**3GPP TSG-CT WG1 Meeting #125-eC1-204899**

**Electronic meeting, 20-28 August 2020**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.0* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  | **24.582** | **CR** | **0016** | **rev** | **-** | **Current version:** | **14.3.0** |  |
|  | | | | | | | | |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* | | | | | | | | |
|  | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME | **X** | Radio Access Network |  | Core Network | **X** |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | Addressing a potential race/ambiguity condition when MSRP is used | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | AT&T | | | | | | | | | |
| ***Source to TSG:*** | C1 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | MCImp-MCDATA-CT | | | | |  | ***Date:*** | | | 9 August 2020 |
|  |  | | | |  | |  | | |  |
| ***Category:*** | **F** |  | | | | | ***Release:*** | | | Rel-14 |
|  | *Use one of the following categories:* ***F*** *(correction)* ***A*** *(mirror corresponding to a change in an earlier release)* ***B*** *(addition of feature),* ***C*** *(functional modification of feature)* ***D*** *(editorial modification)*  Detailed explanations of the above categories can be found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | | | | | | | | *Use one of the following releases: Rel-8 (Release 8) Rel-9 (Release 9) Rel-10 (Release 10) Rel-11 (Release 11) Rel-12 (Release 12)* *Rel-13 (Release 13) Rel-14 (Release 14) Rel-15 (Release 15) Rel-16 (Release 16)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | During C1#124-e (June 2020) the issue of a potential race/ambiguity condition when using MSRP was raised, but at the time, no definite conclusion was reached in terms of a solution.  Essentially, the terminating participating function (TPF) sends an MSRP 200 OK towards the originating side immediately upon receiving an MSRP SEND media package, although the (in)correctness of the packet is only determined by the MCData client, later on, upon receiving that media packet from the TPF. At that time, if the media packet is deemed incorrect, the MCData client sends the error response towards the originator, but it is unclear if this mechansim works, as the originator’s MSRP context may no longer exist after receiving the 200 OK. (The MSRP RFC is silent on this topic).  This CR proposes a solution. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Upon forwarding a received media packet to the terminating MCData Client, the TPF starts a timer set to an implementation determined value. The MCData client is required to send a response towards the TPF. If the timer expires before the response from the MCData client is received, the TPF sends a 200 OK towards the originator. If the response from the MCData client is received while the timer runs, the TPF will forward the response towards the originator. If the response from the MCData client is received after the timer expires and is not 200 OK, the TPF will forward the received response towards the originator. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | Potential for malfunction may lead to transfer failure. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 6.1.1.3.1, 6.1.2.3.1, 6.2.1.5.3, 6.2.2.5.3, 7.1.3.1, 7.2.5.3 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **Y** | **N** |  | | | |  | | |
| ***Other specs*** | |  | **X** | Other core specifications | | | | TS/TR ... CR ... | | |
| ***affected:*** | |  | **X** | Test specifications | | | | TS/TR ... CR ... | | |
| ***(show related CRs)*** | |  | **X** | O&M Specifications | | | | TS/TR ... CR ... | | |
|  | |  | | | | | | | | |
| ***Other comments:*** | |  | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | |  | | | | | | | | |

**\* \* \* \* \* FIRST CHANGE \* \* \* \* \***

##### 6.1.1.3.1 Handling MSRP connection

Upon receiving an indication to establish MSRP connection for standalone SDS using media plane as the terminating client, the MCData client:

1. shall act as an MSRP client according to IETF RFC 6135 [12];

2. shall act either as an active endpoint or as a passive endpoint to open the transport connection, according to IETF RFC 6135 [12];

3. shall establish the MSRP connection according to the MSRP connection parameters in the SDP offer received in the SIP INVITE request according to IETF RFC 4975 [11]; and

4. if acting as an "active" endpoint, shall send an empty MSRP SEND request to bind the MSRP connection to the MSRP session from the perspective of the passive endpoint according to the rules and procedures of IETF RFC 4975 [11] and IETF RFC 6135 [12].

Once the MSRP connection is established, on receipt of an MSRP request in an MSRP session, the MCData client:

1. shall follow the rules and procedures defined in IETF RFC 4975 [11] and in IETF RFC 6714 [13];
2. unless a response to the received MSRP request was already sent to the terminating participating MCData function as part of the previous step, shall generate an MSRP 200 (OK) response and send it to the terminating participating MCData function, according to the rules and procedures of IETF RFC 4975 [11];

3. if an MSRP SEND request indicates the use of chunking, shall wait until all further MSRP SEND requests for the remaining chunks of the MSRP message have been received and shall reassemble the entire set of MSRP requests into the MCData standalone message before delivering the content to the application; and

4. shall handle the received content as described in subclause 6.1.1.3.2.

**\* \* \* \* \* NEXT CHANGE \* \* \* \* \***

##### 6.1.2.3.1 Handling MSRP connection

Upon receiving an indication to establish MSRP connection for SDS session as the terminating MCData client, the MCData client:

1. shall act as an MSRP client according to IETF RFC 6135 [12];

2. shall act either as an active endpoint or as a passive endpoint to open the transport connection, according to IETF RFC 6135 [12];

3. shall establish the MSRP connection according to the MSRP connection parameters in the SDP offer received in the SIP INVITE request according to IETF RFC 4975 [11]; and

4. if acting as an "active" endpoint, shall send an empty MSRP SEND request to bind the MSRP connection to the MSRP session from the perspective of the passive endpoint according to the rules and procedures of IETF RFC 4975 [11] and IETF RFC 6135 [12].

Once the MSRP session is established, on receipt of an MSRP request in the MSRP session, the MCData client:

1. shall follow the rules and procedures defined in IETF RFC 4975 [11] and in IETF RFC 6714 [13];

2. unless a response to the received MSRP request was already sent to the terminating participating MCData function as part of the previous step, shall generate an MSRP 200 (OK) response and send it to the terminating participating MCData function, according to the rules and procedures of IETF RFC 4975 [11];

3. if an MSRP SEND request indicates the use of chunking, shall wait until all further MSRP SEND requests for the remaining chunks of the MSRP message have been received and shall reassemble the entire set of MSRP requests into the MCData SDS message before delivering the content to the application; and

4. shall handle the received content as described in subclause 6.1.2.6.

On receiving MSRP 200 (OK) response to the first MSRP SEND request sent as "active" endpoint, or after sending MSRP 200 (OK) response to the first MSRP SEND request received as "passive" endpoint, the MCData client can generate and send an SDS message as specified in subclause 6.1.2.4, or can generate and send an SDS disposition notification for a received SDS message as specified in subclause 6.1.2.5, if requested.

Received content and disposition requests shall be handled as specified in subclause 6.1.2.6.

**\* \* \* \* \* NEXT CHANGE \* \* \* \* \***

##### 6.2.1.5.3 Handling of received MSRP messages

Upon receiving an MSRP SEND request from the controlling MCData function, the terminating participating MCData function:

1. shall forward the received MSRP SEND request to the terminating MCData client according to the rules and procedures of IETF RFC 4975 [11]; and

2. shall associate a timer with the forwarded MSRP SEND request, shall set the timer to an implementation dependent value and shall start the timer.

Upon the timer expiring without the terminating participating MCData function having received a response from the MCData Client associated with a forwarded MSRP SEND request, the terminating participating MCData function shall generate and send a MSRP 200 (OK) response for the forwarded MSRP SEND request to the controlling MCData function, according to the rules and procedures of IETF RFC 4975 [11].

Upon receiving an MSRP response from the terminating MCData client, the participating MCData function shall forward the MSRP response to the originating MCData client according to the rules and procedures of IETF RFC 4975 [11], unless the received MSRP response is an MSRP 200 (OK) for a recognized previously forwarded MSRP SEND request for which the terminating participating MCData function has already sent an MSRP 200 (OK) response to the controlling MCData function.

**\* \* \* \* \* NEXT CHANGE \* \* \* \* \***

##### 6.2.2.5.3 Handling of received MSRP messages

Upon receiving an MSRP SEND request from the controlling MCData function, the terminating participating MCData function:

1. shall forward the received MSRP SEND request to the terminating MCData client according to the rules and procedures of IETF RFC 4975 [11]; and

2. shall associate a timer with the forwarded MSRP SEND request, shall set the timer to an implementation dependent value and shall start the timer.

Upon the timer expiring without the terminating participating MCData function having received a response from the MCData Client associated with a forwarded MSRP SEND request, the terminating participating MCData function shall generate and send a MSRP 200 (OK) response for the forwarded MSRP SEND request to the controlling MCData function, according to the rules and procedures of IETF RFC 4975 [11].

Upon receiving an MSRP response from the terminating MCData client, the participating MCData function shall forward the MSRP response to the originating MCData client according to the rules and procedures of IETF RFC 4975 [11], unless the received MSRP response is an MSRP 200 (OK) for a recognized previously forwarded MSRP SEND request for which the terminating participating MCData function has already sent an MSRP 200 (OK) response to the controlling MCData function.

**\* \* \* \* \* NEXT CHANGE \* \* \* \* \***

#### 7.1.3.1 Handling MSRP connection

Upon receiving an indication to establish MSRP connection for file distribution as the terminating client, the MCData client:

1. shall act as an MSRP client according to IETF RFC 6135 [12];

2. shall act either as an active endpoint or as a passive endpoint to open the transport connection, according to IETF RFC 6135 [12];

3. shall establish the MSRP connection according to the MSRP connection parameters in the SDP offer received in the SIP INVITE request according to IETF RFC 4975 [11]; and

4. if acting as an "active" endpoint, shall send an empty MSRP SEND request to bind the MSRP connection to the MSRP session from the perspective of the passive endpoint according to the rules and procedures of IETF RFC 4975 [11] and IETF RFC 6135 [12].

Once the MSRP session is established, on receipt of an MSRP request in the MSRP session, the MCData client:

1. shall follow the rules and procedures defined in IETF RFC 4975 [11] and in IETF RFC 6714 [13];

2. unless a response to the received MSRP request was already sent to the terminating participating MCData function as part of the previous step, shall generate an MSRP 200 (OK) response and send it to the terminating participating MCData function, according to the rules and procedures of IETF RFC 4975 [11];

3. If an MSRP SEND request indicates the use of chunking, shall wait until all further MSRP SEND requests for the remaining chunks of the MSRP message have been received and shall reassemble the entire set of MSRP requests into the file before delivering the content to the application; and

4. shall handle the received content as described in subclause 7.1.3.2.

**\* \* \* \* \* NEXT CHANGE \* \* \* \* \***

#### 7.2.5.3 Handling of received MSRP messages

Upon receiving an MSRP SEND request from the controlling MCData function, the terminating participating MCData function:

1. shall forward the received MSRP SEND request to the terminating MCData client according to the rules and procedures of IETF RFC 4975 [11]; and

2. shall associate a timer with the forwarded MSRP SEND request, shall set the timer to an implementation dependent value and shall start the timer.

Upon the timer expiring without the terminating participating MCData function having received a response from the MCData Client associated with a forwarded MSRP SEND request, the terminating participating MCData function shall generate and send a MSRP 200 (OK) response for the forwarded MSRP SEND request to the controlling MCData function, according to the rules and procedures of IETF RFC 4975 [11].

Upon receiving an MSRP response from the terminating MCData client, the participating MCData function shall forward the MSRP response to the originating MCData client according to the rules and procedures of IETF RFC 4975 [11], unless the received MSRP response is an MSRP 200 (OK) for a recognized previously forwarded MSRP SEND request for which the terminating participating MCData function has already sent an MSRP 200 (OK) response to the controlling MCData function.

**\* \* \* \* \* END CHANGES \* \* \* \* \***