/vnd.onem2m-ntfy+json”, “application/vnd.onem2m-attrs+xml” , “application/vnd.onem2m-attrs+json”.

Content-type of the HTTP response should be chosen by the Hosting CSE considering Accept header given in the HTTP request. For example, if the Accept header in a request is set to “application/vnd.onem2m-prsp+xml”, then the response should use one of the oneM2M defined media types in XML serialization (see clause 6.7 [3]).

6.4.4 Content-Location

The Content-Location header shall be set to the URI of the created resource, when responding to a Create request primitive. The URI shall be retrieved from ***Content*** parameter. See clause 7.2.3.11 “Create a success response” [3].

6.4.5 Content-Length

If message-body is included, the Content-Length header shall be included indicating the length of the message-body in octets (8-bit bytes).

6.4.6 Etag

A retrieve response primitive corresponding to a resource retrieval request primitive should include an Etag header together with the resource representation [1].

Etag facilitates the use of conditional requests (i.e. using the if-match and if-none-match HTTP headers).

If a CSE supports the Etag header, then the CSE shall support conditional requests.

6.4.7 X-M2M-Origin

The X-M2M-Origin header shall be mapped to the ***From*** parameter of the request/response primitive.

The X-M2M-Origin header value shall be specified by the composer of the request (e.g. AE or CSE).

6.4.8 X-M2M-RI

The X-M2M-RI header shall be mapped to the ***Request Identifier*** parameter.

6.4.9 X-M2M-NM

The X-M2M-NM header shall be mapped to the ***Name*** parameter if applicable.

6.4.10 X-M2M-GID

The X-M2M-NM header shall be mapped to the ***Group Request Identifier*** parameter if applicable.

6.4.11 X-M2M-RTU

The X-M2M-RTU header shall be mapped to the *notificationURI* element of the ***Response Type*** parameter if applicable. If there are more than one value in the element, then the values shall be combined with “&” character.

6.4.12 X-M2M-OT

The X-M2M-OT header shall be mapped to the ***Originating Timestamp*** parameter if applicable.

6.4.13 X-M2M-RST

The X-M2M-RST header shall be mapped to the ***Result Expiration Timestamp*** parameter if applicable.

6.4.14 X-M2M-RET

The X-M2M-FC header shall be mapped to the ***Request Expiration Timestamp*** parameter if applicable.

6.4.15 X-M2M-OET

The X-M2M-FC header shall be mapped to the ***Operation Execution Time*** parameter if applicable.

6.4.16 X-M2M-EC

The X-M2M-ECT header shall be mapped to the ***Event Category*** parameter if applicable.

6.4.17 X-M2M-RSC

The X-M2M-RSC header shall be mapped to the ***Response Status Code*** parameter in a HTTP response message.

6.5 Message-body

Message-body shall be mapped to the ***Content*** parameter if applicable.

6.6 Message Routing

HTTP request and response message routing shall be performed as described in HTTP/1.1 [1].

7 Security Consideration

7.1 Authentication on HTTP Request Message

When sending the credential to be checked by Registrar CSE, Proxy-Authorization header should be used as specified in HTTP/1.1 (see [4]).

When sending the credential to be checked by Hosting CSE, Authorization header should be used as specified in HTTP/1.1.

When the credential to be checked by Hosting CSE is an Access Token which is compatible with OAuth 2.0 framework (see [5]), the Bearer authentication scheme shall be used as specified in OAuth 2.0 framework.

NOTE: The oneM2M Security Solutions [2] does not provide any details on usage or provisioning the token.

7.2 Transport Layer Security

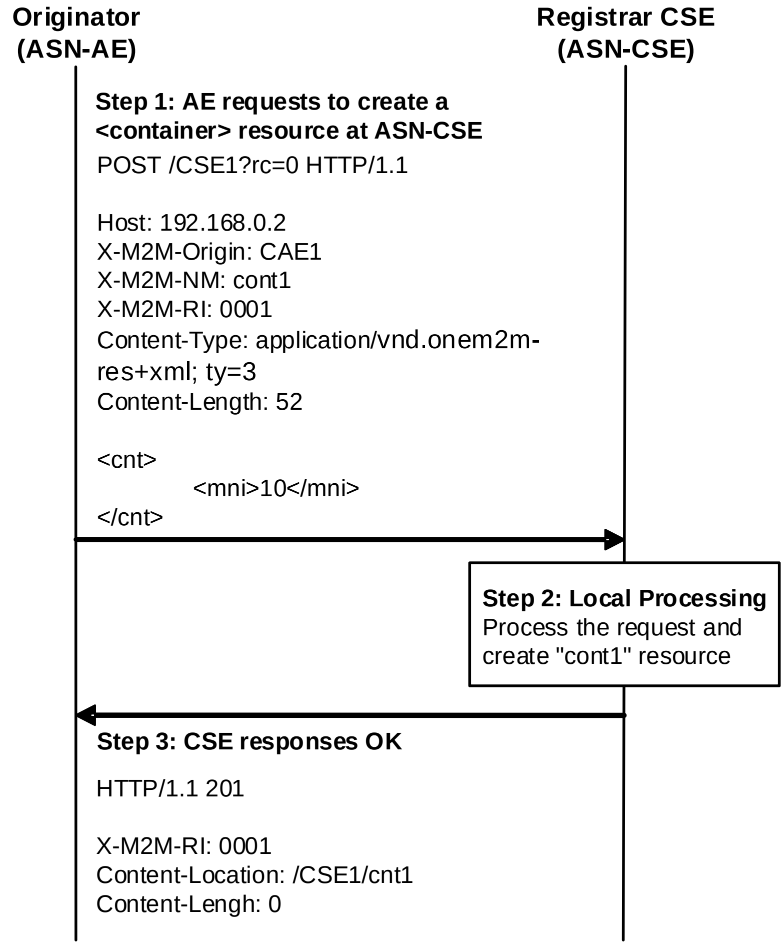
oneM2M primitive parameters contained in HTTP messages may be protected by TLS as hop-by-hop manner. For the details, see the oneM2M Security Solutions specification [2]

NOTE: Some provisioning schemes of oneM2M TS-0003 [2] enable the provisioning of end-to-end credentials, but protocols to establish security associations between non-adjacent nodes are not addressed.by oneM2M in the present release.

Annex A (informative):
Example Procedures

A.1 <container> resource creation

Figure A.1-1 is HTTP mapping of procedure described in clause 7.3.7.2.1.



Step 1: AE requests to create a <container> resource at ASN-CSE

```
POST /CSE1?rc=0 HTTP/1.1

Host: 192.168.0.2
X-M2M-Origin: CAE1
X-M2M-NM: cont1
X-M2M-RI: 0001
Content-Type: application/vnd.onem2m-res+xml; ty=3
Content-Length: 52

<cnt>
        <mni>10</mni>
</cnt>
```

Step 2: Local Processing
Process the request and create "cont1" resource

Step 3: CSE responses OK

```
HTTP/1.1 201

X-M2M-RI: 0001
Content-Location: /CSE1/cnt1
Content-Lengh: 0
```

Figure A.1-1: oneM2M HTTP Binding Example - container creation

Annex B (informative): WebSocket

B.1 Notification using WebSocket

WebSocket [i.4] can be used for transporting notifications to an AE/CSE. This can be useful for an AE/CSE which is not server-capable or cannot be reachable for delivery of unsolicited requests.

For example, when an AE needs to receive a notification message from the CSE, the AE establishes a WebSocket connection to a CSE. When a new notification message is generated, the notification will be sent to the AE as the data frame of the WebSocket.

History

Publication history		
V1.0.1	30 Jan 2015	Release 1 - Publication