|  |
| --- |
|  |

|  |  |
| --- | --- |
| CHANGE REQUEST | |
| Meeting ID:\* | SDS 40.2 |
| Source:\* | Bob Flynn, Convida Wireless , Bob.Flynn@convidawireless.com |
| Date:\* | 2019-06-13 |
| Reason for Change/s:\* | Change the supported http version |
| CR against: Release\* | Rel-3 |
| CR against: WI\* | Active <Work Item number>  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\* | TS-0009V3.2.0 |
| Clauses \* | 6.2.3 |
| 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 |
| Other TS/TR(s) impacted | None |
| 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 2019 (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.

If this is a correction, and the change applies to previous releases, a separate “mirror CR” should be posted at the same time as 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. Include any changes to references, definitions, and abbreviations in the same deliverable.

Follow the drafting rules.

All pictures must be editable.

Check spelling and grammar.

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 proposed new clause is 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 the 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

During recent deployment of a CSE we found out that an nginx proxy server, in its default configuration sends receie http 1.1 messages out as http 1.0 messages.

During resolution of this issue, I determined that this binding does not use 1.1 features and propose that we specify that 1.0 or 1.1 can be received. Responses will remain as 1.1

R01 – Feedback from Peter

We reference RFC 7230.   Section 2.6 of that RFC doesn't explicitly say that an HTTP 1.1 server must be able to accept HTTP 1.0 requests, but that's certainly the implication.  Here are some excerpts from it

The minor version advertises the sender's  
  communication capabilities even when the sender is only using a  
  backwards-compatible subset of the protocol, thereby letting the  
  recipient know that more advanced features can be used in response  
  (by servers) or in future requests (by clients).

When an HTTP/1.1 message is sent to an HTTP/1.0 recipient [[RFC1945](https://tools.ietf.org/html/rfc1945)]  
  or a recipient whose version is unknown, the HTTP/1.1 message is  
  constructed such that it can be interpreted as a valid HTTP/1.0  
  message if all of the newer features are ignored.  This specification  
  places recipient-version requirements on some new features so that a  
  conformant sender will only use compatible features until it has  
  determined, through configuration or the receipt of a message, that  
  the recipient supports HTTP/1.1.

...

A client SHOULD send a request version equal to the highest version  
  to which the client is conformant and whose major version is no  
  higher than the highest version supported by the server, if this is  
  known.  A client MUST NOT send a version to which it is not  
  conformant.

...

A server SHOULD send a response version equal to the highest version  
  to which the server is conformant that has a major version less than  
  or equal to the one received in the request.

  {PN note, this doesn't place a constraint on the minor version so if the request is 1.x the server responds with the highest 1.y that it supports even if y > x}

...

A server MAY send an HTTP/1.0 response to a request if it is known or  
  suspected that the client incorrectly implements the HTTP  
  specification and is incapable of correctly processing later version  
  responses, such as when a client fails to parse the version number  
  correctly or when an intermediary is known to blindly forward the  
  HTTP-version even when it doesn't conform to the given minor version  
  of the protocol.  Such protocol downgrades SHOULD NOT be performed  
  unless triggered by specific client attributes, such as when one or  
  more of the request header fields (e.g., User-Agent) uniquely match  
  the values sent by a client known to be in error.

What this nets out to is that if you conform to this RFC

* HTTP 1.1 Request should cause a 1.1 Response
* HTTP 1.0 Request should cause a Response that's compatible with 1.0
* The Response should say that it is 1.1 in both cases - with the exception that's noted in that last paragraph.

Our current text seems to narrow this by saying "The HTTP version field in HTTP request messages shall be set to “HTTP/1.1**”."    If you are concerned about clients that support 1.0, all you really need to do is to delete that sentence.   I would still just say that we support 1.1 as defined in the RFC.  You might want to add an explanatory non-normative note that says that the RFC provides support for 1.0 clients.**

Changing proposed text to:

This specification defines binding compliant with HTTP 1.1 [1].

NOTE: [RFC 7230] describes how a HTTP/1.0 request is handled.

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

### 6.2.3 HTTP-Version

This specification defines binding compliant with HTTP 1.1 [1].

NOTE: The HTTP version in a request can be HTTP/1.0 or HTTP/1.1. [RFC 7230] describes how a HTTP/1.0 request is handled.

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

### -----------------------Start of change 2-------------------------------------------

### -----------------------End of change 2---------------------------------------------

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.?