IPv6 Ready Logo Phase 2
Session Initiation Protocol

Test Profile
Proxy Server

Version 2.0.1
Modification Record

Version 0.1  Jan. 16, 2007  - First release
Ver.0.1.01  Feb. 23, 2007  - Changed tel-URI in FW-1-1-3 to a number that a nonexistent one:
05011112222 -> 00011112222

Ver.0.1.02  Feb. 26, 2007  - Corrected the misspellings that is pointed in internal review.

Ver.0.1.03  Feb. 26, 2007  - Corrected the other misspellings.

Ver.1.0.0  Apr. 27, 2007  - Modified the parts that UNH-IOL(Timothy Winters) pointed out. Corrected misspellings.
- Added explanation of Index.

Ver.1.0.1  Jul. 31, 2007  Version 1.0.1 release. Modified the parts that UNH-IOL (Timothy Winters) and NICK(Fang-Yu Ling) pointed out.

Ver.1.0.2  May.30, 2008  Remove TS-3-1-3. (The test is same purpose of TS-3-1-1) Collected some missspellings.

Ver.1.1.0  Dec. 12, 2008  Major revision up. (No modification).

Ver.2.0.0  Nov. 27, 2009  - Separated from Server Conformance Test Profile.
- Added Index of BASIC/ADVANCED tests.
- Changed [Judgement] into [Observable Results].
- Modified some incorrect parts.

Ver.2.0.1  Jan. 13, 2010  - Move FW-2-2-1 from “Supported NNI connection Processing of DNS(only AAAA record) of ADVANCED” to Routing of BASIC).
Acknowledgements

IPv6 Forum would like to acknowledge the efforts of the following organizations and commentators in the development of this test specification.

- IPv6 Promotion Council
  Certification Working Group
  SIP IPv6 Sub Working Group

- Commentators:

  Fang-Yu Ling
  NICI Ipv6 Standard and Interoperability Testing Lab. (Telecommunication Laboratories, Chunghwa Telecom Co., Ltd.)
# Table of Contents

[I] IPv6 Ready Logo Phase 2 Test Specification SIP IPv6 Server

1 Overview ........................................................................................................................................ 1

2 Requirement of conformance test .................................................................................................. 3
   2.1 Requirements based on Policy of SIP IPv6 Ready Logo .................................................. 3
   2.2 Other Requirements .............................................................................................................. 3
   2.3 Index of BASIC/ADVANCED tests ..................................................................................... 3

3 Common OBSERVABLE RESULTS ............................................................................................ 5
   3.1 generic_message .................................................................................................................. 5
   3.2 generic_forward_R-URI_non-responsible-domain ............................................................ 5
   3.3 generic_forward_R-URI_responsible-domain ..................................................................... 6
   3.4 generic_forward_from-PX2 ................................................................................................. 6
   3.5 generic_forward_from-UA11 ............................................................................................... 7
   3.6 generic_forward_from-UA12 ............................................................................................... 8
   3.7 generic_forward_request .................................................................................................... 9
   3.8 generic_forward_response ................................................................................................. 10
   3.9 generic_make_ACK_for-non2XX ......................................................................................... 11
   3.10 generic_make_CANCEL ..................................................................................................... 13
   3.11 generic_make_response-200_for-CANCEL ...................................................................... 16
   3.12 generic_make_response .................................................................................................... 17
   3.13 generic_proxy-auth .......................................................................................................... 18
   3.14 generic_status .................................................................................................................. 18
   3.15 generic_www-auth ............................................................................................................ 19

4. Test Profile:Proxy Server operation .......................................................................................... 20
   4.1 Session Establishment on One Proxy .................................................................................. 20
      4.1.1 PX-1-1-1 - SIP Proxy- Session Establishment Through One Proxy in the same domain ................................................................................................................................... 20
      4.1.2 PX-1-1-2 - SIP Proxy- Unsuccessful No Answer [CANCEL] ........................................ 33
      4.1.3 PX-1-1-3 - SIP Proxy- Session establishment and holding with re-INVITE ............... 43
      4.1.4 PX-1-2-1 - SIP Proxy- Unsuccessful Busy .................................................................. 62
      4.1.5 PX-1-2-2 - SIP Proxy- Unsuccessful No Response from User Agent ...................... 70
      4.1.6 PX-1-2-3 - SIP Proxy- Unsuccessful Temporarily Unavailable ................................. 77
   4.2 Session Establishment on Two Proxies ............................................................................. 85
      4.2.1 PX-2-1-1 - SIP Proxy- Session Establishment Through Two Proxies - Callee hanging up [another domain] (Caller) ........................................................................... 85
      4.2.2 PX-2-1-2 - SIP Proxy- Session Establishment Through Two Proxies - Callee hanging up [another domain] (Callee) .................................................................................. 99
      4.2.3 PX-2-1-3 - SIP Proxy- Unsuccessful No Answer [CANCEL] (Caller) ....................... 111
      4.2.4 PX-2-1-4 - SIP Proxy- Unsuccessful No Answer [CANCEL] (Callee) ...................... 122
      4.2.5 PX-2-1-5 - SIP Proxy- Session establishment and call hold by re-INVITE (Caller)131
      4.2.6 PX-2-1-6 - SIP Proxy- Session establishment and call hold by re-INVITE (Callee)151
4.2.7 PX-2-2-1 - SIP Proxy- Unsuccessful Busy (Caller) ..................................................169
4.2.8 PX-2-2-2 - SIP Proxy- Unsuccessful Busy (Callee) ..................................................177
4.2.9 PX-2-2-3 - SIP Proxy- Unsuccessful No Response from the UA through the other proxy ..........................................................................................................................183
4.2.10 PX-2-2-4 - SIP Proxy- Unsuccessful No Response from UA (Callee) .................191
4.2.11 PX-2-2-5 - SIP Proxy- Unsuccessful Temporarily Unavailable (Callee) ..........197

4.3 Routing ...............................................................................................................................204
4.3.1 FW-1-1-1 - SIP Proxy- Request-URI with escaped characters.................................204
4.3.2 FW-1-1-2 - SIP Proxy- Non-allowed parameters in Request-URI .........................215
4.3.3 FW-1-1-3 - SIP Proxy- Update of a Request-URI scheme ........................................220
4.3.4 FW-1-2-1 - SIP Proxy- Request-URI with an unknown scheme ...............................226
4.3.5 FW-1-2-2 - SIP Proxy- Request with an inexistent entity at the proxy ....................230
4.3.6 FW-1-2-3 - SIP Proxy- An unsupported option-tag in a Proxy-Require ..................233
4.3.7 FW-1-2-4 - SIP Proxy- Max-Forwards header field with a value of zero(0) .............237
4.3.8 FW-1-2-5 - SIP Proxy- Request without Max-Forwards header field ....................241
4.3.9 FW-1-2-6 - SIP Proxy- Timestamp header field in a 100 response .......................246
4.3.10 FW-2-1-1 - SIP Proxy- "sent-by" in Via with a domain name and a port ...............250
4.3.11 FW-2-1-2 - SIP Proxy- "sent-by" in Via with a domain name and without a port 258
4.3.12 FW-2-1-3 - SIP Proxy- Multiple 2xx responses .......................................................264
4.3.13 FW-2-2-1 - SIP Proxy- Receipt of 503 (Service Unavailable) response ................279
4.3.14 FW-2-2-2 - SIP Proxy- Receipt of 503 (Service Unavailable) response without Retry-After ..........................................................................................................................283
4.3.15 FW-2-2-3 - SIP Proxy- Forwarding of INVITE to an alternate server upon receipt of 503 response ..................................................................................................................287
4.3.16 FW-2-2-4 - SIP Proxy- Forwarding of INVITE to an alternate server upon receipt of 503 response ..................................................................................................................291
4.3.17 FW-3-1-1 - SIP Proxy- Session Establishment Through Two Proxies with a strict router in separate domains (Callee) ..................295
4.3.18 FW-3-1-2 - SIP Proxy- Session Establishment Through Two Proxies with a strict router in separate domains (Caller) ...........................................................................305
4.3.19 FW-4-1-1 - SIP Proxy- Unsuccessful No Answer with Proxy-Require [CANCEL] 310
4.3.20 FW-4-1-2 - SIP Proxy- processing of CANCEL upon no provisional response 315

4.4 Forwarding Request ..........................................................................................................319
4.4.1 RQ-1-1-1 - SIP Proxy- Receipt of OPTIONS from UAC ............................................319
4.4.2 RQ-1-1-2 - SIP Proxy- Receipt of OPTIONS when a UAS is ready to accept a call 323
4.4.3 RQ-2-1-1 - SIP Proxy- Future extension about new header fields ............................329
4.4.4 RQ-2-1-2 - SIP Proxy- Request without a tag in From field .................................333
4.4.5 RQ-2-1-3 - SIP Proxy- Response without a tag in To field ....................................345
4.4.6 RQ-2-1-4 - SIP Proxy- Unrecognized type of body ..................................................353
4.4.7 RQ-2-1-5 - SIP Proxy- Unrecognized encoding of body .........................................360
4.4.8 RQ-2-1-6 - SIP Proxy- Unrecognized language of body .........................................366
4.4.9 RQ-3-1-1 - SIP Proxy- Receipt of BYE with an unacceptable header field ..........371
4.4.10 RQ-3-1-2 - SIP Proxy- BYE not matching an existing dialog ...............................377
4.4.11 RQ-3-1-3 - SIP Proxy- BYE with a lower CSeq ......................................................385
4.4.12 RQ-4-1-1 - SIP Proxy- receipt of CANCEL with an unacceptable header field ... 389
4.4.13 RG-4-1-2 - SIP Registrar- Forwarding REGISTER request ........................................ 394
4.5 Forwarding Response .................................................................................................. 402
4.5.1 RS-1-1-1 - SIP Proxy- Unrecognized response code (2xx) .................................... 402
4.5.2 RS-1-1-2 - SIP Proxy- Unrecognized response code (4xx) .................................... 406
4.5.3 RS-1-1-3 - SIP Proxy- Unrecognized response code (5xx) .................................... 412
4.5.4 RS-1-1-4 - SIP Proxy- Unrecognized response code (6xx) .................................... 417
4.5.5 RS-1-1-5 - SIP Proxy- Provisional response other than 100 response ................. 422
4.5.6 RS-1-1-6 - SIP Proxy- Receipt of 200 with an unacceptable header field ............ 427
4.6 Forking .................................................................................................................... 432
4.6.1 FK-1-1-1 - SIP Proxy- Forked request with different Via branch parameters ....... 432
4.6.2 FK-1-1-2 - SIP Proxy- Choice of response to forked request (6xx response) ...... 441
4.6.3 FK-1-1-3 - SIP Proxy- Choice of response to forked request (The lowest response class) ..................................................................................... 447
4.6.4 FK-1-1-4 - SIP Proxy- Stateful proxy receiving a CANCEL request ................. 455
4.6.5 FK-1-1-5 - SIP Proxy- Forked request .................................................................. 463
4.7 Transaction .............................................................................................................. 471
4.7.1 TS-1-1-1 - SIP Proxy- INVITE client transaction (Stop of retransmission of INVITE upon Timer B fired) ................................................................. 471
4.7.2 TS-1-1-2 - SIP Proxy- INVITE client transaction (Stop of retransmission upon Receipt of 180 Ringing) ................................................................. 476
4.7.3 TS-1-1-3 - SIP Proxy- INVITE client transaction (Stop of ACK upon Timer D fired) 479
4.7.4 TS-2-1-1 - SIP Proxy- Non-INVITE Client Transaction (Stop of retransmission of CANCEL upon Timer F fired) ................................................................. 483
4.7.5 TS-2-1-2 - SIP Proxy- Non-INVITE Client Transaction (Stop of retransmission of BYE upon Timer F fired) ................................................................. 489
4.7.6 TS-2-1-3 - Non-INVITE Client Transaction (Receipt of 100 response to CANCEL and reset of Timer E with T2) ................................................................. 496
4.7.7 TS-2-1-4 - SIP Proxy- Non-INVITE Client Transaction (Receipt of 100 response to BYE and reset of Timer E with T2) ................................................................. 500
4.7.8 TS-3-1-1 - SIP Proxy- INVITE Server Transaction (Response after Timer H fired) 505
4.7.9 TS-3-1-2 - SIP Proxy- INVITE Server Transaction (Stop of retransmission after Timer H fired) ................................................................. 509
4.7.10 TS-3-1-4 - SIP Proxy- INVITE Server Transaction (Stop of retransmission of 4xx-6xx response upon receipt of ACK) ................................................................. 515
4.7.11 TS-3-1-5 - SIP Proxy- 487 to CANCEL for INVITE request after 64*T1 fired ...... 518
4.7.12 TS-4-1-1 - SIP Proxy- Non-INVITE Server Transaction (Response for CANCEL after Timer J fired) ................................................................. 521
4.7.13 TS-4-1-2 - SIP Proxy- Non-INVITE Server Transaction (Forwarding of BYE upon Timer J fired) ................................................................. 525
4.7.14 TS-5-1-1 - SIP Proxy- Session Establishment Through One Proxy without Provisional Response in the same domain ................................................................. 529
4.7.15 TS-5-1-2 - SIP Proxy- Session Establishment Through One Proxy with Multiple Provisional Responses in the same domain ................................................................. 537
4.7.16 TS-5-1-3 - SIP Proxy- Retransmission of INVITE Request Before Session Establishment Through One Proxy ................................................................. 542

4.8 Session Progress ................................................................. 548
  4.8.1 PG-1-1-1 - SIP Proxy- Session Progress response ...................... 548
  4.8.2 PG-1-1-2 - SIP Proxy- INVITE Client Transaction (Extension of Timer C) ...... 553
  4.8.3 PG-1-2-1 - SIP Proxy- INVITE Client Transaction (Reset of Timer C) .................. 556
  4.8.4 PG-1-2-2 - SIP Proxy- INVITE Client Transaction (No provisional response after Timer C fired) .................................................................................. 560

4.9 Transport .................................................................................. 564
  4.9.1 TP-1-1-1 - SIP Proxy- Receiving INVITE with additional bytes in a transport packet ................................................................. 564
  4.9.2 TP-1-2-1 - SIP Proxy- Transport packet of response ending before the end of the message body ................................................................. 569
  4.9.3 TP-1-2-2 - SIP Proxy- Transport packet of request ending before the end of the message body ................................................................. 574
  4.9.4 TP-2-1-1 - SIP Proxy- Receipt of “ICMP time exceeded” for a sent request....... 578
  4.9.5 TP-2-1-2 - SIP Proxy- Receipt of “ICMP time exceeded” for a sent response ...... 581
  4.9.6 TP-2-2-1 - SIP Proxy- Receipt of the “ICMP destination unreachable” message for a sent request .................................................................................. 585

4.10 Authentication ........................................................................... 588
  4.10.1 AU-1-1-1 - SIP Proxy- BYE request with user authentication .................. 588
  4.10.2 AU-1-1-2 - SIP Proxy- re-INVITE with user authentication ...................... 593

AUTHORS’ LIST ................................................................................ 600
1 Overview

This document describes details of the SIP Conformance Test. The format of the description block is as follows:

Description block

- **[NAME]**
  NAME is a name of the test.

- **[TARGET]**
  TARGET is a target node of the test.

- **[PURPOSE]**
  PURPOSE is a short statement describing what the test attempts to achieve. It is usually phrased as a simple assertion of the feature or capability to be tested.

- **[REQUIREMENT]**
  REQUIREMENT section specifies the functions and conditions that will be needed to perform the test.

- **[PARAMETER]**
  PARAMETER describes SIP URIs on the topology that relates to the test.

- **[ADDRESS]**
  ADDRESS describes IP addresses on the topology that relates to the test.

- **[TOPOLOGY]**
  TOPOLOGY describes the network used in the test.

- **[CONFIGURATION for NUT]**
  CONFIGURATION for NUT describes how to initialize and configure the NUT before starting each test. If a value is not provided, then the protocol's default value is used.

- **[INITIALIZATION]**
  INITIALIZATION describes step-by-step instructions for carrying out the setting before the test.

- **[PROCEDURE]**
  PROCEDURE describes step-by-step instructions for carrying out the test.

- **[OBSERVABLE RESULTS]**
  OBSERVABLE RESULTS describes expected result. If we can observe the same result as the description of Observable Results, the NUT passes the test.

- **[REFERENCE]**
  REFERENCE section contains some parts of specification related to the tests. It also shows the document names and section numbers.

NOTE: There are common Observable Results in the category of OBSERVABLE RESULTS. Refer to Section 3.

Acronyms

<table>
<thead>
<tr>
<th>Acronym</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT</td>
<td>Node Under Test (applicant implementation)</td>
</tr>
<tr>
<td>UA</td>
<td>SIP User Agent</td>
</tr>
<tr>
<td>EP</td>
<td>SIP Endpoint</td>
</tr>
<tr>
<td>B2BUA</td>
<td>SIP Back to Back User Agent</td>
</tr>
</tbody>
</table>
RG - SIP Registrar Server
Server - SIP Sever (Proxy and Registrar server)
IF - Interface
UNI - User-Network Interface
NNI - Network-Network Interface

Reference standards
(2) RFC3264: An Offer/Answer Model with Session Description Protocol
   (http://www.ietf.org/rfc/rfc3264.txt)
(3) RFC4566: SDP: Session Description Protocol (http://www.ietf.org/rfc/rfc4566.txt)
(4) RFC2617: HTTP Authentication: Basic and Digest Access Authentication
   (http://www.ietf.org/rfc/rfc2617.txt)
(5) RFC3665: SIP Basic Call Flow Examples (http://www.ietf.org/rfc/rfc3665.txt)
(6) IPv6 Ready Logo Phase 2 Policy
(7) SIP IPv6 Test Scope

Index
ex. [RFC3261 X.X.X]
   Please refer to the table of contents in RFC3261

[RFC3261-X-X]
   Please refer to the table number in Test-item-Priority
2 Requirement of conformance test

2.1 Requirements based on Policy of SIP IPv6 Ready Logo.

[PRq]
1. Supported transport protocol is only UDP.
2. The path MTU is 1500 bytes.
3. Supported URI scheme is only SIP-URI.
4. Supported media type of the message-body is application/sdp.
5. Only unicast session is supported.

2.2 Other Requirements

[ORq]
1. INVITE requests includes the bodies and any other requests doesn't include the bodies.
2. In case of a tester for conformance test sending message with Record-Route header field.
3. Use the value of Max-Forwards that are configured.
4. Send Digest authentication challenge for establishing a session.
5. Proxy must use location service. If an applicant implementation does not support registrar function, other registrar server is needed.

2.3 Index of BASIC/ADVANCED tests

<table>
<thead>
<tr>
<th>Test rank</th>
<th>Function</th>
<th>Test category</th>
<th>Profile No</th>
</tr>
</thead>
<tbody>
<tr>
<td>BASIC</td>
<td>Digest authentication (initial INVITE)</td>
<td>Session</td>
<td>PX-1-1-1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Establishment on One Proxy</td>
<td>PX-1-1-2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>PX-1-1-3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>PX-1-2-1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>PX-1-2-2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>PX-1-2-3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Routing</td>
<td>FW-1-1-1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FW-1-1-2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FW-1-2-1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FW-1-2-2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FW-1-2-3</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FW-1-2-4</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FW-1-2-5</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FW-1-2-6</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FW-2-1-1</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FW-2-1-2</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>FW-2-2-1</td>
</tr>
<tr>
<td>Module</td>
<td>Requirements</td>
<td></td>
<td></td>
</tr>
<tr>
<td>------------------------------------</td>
<td>--------------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Forwarding Request</td>
<td>FW-2-2-2, FW-4-1-1, FW-4-1-2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transaction</td>
<td>TS-1-1-1, TS-1-1-2, TS-1-1-3, TS-2-1-1, TS-2-1-2, TS-2-1-3, TS-2-1-4, TS-3-1-1, TS-3-1-2, TS-3-1-4, TS-3-1-5, TS-4-1-1, TS-4-1-2, TS-5-1-1, TS-5-1-2, TS-5-1-3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Session Progress</td>
<td>PG-1-1-1, PG-1-1-2, PG-1-2-1, PG-1-2-2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Transport</td>
<td>TP-1-1-1, TP-1-2-1, TP-1-2-2, TP-2-1-1, TP-2-1-2, TP-2-1-3</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Digest authentication for BYE request</td>
<td>AU-1-1-1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Digest authentication for re-INVITE request</td>
<td>AU-1-1-2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Processing of DNS (only AAAA record)</td>
<td>FW-2-2-3</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

ADVANCED
3.1 generic_message

Generic Observable Results for SIP message

- The empty line must be present even if the message-body is not. [RFC3261·7-2]

- Request-Line and Status-Line:
  Must exist as a start-line. [RFC3261.7]
  Must be terminated by a carriage-return line-feed sequence (CRLF). [RFC3261·7-1]
  SIP-version: Must be "SIP/2.0". [RFC3261·7-5,6]

- Header fields:
  Must be terminated by a carriage-return line-feed sequence (CRLF). [RFC3261·7-1]
  Recommended that Via, Route, Record-Route, Proxy-Require, Max-Forwards, and Proxy-Authorization appear towards the top of the message to facilitate rapid parsing. [RFC3261·7-7]

3.2 generic_forward_R-URI_non-responsible-domain

Generic Observable Results for Request-URI forwarded to non-responsible domain
· Request-Line:
  Request-URI: Must equal that of the forwarded request. [RFC3261-16-22,23]

3.3 generic_forward_R-URI_responsible-domain
Generic Observable Results for Request-URI forwarded to responsible domain

· Request-Line:
  Request-URI: Must equal one that the location service provides (i.e. Contact address registered by REGISTER request). [RFC3261-16-29,47]

3.4 generic_forward_from-PX2
Generic Observable Results for SIP message forwarded from PX2

The size of whole message must be less than or equal to 1300 bytes.

· Request-Line:
  Method: Must equal as that in the message from PX2. [RFC3261-16-42,43]
  Status-Line:
    Status-Code: Must equal as that in the message from PX2. [RFC3261-16-42,43]

· Header fields:
  Must not reorder field values with a common field name. [RFC3261-16-45]

  * To
    Must exist. [RFC3261-16-42,43]
    addr-spec: Must equal as that in the message from PX2. [RFC3261-16-42,43]
    tag-param: Must equal as that in the message from PX2 (Must not exist unless original message has any value). [RFC3261-16-42,43,124]

  * From
    Must exist. [RFC3261-16-42,43]
    addr-spec: Must equal as that in the message from PX2. [RFC3161-16-42,43]
    tag-param: Must equal as that in the message from PX2 (Must not exist unless original message has any value). [RFC3261-16-42,43]

  * Call-ID
    Must exist. [RFC3261-16-42,43]
    Must equal as that in the message from PX2. [RFC3261-16-42,43]
* CSeq
   Must exist. [RFC3261-16-42,43]
   seq-no: Must equal as that in the message from PX2. [RFC3261-16-42,43]
   Method: Must match that of the request. [RFC3261-16-42,43]
      Must equal as that in the message from PX2. [RFC3261-16-42,43]

* Content-Length
   Must equal as that in the message from PX2. [RFC3261-16-42,43]

* Bodies:
   Must equal as that in the message from PX2 (not be added, modified, deleted)

[RFC3261-16-46,131]

3.5 generic_forward_from-UA11

Generic Observable Results for SIP message forwarded from UA11

The size of whole message must be less than or equal to 1300 bytes.

* Request-Line:
   Method: Must equal as that in the message from UA11. [RFC3261-16-42,43]

* Status-Line:
   Status-Code: Must equal as that in the message from UA11. [RFC3261-16-42,43]

* Header fields:
   Must not reorder field values with a common field name. [RFC3261-16-45]

* To
   Must exist. [RFC3261-16-42,43]
   addr-spec: Must equal as that in the message from UA11. [RFC3261-16-42,43]
   tag-param: Must equal as that in the message from UA11 (Must not exist unless original message has any value). [RFC3261-16-42,43,124]

* From
   Must exist. [RFC3261-16-42,43]
   addr-spec: Must equal as that in the message from UA11. [RFC3261-16-42,43]
   tag-param: Must equal as that in the message from UA11 (Must not exist unless original message has any value). [RFC3261-16-42,43]

* Call-ID
Must exist. [RFC3261-16-42,43]
Must equal as that in the message from UA11. [RFC3261-16-42,43]

* CSeq
  Must exist. [RFC3261-16-42,43]
  seq-no: Must equal as that in the message from UA11. [RFC3261-16-42,43]
  Method: Must match that of the request. [RFC3261-16-42,43]
  Must equal as that in the message from UA11. [RFC3261-16-42,43]

* Content-Length
  Must equal as that in the message from UA11. [RFC3261-16-42,43]

* Bodies:
  Must equal as that in the message from UA11. [RFC3261-16-46,131]

3.6 generic_forward_from-UA12
Generic Observable Results for SIP message forwarded from UA12.

The size of whole message must be less than or equal to 1300bytes.

* Request-Line:
  Method: Must equal as that in the message from UA12. [RFC3261-16-42,43]
* Status-Line:
  Status-Code: Must equal as that in the message from UA12. [RFC3261-16-42,43]

* Header fields:
  Must not reorder field values with a common field name. [RFC3261-16-45]

* To
  Must exist. [RFC3261-16-42,43]
  addr-spec: Must equal as that in the message from UA12. [RFC3261-16-42,43]
  tag-param: Must equal as that in the message from UA12(Must not exist unless original message has any value). [RFC3261-16-42,43,124]

* From
  Must exist. [RFC3261-16-42,43]
  addr-spec: Must equal as that in the message from UA12. [RFC3261-16-42,43]
  tag-param: Must equal as that in the message from UA12(Must not exist unless original message has any value). [RFC3261-16-42,43]
* Call-ID  
  Must exist. [RFC3261-16-42,43]  
  Must equal as that in the message from UA12. [RFC3261-16-42,43]

* CSeq  
  Must exist. [RFC3261-16-42,43]  
  seq-no: Must equal as that in the message from UA12. [RFC3261-16-42,43]  
  Method: Must match that of the request. [RFC3261-16-42,43,]  
  Must equal as that in the message from UA12. [RFC3261-16-42,43]

* Content-Length  
  Must equal as that in the message from UA12. [RFC3261-16-42,43]

* Bodies:  
  Must equal as that in the message from UA12. [RFC3261-16-46,131]

3.7 generic_forward_request

Generic Observable Results for SIP request that was forwarded

* Header fields:

  * Via  
    Must exist. [RFC3261-42,43]  
    Must add a Via header field value. [RFC3261-8-55, RFC3261-16-81]  

    For the 1st Via header field, Must follow these rules.  
    via-branch:  
      token: Must be different from all values sent by NUT in this sequence.  
      (Excepting ACK for non-2xx response and CANCEL.)  
      [RFC3261-16-87]  

    via-branch:  
      token: Must begin with the characters "z9hG4bK". [RFC3261-8-23,  
      RFC3261-20-46]  
      transport: Must be "UDP". [PRq·1]  
      sent-by:  
        host: Recommended to be specified hostname of NUT.  
        [RFC3261-18·11,12]  

    For the 2nd Via header field, if the host portion of the "sent-by" parameter contains a domain name,  
    via-received: Must be added if the host portion of the "sent-by" parameter
contains a domain name. [RFC3261-18-27]
via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

For the Via header field that was in original message,
via-branch: Must equal as that in the message. [RFC3261-8-22]
sent-by: Must equal as that in the message. [RFC3261-16-42,43]
transport: Must equal as that in the message. [RFC3261-42,43]

* Route
  Must remove the first value in the Route header field from the request,
  if the value indicates NUT. [RFC3261-16-72]

* Record-Route
  Must insert a Record-Route header field value of NUT into the copy
  before any existing Record-Route header field values, when it is INVITE.
  [RFC3261-16-52,53][ORq-2]
The value is SIP URI. [RFC3261-16-54][PRq-3]
Lr-param: Must contain “lr” parameter. [RFC3261-16-55]

* Max-Forwards
  Must exist. [RFC3261-16-42,43]
  Must be decremented by one from that in the message from UA11.
  [RFC3261-16-49]

3.8 generic_forward_response
Generic Observable Results for SIP response that was forwarded

* Header fields:

  * Via
    Must exist. [RFC3261-42,43]
    Must not exist the first Via header field line in original message.
    [RFC3261-16-94]

    For the Via header field that was in original message,
    via-branch: Must equal as that in the message. [RFC3261-8-22,135]
    sent-by: Must equal as that in the message. [RFC3261-16-42,43,135]
    transport: Must equal as that in the message. [RFC3261-42,43,135]

  * Record-Route
    Must include the Record-Route header field value in original message.[ORq-2]
3.9 generic_make_ACK_for-non2XX

Generic Observable Results for ACK request for non-2xx response

- Request-Line:
  Method: Must be "ACK". [RFC3261 7.1].
  Request-URI: Must be the same value of that in the original request.
  Request-URI: Must not contain unescaped spaces or control characters. [RFC3261-7-5][RFC3261-19-11]
  Request-URI: Must not be enclosed in "<>". [RFC3261-7-4]

- Header fields:

  * From
    Must exist. [RFC3261-8-1]
    Must be the same value of that in the original request. [RFC3261-17-32]
    addr-spec: Must be the specified SIP-URI as NUT(AoR). [RFC3261 8.1.1.3],[RFC3261 20.20]
    addr-spec: Must be enclosed in "<>", if a comma, semicolon, or question mark is contained. [RFC3261-20-13]
    tag-param: Must exist. [RFC3261-8-9]

  * Call-ID
    Must exist. [RFC3261-8-1]
    Must be the same value of that in the original request. [RFC3261-17-32]

  * To
    Must exist. [RFC3261-8-1]
    Must equal the To header field in the response being acknowledged. [RFC3261-17-33]
    addr-spec: Must be the specified SIP-URI as UA1(AoR). (Excepting REGISTER request.). [RFC3261-12-34]
    addr-spec: Must be enclosed in "<>", if a comma, semicolon, or question mark is contained. [RFC3261-20-13]
    tag-param: Must equal that in corresponding response, if present. [RFC3261-16-42,43]

  * Via
    MUST be equal to the top Via header field of the original request. [RFC3261-17-35]

  * CSeq
    Must exist. [RFC3261-8-1]
Must be less than $2^{31}$. [RFC3261-8-15,16, RFC3261-20-32]
Method: Must match that of the request. [RFC3261-8-14, RFC3261-17-37]
sequence number: Must be the same value as was present in the original request. [RFC3261-17-36]

* Max-Forwards
  Must exist. [RFC3261-8-1]
  Must be the value specified in the tester configuration. [ORq-3]

* Content-Length
  Must be the size of the message-body, in decimal number of octets. [RFC3261 25.1]

* Require
  Must not exist. [RFC3261-8-80]

* Proxy-Require
  Must not exist. [RFC3261-8-80]

* Accept
  Must not exist. [RFC3261 20],[RFC3261-20-8]

* Accept-Encoding
  Must not exist. [RFC3261 20],[RFC3261-20-8]

* Accept-Language
  Must not exist. [RFC3261 20],[RFC3261-20-8]

* Alert-Info
  Must not exist. [RFC3261 20],[RFC3261-20-8]

* Allow
  Must not exist. [RFC3261 20],[RFC3261-20-8]

* Expires
  Must not exist. [RFC3261 20],[RFC3261-20-8]

* In-Reply-To
  Must not exist. [RFC3261 20],[RFC3261-20-8]

* Organization
  Must not exist. [RFC3261 20],[RFC3261-20-8]
**Priority**
Must not exist. [RFC3261 20], [RFC3261·20·8]

**Reply-To**
Must not exist. [RFC3261 20], [RFC3261·20·8]

**Server**
Must not exist. [RFC3261 20], [RFC3261·20·8]

**Subject**
Must not exist. [RFC3261 20], [RFC3261·20·8]

**Supported**
Must not exist. [RFC3261 20], [RFC3261·20·8]

**Warning**
Must not exist. [RFC3261 20], [RFC3261·20·8]

* Bodies:
  Must not exist. [RFC3261·17·39]

**3.10 generic_makeCANCEL**
Generic Observable Results for CANCEL request

The destination address, port, and transport
  Must be equal those used to send the original request. [RFC3261·9·11]

* Request-Line:
  Method: Must be "CANCEL". [RFC3261 7.1].
  Request-URI: Must be equal that in the request being cancelled. [RFC3261·9·2]

* Header fields:

  * Max-Forwards
    Must exist. [RFC3261·8·1]
    Must be the value specified in the tester configuration. [ORq·3]

  * Via
    Must exist. [RFC3261·8·1]
    MUST have only a single Via header field value. [RFC3261·9·3]
Must be the value matching the top Via value in the request being cancelled. (Tester check the value of sent-by, via-branch, and via-received, respectively.)

`via-branch`: Must exist in each Via header field. [RFC3261·8·21]
`via-branch`:
  token: Must begin with the characters "z9hG4bK". [RFC3261·8·23, RFC3261·20·46]
  transport: Must be "UDP". [PRq-1]
  sent-by:
    host: Recommended to be specified hostname of NUT. [RFC3261·18·11,12]

* Call-ID
  Must equal that in the request being cancelled. [RFC3261·9·2]

* To
  Must equal that in the request being cancelled, including a tag. (Tester check URI and tag, respectively.) [RFC3261·9·2]

* CSeq
  sequence number: Must equal that in the request being cancelled. [RFC3261·9·2]
  Must equal "CANCEL". [RFC3261·9·4]

* From
  Must equal that in the request being cancelled, including a tag. (Tester check the value of URI and tag, respectively.) [RFC3261·9·2]

* Content-Length
  Must be the size of the message-body, in decimal number of octets.
  [RFC3261 25.1]

* Route
  Must exist if the request being cancelled contains a Route header field.
  [RFC3261·9·5]

* Accept
  Must not exist. [RFC3261 20],[RFC3261·20·8]

* Accept-Encoding
  Must not exist. [RFC3261 20],[RFC3261·20·8]

* Accept-Language
  Must not exist. [RFC3261 20],[RFC3261·20·8]
* Alert-Info
  Must not exist. [RFC3261 20], [RFC3261-20-8]

* Allow
  Must not exist. [RFC3261 20], [RFC3261-20-8]

* Contact
  Must not exist. [RFC3261 20], [RFC3261-20-8]

* Content-Disposition
  Must not exist. [RFC3261 20], [RFC3261-20-8]

* Content-Encoding
  Must not exist. [RFC3261 20], [RFC3261-20-8]

* Content-Language
  Must not exist. [RFC3261 20], [RFC3261-20-8]

* Expires
  Must not exist. [RFC3261 20], [RFC3261-20-8]

* In-Reply-To
  Must not exist. [RFC3261 20], [RFC3261-20-8]

* MIME-Version
  Must not exist. [RFC3261 20], [RFC3261-20-8]

* Organization
  Must not exist. [RFC3261 20], [RFC3261-20-8]

* Priority
  Must not exist. [RFC3261 20], [RFC3261-20-8]

* Proxy-Authorization
  Must not exist. [RFC3261 20], [RFC3261-20-8]

* Proxy-Require
  Must not exist. [RFC3261-9-6] [RFC3261 20], [RFC3261-20-8]

* Reply-To
  Must not exist. [RFC3261 20], [RFC3261-20-8]

* Require
Must not exist. [RFC3261-9-6]

* Subject
  Must not exist. [RFC3261 20],[RFC3261-20-8]

- Bodies:
  Must not exist. [ORq-1]

### 3.11 generic_make_response-200_for-CANCEL

Generic Observable Results for 200 response for CANCEL request

The size of whole response message must be less than or equal to 1500 bytes.

- Status-Line:
  Status-Code: Must exist. [RFC3261 7.2]
  Status-Code: Must be three digit integer. [RFC3261 7.2]

- Header fields:

  * From
    Must exist. [RFC3261 20]
    Must equal that of the request. (Tester check the value of URI and tag, respectively.). [RFC3261-8-98]

  * Call-ID
    Must exist. [RFC3261 20]
    Must equal that of the request. [RFC3261·8·99]

  * CSeq
    Must exist. [RFC3261 20]
    Must equal that of the request. [RFC3261·8·100]

  * Via
    Must exist. [RFC3261 20]
    via·branch: Must exist in each Via header field. [RFC3261·8·21]
    via·param: Must equal that in the request. [RFC3261·8·101]
    via·param: Must maintain the same ordering as that in the request.(Tester check the value of sent·by, via·branch, and via·received of expect 1st line, respectively.) [RFC3261·8·102]

  * To
    Must exist. [RFC3261 20]
tag-param: Must equal To tag in the response to the original request.  
[RFC3261-8-103]

addr-spec: Must equal that in the request. [RFC3261-8-103]

* Content-Length
   Must be the size of the message-body, in decimal number of octets.  
   [RFC3261 25.1]

3.12 generic_make_response

Generic Observable Results for SIP response

The size of whole response message must be less than or equal to 1500bytes.

* Status-Line:
   Status-Code: Must exist. [RFC3261 7.2]
   Status-Code: Must be three digit integer. [RFC3261 7.2]

* Header fields:

* From
   Must exist. [RFC3261 20]
   Must equal that of the request. (Tester check the value of URI and tag,  
   respectively.) [RFC3261-8-98]

* Call-ID
   Must exist. [RFC3261.20]
   Must equal that of the request. [RFC3261-8-99]

* CSeq
   Must exist. [RFC3261 20]
   Must equal that of the request. [RFC3261-8-100]

* Via
   Must exist. [RFC3261 20]
   via-branch: Must exist in each Via header field. [RFC3261-8-21]
   via-param: Must equal that in the request. [RFC3261-8-101]
   via-param: Must maintain the same ordering as that in the request.(Tester check  
   the value of sent-by, via-branch, and via-received of expect 1st line,  
   respectively.) [RFC3261-8-102]

* To
   Must exist. [RFC3261 20]
Must equal that of the request if the request contained a tag-param. (Tester check the value of URI and tag, respectively.) [RFC3261-8-103]

addr-spec: Must equal that in the request if the request did not contain a tag-param. [RFC3261-8-104]
tag-param: Must added if the request did not contain a tag-param. (Excepting 100 response.) [RFC3261-8-105]

* Content-Length
  Must be the size of the message-body, in decimal number of octets. [RFC3261 25.1]

3.13 generic_proxy-auth
Generic Observable Results for SIP authentication challenge (Proxy-Authenticate)

* Header fields:
  *
  Proxy-Authenticate

    challenge: Must begin with "Digest". [RFC2617-7-3-1]
    nonce: Must exist. [RFC2617 3.2.1]
    realm: Must exist. [RFC2617 3.2.1]
    qop: Must exist. [RFC3261-22-36]
      Must include “auth”. [RFC3261-22-37]
    uri: Must be enclosed in quotation marks. [RFC3261-22-34]
    algorithm: Must equal "MD5", if algorithm parameter exist. [RFC2617 3.2.1]

3.14 generic_status
Generic Observable Results for SIP response

The size of whole response message must be less than that or equal to 1500bytes.

* Status-Line:
  Status-Code: Must exist. [RFC3261 7.2]
  Status-Code: Must be three digit integer. [RFC3261 7.2]

* Header fields:

  Must exist. [RFC3261 20]
  Must equal that of the request. (Tester check the value of URI and tag, respectively.) [RFC3261-8-98]
* Call-ID
  Must exist. [RFC3261 20]
  Must equal that of the request. [RFC3261-8-99]

* CSeq
  Must exist. [RFC3261 20]
  Must equal that of the request. [RFC3261-8-100]

* Via
  Must exist. [RFC3261 20]
  via-branch: Must exist in each Via header field. [RFC3261-8-21]
  via-param: Must equal that in the request. [RFC3261-8-101]
  via-param: Must maintain the same ordering as that in the request. (Tester check
  the value of sent-by, via-branch, and via-received of expect 1st line,
  respectively.) [RFC3261-8-102]

* To
  Must exist. [RFC3261 20]
  Must equal that of the request if the request contained a tag-param. (Tester check
  the value of URI and tag, respectively.) [RFC3261-8-103]
  addr-spec: Must equal that in the request if the request did not contain a
  tag-param. [RFC3261-8-104]
  tag-param: Must added if the request did not contain a tag-param. (Excepting 100
  response.) [RFC3261-8-105]

3.15 generic_www-auth
Generic Observable Results for SIP authentication challenge (WWW-Authenticate)

- Header fields:
  * WWW-Authenticate
  
  challenge: Must begin with "Digest". [RFC2617-3-1]
  nonce: Must exist. [RFC2617 3.2.1]
  realm: Must exist. [RFC2617 3.2.1]
  qop: Must exist. [RFC3261-22-36]
    Must include auth parameter. [RFC3261-22-37]
  uri: Must be enclosed in quotation marks. [RFC3261-22-34]
  algorithm: Must equal "MD5", if algorithm parameter exist. [RFC2617 3.2.1]
4. Test Profile: Proxy Server operation

4.1 Session Establishment on One Proxy

4.1.1 PX-1-1-1 - SIP Proxy- Session Establishment Through One Proxy in the same domain

**[NAME]**
PX-1-1-1 - SIP Proxy- Session Establishment Through One Proxy in the same domain

**[TARGET]**
SIP Proxy

**[PURPOSE]**
Verify that a NUT properly processes when a session is established through one proxy in the same domain.

**[REQUIREMENT]**
Set up registrar server to use location service, if necessary.

**[PARAMETER]**

| NUT(AOR) | sip:ss.under.test.com |
| Registrar(AOR) | sip:reg.under.test.com |
| UA11(AOR) | sip:UA11@under.test.com |
| UA11(Contact) | sip:UA11@node.under.test.com |
| UA12(AOR) | sip:UA12@under.test.com |
| UA12(Contact) | sip:UA12@node11.under.test.com |

**[ADDRESS]**

| NUT (IPv6) | 3ffe:501:ffff:50::50/64 |
| Registrar (IPv6) | 3ffe:501:ffff:50::60/64 |
| UA11 (IPv6) | 3ffe:501:ffff:1::1/64 |
| UA12 (IPv6) | 3ffe:501:ffff:2::2/64 |
| R (IPv6) | 3ffe:501:ffff:50::1/64 |

**[TOPOLOGY]**

```
+----------+----------+----------|
|          |          |          |
|          |          |          |
|          |          |          |
|          |          |          |
|          |          |          |
|          |          |          |
|          |          |          |
|          |          |          |
|          |          |          |
|          |          |          |
|          |          |          |
|          |          |          |
|          |          |          |
```

IPv6 FORUM TECHNICAL DOCUMENT

IPv6 Ready Logo Program

Phase 2 Test Specification

SIP IPv6
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip: ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
[PROCEDURE]

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required. (*1)
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE. (*2)
6. UA11 Receive 100 Trying. (*3)
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing. (*4)
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK. (*5)
11. UA11 Send ACK.
12. UA12 Receive ACK. (*6)
13. UA12 Send BYE.

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required. (*1)
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE. (*2)
6. UA11 Receive 100 Trying. (*3)
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing. (*4)
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK. (*5)
11. UA11 Send ACK.
12. UA12 Receive ACK. (*6)
13. UA12 Send BYE.
14. UA11 Receive BYE. (*7)  
15. UA11 Send 200.  
16. UA12 Receive 200. (*8)  

=== Message example ===

1. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0  
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43  
Max-Forwards: 70  
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl  
To: UA12 <sip:UA12@under.test.com>  
Call-ID: 3848276298220188511@under.test.com  
CSeq: 1 INVITE  
Contact: <sip:UA11@node.under.test.com>  
Supported: none  
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE  
Content-Type: application/sdp  
Content-Length: 151

v=0  
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1  
s=-  
c=IN IP6 3ffe:501:ffff:1::1  
t=0 0  
m=audio 49172 RTP/AVP 0  
a=rtpmap:0 PCMU/8000

/* NUT challenges UA11 for authentication */

2. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required  
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43  
;received=3ffe:501:ffff:1::1  
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl  
To: UA12 <sip:UA12@under.test.com>;tag=3flal12sf  
Call-ID: 3848276298220188511@under.test.com  
CSeq: 1 INVITE  
Proxy-Authenticate: Digest realm="under.test.com", qop="auth",  
nonce="f84f1ce41e6cbe5aa9c8e88d359",  
opaque="", stale=FALSE, algorithm=MD5
3. ACK UA11 -> NUT

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=3flal12sf
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 ACK
Content-Length: 0

/* UA11 responds by re-sending the INVITE with authentication credentials in it. */

4. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1ce41e6be5aae9c8e88d3d59", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA12@under.test.com",
response="b51e504e73af5429e4f2bd7f8dc4654"
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

/* Proxy(NUT) accepts the credentials and forwards the INVITE to UA12
2. Client for UA11 prepares to receive data on port 49172 from the
network. */

5. INVITE NUT -> UA12

INVITE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 384827629822018511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

6. 100 Trying NUT -> UA11

SIP/2.0 100 Trying
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Length: 0
7. 180 Ringing UA11 -> NUT

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501::ff50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501::ff1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

8. 180 Ringing NUT -> UA11

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501::ff1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

9. 200 OK UA12 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501::ff50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501::ff1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA12 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

10. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 384827629822018511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 147

v=0
do=UA12 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

11. ACK UA11 -> NUT

ACK sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf76
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1cece41e6cbe5a9c8e88d359", opaque=""
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA12@under.test.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 ACK
Content-Length: 0

12. ACK NUT -> UA12

ACK sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b76
:received=3ffe:501:fff1::1
Max-Forwards: 69
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 ACK
Content-Length: 0

13. BYE UA12 -> NUT

BYE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA12@under.test.com>;tag=314159
To: NUT <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

14. BYE NUT -> UA11

BYE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK74b43
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:fff2::2
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
15. 200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK74b43
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:2::2
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

16. 200 OK NUT -> UA12

SIP/2.0 200 OK
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:2::2
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

[OBSERVABLE RESULTS]
*1:407 response from NUT to UA11.
    As a SIP Message,
    See generic_message

    As a SIP response,

    · Status-Line:
      See generic_make_response
      Status-Code: Must be "407". [RFC3261 22.3]

    · Header fields:
      See generic_make_response
See generic_proxy-auth

* Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

*2: INVITE request from NUT to UA12.

As a SIP Message,
  See generic_message

As a SIP request,

* Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain

* Header fields:
  * outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

* Bodies:
  See generic_forward_from-UA11

*3: 100 response from NUT to UA11. (Optional)

As a SIP Message,
  See generic_message

As a SIP response,

* Status-Line:
  See generic_make_response
  Status-Code: Must be "100". [RFC3261 4]

* Header fields:
  See generic_make_response
* Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]
*4:180 response from NUT to UA11.

As a SIP Message,

See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA12
  Status-Code: Must be "180". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA12
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter
    contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
    received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from-UA12

*5:200 response from NUT to UA11.

As a SIP Message,

See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA12
  Status-Code: Must be "200". [RFC3261-104]

- Header fields:
  See generic_forward_from-UA12
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter
    contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
    received. [RFC3261-18-28]
- Bodies:
  See generic_forward_from-UA12

*6: ACK request from NUT to UA12.

As a SIP Message,
  See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_responsible-domain

- Header fields:
  - outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

- Bodies:
  See generic_forward_from-UA11

*7: BYE request from NUT to UA11.

As a SIP Message,
  See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA12

- Header fields:
  - inside of a dialog
    See generic_forward_from-UA12
    See generic_forward_request

- Bodies:
  See generic_forward_from-UA12

*8: 200 response from NUT to UA12.
As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA11
  Status-Code: Must be "200". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA11
  See generic_forward_response

- Bodies:
  See generic_forward_from-UA12

[REFERENCE]
Sequence from RFC3665 Section 3.2.

4.1.2 PX-1-1-2 - SIP Proxy- Unsuccessful No Answer [CANCEL]

[NAME]
PX-1-1-2 - SIP Proxy- Unsuccessful No Answer [CANCEL]

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when a UA doesn’t send any response and receives a CANCEL request from the other UA.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]
<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]
Phase 2 Test Specification

SIP IPv6

---+-----------+---------
|           |
|          UA11 |
|           R |
---+---R-------+-----------+---------
|           |           |
|         NUT       Registrar |
| R12 |
---+-----------+---------
|           |
|          UA12 |

[CONFIGURATION for NUT]

| NUT | sip: ss.under.test.com |
| NUT(IPADDRESS) | 3ffe:501::ffff:50::50/64 (IPv6) |

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.
**1. REGISTER Request.**
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:----&gt;</td>
<td></td>
<td>1. INVITE</td>
</tr>
<tr>
<td>&lt;-----:------&gt;</td>
<td></td>
<td>2. 407</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:----&gt;</td>
<td></td>
<td>3. ACK</td>
</tr>
<tr>
<td>&lt;-----:------&gt;</td>
<td></td>
<td>4. INVITE</td>
</tr>
<tr>
<td>&lt;-----:------&gt;</td>
<td></td>
<td>5. INVITE</td>
</tr>
<tr>
<td>&lt;-----:------&gt;</td>
<td></td>
<td>6. 100</td>
</tr>
<tr>
<td>&lt;-----:------&gt;</td>
<td></td>
<td>7. 180</td>
</tr>
<tr>
<td>&lt;-----:------&gt;</td>
<td></td>
<td>8. 180</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:----&gt;</td>
<td></td>
<td>9. CANCEL</td>
</tr>
<tr>
<td>&lt;-----:------&gt;</td>
<td></td>
<td>10. 200 (*1)</td>
</tr>
<tr>
<td>&lt;-----:------&gt;</td>
<td></td>
<td>11. CANCEL (*2)</td>
</tr>
<tr>
<td>&lt;-----:------&gt;</td>
<td></td>
<td>12. 200</td>
</tr>
<tr>
<td>&lt;-----:------&gt;</td>
<td></td>
<td>13. 487</td>
</tr>
<tr>
<td>&lt;-----:------&gt;</td>
<td></td>
<td>14. ACK (*3)</td>
</tr>
<tr>
<td>&lt;-----:------&gt;</td>
<td></td>
<td>15. 487 (*4)</td>
</tr>
</tbody>
</table>
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA11 Send CANCEL.
10. UA11 Receive 200 OK. (*1)
11. UA12 Receive CANCEL. (*2)
12. UA12 Send 200 OK.
13. UA12 Send 487 Request Terminated.
14. UA12 Receive ACK. (*3)
15. UA11 Receive 487 Request Terminated. (*4)
16. UA11 Send ACK.

=== Message example ===

1. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
2. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
:received=3ffe:501::ff1:1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=3ffal12sf
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 INVITE
Proxy-Authenticate: Digest realm="under.test.com", qop="auth",
    nonce="f84f1ce41e6cbe5a9c8e88d359", opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. ACK UA11 -> NUT

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=3fla12sf
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 ACK
Content-Length: 0

4. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Proxy-Authentication: Digest username="UA11",
    realm="under.test.com",
    nonce="f84f1ce41e6cbe5a9c8e88d359", opaque="", qop=auth, nc=00000004, cnonce="6f54a149",
    uri="sip:UA12@under.test.com",
    response="b51e504e73af54829e4f2bd7f8dc4654"
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501::ffff:1::1
s=-
c=IN IP6 3ffe:501::ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

5. INVITE NUT -> UA12

INVITE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501::ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501::ffff:1::1
s=-
c=IN IP6 3ffe:501::ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

6. 100 Trying NUT -> UA11

SIP/2.0 100 Trying
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501::ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

7. 180 Ringing UA12 -> NUT

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Content-Length: 0

8. 180 Ringing NUT -> UA11

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Content-Length: 0

9. CANCEL UA11 -> NUT

CANCEL sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Route: <sip:ss.under.test.com;lr>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 CANCEL
Content-Length: 0

10. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xB9vxSit55XU7p8@under.test.com
CSeq: 2 CANCEL
Content-Length: 0

11. CANCEL NUT -> UA12

CANCEL sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xB9vxSit55XU7p8@under.test.com
CSeq: 2 CANCEL
Content-Length: 0

12. 200 OK UA12 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:50::50
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xB9vxSit55XU7p8@under.test.com
CSeq: 2 CANCEL
Content-Length: 0

13. 487 Request Terminated UA12 -> NUT

SIP/2.0 487 Request Terminated
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:1::1
14. ACK NUT -> UA12

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

15. 487 Request Terminated NUT -> UA11

SIP/2.0 487 Request Terminated
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

16. ACK UA11 -> NUT

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

[OBSERVABLE RESULTS]

*1:200 response from NUT to UA11.
   As a SIP Message,
   See generic_message
As a SIP response,

- **Status-Line:**
  See generic_make_response-200_for-CANCEL
  Status-Code: Must be "200". [RFC3261 16.10]

- **Header fields:**
  See generic_make_response-200_for-CANCEL
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

*2: CANCEL request from NUT to UA12.

As a SIP Message,
  See generic_message

As a SIP request,

- **Request-Line:**
  See generic_make_CANCEL

- **Header fields:**
  - outside of a dialog
    See generic_make_CANCEL

- **Bodies:**
  See generic_make_CANCEL

*3: ACK request from NUT to UA12.

As a SIP Message,
  See generic_message

As a SIP request,

- **Request-Line:**
  See generic_make_ACK_for-non2XX
- Header fields:
  - outside of a dialog
    See generic_make_ACK_for-non2XX

- Bodies:
  See generic_make_ACK_for-non2XX

*4:487 response from NUT to UA11.

As a SIP Message,
  See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response
  Status-Code: Must be "487". [RFC3261-9-15]

- Header fields:
  See generic_make_response

* To
tag-param: Should be the same value of "10. 200 response". [RFC3261-9-16]

* Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
  Sequence from RFC3665 Section 3.8.

4.1.3 PX-1-1-3 - SIP Proxy- Session establishment and holding with re-INVITE

[NAME]
PX-1-1-3 - SIP Proxy- Session establishment and holding with re-INVITE

[TARGET]
SIP Proxy
[PURPOSE]
Verify that a NUT properly processes when a session is established and on holding with re-INVITE.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

---+-----------+---------
|           |          |
|          | UA11    |
| R11      |   |        |
| R---------+-----------+---------|
|           |          |
|         | NUT      | Registrar|
| R12      |   |        |
|          |           |
|          |           |
|          | UA12     |        |

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

```
UA11 : NUT : UA12
| | |
| | |
|———> | : | 1. INVITE
|<———> | : | 2. 407
|———> | : | 3. ACK
| | |
|———> | : | 4. INVITE
| | ———> | 5. INVITE
|<———> | : | 6. 100 Trying
| : |<———> | 7. 180 Ringing
|<———> | : | 8. 180 Ringing
| : | <———> | 9. 200 OK
<———> | : | 10. 200 OK
```
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK.
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA12 Send INVITE.
14. UA11 Receive INVITE. (*1)
15. UA11 Send 200 OK.
16. UA12 Receive 200 OK. (*2)
17. UA12 Send ACK.
18. UA11 Receive ACK. (*3)
19. UA12 Send INVITE.
20. UA11 Receive INVITE. (*4)
21. UA11 Send 200 OK.
22. UA12 Receive 200 OK. (*5)
23. UA12 Send ACK.
24. UA11 Receive ACK. (*6)
25. UA11 Send BYE.
26. UA12 Receive BYE.
27. UA12 Send 200 OK.
28. UA11 Receive 200 OK.

== Message example ==

1. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

2. 407 Proxy Authentication Required NUT -> UA11
SIP/2.0 407 Proxy Authentication Required
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
:received=3ffe:501::ff1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=3flal12sf
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 INVITE
Proxy-Authenticate: Digest realm="under.test.com", qop="auth",
nonce="f84f1ce41e6cb5aee9c8e88d359",
opaque=""", stale=FALSE, algorithm=MD5
Content-Length: 0

3. ACK UA11 -> NUT

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=3flal12sf
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 ACK
Content-Length: 0

4. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1ce41e6cb5aee9c8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA12@under.test.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
5. INVITE NUT -> UA12

INVITE sip:UA11@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

6. 100 Trying NUT -> UA11

SIP/2.0 100 Trying
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
7. 180 Ringing UA12 -> NUT

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

8. 180 Ringing NUT -> UA11

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

9. 200 OK UA12 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA12 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

10. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA12 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

11. ACK UA11 -> NUT

ACK sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b7b
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
   realm="under.test.com",
   nonce="f84f1e4e41e6cbe5aeea9c8e88d359", opaque="",
   qop=auth, nc=00000004, cnonce="6f54a149",
   uri="sip:UA12@under.test.com",
   response="b51e504e73af54829e4f2bd7f8dc4654"
Route: <sip:ss.under.test.com;lr>,
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55UX7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

12. ACK NUT -> UA12

ACK sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKgs24u
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b7b
   ;received=3ffe:501:ffff:1::1
Max-Forwards: 69
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55UX7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

13. INVITE UA12 -> NUT

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnashds7
   ;received=3ffe:501:ffff:2::2
Max-Forwards: 70
Route: <sip:ss.under.test.com;lr>
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55UX7p8@under.test.com
CSeq: 15 INVITE
Content-Type: application/sdp
Content-Length: 149

v=0
14. INVITE NUT -> UA11

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKjgk15
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:2::2
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 15 INVITE
Contact: <sip:UA12@node11.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 149

v=0
o=UA12 2890844527 2890844528 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=sendonly

15. 200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKjgk15
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 15 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 150

v=0
o=UA11 2890844526 2890844527 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=recvonly

16. 200 OK NUT -> UA12

SIP/2.0 200 OK
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 15 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 150

v=0
o=UA11 2890844526 2890844527 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=recvonly
17. ACK UA12 -> NUT

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKhyu5r
:received=3ffe:501::ff2::2
Max-Forwards: 70
Route: <sip:ss.under.test.com;lr>
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 15 ACK
Content-Length: 0

18. ACK NUT -> UA11

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKjugh6
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKhyu5r
:received=3ffe:501::ff2::2
Max-Forwards: 69
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 15 ACK
Content-Length: 0

19. INVITE UA12 -> NUT

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKhgt56
:received=3ffe:501::ff2::2
Max-Forwards: 70
Route: <sip:ss.under.test.com;lr>
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 16 INVITE
Contact: <sip:UA12@node11.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 149
v=0
o=UA12 2890844527 2890844529 IN IP6 3ffe:501::ffff:2::2
s=-
c=IN IP6 3ffe:501::ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

20. INVITE NUT -> UA11

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKsdp9i
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKhgt56
;received=3ffe:501:ffff:2::2
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 16 INVITE
Contact: <sip:UA12@node11.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 149

v=0
o=UA12 2890844527 2890844529 IN IP6 3ffe:501::ffff:2::2
s=-
c=IN IP6 3ffe:501::ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

21. 200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKsdp9i
;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKhgt56
;received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 16 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 150

v=0
o=UA11 2890844526 2890844528 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

22. 200 OK NUT -> UA12

SIP/2.0 200 OK
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKhgt56
:received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 16 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 150

v=0
o=UA11 2890844526 2890844528 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
23. ACK UA12 -> NUT

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKjyt4
  ;received=3ffe:501:ffff:2::2
Max-Forwards: 69
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 16 ACK
Content-Length: 0

24. ACK NUT -> UA11

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKjyt4
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKjyt4
  ;received=3ffe:501:ffff:2::2
Max-Forwards: 69
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 16 ACK
Content-Length: 0

25. BYE UA11 -> NUT

BYE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKjyt4
Route: <sip:ss.under.test.com;lr>
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 3 BYE
Content-Length: 0

26. BYE NUT -> UA12

BYE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKjyt4
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKjyt4
27. 200 OK UA12 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKju6y8
:received=3ffe:501::ff:1::1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK78jk6
:received=3ffe:501::ff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 3 BYE
Content-Length: 0

28. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKju6y8
:received=3ffe:501::ff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 3 BYE
Content-Length: 0

[OBSERVABLE RESULTS]
*1:INVITE request from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP request,
- Request-Line:
  See generic_forward_from-UA12
  See generic_forward_R-URI_responsible-domain

- Header fields:
  - inside of a dialog
    See generic_forward_from-UA12
    See generic_forward_request

- Bodies:
  See generic_forward_from-UA12

*2:200 response from NUT to UA12.
  As a SIP Message,
  See generic_message

  As a SIP response,

  - Status-Line:
    See generic_forward_from-UA11
    Status-Code: Must be "200". [RFC3261-16-104]

  - Header fields:
    See generic_forward_from-UA11
    See generic_forward_response
    * Via
      via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
      via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

  - Bodies:
    See generic_forward_from-UA11

*3:ACK request from NUT to UA11.

  As a SIP Message,
  See generic_message

  As a SIP request,

  - Request-Line:
    See generic_forward_from-UA12
See generic_forward_R-URI_responsible-domain

- Header fields:
  - inside of a dialog
    See generic_forward_from-UA12
    See generic_forward_request

- Bodies:
  See generic_forward_from-UA12

*4:*INVITE request from NUT to UA11.

As a SIP Message,
  See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA12
  See generic_forward_R-URI_responsible-domain

- Header fields:
  - inside of a dialog
    See generic_forward_from-UA12
    See generic_forward_request

- Bodies:
  See generic_forward_from-UA12

*5:*200 response from NUT to UA12.

As a SIP Message,
  See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA11
  Status-Code: Must be "200". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA11
  See generic_forward_response
* Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from-UA11

*6:ACK request from NUT to UA11.

  As a SIP Message,
  See generic_message

  As a SIP request,

  - Request-Line:
    See generic_forward_from-UA12
    See generic_forward_R-URI_responsible-domain

  - Header fields:
    - inside of a dialog
      See generic_forward_from-UA12
      See generic_forward_request

  - Bodies:
    See generic_forward_from-UA12

[REFERENCE]
Sequence from RFC3665 Section 3.7.

4.1.4 PX-1-2-1 - SIP Proxy- Unsuccessful Busy

[NAME]
PX-1-2-1 - SIP Proxy- Unsuccessful Busy

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when the callee is busy.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT (AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11 (AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11 (Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12 (AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12 (Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501::fff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501::fff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501::fff:1::1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501::fff:2::2/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501::fff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
---+-----------+---------
|           |         |
|          UA11  |
R11
|           |
---+---R-------+-----------+---------
|           |         |
|         NUT       Registrar |
R12
|           |
---+-----------+---------
|         |
---+-----------+---------
|           |
|         |
UA12
```

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPADDRESS)</td>
<td>3ffe:501::fff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

```
UA11      R        NUT
<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

1. ICMP Echo Request
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

<table>
<thead>
<tr>
<th>UA11</th>
<th>UA12</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>------</td>
<td></td>
<td>1. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>------</td>
<td></td>
<td>2. 200 OK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>------</td>
<td></td>
<td>4. 200 OK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

<table>
<thead>
<tr>
<th align="right">UA11 :</th>
<th align="right">NUT :</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td align="right"></td>
<td align="right"></td>
<td></td>
</tr>
<tr>
<td align="right">&lt;------:---------</td>
<td align="right">1. INVITE</td>
<td></td>
</tr>
<tr>
<td align="right"></td>
<td align="right"></td>
<td></td>
</tr>
<tr>
<td align="right">&lt;------:---------</td>
<td align="right">2. 407</td>
<td></td>
</tr>
<tr>
<td align="right"></td>
<td align="right"></td>
<td></td>
</tr>
<tr>
<td align="right"></td>
<td align="right"></td>
<td></td>
</tr>
<tr>
<td align="right"></td>
<td align="right"></td>
<td></td>
</tr>
<tr>
<td align="right"></td>
<td align="right"></td>
<td>3. ACK</td>
</tr>
<tr>
<td align="right"></td>
<td align="right"></td>
<td></td>
</tr>
<tr>
<td align="right">&lt;------:---------</td>
<td align="right">4. INVITE</td>
<td></td>
</tr>
<tr>
<td align="right"></td>
<td align="right"></td>
<td></td>
</tr>
<tr>
<td align="right">&lt;------:---------</td>
<td align="right">5. INVITE</td>
<td></td>
</tr>
<tr>
<td align="right"></td>
<td align="right"></td>
<td></td>
</tr>
<tr>
<td align="right">&lt;------:---------</td>
<td align="right">6. 100</td>
<td></td>
</tr>
<tr>
<td align="right"></td>
<td align="right"></td>
<td></td>
</tr>
<tr>
<td align="right">&lt;------:---------</td>
<td align="right">7. 486</td>
<td></td>
</tr>
<tr>
<td align="right"></td>
<td align="right"></td>
<td></td>
</tr>
<tr>
<td align="right">&lt;------:---------</td>
<td align="right">8. ACK (*)1</td>
<td></td>
</tr>
<tr>
<td align="right"></td>
<td align="right"></td>
<td></td>
</tr>
<tr>
<td align="right">&lt;------:---------</td>
<td align="right">9. 486 (*)2</td>
<td></td>
</tr>
<tr>
<td align="right"></td>
<td align="right"></td>
<td></td>
</tr>
<tr>
<td align="right"></td>
<td align="right"></td>
<td></td>
</tr>
<tr>
<td align="right"></td>
<td align="right"></td>
<td>10. ACK</td>
</tr>
</tbody>
</table>
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 486 Busy Here.
8. UA12 Receive ACK. (*1)
9. UA11 Receive 486 Busy Here. (*2)
10. UA11 Send ACK.

--- Message example ---
1. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff1:1
i=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

2. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
;received=3ffe:501:ffff1:1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=3flal12sf
Call-ID: 2xB9vSxSit55XU7p8@under.test.com
CSeq: 1 INVITE
Proxy-Authenticate: Digest realm="under.test.com", qop="auth",
nonce="dc3a5ab2530aa93112cf5904ba7d88fa", opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. ACK UA11 -> NUT

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=3flal12sf
Call-ID: 2xB9vSxSit55XU7p8@under.test.com
CSeq: 1 ACK
Content-Length: 0

4. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xB9vSxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Proxy-Authorization: Digest username="UA11",
realm="under.test.com", nonce="dc3a5ab2530aa93112cf5904ba7d88fa", opaque="", qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA12@under.test.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

5. INVITE NUT -> UA12

INVITE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lro>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Type: application/sdp

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

/*Client for NUT prepares to receive data on port 49172 from the network.*/

6. 100 Trying NUT -> UA11

SIP/2.0 100 Trying
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0
7. 486 Busy Here UA12 -> NUT

SIP/2.0 486 Busy Here
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
  ;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
  ;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

8. ACK NUT -> UA12

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

9. 486 Busy Here NUT -> UA11

SIP/2.0 486 Busy Here
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK721e418c4.1
  ;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

10. ACK UA11 -> NUT

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

[OBSERVABLE RESULTS]

*1: ACK request from NUT to UA12.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_make_ACK_for-non2XX

- Header fields:
  - outside of a dialog
  See generic_make_ACK_for-non2XX

- Bodies:
  See generic_make_ACK_for-non2XX

*2: 486 response from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response
Status-Code: Must be "486". [RFC3261 16.7.6]

- Header fields:
  See generic_make_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter
    contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
    received. [RFC3261-18-28]

[REFERENCE]
Sequence from RFC3665 Section 3.9.
4.1.5 PX-1-2-2 - SIP Proxy- Unsuccessful No Response from User Agent

[NAME]
PX-1-2-2 - SIP Proxy- Unsuccessful processing by No Response from User Agent

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when receiving no response.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:ss.under.test.com</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
              +-----------+---------
             |           |         |
             |          UA11 |
R11          |
---+---R-------+-----------+---------
             |           |         |
             |         NUT       Registrar |
R12          |
             |           |         |
R             +-----------+---------
             |           |         |
             |          R      |
R12          |
             |           |         |
R             +-----------+---------
```
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501::ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>---</td>
<td>-----</td>
</tr>
<tr>
<td>&lt;-----</td>
<td>------</td>
<td>1. ICMP Echo Request</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-----</td>
<td>------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

[PROCEDURE]

<table>
<thead>
<tr>
<th>UA11</th>
<th>UA12</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>---</td>
<td>-----------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>---------</td>
<td>2. 200 OK</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>---------</td>
<td>4. 200 OK</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE. (*1)
6. UA11 Receive 100 Trying. (*2)
7. UA12 Receive INVITE. (*3)
8. UA12 Receive INVITE. (*4)
9. UA12 Receive INVITE. (*5)
10. UA12 Receive INVITE. (*6)
11. UA12 Receive INVITE. (*7)
12. UA12 Receive INVITE. (*8)
13. UA11 Receive 480 No Response. (*9)
14. UA11 Send ACK.

=== Message example ===

1. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

2. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=3flal12sf
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 INVITE
Proxy-Authenticate: Digest realm="under.test.com", qop="auth",
nonce="cf5904ba7d8dc3a5ab2530aa931128fa",
opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. ACK UA11 -> NUT

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=3flal12sf
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 ACK
Content-Length: 0
4. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="cf5904ba7d8dc3a5ab2530aa931128fa", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA12@under.test.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

5. INVITE NUT -> UA12

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

6. 100 Trying NUT -> UA11

SIP/2.0 100 Trying
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

7. INVITE NUT -> UA12

Resend of message 5.

8. INVITE NUT -> UA12

Resend of message 5.

9. INVITE NUT -> UA12

Resend of message 5.

10. INVITE NUT -> UA12

Resend of message 5.

11. INVITE NUT -> UA12

Resend of message 5.
12. INVITE NUT -> UA12
Resend of message 5.

13. 480 No Response NUT -> UA11

SIP/2.0 480 No Response
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fced676l
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

14. ACK UA11 -> NUT

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fced676l
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

[OBSERVABLE RESULTS]
*9:480 response from NUT to UA12
As a SIP Message,
See generic_message
As a SIP response,

· Status-Line:
See generic_make_response
Status-Code: Must be "480". [RFC3261 16.7.6], [RFC3261 21.4.18]

· Header fields:
See generic_make_response
* Via
via-received: Must be added if the host portion of the "sent-by" parameter
contains a domain name. [RFC3261-18-27]
via-received: Must contain the source address from which the packet was
[REFERENCE]
Sequence from RFC3665 Section 3.10.

4.1.6 PX-1-2-3 - SIP Proxy- Unsuccessful Temporarily Unavailable

[NAME]
PX-1-2-3 - SIP Proxy- Unsuccessful Temporarily Unavailable

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly forwards a 480 (Temporarily Unavailable) response.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:ss.under.test.com</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>Address</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501::50:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501::50:60::64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501::50:1::1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501::50:2::2/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501::50:1::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
+-----------+---------
|           |
|          UA11
+---+---R---+---------
|   |   |   |
| R11 |
```

---

IPv6 FORUM TECHNICAL DOCUMENT
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:\ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]
1. UA11 Send INVITE.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 480 Temporarily Unavailable.
10. UA12 Receive ACK. (*1)
11. UA11 Receive 480 Temporarily Unavailable. (*2)
12. UA11 Send ACK.

--- Message example ---

1. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
;received=3ffe:501:ffff:1::1
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com  
CSeq: 1 INVITE  
Contact: <sip:UA11@node.under.test.com>  
Supported: none  
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE  
Content-Type: application/sdp  
Content-Length: 151

v=0  
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1  
s=  
c=IN IP6 3ffe:501:ffff:1::1  
t=0 0  
m=audio 49172 RTP/AVP 0  
a=rtpmap:0 PCMU/8000

2. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required  
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43  
:received=3ffe:501:ffff:1::1  
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl  
To: UA12 <sip:UA12@under.test.com>;tag=3flal12sf  
Call-ID: 2xTb9vxSit55XU7p8@under.test.com  
CSeq: 1 INVITE  
Proxy-Authenticate: Digest realm="under.test.com", qop="auth", nonce="cf5904ba7d8dc3a5ab2530aa931128fa", opaque="", stale=FALSE, algorithm=MD5  
Content-Length: 0

3. ACK UA11 -> NUT

ACK sip:UA12@under.test.com SIP/2.0  
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43  
Max-Forwards: 70  
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl  
To: UA12 <sip:UA12@under.test.com>;tag=3flal12sf  
Call-ID: 2xTb9vxSit55XU7p8@under.test.com  
CSeq: 1 ACK  
Content-Length: 0

4. INVITE UA11 -> NUT
INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="cf5904ba7d8dc3a5ab2530aa931128fa", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA12@under.test.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=
\nc=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

5. INVITE NUT -> UA12

INVITE sip:UA12@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
\:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com；lr>.
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported: none
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

6. 100 Trying NUT -> UA11

SIP/2.0 100 Trying
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

7. 180 Ringing UA12 -> NUT

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Content-Length: 0

8. 180 Ringing NUT -> UA11

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Content-Length: 0

9. 480 Temporarily Unavailable UA12 -> NUT

SIP/2.0 480 Temporarily Unavailable
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

10. ACK NUT -> UA12

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

11. 480 Temporarily Unavailable NUT -> UA11

SIP/2.0 480 Temporarily Unavailable
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

12. ACK UA11 -> NUT
ACK sip:UA12@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

[OBSERVABLE RESULTS]
*1: ACK request from NUT to UA12

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_make_ACK_for-non2XX

- Header fields:
  - outside of a dialog
  See generic_make_ACK_for-non2XX

- Bodies:
  See generic_make_ACK_for-non2XX

*2: 480 response from NUT to UA11

As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response
  Status-Code: Must be "480". [RFC3261 16.7.6], [RFC3261 21.4.18]

- Header fields:
  See generic_make_response
  * Via
  via-received: Must be added if the host portion of the "sent-by" parameter
contains a domain name. [RFC3261-18-27]
via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
Sequence from RFC3665 Section 3.11.

4.2 Session Establishment on Two Proxies

4.2.1 PX-2-1-1 - SIP Proxy- Session Establishment Through Two Proxies
- Callee hanging up [another domain] (Caller)

[NAME]
PX-2-1-1 - SIP Proxy- Session Establishment Through Two Proxies - Callee hanging up [another domain] (Caller)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly forwards to another domain when a session is established through two proxies.

[REQUIREMENT]
Only when a proxy can forward requests/responses on NNI connection.
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA21(Contact)</td>
<td>sip:<a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>PX2</td>
<td>sip:ss2.biloxi.example.com</td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501::ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501::ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501::ffff:1::1/64</td>
</tr>
<tr>
<td>UA21 (IPv6)</td>
<td>3ffe:501::ffff:2::2/64</td>
</tr>
<tr>
<td>PX2 (IPv6)</td>
<td>3ffe:501::ffff:20::20/64</td>
</tr>
</tbody>
</table>
**Phase 2 Test Specification**

**SIP IPv6**

---

**[TOPOLOGY]**

```
| R(I Pv6) | 3ffe:501:ffff:50::1/64 |
```

---

**[CONFIGURATION for NUT]**

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

---

**[INITIALIZATION]**

- 1. Send ICMP Echo Request.
- 2. Receive ICMP Echo Reply.

---

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

---

**IPv6 FORUM TECHNICAL DOCUMENT**

IPv6 Ready Logo Program

Phase 2 Test Specification

SIP IPv6
1. Send REGISTER Request.
2. Receive 200 OK response.

[PROCEDURE]

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>PX2</th>
<th>UA21</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;-------</td>
<td>------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required. (*1)
3. UA11 Send ACK.
4. UA11 Send INVITE. (*2)
5. PX2 Receive INVITE. (*2)
6. UA11 Receive 100 Trying. (*3)
7. PX2 Send 100 Trying. (*4)
8. PX2 Send 180 Ringing.
9. UA11 Receive 180 Ringing. (*5)
10. PX2 Send 200 OK.
11. UA11 Receive 200 OK. (*6)
12. UA11 Send ACK.
13. PX2 Receive ACK. (*7)
14. PX2 Send BYE.
15. UA11 Receive BYE. (*8)
16. UA11 Send 200 OK.
17. PX2 Receive 200 OK. (*9)

=== Message example ===

1. INVITE UA11 -> NUT

INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 384276298220188511@under.test.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported:
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=
 c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

/* Proxy(NUT) challenges UA11 for authentication */
2. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
:received=3ffe:501::fff1:1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=3flal12sf
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 INVITE
Proxy-Authenticate: Digest realm="under.test.com", qop="auth",
nonce="884f1ce41e6cbe5ae9c8e88d359", opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. ACK UA11 -> NUT

ACK sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=3flal12sf
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 ACK
Content-Length: 0

/* UA11 responds re-sending the INVITE with authentication credentials in it. */

4. INVITE UA11 -> NUT

INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="884f1ce41e6cbe5ae9c8e88d359", opaque="", qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA21@biloxi.example.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported:
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

/* Proxy(NUT) accepts the credentials and forwards the INVITE to Proxy */
2. Client for UA11 prepares to receive data on port 49172 from the network. */

5. INVITE NUT -> Proxy 2

INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Supported:
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
IPv6 Forum Technical Document
Phase 2 Test Specification
SIP IPv6

a=rtpmap:0 PCMU/8000

6. 100 Trying NUT -> UA11

SIP/2.0 100 Trying
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Length: 0

7. 100 Trying Proxy 2 -> NUT

SIP/2.0 100 Trying
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Length: 0

8. 180 Ringing Proxy 2 -> NUT

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA21@client.biloxi.example.com>
CSeq: 2 INVITE
Content-Length: 0

9. 180 Ringing NUT -> UA11
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA21@client.biloxi.example.com>
CSeq: 2 INVITE
Content-Length: 0

10. 200 OK Proxy 2 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Supported:
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA21 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

11. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>,

IPv6 FORUM TECHNICAL DOCUMENT IPv6 Ready Logo Program Phase 2 Test Specification SIP IPv6
<sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Supported:
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA21 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

12. ACK UA11 -> NUT

ACK sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b76
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1ccc416cbe5a9c8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA21@biloxi.example.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
Route: <sip:ss.under.test.com;lr>,
<sip:ss2.biloxi.example.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 ACK
Content-Length: 0

13. ACK NUT -> Proxy 2

ACK sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b76
:received=3ffe:501::1
Max-Forwards: 69
Route: <sip:ss2.biloxi.example.com:lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 ACK
Content-Length: 0

14. BYE Proxy 2 -> NUT

BYE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501::2
Max-Forwards: 69
Route: <sip:ss.under.test.com:lr>
Record-Route: <sip:ss2.biloxi.example.com:lr>,
From: UA11 <sip:UA21@biloxi.example.com>;tag=314159
To: NUT <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

15. BYE NUT -> UA11

BYE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501::2
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501::2
Max-Forwards: 68
Record-Route: <sip:ss.under.test.com:lr>, <sip:ss2.biloxi.example.com:lr>
From: UA11 <sip:UA21@biloxi.example.com>;tag=314159
To: NUT <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

16. 200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
  ;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
  ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
  ;received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

17. 200 OK NUT -> Proxy 2

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
  ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
  ;received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

[OBSERVABLE RESULTS]
*1:407 response from NUT to UA11.
  As a SIP Message,
  See generic_message

  As a SIP response,
  
  · Status-Line:
  See generic_make_response
  Status-Code: Must be "407". [RFC3261 22.3]

  · Header fields:
  See generic_make_response
  * Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

*2:INVITE request from NUT to PX2.

As a SIP Message,
See generic_message

As a SIP request,

· Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain

· Header fields:
  · outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

· Bodies:
  See generic_forward_from-UA11

*3:After 100 response from PX2 to NUT
  Must not forward this message. [RFC3261-16-109]

*3:100 response from NUT to UA11.(Optional)
  As a SIP Message,
  See generic_message

As a SIP response,

· Status-Line:
  See generic_make_response
  Status-Code: Must be "100". [RFC3261 4]

· Header fields:
  See generic_make_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]
*4:180 response from NUT to UA11.
As a SIP Message,
See generic_message

As a SIP response,
· Status-Line:
  See generic_forward_from-PX2
  Status-Code: Must be "180". [RFC3261-16-104]
· Header fields:
  See generic_forward_from-PX2
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]
· Bodies:
  See generic_forward_from-PX2

*5:200 response from NUT to UA11.
As a SIP Message,
See generic_message

As a SIP response,
· Status-Line:
  See generic_forward_from-PX2
  Status-Code: Must be "200". [RFC3261-16-104]
· Header fields:
  See generic_forward_from-PX2
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]
- Bodies:
  See generic_forward_from-PX2

*6:ACK request from NUT to PX2.

As a SIP Message,
  See generic_message

As a SIP request,

  - Request-Line:
    See generic_forward_from-UA11
    See generic_forward_R-URI_non-responsible-domain

  - Header fields:
    - outside of a dialog
      See generic_forward_from-UA11
      See generic_forward_request

  - Bodies:
    See generic_forward_from-UA11

*7:BYE request from NUT to UA11.

As a SIP Message,
  See generic_message

As a SIP request,

  - Request-Line:
    See generic_forward_from-PX2

  - Header fields:
    - outside of a dialog
      See generic_forward_from-PX2
      See generic_forward_request

  - Bodies:
    See generic_forward_from-PX2

*8:200 response from NUT to PX2.

As a SIP Message,
As a SIP response,

- Status-Line:
  See generic_forward_from-UA11
  Status-Code: Must be "200". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA11
  See generic_forward_response

- Bodies:
  See generic_forward_from-PX2

[REFERENCE]
  Sequence from RFC3665 Section 3.2.

[RFC3261-16-108, 109]

16.7 Response Processing

5. Check response for forwarding

A stateful proxy MUST NOT immediately forward any other responses. In particular, a stateful proxy MUST NOT forward any 100 (Trying) response. Those responses that are candidates for forwarding later as the "best" response have been gathered as described in step "Add Response to Context".

4.2.2 PX-2-1-2 - SIP Proxy- Session Establishment Through Two Proxies - Callee hanging up [another domain] (Callee)

[NAME]
PX-2-1-2 · SIP Proxy- Session Establishment Through Two Proxies · Callee hanging up [another domain] (Callee)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes that upstream that is forwarded from another domain when a session is established through two proxies.
[REQUIREMENT]
Only when a proxy supports the architecture with two proxies
Set up registrar server to use location service, if necessary.

[PARAMETER]
<table>
<thead>
<tr>
<th></th>
<th>sip:ss.under.test.com</th>
<th>sip:reg.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
<td></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
<td></td>
</tr>
<tr>
<td>UA21(Contact)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PX2</td>
<td>sip:ss2.biloxi.example.com</td>
<td></td>
</tr>
</tbody>
</table>

[ADDRESS]
<table>
<thead>
<tr>
<th></th>
<th>3ffe:501:ffff:50::50/64</th>
<th>3ffe:501:ffff:50::60/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td></td>
<td>Registrar (IPv6)</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>UA21(IPv6)</td>
<td>3ffe:501:ffff:20::20/64</td>
<td>R(IPv6)</td>
</tr>
<tr>
<td>PX2(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

---+-----------+---------
|           |          |
|          UA11 |
| R11    |
---+---R-------+-----------+---------
|           |          |
|          NUT       Registrar |
| R12    |
---+-----------+---------
|           |          |
|          PX2 |
| R13    |
---+-----------+---------
|           |          |
|          UA21 |
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip: ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>-----&gt; 1. ICMP Echo Request</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>------&gt; 2. ICMP Echo Reply</td>
<td></td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>------&gt;</td>
<td>1. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>------&gt; 2. 200 OK</td>
<td></td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.

[PROCEDURE]

<table>
<thead>
<tr>
<th>UA21</th>
<th>PX2</th>
<th>NUT</th>
<th>UA11</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;--------</td>
<td>-----</td>
<td>-----</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&lt;------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1. PX2 Send INVITE.
2. UA11 Receive INVITE. (*)
3. PX2 Receive 100 Trying. (*)
4. UA11 Send 180 Ringing.
5. PX2 Receive 180 Ringing. (*)
6. UA11 Send 200 OK.
7. PX2 Receive 200 OK. (*)
8. PX2 Send ACK.
9. UA11 Receive ACK. (*)
10. UA11 Send BYE.
11. PX2 Receive BYE. (*)
12. PX2 Send 200 OK.
13. UA11 Receive 200 OK. (*)

Message example

INVITE Proxy 2 -> NUT

INVITE sip:UA11@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=ffe:501:ff:2::2
Max-Forwards: 69
Record-Route: <sip:ss2.biloxi.example.com:lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 2 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

2.INVITE NUT -> UA11

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
 Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:20::20
Max-Forwards: 68
Record-Route: <sip:ss.under.test.com;lr>,
 <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 2 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
3.100 Trying NUT -> Proxy 2

SIP/2.0 100 Trying
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
    ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
    ;received=3ffe:501:ffff:2::2
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 38482762958280188511@biloxi.example.com
CSeq: 2 INVITE
Content-Length: 0

4.180 Ringing UA11 -> NUT

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e48c4.1
    ;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
    ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
    ;received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>,
    <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
Contact: <sip:UA11@node.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

5.180 Ringing NUT -> Proxy 2

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
    ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
    ;received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>,
    <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
Contact: <sip:UA11@node.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

6.200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>,
<sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA11 289084527 289084527 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

7.200 OK NUT -> Proxy 2

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>,
<sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA11 2890844527 2890844527 IN IP6 3ffe:501:ffff:1::1
s=
:c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

8.ACK Proxy 2 -> NUT

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74b76
:received=3ffe:501:ffff:2::2
Max-Forwards: 69
Route: <sip:ss.under.test.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@node.under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 2 ACK
Content-Length: 0

9.ACK NUT -> UA11

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74b76
:received=3ffe:501:ffff:2::2
Max-Forwards: 68
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@node.under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 2 ACK
Content-Length: 0
/* RTP streams are established between UA11 and UA21 */

/* UA21 Hangs Up with UA11. */

/* Again, note that the CSeq is NOT 3. UA11 and UA21 maintain their own separate CSeq counts */

10.BYE UA11 -> NUT

BYE sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
Route: <sip:ss.under.test.com;lr>,
     <sip:ss2.biloxi.example.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 1 BYE
Content-Length: 0

11.BYE NUT -> Proxy 2

BYE sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=9hG4bK721e418c4.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 :received=3ffe:501:ffff:1::1
Max-Forwards: 69
Route: <sip:ss2.biloxi.example.com;lr>
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 1 BYE
Content-Length: 0

12.200 OK Proxy 2 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=9hG4bK721e418c4.1
 :received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 :received=3ffe:501:ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 1 BYE
Content-Length: 0

13.200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501::1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 1 BYE
Content-Length: 0

[OBSERVABLE RESULTS]
*1:INVITE request from NUT to UA11.

As a SIP Message,
   See generic_message

As a SIP request,

   · Request-Line:
     See generic_forward_from-PX2
     See generic_forward_R-URI_non-responsible-domain

   · Header fields:
     · outside of a dialog
     See generic_forward_from-PX2
     See generic_forward_request

   · Bodies:
     See generic_forward_from-PX2

*2:100 response from NUT to PX2.(Optional)

As a SIP Message,
   See generic_message

As a SIP response,
- Status-Line:
  See generic_make_response
  Status-Code: Must be "100". [RFC3261 4]

- Header fields:
  See generic_make_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter
    contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
    received. [RFC3261-18-28]

*3:180 response from NUT to PX2.
  As a SIP Message,
  See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA11
  Status-Code: Must be "180". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA11
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter
    contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
    received. [RFC3261-18-28]

*4:200 response from NUT to PX1.
  As a SIP Message,
  See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA11
  Status-Code: Must be "200". [RFC3261-16-104]
- Header fields:
  See generic_forward_from-UA11
  See generic_forward_response
* Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from-UA11

*5:ACK request from NUT to UA11.

  As a SIP Message,
  See generic_message

  As a SIP request,

  - Request-Line:
    See generic_forward_from-PX2
    See generic_forward_R-URI_non-responsible-domain

  - Header fields:
    - outside of a dialog
      See generic_forward_from-PX2
      See generic_forward_request

  - Bodies:
    See generic_forward_from-PX2

*6:BYE request from NUT to PX2.

  As a SIP Message,
  See generic_message

  As a SIP request,

  - Request-Line:
    See generic_forward_from-UA11
- Header fields:
  - outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

- Bodies:
  See generic_forward_from-UA11

*7:200 response from NUT to UA11.
   As a SIP Message,
   See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-PX2
  Status-Code: Must be "200". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-PX2
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
Sequence from RFC3665 Section 3.2.

4.2.3 PX-2-1-3 - SIP Proxy- Unsuccessful No Answer [CANCEL] (Caller)

[NAME]
PX-2-1-3 - SIP Proxy- Unsuccessful No Answer [CANCEL] (Caller)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when the UA1 receives no response and sends a CANCEL request.

[REQUIREMENT]
Only when a proxy supports the architecture with two proxies
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:ss.under.test.com</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA21(Contact)</td>
<td>sip:<a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>PX2</td>
<td>sip:ss2.biloxi.example.com</td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>Address</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA21(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>PX2(IPv6)</td>
<td>3ffe:501:ffff:20::20/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
+-----------+--------+
|           |        |
|          UA11 |
| R11       |
+-----------+--------+
|        |        |
|        R    |
|        |        |
|        | NUT    Registrar |
| R12       |
+-----------+--------+
|        |        |
|        |        |
|        |         PX2 |
| R13       |
+-----------+--------+
|        |        |
|        |        |
|        |            |
|        | UA21      |
```

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
</table>
[INITIALIZATION]

**UA11** R NUT

| | | |
| | | |
|<----------->| 1. ICMP Echo Request
| | | |
| | | |

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

**UA11** R Registrar

| | | |
| | | |
|<----------->| 1. REGISTER
| | | |
| | | |

1. Send REGISTER Request.
2. Receive 200 OK response.

[PROCEDURE]

**UA11** : NUT : PX2 : UA21

| : | | : |
| : | | : |

|<----------->| 1. INVITE
| | | |
| | | |

| | | |
| | | |

|<----------->| 2. 407
| | | |
| | | |

| | | |
| | | |

| | | |
| | | |

|<----------->| 3. ACK
| | | |
| | | |

| | | |
| | | |

|<----------->| 4. INVITE
| | | |
| | | |

|<----------->| 5. INVITE
| | | |
| | | |

|<----------->| 6. 100
| | | |
| | | |

|<----------->| INVITE
| | | |
| | | |

|<----------->| 7. 100
| | | |
| | | |

|<----------->| 8. 180
| | | |
| | | |

|<----------->| 9. 180
| | | |
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. PX2 Receive INVITE.
6. UA11 Receive 100 Trying.
7. PX2 Send 100 Trying.
8. PX2 Send 180 Ringing.
9. UA11 Receive 180 Ringing.
10. UA11 Send CANCEL.
11. UA11 Receive 200 OK. (*1)
12. PX2 Receive CANCEL. (*2)
13. PX2 Send 200 OK.
14. PX2 Send 487 Request Terminated.
15. PX2 Receive ACK. (*3)
16. UA11 Receive 487 Request Terminated. (*4)
17. UA11 Send ACK.

=== Message example ===

1. INVITE UA11 -> NUT
INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

2. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=3flal12sf
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 INVITE
Proxy-Authenticate: Digest realm="under.test.com", qop="auth",
nonce="f84f1ceedc41e6be5aa9c8e88d359",
opaque=""", stale=FALSE, algorithm=MD5
Content-Length: 0

3. ACK UA11 -> NUT

ACK sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=3flal12sf
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 ACK
Content-Length: 0
4. INVITE UA11 -> NUT

INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1ece41e6cbe5a9c8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA21@biloxi.example.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

5. INVITE NUT -> PX2

INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151
v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501::ffff:1:1
s=
\r
\r
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

6. 100 Trying NUT -> UA11

SIP/2.0 100 Trying
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1:1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

7. 100 Trying PX2 -> NUT

SIP/2.0 100 Trying
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1:1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

8. 180 Ringing PX2 -> NUT

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:50:50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1:1
Record-Route: <sip:ss2.biloxi.example.com;lr>,
<sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
9. 180 Ringing NUT -> UA11

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501::ff1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>,
<sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@client.biloxi.example.com>
Content-Length: 0

10. CANCEL UA11 -> NUT

CANCEL sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Route: <sip:ss.under.test.com;lr>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 CANCEL
Content-Length: 0

11. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501::ff1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 CANCEL
Content-Length: 0

12. CANCEL NUT -> PX2
CANCEL sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxcd76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSIt55XU7p8@under.test.com
CSeq: 2 CANCEL
Content-Length: 0

13. 200 OK PX2 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501::50
From: UA11 <sip:UA11@under.test.com>;tag=9fxcd76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSIt55XU7p8@under.test.com
CSeq: 2 CANCEL
Content-Length: 0

14. 487 Request Terminated PX2 -> NUT

SIP/2.0 487 Request Terminated
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxcd76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSIt55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

15. ACK NUT -> PX2

ACK sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxcd76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSIt55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0
16. 487 Request Terminated NUT -> UA11

SIP/2.0 487 Request Terminated
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

17. ACK UA11 -> NUT

ACK sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

[OBSERVABLE RESULTS]
*1:200 response from NUT to UA11.
   As a SIP Message,
      See generic_message

   As a SIP response,
      · Status-Line:
         See generic_make_response-200_for-CANCEL
         Status-Code: Must be “200”. [RFC3261 16.10]
      · Header fields:
         See generic_make_response-200_for-CANCEL
         * Via
            via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
            via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]
*2: CANCEL request from NUT to PX2.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_makeCANCEL

- Header fields:
  - outside of a dialog
  See generic_makeCANCEL

- Bodies:
  See generic_makeCANCEL

*3: ACK request from NUT to PX2.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_make_ACK_for-non2XX

- Header fields:
  - outside of a dialog
  See generic_make_ACK_for-non2XX

- Bodies:
  See generic_make_ACK_for-non2XX

*4: 487 response from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
See generic_make_response
Status-Code: Must be "487". [RFC3261-9-15]

* Header fields:
  See generic_make_response

* To
  tag-param: Should be the same value of “11. 200 response”. [RFC3261-9-16]

* Via
  via-received: Must be added if the host portion of the "sent-by" parameter
  contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was
  received. [RFC3261-18-28]

[REFERENCE]
Sequence from RFC3665 Section 3.8.

4.2.4 PX-2-1-4 - SIP Proxy- Unsuccessful No Answer [CANCEL] (Callee)

[NAME]
PX-2-1-4 - SIP Proxy- Unsuccessful No Answer [CANCEL] (Callee)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when the UA1 receives no response and sends a
CANCEL request.

[REQUIREMENT]
Only when a proxy supports the architecture with two proxies
Set up registrar server to use location service, if necessary.

[PARAMETER]
<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA21(Contact)</td>
<td>sip:<a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>PX2</td>
<td>sip:ss2.biloxi.example.com</td>
</tr>
</tbody>
</table>
**[ADDRESS]**

<table>
<thead>
<tr>
<th></th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA21(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>PX2(IPv6)</td>
<td>3ffe:501:ffff:20::20/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

**[TOPOLOGY]**

```
---+-----------+---------
 |           |
 |          UA11 |
 R11 |
---+---R-------+-----------+---------
 |           |           |
 |         NUT       Registrar |
 R12 |
---+-----------+---------
 |           |
 |          PX2 |
 R13 |
---+-----------+---------
 |           |
 |           |
 |     UA21 |
```

**[CONFIGURATION for NUT]**

<table>
<thead>
<tr>
<th></th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT</td>
<td>sip: ss.under.test.com</td>
</tr>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

**[INITIALIZATION]**

```
UA11      R      NUT
|        |         |   |
|        |         |   |
|--------|--------|> 1. ICMP Echo Request
|        |         |   |
|<-------|--------| 2. ICMP Echo Reply
|        |         |   |
```

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.

**[PROCEDURE]**

```
<table>
<thead>
<tr>
<th>UA21</th>
<th>PX2</th>
<th>NUT</th>
<th>UA11</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```
1. PX2 Send INVITE.
2. UA11 Receive INVITE.
3. PX2 Receive 100 Trying.
4. UA11 Send 180 Ringing.
5. PX2 Receive 180 Ringing.
6. PX2 Send CANCEL.
7. PX2 Receive 200 OK. (*1)
8. UA11 Receive CANCEL. (*2)
9. UA11 Send 200 OK.
10. UA11 Send 487 Request Terminated.
11. UA11 Receive ACK. (*3)
12. PX2 Receive 487 Request Terminated. (*4)
13. PX2 Send ACK.

=== Message example ===

1. INVITE PX2 -> NUT

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501:ffff:2::2
Record-Route: <sip:ss2.biloxi.example.com;lr>
Max-Forwards: 69
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

2. INVITE NUT -> UA11

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff2:2::2
Record-Route: <sip:ss.under.test.com;lr>,
<sip:ss2.biloxi.example.com:lr>
Max-Forwards: 68
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

3. 100 Trying NUT -> PX 2

SIP/2.0 100 Trying
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff2:2::2
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Content-Length: 0

4. 180 Ringing UA11 -> NUT

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss2.node.under.test.com:5060;branch=z9hG4bK721e418c4.1
  ;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
  ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
  ;received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>,
  <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Content-Length: 0

5. 180 Ringing NUT -> PX2

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
  ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
  ;received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>,
  <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Content-Length: 0

6. CANCEL PX2 -> NUT

CANCEL sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Max-Forwards: 70
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 CANCEL
Content-Length: 0

7. 200 OK NUT -> PX2

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 CANCEL
Content-Length: 0

8. CANCEL NUT -> UA11

CANCEL sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Max-Forwards: 70
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 CANCEL
Content-Length: 0

9. 200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:ffff:50::50
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 CANCEL
Content-Length: 0

10. 487 Request Terminated UA11 -> NUT

SIP/2.0 487 Request Terminated
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:2::2
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Content-Length: 0

11. ACK NUT -> UA11

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Max-Forwards: 70
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 ACK
Content-Length: 0

12. 487 Request Terminated NUT -> PX2

SIP/2.0 487 Request Terminated
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:2::2
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Content-Length: 0

13. ACK PX2 -> NUT

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Max-Forwards: 70
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 ACK
Content-Length: 0

[OBSERVABLE RESULTS]
*1:200 response from NUT to PX2.
As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response-200_for-CANCEL
  Status-Code: Must be "200". [RFC3261 16.10]

- Header fields:
  See generic_make_response-200_for-CANCEL
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter
    contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
    received. [RFC3261-18-28]

*2:CANCEL request from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_make_CANCEL

- Header fields:
  - outside of a dialog
    See generic_make_CANCEL

- Bodies:
  See generic_make_CANCEL

*3:ACK request from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_make_ACK_for-non2XX

- Header fields:
  - outside of a dialog
  See generic_make_ACK_for-non2XX

- Bodies:
  See generic_make_ACK_for-non2XX

*4:487 response from NUT to PX2.
As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response
  Status-Code: Must be "487". [RFC3261-9-15]

- Header fields:
  See generic_make_response

  * To
tag-param: Should be the same value of “7. 200 response”. [RFC3261-9-16]

  * Via
via-received: Must be added if the host portion of the "sent-by" parameter
contains a domain name. [RFC3261-18-27]
via-received: Must contain the source address from which the packet was
received. [RFC3261-18-28]

[REFERENCE]
Sequence from RFC3665 Section 3.8.

4.2.5 PX-2-1-5 - SIP Proxy- Session establishment and call hold by re-INVITE (Caller)
[NAME]
PX-2-1-5 - SIP Proxy- Session establishment and call hold by re-INVITE (Caller)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when the UA2 sends a re-INVITE for holding.

[REQUIREMENT]
Only when a proxy supports the architecture with two proxies
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA21(Contact)</td>
<td>sip:<a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>PX2</td>
<td>sip:ss2.biloxi.example.com</td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA21(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>PX2(IPv6)</td>
<td>3ffe:501:ffff:20::20/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
---+-----------+---------
|           |          |
|          UA11 |
R11--------

---+-----------+---------
|           |          |
|         NUT        Registrar |
R12---
```

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip: ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

[PROCEDURE]

1. Send REGISTER Request.
2. Receive 200 OK response.
Phase 2 Test Specification

SIP IPv6

|      :      |------:------>|            | 5. INVITE
|<-----:------|      :       |            | 6. 100 Trying
|      :      |<-----:------|            | 7. 100 Trying
|      :      |      :       |----------->|   INVITE
|      :      |      :       |<-----------|   180 Ringing
|<-----:------|      :       |            | 9. 180 Ringing
|      :      |      :       |<-----------|   200 OK
|<-----:------|      :       |            | 10. 200 OK
|      :      |<-----:-------|            | 11. 200 OK
|<-----:------|      :       |            | 12. ACK
|      :      |      :       |            | 13. ACK
|      :      |      :       |            |   HOLD
|      :      |<-----:-------|            |14. INVITE
|<-----:------|      :       |            |15. INVITE (*1)
|------:----->|      :       |            |16. 200 OK
|      :      |------:------>|            |17. 200 OK (*2)
|      :      |      :       |----------->|   200 OK
|      :      |      :       |<-----------|   ACK
|      :      |<-----:-------|            |18. ACK
|<-----:------|      :       |            |19. ACK (*3)
|      :      |      :       |            |   HOLD Release
|      :      |<-----:-------|            |20. INVITE
|<-----:------|      :       |            |21. INVITE (*4)
|------:----->|      :       |            |22. 200 OK
|      :      |------:------>|            |23. 200 OK (*5)
|      :      |      :       |----------->|   200 OK
|      :      |      :       |<-----------|   ACK
|      :      |<-----:-------|            |24. ACK
|<-----:------|      :       |            |25. ACK (*6)
|<=======================================>|   Both Way RTP Media Established

|<=======================================>|   New RTP Media Stream
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. PX2 Receive INVITE.
6. UA11 Receive 100 Trying.
7. PX2 Send 100 Trying.
8. PX2 Send 180 Ringing.
9. UA11 Receive 180 Ringing.
10. PX2 Send 200 OK.
11. UA11 Receive 200 OK.
12. UA11 Send ACK.
13. PX2 Receive ACK.
14. PX2 Send INVITE.
15. UA11 Receive INVITE. (*1)
16. UA11 Send 200 OK.
17. PX2 Receive 200 OK. (*2)
18. PX2 Send ACK.
19. UA11 Receive ACK. (*3)
20. PX2 Send INVITE.
21. UA11 Receive INVITE. (*4)
22. UA11 Send 200 OK.
23. PX2 Receive 200 OK. (*5)
24. PX2 Send ACK.
25. UA11 Receive ACK. (*6)
26. UA11 Send BYE.
27. PX2 Receive BYE.
28. PX2 Send 200 OK.
29. UA11 Receive 200 OK.

=== Message example ===

1. INVITE UA11 -> NUT

INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxsit55XU7p8@under.test.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501::1
s=-
c=IN IP6 3ffe:501::1
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

2. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
:received=3ffe:501::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=3flal12sf
Call-ID: 2xTb9vxsit55XU7p8@under.test.com
CSeq: 1 INVITE
Proxy-Authenticate: Digest realm="under.test.com", qop="auth",
nonce="f84f1cec41e6cbe5aea9c8e88d359",
opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. ACK UA11 -> NUT

ACK sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=3flal12sf
Call-ID: 2xTb9vxsit55XU7p8@under.test.com
CSeq: 1 ACK
Content-Length: 0
4. INVITE UA11 -> NUT

INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1ce41e6cbe9aea9c8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA21@biloxi.example.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

5. INVITE NUT -> PX2

INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151
v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501::ff1::1
s=
c=IN IP6 3ffe:501::ff1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

6. 100 Trying NUT -> UA11

SIP/2.0 100 Trying
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501::ff1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

7. 100 Trying PX2 -> NUT

SIP/2.0 100 Trying
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501::50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501::ff1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

8. 180 Ringing PX2 -> NUT

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501::ff1::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501::ff1::1
Record-Route: <sip:ss2.biloxi.example.com;l r>,
<sip:ss.under.test.com;l r>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Length: 0

9. 180 Ringing NUT -> UA11

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
   ;received=3ffe:501::ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Length: 0

10. 200 OK PX2 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
   ;received=3ffe:501::ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
   ;received=3ffe:501::ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA21 2890844527 2890844527 IN IP6 3ffe:501::ffff:2::2
s=-
c=IN IP6 3ffe:501::ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
11. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>,
<sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 147

v=0
ox=UA21 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

12. ACK UA11 -> NUT

ACK sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1ce41e6cb6e9a9c8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA21@biloxi.example.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
Route: <sip:ss.under.test.com;lr>,
<sip:ss2.biloxi.example.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0
13. ACK NUT -> PX2

ACK sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b7b
:received=3ffe:501::ff:1:1
Max-Forwards: 70
Route: <sip:ss2.biloxi.example.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

14. INVITE PX2 -> NUT

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501::ff:2:2
Max-Forwards: 69
Route: <sip:ss.under.test.com;lr>
Record-Route: <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 15 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 149

v=0
o=UA21 2890844527 2890844528 IN IP6 3ffe:501::ff:2:2
s=
C=IN IP6 3ffe:501::ff:2:2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=sendonly

15. INVITE NUT -> UA11

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 68
Record-route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 15 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 149

v=0
o=UA21 2890844527 2890844528 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=sendonly

16. 200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
Record-route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 15 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 150

v=0
17. 200 OK NUT -> PX2

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:ffff:200::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:2::2
Record-route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 15 INVITE
Content-Type: application/sdp
Content-Length: 150

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=recvonly

18. ACK PX2-> NUT

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:2::2
Max-Forwards: 69
Route: <sip:ss.under.test.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
19. ACK NUT -> UA1

ACK sip:UA11@node.under.test.com SIP/2.0  
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1  
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1  
:received=3ffe:501:ffff:20::20  
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7  
:received=3ffe:501:ffff:2::2  
Max-Forwards: 68  
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159  
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl  
Call-ID: 2xTb9vxSit55XU7p8@under.test.com  
CSeq: 15 ACK  
Content-Length: 0

20. INVITE PX2 -> NUT

INVITE sip:UA11@node.under.test.com SIP/2.0  
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1  
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7  
:received=3ffe:501:ffff:2::2  
Max-Forwards: 69  
Route: <sip:ss.under.test.com;lr>  
Record-route: <sip:ss2.biloxi.example.com;lr>  
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159  
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl  
Call-ID: 2xTb9vxSit55XU7p8@under.test.com  
CSeq: 16 INVITE  
Contact: <sip:UA21@client.biloxi.example.com>  
Content-Type: application/sdp  
Content-Length: 149

v=0  
o=UA21 2890844527 2890844529 IN IP6 3ffe:501:ffff:2::2  
s=  
c=IN IP6 3ffe:501:ffff:2::2  
t=0 0  
m=audio 3456 RTP/AVP 0  
a=rtpmap:0 PCMU/8000
21. INVITE NUT -> UA11

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
;received=3ffe:501::50
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
;received=3ffe:501::50
Max-Forwards: 68
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76si
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 16 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 149

v=0
o=UA21 2890844527 2890844529 IN IP6 3ffe:501::50
s=-
c=IN IP6 3ffe:501::50
 t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

22. 200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
;received=3ffe:501::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
;received=3ffe:501::50
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
;received=3ffe:501::50
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76si
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 16 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 150

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

23. 200 OK NUT -> PX2

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
 :received=3ffe:501:ffff:20:20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
 :received=3ffe:501:ffff2:2
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9xed76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 16 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 150

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

24. ACK PX2 -> NUT

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
 :received=3ffe:501:ffff2:2
Max-Forwards: 69
Route: <sip:ss.under.test.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 16 ACK
Content-Length: 0

25. ACK NUT -> UA11

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:2::2
Max-Forwards: 68
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 16 ACK
Content-Length: 0

26. BYE UA11 -> NUT

BYE sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bo4
Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 3 BYE
Content-Length: 0

27. BYE NUT -> PX2

BYE sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bo4
:received=3ffe:501:ffff:1::1
Route: <sip:ss2.biloxi.example.com;lr>
Record-Route: <sip:ss.under.test.com;lr>
Max-Forwards: 69
**[OBSERVABLE RESULTS]**

*1: INVITE request from NUT to UA11.

As a SIP Message,

See generic_message

As a SIP request,

- Request-Line:
  
  See generic_forward_from-PX2

  See generic_forward_R-URI_responsible-domain
- Header fields:
  - outside of a dialog
    See generic_forward_from-PX2
    See generic_forward_request

- Bodies:
  See generic_forward_from-PX2

*2:200 response from NUT to PX2.
  As a SIP Message,
    See generic_message

  As a SIP response,

  - Status-Line:
    See generic_forward_from-UA11
    Status-Code: Must be "200". [RFC3261-16-104]

  - Header fields:
    See generic_forward_from-UA11
    See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

  - Bodies:
    See generic_forward_from-UA11

*3:ACK request from NUT to UA11.

  As a SIP Message,
    See generic_message

  As a SIP request,

  - Request-Line:
    See generic_forward_from-PX2
    See generic_forward_R-URI_responsible-domain

  - Header fields:
· outside of a dialog
  See generic_forward_from-PX2
  See generic_forward_request

· Bodies:
  See generic_forward_from-PX2

*4: INVITE request from NUT to UA11.

  As a SIP Message,
  See generic_message

  As a SIP request,

  · Request-Line:
    See generic_forward_from-PX2
    See generic_forward_R-URI_responsible-domain

  · Header fields:
    · outside of a dialog
    See generic_forward_from-PX2
    See generic_forward_request

  · Bodies:
    See generic_forward_from-PX2

*5: 200 response from NUT to PX2.

  As a SIP Message,
  See generic_message

  As a SIP response,

  · Status-Line:
    See generic_forward_from-UA11
    Status-Code: Must be "200". [RFC3261-16-104]

  · Header fields:
    See generic_forward_from-UA11
    See generic_forward_response
    * Via
      via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from-UA11

*6:ACK request from NUT to UA11.

As a SIP Message,
  See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-PX2
  See generic_forward_R-URI_responsible-domain

- Header fields:
  - outside of a dialog
    See generic_forward_from-PX2
    See generic_forward_request

- Bodies:
  See generic_forward_from-PX2

[REFERENCE]
Sequence from RFC3665 Section 3.7.

4.2.6 PX-2-1-6 - SIP Proxy- Session establishment and call hold by re-INVITE (Callee)

[NAME]
PX-2-1-6 - SIP Proxy- Session establishment and call hold by re-INVITE (Callee)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when the UA2 sends a re-INVITE for holding.

[REQUIREMENT]
Only when a proxy supports the architecture with two proxies
Set up registrar server to use location service, if necessary.
### [PARAMETER]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:ss.under.test.com</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA21(Contact)</td>
<td>sip:<a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>PX2</td>
<td>sip:ss2.biloxi.example.com</td>
</tr>
</tbody>
</table>

### [ADDRESS]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501::1/64</td>
</tr>
<tr>
<td>UA21 (IPv6)</td>
<td>3ffe:501::2/64</td>
</tr>
<tr>
<td>PX2 (IPv6)</td>
<td>3ffe:501::20/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501::1/64</td>
</tr>
</tbody>
</table>

### [TOPOLOGY]

```
  UA11
 /   \
|     |
| R11 |
|     |
  NUT Registar
 /     \
|       |
| R12   |
 /     \
|       |
| PX2   |
 |     |
| R13   |
 |     |
| UA21  |
```

### [CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT</td>
<td>sip:ss.under.test.com</td>
</tr>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501::50/64 (IPv6)</td>
</tr>
</tbody>
</table>
[INITIALIZATION]

UA11 R NUT
|
|
|--------|--------| 1. ICMP Echo Request
|
|<-------|---------| 2. ICMP Echo Reply

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11 R Registrar
|
|
|--------|--------| 1. REGISTER
|
|<-------|---------| 2. 200 OK

1. Send REGISTER Request.
2. Receive 200 OK response.

[PROCEDURE]

UA21 PX2 : NUT : UA11
|
|
|--------> : : : | INVITE
|<---------- : : : | 407
|--------> : : : | ACK
|--------> : : : | INVITE
|<--------> : : : | 1. INVITE
|<--------> : : : | 2. INVITE
|<--------> : : : | 3. 100 Trying
|<--------> : : : | 4. 180 Ringing
|<--------> : : : | 5. 180 Ringing
<--------> : : : | 180 Ringing
<--------> : : : | 200 OK
<--------> : : : | 7. 200 OK
<--------> : : : | 200 OK
<--------> : : : | ACK
1. PX2 Send INVITE.
2. UA11 Receive INVITE.
3. PX2 Receive 100 Trying.
4. UA11 Send 180 Ringing.
5. PX2 Receive 180 Ringing.
6. UA11 Send 200 OK.
7. PX2 Receive 200 OK.
8. PX2 Send ACK.
9. UA11 Receive ACK.
10. UA11 Send INVITE.
11. PX2 Receive INVITE. (*1)
12. PX2 Send 200 OK.
13. UA11 Receive 200 OK. (*2)
14. UA11 Send ACK.
15. PX2 Receive ACK. (*3)
16. UA11 Send INVITE.
17. PX2 Receive INVITE. (*4)
18. PX2 Send 200 OK.
19. UA11 Receive 200 OK. (*5)
20. UA11 Send ACK.
21. PX2 Receive ACK. (*6)
22. PX2 Send BYE.
23. UA11 Receive BYE.
24. UA11 Send 200 OK.
25. PX2 Receive 200 OK.

== Message example ==

1. INVITE PX2 -> NUT

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501:ffff:2::2
Max-Forwards: 69
Record-Route: <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxsit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

2. INVITE NUT -> UA11

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 68
Record-Route: <sip:ss.under.test.com;lr>,
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

3. 100 Trying NUT -> PX2

SIP/2.0 100 Trying
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
From: UA21 <sip:UA21@node.under.test.com>;tag=9fxced76sl
To: UA11 <sip:UA11@node.under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA21@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151
4. 180 Ringing UA11 -> NUT

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:ffff:50:::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>,
<sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Content-Length: 0

5. 180 Ringing NUT -> PX2

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>,
<sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Length: 0

6. 200 OK UA11 -> NUT
SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501::50::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bKd4790.1
:received=3ffe:501:fff20:20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:fff2::2
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9f9ceb76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=NUT 2890844527 2890844527 IN IP6 3ffe:501:fff1:1
s=
c=IN IP6 3ffe:501:fff1:1
a=rtmap:0 PCMU/8000

7. 200 OK NUT -> PX2

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:fff20:20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:fff2::2
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9f9ceb76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 147

v=0
8. ACK PX2 -> NUT

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74b76
:received=3ffe:501:ffff:1::1
Max-Forwards: 68
Route: <sip:ss.under.test.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 ACK
Content-Length: 0

9. ACK NUT -> UA11

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:2::2
Max-Forwards: 68
Route: <sip:ss.under.test.com;lr>,
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 ACK
Content-Length: 0

10. INVITE UA11 -> NUT

INVITE sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKlkld5l
Max-Forwards: 70
Route: <sip:ss.under.test.com;lr>,
<sip:ss2.biloxi.example.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 14 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 149

v=0
o=NUT 2890844527 2890844528 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=sendonly

11. INVITE NUT -> PX2

INVITE sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKlkld5l
 :received=3ffe:501:ffff:1::1
Max-Forwards: 69
Route: <sip:ss2.biloxi.example.com;lr>
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 14 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 149

v=0
o=NUT 2890844527 2890844528 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=sendonly
12. 200 OK PX2 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
 :received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKlkld5l
 :received=3ffe:501:ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 14 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 150

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=recvonly

13. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKlkld5l
 :received=3ffe:501:ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 14 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 150

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
a=recvonly

14. ACK UA11 -> NUT

ACK sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKlkldcc
Max-Forwards: 70
Route: <sip:ss.under.test.com;lr>,
      <sip:ss2.biloxi.example.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 14 ACK
Content-Length: 0

15. ACK NUT -> PX2

ACK sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
     ;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKlkldcc
     ;received=3ffe:501:ffff:1::1
Max-Forwards: 69
Route: <sip:ss2.biloxi.example.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 14 ACK
Content-Length: 0

16. INVITE UA11 -> NUT

INVITE sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKlkdbq
Max-Forwards: 70
Route: <sip:ss.under.test.com;lr>,
      <sip:ss2.biloxi.example.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 15 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 149

v=0
o=NUT 2890844527 2890844529 IN IP6 3ffe:501::1
s=-
c=IN IP6 3ffe:501::1
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

17. INVITE NUT -> PX2

INVITE sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKlkbc
Max-Forwards: 69
Route: <sip:ss2.biloxi.example.com;lr>
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 15 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 149

v=0
o=NUT 2890844527 2890844529 IN IP6 3ffe:501::1
s=-
c=IN IP6 3ffe:501::1
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

18. 200 OK PX2 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKIklbqc
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxxed76sl
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 15 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 150

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

19. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKIklbqc
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxxed76sl
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 15 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 150

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

20. ACK UA11 -> NUT
ACK sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKlklbqc
Max-Forwards: 70
Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 15 ACK
Content-Length: 0

21. ACK NUT -> PX2

ACK sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKlklbqc
:received=3ffe:501:ffff:1::1
Max-Forwards: 70
Route: <sip:ss2.biloxi.example.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 15 ACK
Content-Length: 0

22. BYE PX2 -> NUT

BYE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
Max-Forwards: 68
Route: <sip:ss.under.test.com;lr>
Record-Route: <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 2 BYE
Content-Length: 0

23. BYE NUT -> UA11

BYE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
  ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
  ;received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
Max-Forwards: 68
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 2 BYE
Content-Length: 0

24. 200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
  ;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
  ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
  ;received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 2 BYE
Content-Length: 0

25. 200 OK NUT -> PX2

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
  ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
  ;received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 2 BYE
Content-Length: 0
[OBSERVABLE RESULTS]

*1: INVITE request from NUT to PX2.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain

- Header fields:
  - outside of a dialog
  See generic_forward_from-UA11
  See generic_forward_request

- Bodies:
  See generic_forward_from-UA11

*2: 200 response from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-PX2
  Status-Code: Must be "200". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-PX2
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from-PX2

*3: ACK request from NUT to PX2.
As a SIP Message,
   See generic_message

As a SIP request,

   · Request-Line:
     See generic_forward_from-UA11
     See generic_forward_R-URI_responsible-domain

   · Header fields:
     · outside of a dialog
       See generic_forward_from-UA11
       See generic_forward_request

   · Bodies:
     See generic_forward_from-UA11

*4:INVITE request from NUT to PX2.

As a SIP Message,
   See generic_message

As a SIP request,

   · Request-Line:
     See generic_forward_from-UA11
     See generic_forward_R-URI_non-responsible-domain

   · Header fields:
     · outside of a dialog
       See generic_forward_from-UA11
       See generic_forward_request

   · Bodies:
     See generic_forward_from-UA11

*5:200 response from NUT to UA11.

As a SIP Message,
   See generic_message

As a SIP response,
· Status-Line:
  See generic_forward_from-PX2
  Status-Code: Must be "200". [RFC3261-16-104]

· Header fields:
  See generic_forward_from-PX2
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter
    contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
    received. [RFC3261-18-28]

· Bodies:
  See generic_forward_from-PX2

*6:ACK request from NUT to PX2.

As a SIP Message,
  See generic_message

As a SIP request,

· Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_responsible-domain

· Header fields:
  * outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

· Bodies:
  See generic_forward_from-UA11

[REFERENCE]
  Sequence from RFC3665 Section 3.7.

4.2.7 PX-2-2-1 - SIP Proxy- Unsuccessful Busy (Caller)

[NAME]
PX-2-2-1 · SIP Proxy- Unsuccessful Busy (Caller)
[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when the UA2 is busy.

[REQUIREMENT]
Only when a proxy supports the architecture with two proxies
Set up registrar server to use location service, if necessary.

PARAMETER

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA21(Contact)</td>
<td>sip:<a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>PX2</td>
<td>sip:ss2.biloxi.example.com</td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA21(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>PX2(IPv6)</td>
<td>3ffe:501:ffff:20::20/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
---+-----------+---------
|           |          |
|          UA11 |
R11        |
---+---R-------+-----------+---------
|           |          |
|         NUT       Registrar |
R12        |
---+-----------+---------
|           |          |
|          PX2 |
R13        |
```
Phase 2 Test Specification

**SIP IPv6**

---

**[CONFIGURATION for NUT]**

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:fff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

**[INITIALIZATION]**

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>-----</td>
<td></td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>-----</td>
<td>-----------</td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.

**[PROCEDURE]**

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>PX2</th>
<th>UA21</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;-------</td>
<td>-------</td>
<td>-------</td>
<td>------</td>
</tr>
</tbody>
</table>

1. INVITE
2. 407
3. ACK
4. INVITE
5. INVITE
6. 100

-----------> INVITE
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. PX2 Receive INVITE.
6. UA11 Receive 100 Trying.
7. PX2 Send 100 Trying.
8. PX2 Send 486 Busy Here.
9. PX2 Receive ACK. (*1)
10. UA11 Receive 486 Busy Here. (*2)
11. UA11 Send ACK.

=== Message example ===

1. INVITE UA11 -> NUT

INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxsit55XU7p8@under.test.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

2. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
:received=3ffe:501::ff1:1
From: UA11 <sip:UA11@under.test.com>;tag=9fcxed76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=3flal12sf
Call-ID: 2xTb9vxt55XU7p8@under.test.com
CSeq: 1 INVITE
Proxy-Authenticate: Digest realm="under.test.com", qop="auth",
nonce="dc3a5ab2530aa93112cf5904ba7d88fa",
opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. ACK UA11 -> NUT

ACK sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fcxed76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=3flal12sf
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 ACK
Content-Length: 0

4. INVITE UA11 -> NUT

INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fcxed76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxt55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="dc3a5ab2530aa93112cf5904ba7d88fa", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA21@biloxi.example.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

5. INVITE NUT -> PX2

INVITE sip:UA21@biloxi.example.com SIP/2.0
 Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
 Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501:ffff:1::1
 Max-Forwards: 69
 Record-Route: <sip:ss.under.test.com;lr>
 From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
 To: UA21 <sip:UA21@biloxi.example.com>
 Call-ID: 2xTb9vxSit55XU7p8@under.test.com
 CSeq: 2 INVITE
 Contact: <sip:UA11@node.under.test.com>
 Content-Type: application/sdp
 Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

/*Client for NUT prepares to receive data on port 49172 from the network.*/

6. 100 Trying NUT -> UA11

SIP/2.0 100 Trying
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
  ;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

7. 100 Trying PX2 -> NUT

SIP/2.0 100 Trying
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
  ;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
  ;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

8. 486 Busy Here PX2 -> NUT

SIP/2.0 486 Busy Here
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
  ;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
  ;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

9. ACK NUT -> PX2

ACK sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

10. 486 Busy Here NUT -> UA11

SIP/2.0 486 Busy Here
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

11. ACK UA11 -> NUT

ACK sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

[OBSERVABLE RESULTS]
*1: ACK request from NUT to PX2.

As a SIP Message,
   See generic_message

As a SIP request,

   - Request-Line:
     See generic_make_ACK_for-non2XX

   - Header fields:
     - outside of a dialog
     See generic_make_ACK_for-non2XX

   - Bodies:
     See generic_make_ACK_for-non2XX
2:486 response from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
See generic_make_response
Status-Code: Must be "486". [RFC3261 16.7.6]

- Header fields:
See generic_make_response
* Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
Sequence from RFC3665 Section 3.9.

4.2.8 PX-2-2-2 - SIP Proxy- Unsuccessful Busy (Callee)

[NAME]
PX-2-2-2 - SIP Proxy- Unsuccessful Busy (Callee)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when the UA2 is busy.

[REQUIREMENT]
Only when a proxy supports the architecture with two proxies
Set up registrar server to use location service, if necessary.

PARAMETER

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA21(Contact)</td>
<td>sip:<a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
</tbody>
</table>
**[ADDRESS]**

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA21 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>PX2 (IPv6)</td>
<td>3ffe:501:ffff:20::20/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

**[TOPOLOGY]**

```
---+-----------+---------
|           |
|          UA11|
|           |
---+---R-------+-----------+---------
|           |           |
|         NUT   Registrar|
|           |
|           |
---+-----------+---------
|           |
|          PX2|
|           |
---+-----------+---------
|           |
|          UA21|
```

**[CONFIGURATION for NUT]**

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64  (IPv6)</td>
</tr>
</tbody>
</table>

**[INITIALIZATION]**

```
UA11  R  NUT
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
</table>
|-----|-----| 1.ICMP Echo Request
|     |     |
|<-----|-----| 2.ICMP Echo Reply
|     |     |
```
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

```
|   |   | Registrar |
|-------------------------|---------|
|   |   |           |
|   |   |           |
|<----------|--->|1. REGISTER|
|   |   |           |
|<----------|--->|2. 200 OK |
```

1. Send REGISTER Request.
2. Receive 200 OK response.

**[PROCEDURE]**

```
UA11   R   Registrar
|   |   |           |
|-------------------------|---------|
|   |   |           |
|   |   |           |
|<----------|--->|1. REGISTER|
|   |   |           |
|   |   |           |
```

1. PX2 Send INVITE.
2. UA11 Receive INVITE.
3. PX2 Receive 100 Trying.
4. UA11 Send 486 Busy Here.
5. UA11 Receive ACK. (*1)
6. PX2 Receive 486 Busy Here. (*2)
7. PX2 Send ACK.

=== Message example ===

1. INVITE PX2 -> NUT

INVITE sip:UA11@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:2::2
Max-Forwards: 69
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

2. INVITE NUT -> UA11

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
;received=3ffe:501:ffff:2::2
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:2::2
Max-Forwards: 68
Record-Route: <sip:ss.under.test.com;lr>,
<sip:ss2.biloxi.example.com;lr>,
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA21@biloxi.example.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

3. 100 Trying NUT -> PX

SIP/2.0 100 Trying
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Content-Length: 0

4. 486 Busy Here UA11 -> NUT

SIP/2.0 486 Busy Here
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Content-Length: 0

5. ACK NUT -> UA11
ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Max-Forwards: 70
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 ACK
Content-Length: 0

6. 486 Busy Here NUT -> PX2

SIP/2.0 486 Busy Here
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Content-Length: 0

7. ACK PX2 -> NUT

ACK sip:UA11@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Max-Forwards: 70
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 ACK
Content-Length: 0

[OBSERVABLE RESULTS]

*1:ACK request from NUT to UA11

As a SIP Message,
See generic_message

As a SIP request,
- Request-Line:
  See generic_make_ACK_for-non2XX
- Header fields:
  - outside of a dialog
    See generic_make_ACK_for-non2XX

- Bodies:
  See generic_make_ACK_for-non2XX

*2:486 response from NUT to PX2
As a SIP Message,
  See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response
  Status-Code: Must be "486". [RFC3261 16.7.6]

- Header fields:
  See generic_make_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
Sequence from RFC3665 Section 3.9.

4.2.9 PX-2-2-3 - SIP Proxy- Unsuccessful No Response from the UA through the other proxy

[NAME]
PX-2-2-3 - SIP Proxy- Unsuccessful No Response from the UA through the other proxy

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly executes an unsuccessful processing when the UA through the other proxy has no response.
[REQUIREMENT]
Only when a proxy supports the architecture with two proxies
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip: ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip: reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip: <a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip: <a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip: <a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA21(Contact)</td>
<td>sip: <a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>PX2</td>
<td>sip: ss2.biloxi.example.com</td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA21(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>PX2(IPv6)</td>
<td>3ffe:501:ffff:20::20/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
---------
|         |
|         |
|         |
| UA11    |
|         |

----------
|         |
|         |
| NUT    |
| Registrar |
|         |

----------
|         |
|         |
| PX2    |
|         |

---------
|         |
|         |
| UA21   |
```

[CONFIGURATION for NUT]
[INITIALIZATION]

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>---</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>---</td>
<td>-----</td>
</tr>
<tr>
<td>&lt;-----</td>
<td>---</td>
<td>1. ICMP Echo Request</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>---</td>
<td>2. ICMP Echo Reply</td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

[PROCEDURE]

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>PX2</th>
<th>UA21</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td>&lt;-----</td>
<td>---</td>
<td></td>
<td>1. INVITE</td>
</tr>
<tr>
<td>&lt;-----</td>
<td>---</td>
<td></td>
<td>2. 407</td>
</tr>
<tr>
<td>&lt;-----</td>
<td>---</td>
<td></td>
<td>3. ACK</td>
</tr>
<tr>
<td>&lt;-----</td>
<td>---</td>
<td></td>
<td>4. INVITE</td>
</tr>
<tr>
<td></td>
<td>&lt;-----</td>
<td></td>
<td>5. INVITE</td>
</tr>
<tr>
<td>&lt;-----</td>
<td>---</td>
<td></td>
<td>6. 100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;-----</td>
<td>7. 100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;-----</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>11. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>12. INVITE</td>
</tr>
</tbody>
</table>
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. PX2 Receive INVITE.
6. UA11 Receive 100 Trying.
7. PX2 Send 100 Trying.
8. PX2 Send 480 No Response.
9. PX2 Receive ACK. (*1)
10. UA11 Receive 480 No Response. (*2)
11. UA11 Send ACK.

--- Message example ---
1. INVITE UA11 -> NUT

INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

2. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=3fllal12sf
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 INVITE
Proxy-Authenticate: Digest realm="under.test.com", qop="auth",
nonce="cf5904ba7d8dc3a5ab2530aa931128fa",
opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. ACK UA11 -> NUT

ACK sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=3fllal12sf
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 1 ACK
Content-Length: 0

4. INVITE UA11 -> NUT

INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="cf5904ba7d8dc3a5ab2530aa931128fa", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA21@biloxi.example.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501::fff1::1
s=-
c=IN IP6 3ffe:501::fff1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

5. INVITE NUT -> PX2

INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501::fff1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501::fff1::1
s=-
c=IN IP6 3ffe:501::fff1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

6. 100 Trying NUT -> UA11

SIP/2.0 100 Trying
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501::fff1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

7. 100 Trying PX2 -> NUT

SIP/2.0 100 Trying
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1  
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9  
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

8. 480 No Response PX2 -> NUT

SIP/2.0 480 No Response
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1  
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9  
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

9. ACK NUT -> PX2

ACK sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0
10. 480 No Response NUT -> UA11

SIP/2.0 480 No Response
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

11. ACK UA11 -> NUT

ACK sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

**[OBSERVABLE RESULTS]**

*1: ACK request from NUT to UA11

As a SIP Message,
   See generic_message

As a SIP request,
   - Request-Line:
     See generic_make_ACK_for-non2XX
   - Header fields:
     - outside of a dialog
     See generic_make_ACK_for-non2XX
   - Bodies:
     See generic_make_ACK_for-non2XX

*2: 480 response from NUT to PX2

As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response
  Status-Code: Must be "480". [RFC3261 16.7.6], [RFC3261 21.4.18]

- Header fields:
  See generic_make_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
Sequence from RFC3665 Section 3.10.

4.2.10 PX-2-2-4 - SIP Proxy- Unsuccessful No Response from UA (Callee)

[NAME]
PX-2-2-4 - SIP Proxy- Unsuccessful No Response from UA (Callee)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly sends a 480 (No Response) response to the other proxy when the UA2 has no response.

[REQUIREMENT]
Only when a proxy supports the architecture with two proxies
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
</tbody>
</table>
[ADDRESS]

<table>
<thead>
<tr>
<th>Address</th>
<th>IPv6 Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA21 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>PX2 (IPv6)</td>
<td>3ffe:501:ffff:20::20/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

---+-----------+---------
   |           |
   |          UA11
R11 |     |   |
   |     |   |
 ---+---R-------+-----------+---------
   |           |           |
   |         NUT       Registrar
R12 |     |   |
   |     |   |
 ---+-----------+---------
   |           |
   |          PX2
R13 |     |   |
   |     |   |
 ---+-----------+---------
   |           |
   |          UA21

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip: ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

UA11  R  NUT
<p>| | | |
|     |   |   |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1. ICMP Echo Request</td>
<td></td>
<td></td>
</tr>
<tr>
<td>2. ICMP Echo Reply</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.

[PROCEDURE]

UA11   R   Registrar
<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-------</td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

UA11   PX2    NUT    UA11
<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td></td>
<td>-------</td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>-------</td>
<td></td>
</tr>
</tbody>
</table>

1. INVITE
2. INVITE (*1)
3. 100 (*2)
4. INVITE (*3)
5. INVITE (*4)
6. INVITE (*5)
7. INVITE (*6)
8. INVITE (*7)
9. INVITE (*8)
10. 480 (*9)
11. ACK
1. PX2 Send INVITE.
2. UA11 Receive INVITE. (*1)
3. PX2 Receive 100 Trying. (*2)
4. UA11 Receive INVITE. (*3)
5. UA11 Receive INVITE. (*4)
6. UA11 Receive INVITE. (*5)
7. UA11 Receive INVITE. (*6)
8. UA11 Receive INVITE. (*7)
9. UA11 Receive INVITE. (*8)
10. PX2 Receive 480 No Response. (*9)
11. PX2 Send ACK.

=== Message example ===

1. INVITE PX2 -> NUT

INVITE sip:UA11@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
Max-Forwards: 69
Record-Route: <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

2. INVITE NUT -> UA11
INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
 :received=3ffe:501::fff:20:20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501::fff:2:2
Max-Forwards: 68
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

3. 100 Trying NUT -> PX2

SIP/2.0 100 Trying
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
 :received=3ffe:501::fff:20:20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501::fff:2:2
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Content-Length: 0

4. INVITE NUT -> UA11

Resend of Message 2.
5. INVITE NUT -> UA11
   Resend of Message 2.
6. INVITE NUT -> UA11
   Resend of Message 2.
7. INVITE NUT -> UA11
   Resend of Message 2.
8. INVITE NUT -> UA11
   Resend of Message 2.
9. INVITE NUT -> UA11
   Resend of Message 2.
/* NUT gives up */
10. 480 No Response NUT -> PX2
    
    SIP/2.0 480 No Response
    Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
        ;received=3ffe:501:ffff:20::20
    Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
        ;received=3ffe:501:ffff:2::2
    From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
    To: UA11 <sip:UA11@under.test.com>;tag=314159
    Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
    CSeq: 1 INVITE
    Content-Length: 0

11. ACK PX2 -> NUT
    
    ACK sip:UA11@under.test.com SIP/2.0
    Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
    Max-Forwards: 70
    From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
    To: UA11 <sip:UA11@under.test.com>;tag=314159
    Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 ACK
Content-Length: 0

[OBSERVABLE RESULTS]
** The response code for INVITE would be different from your implementation.
Test supports any of 4xx response here.

*9:480 response from NUT to PX2
As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response
  Status-Code: Must be "480". [RFC3261-16-40,41], [RFC3261 16.7.6], [RFC3261 21.4.18]

- Header fields:
  See generic_make_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
Sequence from RFC3665 Section 3.10.

4.2.11 PX-2-2-5 - SIP Proxy- Unsuccessful Temporarily Unavailable (Callee)

[NAME]
PX-2-2-5 - SIP Proxy- Unsuccessful Temporarily Unavailable (Callee)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when receiving a 480 (Temporarily Unavailable.) from the UA2.

[REQUIREMENT]
Only when a proxy supports the architecture with two proxies
Set up registrar server to use location service, if necessary.

**[PARAMETER]**

<table>
<thead>
<tr>
<th></th>
<th>sip: ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (AOR)</td>
<td>sip: ss.under.test.com</td>
</tr>
<tr>
<td>Registrar (AOR)</td>
<td>sip: reg.under.test.com</td>
</tr>
<tr>
<td>UA11 (AOR)</td>
<td>sip: <a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11 (Contact)</td>
<td>sip: <a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA21 (AOR)</td>
<td>sip: <a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA21 (Contact)</td>
<td>sip: <a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>PX2</td>
<td>sip: ss2.biloxi.example.com</td>
</tr>
</tbody>
</table>

**[ADDRESS]**

<table>
<thead>
<tr>
<th></th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA21 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>PX2 (IPv6)</td>
<td>3ffe:501:ffff:20::20/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

**[TOPOLOGY]**

```
+-----+-----+-----
|     |     |     |
|     |     |     |
|     |   R |     |
|     |     |     |
|     |     |     |
+-----+-----+-----
|     |     |     |
|     |     |     |
+-----+-----+-----
|     |     |     |
|     |     |     |
+-----+-----+-----
```

**[CONFIGURATION for NUT]**

<table>
<thead>
<tr>
<th></th>
<th>sip: ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT</td>
<td>sip: ss.under.test.com</td>
</tr>
</tbody>
</table>
[INITIALIZATION]

UA11  R  NUT

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11  R  Registrar

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.

[PROCEDURE]

UA21  PX2  :  NUT  :  UA11

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
</tbody>
</table>
|<--------|-------:------> 1. INVITE
|        |        |        |
|        |        |        |
|<--------|-------:------> 2. INVITE
|        |        |        |
|        |        |        |
|        |        |        |
|        |        |        |
|        |        |        |
|        |        |        |
|        |        |        |
|        |        |        |

1. INVITE
2. INVITE
3. 100
4. 180
5. 180
6. 480
7. ACK (*1)
1. PX2 Send INVITE.
2. UA11 Receive INVITE.
3. PX2 Receive 100 Trying.
4. UA11 Send 180 Ringing.
5. PX2 Receive 180 Ringing.
6. UA11 Send 480 Temporarily Unavailable.
7. UA11 Receive ACK. (*1)
8. PX2 Receive 480 Temporarily Unavailable. (*2)
9. PX2 Send ACK.

=== Message example ===

1. INVITE PX2 -> NUT

INVITE sip:UA11@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9;received=3ffe:501:ffff:2::2
Max-Forwards: 69
Record-Route: <sip:ss2.biloxi.example.com;l>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
2. INVITE NUT -> UA11

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
Max-Forwards: 68
Record-Route: <sip:ss.under.test.com;lr>,<sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

3. 100 Trying NUT -> PX2

SIP/2.0 100 Trying
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Length: 0

4. 180 Ringing UA11 -> NUT
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff2::2
Record-Route: <sip:ss.under.test.com;lr>,
<sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Length: 0

5. 180 Ringing NUT -> PX2

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff2::2
Record-Route: <sip:ss.under.test.com;lr>,
<sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Length: 0

6. 480 Temporarily Unavailable UA11 -> NUT

SIP/2.0 480 Temporarily Unavailable
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff2::2
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Content-Length: 0

7. ACK NUT -> UA11

ACK sip:UA11@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Max-Forwards: 70
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 ACK
Content-Length: 0

8. 480 Temporarily Unavailable NUT -> PX2

SIP/2.0 480 Temporarily Unavailable
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 INVITE
Content-Length: 0

9. ACK PX2 -> NUT

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Max-Forwards: 70
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@biloxi.example.com
CSeq: 1 ACK
Content-Length: 0

[OBSERVABLE RESULTS]
*1:ACK request from NUT to UA11

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_make_ACK_for-non2XX

- Header fields:
  - outside of a dialog
    See generic_make_ACK_for-non2XX

- Bodies:
  See generic_make_ACK_for-non2XX

*2:480 response from NUT to PX2
As a SIP Message,
  See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response
  Status-Code: Must be "480". [RFC3261 16.7.6], [RFC3261 21.4.18]

- Header fields:
  See generic_make_response
  * Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

**REFERENCE**
Sequence from RFC3665 Section 3.11.

4.3 Routing

4.3.1 FW-1-1-1 - SIP Proxy- Request-URI with escaped characters

**NAME**
FW-1-1-1 - SIP Proxy- Request-URI with escaped characters
[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT doesn’t change escaped characters in a Request-URI into unescaped characters and sends the request the other UA when receiving a request with escaped characters.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node.11.under.test.com">UA12@node.11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
---+-----------+---------
|           |
|          | UA11    |
| R11      |
---+---R-------+-----------+---------
|           |           |
|         NUT       Registrar |
| R12      |
---+-----------+---------
|           |
|          | UA12    |
```

[CONFIGURATION for NUT]
**INITIALIZATION**

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

<table>
<thead>
<tr>
<th>UA11</th>
<th>UA12</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**PROCEDURE**

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
</table>

1. 1. INVITE
2. 2. 407
3. 3. ACK
4. 4. INVITE
5. 5. INVITE (*1)
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE. (*1)
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK.
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA12 Send BYE.
14. UA11 Receive BYE.
15. UA11 Send 200.
16. UA12 Receive 200.

=== Message example ===

1. INVITE UA11 -> NUT

INVITE sip:U%6512@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:U%6512@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

* userinfo of Request-URI has escaped character (SP(0x20)).

/* Proxy(NUT) challenges UA11 for authentication */

2. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:U%6512@under.test.com>;tag=3flal12sf
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 INVITE
Proxy-Authenticate: Digest realm="under.test.com", qop="auth",
nonce="f84f1ce41e6b5e5a9e8e88d359",
opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. ACK UA11 -> NUT

ACK sip:U%6512@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:U%6512@under.test.com>;tag=3flal12sf
Call-ID: 3848276298220188511@under.test.com
209

CSeq: 1 ACK
Content-Length: 0

* Request-URI does not change.

4. INVITE UA11 -> NUT

INVITE sip:U%6512@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1cec41e6cbe5aea9c8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:U%6512@under.test.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:U%6512@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

* userinfo of Request-URI has escaped character (SP(0x20)).

5. INVITE NUT -> UA12

INVITE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

* Request-URI: userinfo part does not change.

6. 100 Trying NUT -> UA11

SIP/2.0 100 Trying
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxc6d76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Length: 0

7. 180 Ringing UA12 -> NUT

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:50:50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:U%6512@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

8. 180 Ringing NUT -> UA11

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
  ;received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:U%6512@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

9. 200 OK UA12 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
  ;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
  ;received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:U%6512@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA12 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

10. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:U%6512@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA12 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

11. ACK UA11 -> NUT

ACK sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b76
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1ce41e6cbe5aea9c8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:U%6512@under.test.com",
response="b51e504e73af54829e4f2bd7f8de4654"
Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:U%6512@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 ACK
Content-Length: 0

12. ACK NUT -> UA12

ACK sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b76
     ;received=3ffe:501:ffff:1::1
Max-Forwards: 69
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:U%6512@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 ACK
Content-Length: 0

13. BYE UA12 -> NUT

BYE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
Route: <sip:ss.under.test.com:lr>
From: UA12 <sip:U%6512@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

* userinfo of From header field has escaped character (SP(0x20)).

14. BYE NUT -> UA11

BYE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK74b43
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnashds7
     ;received=3ffe:501:ffff:2::2
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com:lr>
From: UA12 <sip:U%6512@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

15. 200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK74b43
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:2::2
From: UA12 <sip:U%6512@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

16. 200 OK NUT -> UA12

SIP/2.0 200 OK
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:2::2
From: UA12 <sip:U%6512@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

[OBSERVABLE RESULTS]
*2:INVITE request from NUT to UA12.

As a SIP Message,
- Must be transmitted. [RFC3261-16-31]
  See generic_message

As a SIP request,
- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain
  Request-URI
userinfo: Must convert from escaped characters to unescaped characters.
[RFC3261-16-31][RFC3261-19-12]

- Header fields:
- outside of a dialog
  See generic_forward_from-UA11
  See generic_forward_request

* To
  Must not convert escaped characters into unescaped characters.
  [RFC3261-16-31]

- Bodies:
  See generic_forward_from-UA11

[REFERENCE]
[RFC3261-16-31]
16.5 Determining Request Targets

When accessing the location service constructed by a registrar, the Request-URI MUST first be canonicalized as described in Section 10.3 before being used as an index. The output of these mechanisms is used to construct the target set.

4.3.2 FW-1-1-2 - SIP Proxy- Non-allowed parameters in Request-URI

[NAME]
FW-1-1-2 - SIP Proxy- Non-allowed parameters in Request-URI

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly removes the parameters in a Request-URI which are not allowed.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
</tbody>
</table>
Phase 2 Test Specification

<table>
<thead>
<tr>
<th>UA11(Contact)</th>
<th>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

**[ADDRESS]**

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

**[TOPOLOGY]**

```
---+-----------+---------
   |           |
   |          UA11|
R11
---+---R-------+-----------+---------
   |           |           |
   |         NUT       Registrar|
R12
---+-----------+---------
   |           |           |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
   |
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11 : NUT : UA12

1. INVITE

UA11 : NUT : UA12

Both Way RTP Media
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE. (*1)
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK.
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA12 Send BYE.
14. UA11 Receive BYE.
15. UA11 Send 200.
16. UA12 Receive 200.

== Message example ==

4. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com;method=INVITE?Subject=test SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1cecc41e6cbe5ae9c8e88d359", opaque="",
quop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA12@under.test.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

/* There are two components that are not allowed to exist in Request-URI, one is method component, another is header field component (Subject). */

5. INVITE NUT -> UA12

INVITE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

[OBSERVABLE RESULTS]
*1:INVITE request from NUT to UA12.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain
* Request-URI
  Must remove any parameters not allowed in a Request-URI.
  \[RFC3261-16-48][RFC3261-19-9\]

- Header fields:
  - outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

- Bodies:
  See generic_forward_from-UA11

[REFERENCE]
\[RFC3261-16-47, 48\]
16.6 Request Forwarding

2. Request-URI

The Request-URI in the copy's start line MUST be replaced with
the URI for this target. If the URI contains any parameters
not allowed in a Request-URI, they MUST be removed.

4.3.3 FW-1-1-3 - SIP Proxy- Update of a Request-URI scheme

(NAME)
FW-1-1-3 - SIP Proxy- Update of a Request-URI scheme

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly changes the scheme into the SIP scheme and sends that request when receiving a 416 (Unsupported URI Scheme) response because the Request-URI scheme is tel URL, not the SIP scheme.

[REQUIREMENT]
Only when the proxy can forward tel-URL.
Set up registrar server to use location service, if necessary.

[PARAMETER]

<p>| NUT(AOR) | sip:ss.under.test.com:lr |
| Registrar(AOR) | sip-reg.under.test.com |</p>
<table>
<thead>
<tr>
<th>UA11(AOR)</th>
<th>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:00011112222</td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:00011112222</td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>tel:00011112222;user=phone</td>
</tr>
</tbody>
</table>

**[ADDRESS]**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

**[TOPOLOGY]**

```
---+-----------+---------
   |           |
   |          UA11 |
R11
   | R
---+-----------+---------
   | NUT       Registrar |
R12
   |
---+-----------+---------
         |           |
         |         UA12 |
```

**[CONFIGURATION for NUT]**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT</td>
<td>sip:ss.under.test.com:lr</td>
</tr>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

**[INITIALIZATION]**

```
UA11  R  NUT
  |   |   |
---|---|--->
  1. ICMP Echo Request
  |   |   
|<---|---|---|
  2. ICMP Echo Reply
  |   |   |
1. Send ICMP Echo Request.
```
2. Receive ICMP Echo Reply.

* NUT changes SIP-URI scheme to tel-URL scheme.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11 : NUT : UA12

|-------------------|--------|1. REGISTER
|                   |<------|2. 200 OK
|                   |--------|3. REGISTER
|<------------------|-------|4. 200 OK
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 416 Unsupported URI Scheme.
8. UA12 Receive ACK.
9. UA12 Receive INVITE (*1)
10. UA12 Send 180 Ringing.
11. UA11 Receive 180 Ringing.
12. UA12 Send 200 OK.
13. UA11 Receive 200 OK.
14. UA11 Send ACK.
15. UA12 Send ACK.
16. UA12 Send BYE.
17. UA11 Receive BYE.
18. UA11 Send 200.
19. UA12 Receive 200.

==== Message example ====

4. INVITE UA11 -> NUT

INVITE sip:00011112222 SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1c9c41e66c5ae9c8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:00011112222",
response="b51e504e73af54829e4f2bd7f8dc4654"
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:00011112222>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

/* Proxy(NUT) accepts the credentials and forwards the INVITE to UA12 */
5. INVITE NUT -> UA12

INVITE tel:00011112222;user=phone SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <tel:00011112222;user=phone>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

7. 416 Unsupported URI Scheme UA12 -> NUT

SIP/2.0 416 Unsupported Scheme
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:00011112222>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:00011112222>
CSeq: 2 INVITE
Content-Length: 0

9. INVITE NUT -> UA12

INVITE sip:00011112222 SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:00011112222>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
[OBSERVABLE RESULTS]

*1: INVITE request from NUT to UA12.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain
  Request-URI: Must be the original SIP URI. [RFC3261-16-101, RFC3261-16-102]

- Header fields:
  - outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

- Bodies:
  See generic_forward_from-UA11

[REFERENCE]

[RFC3261-16-101, 102]
16.7 Response Processing

If a proxy receives a 416 (Unsupported URI Scheme) response to a request whose Request-URI scheme was not SIP, but the scheme in the original received request was SIP or SIPS (that is, the proxy changed the scheme from SIP or SIPS to something else when it proxied a request), the proxy SHOULD add a new URI to the target set. This URI SHOULD be a SIP URI version of the non-SIP URI that was just tried. In the case of the tel URL, this is accomplished by placing the telephone-subscriber part of the tel URL into the user part of the SIP URI, and setting the hostpart to the domain where the prior request was sent. See Section 19.1.6 for more detail on forming SIP URIs from tel URLs.

4.3.4 FW-1-2-1 - SIP Proxy- Request-URI with an unknown scheme

[NAME]
FW-1-2-1 · SIP Proxy- Request-URI with an unknown scheme
[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT rejects the request with a 416 (Unsupported URI Scheme) response when a request contains the Request-URI whose scheme is not understood.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>Specification</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip: ss.under.test.com:lr</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip: reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip: <a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip: <a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip: <a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip: <a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>Specification</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
    +----------+
    |          |
    |   UA11   |
    |          |
    |
    +----------+
    |          |
    |     R    |
    |          |
    |          |
    |   UA11   |
    |          |
    |          |
    |          |
    +----------+
    |          |
    |          |
    |   NUT    |
    | Registrar|
    |          |
    |
    +----------+
    |          |
    |   R     |
    |          |
    |          |
    |          |
    |   UA12   |
```

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>Specification</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT</td>
<td>sip: ss.under.test.com:lr</td>
</tr>
</tbody>
</table>
[INITIALIZATION]

UA11  |  R  |  NUT

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11  |  UA12  |  R  |  Registrar

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11  |  NUT  |  UA12

1. UA11 Send INVITE.
2. UA11 Receive 416 Unsupported URI Scheme. (*1)
3. UA11 Send ACK.

--- Message example ---

1. INVITE UA11 -> NUT

INVITE nobodyKnowsThisScheme:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxc6d76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

[OBSERVABLE RESULTS]

*1:416 response from NUT to UA11.
   As a SIP Message,
   See generic_message

   As a SIP response,

   · Status-Line:
      See generic_make_response
      Status-Code: Should be "416". [RFC3261-16-15]

   · Header fields:
      See generic_make_response
      * Via
         via-received: Must be added if the host portion of the "sent-by" parameter
         contains a domain name. [RFC3261-18-27]
         via-received: Must contain the source address from which the packet was
         received. [RFC3261-18-28]
16.3 Request Validation

2. URI scheme check

   If the Request-URI has a URI whose scheme is not understood by the proxy, the proxy SHOULD reject the request with a 416 (Unsupported URI Scheme) response.

4.3.5 FW-1-2-2 - SIP Proxy- Request with an inexistent entity at the proxy

[NAME]
FW-1-2-2 - SIP Proxy- Request with an inexistent entity at the proxy

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly returns a 404 (Not Found) response when receiving a Request-URI with the entity that doesn’t exist at the proxy.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>
[TOPOLOGY]

---+-----------+---------
 |           |
 |          UA11
R11
 |
---+---R-------+-----------+---------
 |           |           |
 |         NUT       Registrar
R12
 |
---+-----------+---------
 |           |
 |            |
 |            |
 |           |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |
 |...
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------:------:</td>
<td></td>
<td>1. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------:------:</td>
<td></td>
<td>2. 404 (*1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------:------:</td>
<td></td>
<td>3. ACK</td>
</tr>
</tbody>
</table>

1. UA11 Send INVITE.
2. UA11 Receive 404 Not Found. (*1)
3. UA11 Send ACK.

**== Message example ==**

1. INVITE UA11 -> NUT

```
INVITE sip:NotExist@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: NotExist <sip:NotExist@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501::fff1:1
s=
Rt=IN IP6 3ffe:501::fff1:1
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```
* UA11 send INVITE to no existing entity "NotExist".

2. 404 Not Found NUT -> UA11

SIP/2.0 404 Not Found
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
   ;received=3ffe:501::ff1:1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: NotExist <sip:NotExist@under.test.com>;tag=3flal12sf
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 INVITE
Content-Length: 0

[OBSERVABLE RESULTS]
*1:404 response from NUT to UA11.
   As a SIP Message,
      See generic_message

   As a SIP response,
      · Status-Line:
         See generic_make_response
         Status-Code: Must be "404". [RFC3261-16-39]

      · Header fields:
         See generic_make_response
         See generic_proxy-auth
      * Via
         via-received: Must be added if the host portion of the "sent-by" parameter
         contains a domain name. [RFC3261-18-27]
         via-received: Must contain the source address from which the packet was
         received. [RFC3261-18-28]

[REFERENCE]
[RFC3261-16-39]
16.5 Determining Request Targets

If the Request-URI indicates a resource at this proxy that does not
exist, the proxy MUST return a 404 (Not Found) response.

4.3.6 FW-1-2-3 - SIP Proxy- An unsupported option-tag in a
Proxy-Require

(NAME) FW-1-2-3 - SIP Proxy: An unsupported option-tag in a Proxy-Require

[TARGET] SIP Proxy

[PURPOSE] Verify that a NUT properly rejects an unsupported option-tag in a Proxy-Require header field with 420 (Bad Extension) response.

[REQUIREMENT] Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT (AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11 (AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11 (Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12 (AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12 (Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
                         +-----------+---------
                         |           |
                         |          UA11 |
                         R11 ---+-----------+---------
                         |           |
                         |         NUT       Registrar |
                         R12 ---+-----------+---------
```

IPv6 FORUM TECHNICAL DOCUMENT
IPv6 Ready Logo Program
Phase 2 Test Specification
SIP IPv6
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501::ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

UA11      R        NUT
|        |         |
|        |         |
|--------|--------| 1. ICMP Echo Request
|        |         |
|<--------|--------| 2. ICMP Echo Reply

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11     UA12        R     Registrar
|                   |         |
|                   |         |
|-------------------|--------| 1. REGISTER
|                   |         |
|<------------------|---------| 2. 200 OK

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11 : NUT : UA12
| :   :   |
|-------:------| 1. INVITE
|       :       |
|<--------:------| 2. 420 (*1)

---

IPv6 FORUM TECHNICAL DOCUMENT
IPv6 Ready Logo Program
Phase 2 Test Specification
SIP IPv6

235
1. UA11 Send INVITE.
2. UA11 Receive 420 Bad Extension. (*1)
3. UA11 Send ACK.

--- Message example ---

1. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 INVITE
Proxy-Require: 999rel
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

[OBSERVABLE RESULTS]

*1:420 response from NUT to UA11.
As a SIP Message,
   See generic_message

As a SIP response,

   · Status-Line:
      See generic_make_response
      Status-Code: Must be "420". [RFC3261-16-18]

   · Header fields:
See generic_make_response

via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]

via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

* Unsupported
Must exist.
option-tag: Must include "999rel". [RFC3261-16-19]

[REFERENCE]
[RFC3261-16-18, 19]

16.3 Request Validation

5. Proxy-Require check

If the request contains a Proxy-Require header field (Section 20.29) with one or more option-tags this element does not understand, the element MUST return a 420 (Bad Extension) response. The response MUST include an Unsupported (Section 20.40) header field listing those option-tags the element did not understand.

4.3.7 FW-1-2-4 - SIP Proxy- Max-Forwards header field with a value of zero(0)

[NAME]
FW-1-2-4 - SIP Proxy- Max-Forwards header field with a value of zero(0)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT sends a 483 (Too many hops) response and doesn't forward the request when receiving a request containing a Max-Forwards header field with a field value of zero(0).

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td></td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>--------------</td>
<td>-------------------------</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

**[ADDRESS]**

<table>
<thead>
<tr>
<th></th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td></td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

**[TOPOLOGY]**

```
---+-----------+---------
|           |          |
|           |  UA11    |
R11        |
---+-----------+---------
|           |          |
|           |  R       |
|           |          |
|           |  NUT     |
|           | Registrar|
|           |          |
R12        |
---+-----------+---------
|           |          |
|           |          |
|           |  UA12    |
```

**[CONFIGURATION for NUT]**

<table>
<thead>
<tr>
<th></th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT</td>
<td></td>
</tr>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64  (IPv6)</td>
</tr>
</tbody>
</table>

**[INITIALIZATION]**

```
UA11  R  NUT
<p>| | |
|    |    |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1. ICMP Echo Request</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------------------</td>
<td>2. ICMP Echo Reply</td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

1. UA11 Send INVITE.
   (*1)
2. UA11 Receive 483 Too many hops. (*2)
3. UA11 Send ACK.

== Message example ==

1. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 0
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

[OBSERVABLE RESULTS]
*1: after NUT received INVITE request.

Must not forward this request to UA12. [RFC3261-16-16]

*2: 483 response from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP response,

* Status-Line:
See generic_make_response
Status-Code: Must be "483". [RFC3261-16-17]

* Header fields:
See generic_make_response
* Via
via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
[RFC3261-16-16, 17]

16.3 Request Validation
3. Max-Forwards check

If the request contains a Max-Forwards header field with a field value of zero (0), the element MUST NOT forward the request. If the request was for OPTIONS, the element MAY act as the final recipient and respond per Section 11. Otherwise, the element MUST return a 483 (Too many hops) response.

### 4.3.8 FW-1-2-5 - SIP Proxy- Request without Max-Forwards header field

**[NAME]**
FW-1-2-5 - SIP Proxy- Request without Max-Forwards header field

**[TARGET]**
SIP Proxy

**[PURPOSE]**
Verify that a NUT adds a Max-Forwards header field with a field value, which should be 70, when a request doesn’t have a Max-Forwards header field.

**[REQUIREMENT]**
Set up registrar server to use location service, if necessary.

**[PARAMETER]**
| NUT(AOR)   | sip:.ss.under.test.com;lr |
| Registrar(AOR) | sip:reg.under.test.com |
| UA11(AOR)   | sip:UA11@under.test.com  |
| UA11(Contact) | sip:UA11@node.under.test.com |
| UA12(AOR)   | sip:UA12@under.test.com  |
| UA12(Contact) | sip:UA12@node11.under.test.com |

**[ADDRESS]**
| NUT (IPv6)   | 3ffe:501:ffff:50::50/64 |
| Registrar (IPv6) | 3ffe:501:ffff:50::60/64 |
| UA11(IPv6)   | 3ffe:501:ffff:1::1/64   |
| UA12(IPv6)   | 3ffe:501:ffff:2::2/64   |
| R(IPv6)      | 3ffe:501:ffff:50::1/64  |

**[TOPOLOGY]**
```
| | | |
| | UA11 |
```
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------:-------&gt;</td>
<td></td>
<td>1. INVITE</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>2. 407</td>
</tr>
<tr>
<td>------:-------&gt;</td>
<td></td>
<td>3. ACK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------:-------&gt;</td>
<td></td>
<td>4. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5. INVITE (*1)</td>
</tr>
<tr>
<td>------:-------&gt;</td>
<td></td>
<td>6. 100</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7. 180</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8. 180</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9. 200</td>
</tr>
<tr>
<td>------:-------&gt;</td>
<td></td>
<td>10. 200</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------:------&gt;</td>
<td></td>
<td>11. ACK</td>
</tr>
<tr>
<td></td>
<td>------:-------&gt;</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;===============================&gt;</td>
<td></td>
<td>Both Way RTP Media</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>14. BYE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE. (*1)
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK.
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA12 Send BYE.
14. UA11 Receive BYE.
15. UA11 Send 200.
16. UA12 Receive 200.

=== Message example ===

4. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f8f1cece41e6cbe5aa9e8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA12@under.test.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 289084526 289084526 IN IP6 3ffe:501:ffff:1::1
s=
C=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
/* There are no Max-Forwards header field. */

5. INVITE NUT -> UA12

INVITE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 70
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

[OBSERVABLE RESULTS]
*1:INVITE request from NUT to UA12.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain

- Header fields:
  - outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

* Max-Forwards
Must exist. [RFC3261-16-50]
1*DIGIT: the value Should be "70". [RFC3261-16-51]

- Bodies:
  See generic_forward_from-UA11

[REFERENCE]
[RFC3261-16-50, 51]

16.6 Request Forwarding
3. Max-Forwards

If the copy does not contain a Max-Forwards header field, the proxy MUST add one with a field value, which SHOULD be 70.

Some existing UAs will not provide a Max-Forwards header field in a request.

4.3.9 FW-1-2-6 - SIP Proxy- Timestamp header field in a 100 response

**[NAME]**
FW-1-2-6 · SIP Proxy- Timestamp header field in a 100 response

**[TARGET]**
SIP Proxy

**[PURPOSE]**
Verify that a NUT properly copies any Timestamp header field into the 100 response when 100 (trying) response is generated.

**[REQUIREMENT]**
Only when a proxy can process a Timestamp header field.
Set up registrar server to use location service, if necessary.

**[PARAMETER]**

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

**[ADDRESS]**

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

**[TOPOLOGY]**

---+-----------+---------
|           |         |

---
Phase 2 Test Specification

SIP IPv6

[CONFIGURATION for NUT]

| NUT | sip:ss.under.test.com:lr |
| NUT(IPADDRESS) | 3ffe:501:ffff:50::50/64 (IPv6) |

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

```
UA11    :     NUT     :     UA12
|       :       |       :       |
|       :       |       :       |
|-------:------>|       :       | 1. INVITE
|<------:--------|       :       | 2. 407
|-------:------->|       :       | 3. ACK
|       :       |       :       |
|-------:------->|       :       | 4. INVITE
|       :       |-------:------->| 5. INVITE
|<-------:--------|       :       | 6. 100 (*1)
|       :       |<-------:--------| 7. 180
|-------:------->|       :       | 8. 180
|       :       |<-------:--------| 9. 200
|-------:------->|       :       | 10. 200
|       :       |       :       |
|-------:------->|       :       | 11. ACK
|       :       |-------:------->| 12. ACK
|<===================================>| Both Way RTP Media
|       :       |       :       |
|       :       |<-------:--------| 13. BYE
|<-------:--------|       :       | 14. BYE
|       :       |       :       |
|-------:------->|       :       | 15. 200
|       :       |-------:------->| 16. 200
```

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying. (*1)
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK.
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA12 Send BYE.
14. UA11 Receive BYE.
15. UA11 Send 200.
16. UA12 Receive 200.

=== Message example ===

4. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1cecc41e6cbe5aa9c8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA12@under.test.com",
response="b51e504e73af54829e4f2bd78dc4654"
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151
Timestamp: 54

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

6. 100 Trying NUT -> UA11

SIP/2.0 100 Trying
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Length: 0
Timestamp: 54 1.5

* "1.5" in Timestamp header field is the delay value, if there is a delay in generating the response

[OBSERVABLE RESULTS]
* 1:100 response from NUT to UA11 (Optional)

As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response
  Status-Code: Must be "100". [RFC3261-4]

- Header fields:
  See generic_make_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

  * Timestamp
    Must exist.
    Must be the same as "4.INVITE" (except delay). [RFC3261-8-95]

[REFERENCE]
[RFC3261-8-95, 96, 97]

8.2.6.1 Sending a Provisional Response

When a 100 (Trying) response is generated, any Timestamp header field present in the request MUST be copied into this 100 (Trying) response. If there is a delay in generating the response, the UAS SHOULD add a delay value into the Timestamp value in the response. This value MUST contain the difference between the time of sending of the response and receipt of the request, measured in seconds.

4.3.10 FW-2-1-1 - SIP Proxy- "sent-by" in Via with a domain name and a
[NAME]  
FW-2.1-1  ·  SIP Proxy  ·  SIP Proxy  ·  "sent-by" in Via with a domain name and a port

[TARGET]  
SIP Proxy

[PURPOSE]  
Verify that a NUT properly forwards to the port of the element when a "sent-by" parameter in a Via header field contains a domain name and a port.

[REQUIREMENT]  
Set up registrar server to use location service, if necessary.

[PARAMETER]  

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node.11.under.test.com">UA12@node.11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]  

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>
[TOPOLOGY]

--- +-----------+---------
|           |
|          UA11 |
R11
|           |
---+---R-------+-----------+---------
|           |           |
|         NUT       Registrar |
R12
|           |
---+-----------+---------
|           |
|           |
|           |
|           |

UA12

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ff:50:50/64   (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

UA11      R     NUT

|--------|--------| 1. ICMP Echo Request |
|<-------|---------| 2. ICMP Echo Reply |

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11      UA12      R     Registrar

|---------|--------|-------------------| 1. REGISTER |
|<--------|---------|--------------------| 2. 200 OK |
|---------|--------|-------------------| 3. REGISTER |
|<--------|---------|--------------------| 4. 200 OK |
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

```
<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

1. INVITE
2. 407
3. ACK
4. INVITE
5. INVITE
6. 100 (*1)
7. 180
8. 180 (*2)
9. 200
10. 200 (*3)
11. ACK
12. ACK

```
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:---------</td>
<td>:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

13. BYE
14. BYE
15. 200
16. 200

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying. (*1)
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing. (*2)
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK. (*3)
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA12 Send BYE.
14. UA11 Receive BYE.
15. UA11 Send 200 OK.
16. UA12 Receive 200 OK.

=== Message example ===

4. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5070;branch=z9hG4bK74b43
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1ce41e6ce5ae9e8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip: UA12@under.test.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip: UA12@under.test.com >
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1:1
s=-
c=IN IP6 3ffe:501:ffff:1:1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

* Source port number = 5070

8. 180 Ringing NUT -> UA11
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP node.under.test.com:5070;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@under.test.com>
CSeq: 2 INVITE
Content-Length: 0

10. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5070;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA21 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

[OBSERVABLE RESULTS]
*1:100 response from NUT to UA11.(Optional)
The destination address of this message Must be equal to 3ffe:501:ffff:50::50. [RFC3261-18-35]
The destination port of this message Must be equal to 5070. [RFC3261-18-35]

As a SIP Message,
See generic_message

As a SIP response,
- Status-Line:
  See generic_make_response
  Status-Code: Must be "100". [RFC3261 4]

- Header fields:
  See generic_make_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

*2:180 response from NUT to UA11.
  The destination address of this message Must be equal to 3ffe:501:fff:50::50. [RFC3261-18-35]
  The destination port of this message Must be equal to 5070. [RFC3261-18-35]

As a SIP Message,
  See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA12
  Status-Code: Must be "180". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA12
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from-UA12

*3:200 response from NUT to UA11.
  The destination address of this message Must be equal to 3ffe:501:fff:50::50. [RFC3261-18-35]
  The destination port of this message Must be equal to 5070. [RFC3261-18-35]
As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from- UA12
  Status-Code: Must be "200". [RFC3261-16-104]

- Header fields:
  See generic_forward_from- UA12
  See generic_forward_response
* Via
  via-received: Must be added if the host portion of the "sent-by" parameter
  contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was
  received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from- UA12

[REFERENCE]
[RFC3261-18-26, 27, 28]

18.2.1 Receiving Requests

When the server transport receives a request over any transport, it
MUST examine the value of the "sent-by" parameter in the top Via
header field value. If the host portion of the "sent-by" parameter
contains a domain name, or if it contains an IP address that differs
from the packet source address, the server MUST add a "received"
parameter to that Via header field value. This parameter MUST
contain the source address from which the packet was received. This
is to assist the server transport layer in sending the response,
since it must be sent to the source IP address from which the request
came.

[RFC3261-18-35]

18.2.2 Sending Responses

  o Otherwise (for unreliable unicast transports), if the top Via
    has a "received" parameter, the response MUST be sent to the
    address in the "received" parameter, using the port indicated
in the "sent-by" value, or using port 5060 if none is specified explicitly.

4.3.11 FW-2-1-2 - SIP Proxy- "sent-by" in Via with a domain name and without a port

[NAME]
FW-2-1-2 - SIP Proxy- "sent-by" in Via with a domain name and without a port

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly forwards to the element that has the port number of 5060 when a "sent-by" parameter in a Via header field contains a domain name and no port.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501::ff50:50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501::ff50:60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501::ff1:1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501::ff2:2/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501::ff50:1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
+--------------------------+
|                          |
|                          |
|                          |
|                          |
|                          |
|                          |
|                          |
|                          |
|                          |
|                          |
|                          |
|                          |
|                          |
|                          |
|                          |
|                          |
|                          |
|                          |
|                          |
+--------------------------+
```

UA11

R11
### Phase 2 Test Specification

**SIP IPv6**

<table>
<thead>
<tr>
<th>NUT</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td>R12</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**[CONFIGURATION for NUT]**

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT</td>
<td>sip:ss.under.test.com:lr</td>
</tr>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501::ff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

**[INITIALIZATION]**

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-------&gt;</td>
<td>1. ICMP Echo Request</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-------&gt;</td>
<td>2. ICMP Echo Reply</td>
<td></td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

<table>
<thead>
<tr>
<th>UA11</th>
<th>UA12</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;---------&gt;</td>
<td>1. REGISTER</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;---------&gt;</td>
<td>2. 200 OK</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;---------&gt;</td>
<td>3. REGISTER</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;---------&gt;</td>
<td>4. 200 OK</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT :</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying. (*1)
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing. (*2)
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK. (*3)
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA12 Send BYE.
14. UA11 Receive BYE.
15. UA11 Send 200 OK.
16. UA12 Receive 200 OK.

Both Way RTP Media

1. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxc676sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 INVITE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

8. 180 Ringing NUT -> UA11

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP node.under.test.com;branch=z9hG4bK74b9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxc676sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Length: 0

10. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com;branch=z9hG4bK74b9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxc676sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@under.test.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA21 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

[OBSERVABLE RESULTS]
The destination address of this message Must be equal to 3ffe:501:ffff:50::50. [RFC3261-18-35]
The destination port of this message Must be equal to 5060. [RFC3261-18-35]

*1:100 response from NUT to UA11.(Optional)
As a SIP Message,
See generic_message

As a SIP response,

· Status-Line:
  See generic_make_response
  Status-Code: Must be "100". [RFC3261 4]

· Header fields:
  See generic_make_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

*2:180 response from NUT to UA11.
As a SIP Message,
See generic_message

As a SIP response,
- Status-Line:
  See generic_forward_from-UA12
  Status-Code: Must be "180". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA12
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter
    contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
    received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from-UA12

*3:200 response from NUT to UA11.
  As a SIP Message,
  See generic_message

  As a SIP response,

- Status-Line:
  See generic_forward_from-UA12
  Status-Code: Must be "200". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA12
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter
    contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
    received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from-UA12

[REFERENCE]
[RFC3261-18-35]

18.2.2 Sending Responses
o Otherwise (for unreliable unicast transports), if the top Via has a "received" parameter, the response MUST be sent to the address in the "received" parameter, using the port indicated in the "sent-by" value, or using port 5060 if none is specified explicitly.

4.3.12 FW-2-1-3 - SIP Proxy- Multiple 2xx responses

[NAME]  
FW-2-1-3 - SIP Proxy- Multiple 2xx responses

[TARGET]  
SIP Proxy

[PURPOSE]  
Verify that a NUT properly processes when each session is established and the proxy receives multiple 2xx responses.

[REQUIREMENT]  
Only when a proxy supports two-proxy architecture.  
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th></th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip:<a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>UA22(Contact)</td>
<td>sip:<a href="mailto:UA22@client2.biloxi.example.com">UA22@client2.biloxi.example.com</a></td>
</tr>
<tr>
<td>PX2</td>
<td>sip:ss2.biloxi.example.com;lr</td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th></th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50:60/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:1:1/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:2:2/64</td>
</tr>
<tr>
<td>UA21 (IPv6)</td>
<td>3ffe:501:ffff:22:22/64</td>
</tr>
<tr>
<td>PX2 (IPv6)</td>
<td>3ffe:501:ffff:20:20/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50:1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64  (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

---
1. Send REGISTER Request.
2. Receive 200 OK response.

**[PROCEDURE]**

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>PX2</th>
<th>UA21</th>
<th>UA22</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:------&gt;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:------&gt;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:------&gt;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:------&gt;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:------&gt;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:------&gt;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. INVITE
2. 407
3. ACK
4. INVITE
5. INVITE
6. 100
7. 100
8. 180
9. 180
10. 200
11. 200
12. ACK
13. ACK

Both Way RTP Media
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. PX2 Receive INVITE.
6. UA11 Receive 100 Trying.
7. PX2 Send 100 Trying.
8. PX2 Send 180 Ringing.
9. UA11 Receive 180 Ringing.
10. PX2 Send 200 OK.
11. UA11 Receive 200 OK.
12. UA11 Send ACK.
13. PX2 Receive ACK.
14. PX2 Send 180 Ringing.
15. UA11 Receive 180 Ringing.
16. PX2 Send 200 OK.
17. UA11 Receive 200 OK. (*1)
18. UA11 Send ACK.
19. PX2 Receive ACK. (*2)
20. PX2 Send BYE.
21. UA11 Receive BYE. (*3)
22. UA11 Send 200 OK.
23. PX2 Receive 200 OK. (*4)
24. PX2 Send BYE.
25. UA11 Receive BYE. (*5)
26. UA11 Send 200 OK.
27. PX2 Receive 200 OK. (*6)

=== Message example ===

4. INVITE UA11 -> NUT

```
INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1cecc41e6cbe5aea9e8e8d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA21@biloxi.example.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=
 c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```
5. INVITE NUT -> Proxy 2
INVITE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

10. 200 OK Proxy 2 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA21 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

11. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA21 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

12. ACK UA11 -> NUT

ACK sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf76
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1ece41e6cbe5ae9c8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA21@biloxi.example.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 ACK
Content-Length: 0
13. ACK NUT -> Proxy 2

ACK sip:UA21@client.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b76
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Route: <sip:ss2.biloxi.example.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 ACK
Content-Length: 0

16. 200 OK Proxy 2 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=r98765
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA22@client2.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA22 2890844527 2890844527 IN IP6 3ffe:501:ffff:22::22
s=-
c=IN IP6 3ffe:501:ffff:22::22
i=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

* To tag and Contact URI are different from these in 10.200 OK.

17. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:fff:1
Record-Route: <sip:ss2.biloxi.example.com;lr>, <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=r98765
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA22@client2.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA22 2890844527 2890844527 IN IP6 3ffe:501:ffff:22::22
s=-
c=IN IP6 3ffe:501:ffff:22::22
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

18. ACK UA11 -> NUT

ACK sip:UA22@client2.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKiLv329
Max-Forwards: 70
Proxy: Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1ce41e6be5aea9c8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA21@biloxi.example.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=r98765
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 ACK
Content-Length: 0

19. ACK NUT -> Proxy 2

ACK sip:UA22@client2.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK5yE0nza
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKiLv329
:received=3ffe:501:fff:1
Max-Forwards: 69
Route: <sip:ss2.biloxi.example.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=r98765
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 ACK
Content-Length: 0

20. BYE Proxy 2 -> NUT

BYE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bKai8mVsO
Via: SIP/2.0/UDP client2.biloxi.example.com:5060;branch=z9hG4bKcei8Ww3
:received=3ffe:501:ffff:20::20
Max-Forwards: 68
Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=r98765
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

21. BYE NUT -> UA11

BYE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKbsO93jH
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bKai8mVsO
:received=3ffe:501:ffff:22::22
Max-Forwards: 68
Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=r98765
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

22. 200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKbsO93jH
Phase 2 Test Specification

SIP IPv6

;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bKai8mVso
;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client2.biloxi.example.com:5060;branch=z9hG4bKcei8Ww3
;received=3ffe:501:ffff:22::22
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=r98765
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

23. 200 OK NUT -> Proxy 2

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bKai8mVso
;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client2.biloxi.example.com:5060;branch=z9hG4bKcei8Ww3
;received=3ffe:501:ffff:22::22
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=r98765
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

24. BYE Proxy 2 -> NUT

BYE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnaahds7
;received=3ffe:501:ffff:2::2
Max-Forwards: 69
Route: <sip:ss.under.test.com;lr>
Record-Route: <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

25. BYE NUT -> UA11
BYE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
    ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
    ;received=3ffe:501:ffff:2::2
Max-Forwards: 68
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

26. 200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
    ;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
    ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
    ;received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0

27. 200 OK NUT -> Proxy 2

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
    ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
    ;received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com;lr>
From: UA21 <sip:UA21@biloxi.example.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
CSeq: 1 BYE
Content-Length: 0
[OBSERVABLE RESULTS]

*1:200 response from NUT to UA11.
   As a SIP Message,
   See generic_message

As a SIP response,

   · Status-Line:
     See generic_forward_from·PX2
     Status-Code: Must be "200". [RFC3261-16-104]

   · Header fields:
     See generic_forward_from·PX2
     See generic_forward_response
     * Via
     via-received: Must be added if the host portion of the "sent-by" parameter
                    contains a domain name. [RFC3261-18-27]
     via-received: Must contain the source address from which the packet was
                    received. [RFC3261-18-28]

   · Bodies:
     See generic_forward_from·PX2

*2:ACK request from NUT to PX2.

As a SIP Message,
   See generic_message

As a SIP request,

   · Request-Line:
     See generic_forward_from·UA11
     See generic_forward_R·URI_non·responsible·domain

   · Header fields:
     · outside of a dialog
     See generic_forward_from·UA11
     See generic_forward_request

   · Bodies:
     See generic_forward_from·UA11
*3: BYE request from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-PX2

- Header fields:
  - outside of a dialog
    See generic_forward_from-PX2
    See generic_forward_request

- Bodies:
  See generic_forward_from-PX2

*4: 200 response from NUT to PX2.

As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA11
  Status-Code: Must be "200". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA11
  See generic_forward_response

- Bodies:
  See generic_forward_from-PX2

*5: BYE request from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP request,
Phase 2 Test Specification
SIP IPv6

- Request-Line:
  See generic_forward_from-PX2

- Header fields:
  - outside of a dialog
    See generic_forward_from-PX2
    See generic_forward_request

- Bodies:
  See generic_forward_from-PX2

*6:200 response from NUT to PX2.
  As a SIP Message,
  See generic_message

  As a SIP response,

  - Status-Line:
    See generic_forward_from-UA11
    Status-Code: Must be "200". [RFC3261-16-104]

  - Header fields:
    See generic_forward_from-UA11
    See generic_forward_response

  - Bodies:
    See generic_forward_from-PX2

[REFERENCE]
[RFC3261-16-92]

16.7 Response Processing

When a response is received by an element, it first tries to locate a client transaction (Section 17.1.3) matching the response. If none is found, the element MUST process the response (even if it is an informational response) as a stateless proxy (described below). If a match is found, the response is handed to the client transaction.

Forwarding responses for which a client transaction (or more generally any knowledge of having sent an associated with request) is not found improves robustness. In particular, it ensures that "late" 2xx responses to INVITE requests are forwarded properly.
4.3.13 FW-2-2-1 - SIP Proxy- Receipt of 503 (Service Unavailable) response

[NAME]
FW-2-2-1 - SIP Proxy- Receipt of 503 (Service Unavailable) response

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT generates a 500 (Server Internal Error) response and forwards that upstream when receiving a 503 (Service Unavailable) response.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip: ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip: reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip: <a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip: <a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip: <a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip: <a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
                     -------
                    |       |
                    |       |
                    |       |
R11                  |
                     |-------
                     |       |
                     |       |
UA11                 |
                     |       |
```

```
                     -------
                    |       |
                    |       |
                    |       |
NUT                    |
                     |       |
                     |       |
Registrar               |
                     |-------
                    |       |
                    |       |
R12                  
```
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
1. UA11 Send INVITE.
2. UA11 Receive 407.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 503 Service Unavailable.
8. UA11 Receive 500 Server Internal Error. (*1)
9. UA11 Send ACK.

--- Message example ---

7.503 Service Unavailable UA12 -> NUT

SIP/2.0 503 Service Unavailable
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
  ;received=3ffe:501:ff:50::50
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bK74bf9
  ;received=3ffe:501::50:2
Record-Route: <sip:ss.under.test.com;lr>
From: UA12 <sip:UA12@under.test.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-Id: 3848276298220188511@under.test.com
Contact: <sip:UA11@node.under.test.com>
Retry-After: 3600
CSeq: 2 INVITE
Content-Length: 0

8.500 Server Internal Error NUT -> UA11

SIP/2.0 500 Server Internal Error
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>
From: UA12 <sip:UA12@under.test.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA11@node.under.test.com>
Retry-After: 3600
CSeq: 2 INVITE
Content-Length: 0

[OBSERVABLE RESULTS]
*1:500 response from NUT to UA11.
   As a SIP Message,
   See generic_message

   As a SIP response,

   · Status-Line:
     See generic_forward_from-UA12
     Status-Code: Must be "500". [RFC3261-16-119]

   · Header fields:
     See generic_forward_from-UA12
     See generic_forward_response
     * Via
       via-received: Must be added if the host portion of the "sent-by" parameter
       contains a domain name. [RFC3261-18-27]
       via-received: Must contain the source address from which the packet was
       received. [RFC3261-18-28]

[REFERENCE]
[RFC3261-16-118, 119]

16.7 Response Processing

   A proxy which receives a 503 (Service Unavailable) response
   SHOULD NOT forward it upstream unless it can determine that any
   subsequent requests it might proxy will also generate a 503.
   In other words, forwarding a 503 means that the proxy knows it
   cannot service any requests, not just the one for the Request-
   URI in the request which generated the 503. If the only
   response that was received is a 503, the proxy SHOULD generate
4.3.14 FW-2-2-2 - SIP Proxy- Receipt of 503 (Service Unavailable) response without Retry-After

**[NAME]**
FW-2-2-2 · SIP Proxy- Receipt of 503 (Service Unavailable) response without Retry-After

**[TARGET]**
SIP Proxy

**[PURPOSE]**
Verify that a NUT generates a 500 (Server Internal Error) response and sends that to that upstream when receiving a 503 (Service Unavailable) response without a Retry-After header field.

**[REQUIREMENT]**
Set up registrar server to use location service, if necessary.

**[PARAMETER]**

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

**[ADDRESS]**

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50:50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50:60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1:1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501:ffff:2:2/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50:1/64</td>
</tr>
</tbody>
</table>

**[TOPOLOGY]**

```
---+-----------+---------
|           |          |
|          UA11 |
R11        |
---+------------+---------
|           |          |
| R------------|         |
|           |          |
```
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11 : NUT : UA12
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 503 Service Unavailable.
8. UA11 Receive 500 Server Internal Error. (*1)
9. UA11 Send ACK.

=== Message example ===

7.503 Service Unavailable UA12 -> NUT

SIP/2.0 503 Service Unavailable
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA12 <sip:UA12@under.test.com>;tag=9fxcde76sl
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA11@node.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

8.500 Server Internal Error NUT -> UA11
SIP/2.0 500 Server Internal Error
Via: SIP/2.0/UDP node.under.test.com;5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:50::2
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=314159
To: UA12 <sip:UA12@under.test.com>;tag=9fced761
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

[OBSERVABLE RESULTS]
*1:500 response from NUT to UA11.

destinatin IP address Must be different from that in 1.INVITE.
As a SIP Message,
    See generic_message
    The destination address shouleld be equal to 3ffe:501:ffff:50::51.
    [RFC3261-21-27]

As a SIP Message,
    See generic_message

As a SIP response,

· Status-Line:
    See generic_forward_from-UA12
    Status-Code: Must be "500". [RFC3261-16-119]

· Header fields:
    See generic_forward_from-UA12
    See generic_forward_response
* Via
    via-received: Must be added if the host portion of the "sent-by" parameter
    contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
    received. [RFC3261-18-28]

[REFERENCE]
[RFC3261-16-118, 119]

16.7 Response Processing
A proxy which receives a 503 (Service Unavailable) response SHOULD NOT forward it upstream unless it can determine that any subsequent requests it might proxy will also generate a 503. In other words, forwarding a 503 means that the proxy knows it cannot service any requests, not just the one for the Request-URI in the request which generated the 503. If the only response that was received is a 503, the proxy SHOULD generate a 500 response and forward that upstream.

4.3.15 FW-2-2-3 - SIP Proxy- Forwarding of INVITE to an alternate server upon receipt of 503 response

[NAME]
FW-2-2-3 - SIP Proxy- Forwarding of INVITE to an alternate server upon receipt of 503 response

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly forwards an INVITE request to an alternate server when receiving a 503 (Service Unavailable) response.

[REQUIREMENT]
Only when a proxy supports DNS.
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>UA12'(IPv6)</td>
<td>3ffe:501:ffff:2::3/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>
[TOPOLOGY]

```
---+-----------+---------
|           |         |
|          UA11         |
R11
---+---R-------+-----------+---------
|           |           |         |
|         NUT       Registrar       |
R12
---+-----------+---------
|           |         |
|           |
UA12
```

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:fff:50::50/64   (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

```plaintext

1. DNS Query
2. DNS Response

ss2.biloxi.example.com resolves to two IP addresses.
(node11.under.test.com 3ffe:501:fff:2::2)
(node11.under.test.com 3ffe:501:fff:2::3)
```
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

1. UA11 Send INVITE.
2. UA12 Receive INVITE.
3. UA12 Send 503 Service Unavailable.
4. UA12' Receive INVITE. (*1)
5. UA12' Send 486 Busy Here.
6. UA12' Receive ACK.
7. UA11 Receive 486 Busy Here.
8. UA11 Send ACK.

=== Message example ===
3. 503 Service Unavailable UA12 -> NUT

SIP/2.0 503 Service Unavailable
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKbf9f45
 ;received=3ffe:501:ffff:50::50
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@biloxi.example.com>;tag=53fHqlQ3
Call-ID: 2xBt9vxSit55XU7p8@under.test.com
Retry-After: 3600
CSeq: 1 INVITE
Content-Length: 0

**precondition for testing this scenario:**
- NUT must resolve PX2 to two IP addresses.

*1:INVITE request from NUT to UA12.'

destination IP address Must be different from that in 1.INVITE.
As a SIP Message,
See generic_message
The destination address should be equal to 3ffe:501:ffff:50::51.
[RFC3261-21-27]

As a SIP Message,
See generic_message

As a SIP request,
- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain

- Header fields:
  - outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

- Bodies:
  See generic_forward_from-UA11
21.5.4 503 Service Unavailable

A client (proxy or UAC) receiving a 503 (Service Unavailable) SHOULD attempt to forward the request to an alternate server. It SHOULD NOT forward any other requests to that server for the duration specified in the Retry-After header field, if present.

4.3.16 FW-2-2-4 - SIP Proxy- Forwarding of INVITE to an alternate server upon receipt of 503 response

[NAME]
FW-2-2-4 · SIP Proxy- Forwarding of INVITE to an alternate server upon receipt of 503 response

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly forwards a 500 (Server Internal Error) response to that upstream when receiving two 503 (Server Internal Error) responses.

[REQUIREMENT]
Only when a proxy supports two-proxy architecture and DNS. Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA21(Contact)</td>
<td>sip:<a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>PX2</td>
<td>sip:ss2.biloxi.example.com:lr</td>
</tr>
<tr>
<td>PX2'</td>
<td>sip:ss2.biloxi.example.com:lr</td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
</tbody>
</table>
### Phase 2 Test Specification

**SIP IPv6**

<table>
<thead>
<tr>
<th>UA21(IPv6)</th>
<th>3ffe:501::ff:22/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>PX2(IPv6)</td>
<td>3ffe:501::ff:20/64</td>
</tr>
<tr>
<td>PX2'(IPv6)</td>
<td>3ffe:501::ff:20/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501::ff:50/64</td>
</tr>
</tbody>
</table>

#### [TopoLory]

```
---+-----------+---------
   |           |         |
   |          UA11          |
   | R           |
   | NUT       Registrar |
   | PX2   PX2'|
   | R13      |
---+-----------+---------
   |           |         |
   |         UA21          |
```

#### [Configuration for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501::ff:50/64 (IPv6)</td>
</tr>
</tbody>
</table>

#### [Initialization]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

<table>
<thead>
<tr>
<th>NUT</th>
<th>R</th>
<th>DNS Server</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

IPv6 FORUM TECHNICAL DOCUMENT
IPv6 Ready Logo Program
Phase 2 Test Specification
SIP IPv6
ss2.biloxi.example.com resolves to two IP addresses.

(ss2.biloxi.example.com 3ffe:501:ffff:20::20)
(ss2.biloxi.example.com 3ffe:501:ffff:20::21)

UA11 R Registrar

1. Send REGISTER Request.
2. Receive 200 OK response.

1. UA11 Send INVITE.
2. PX2 Receive INVITE.
3. PX2 Send 503 Service Unavailable.
4. PX2' Receive INVITE.
5. PX2' Send 503 Service Unavailable.
6. UA11 Receive 500 Server Internal Error. (*1)
7. UA11 Send ACK.

=== Message example ===

3. 503 Service Unavailable PX2 -> NUT
SIP/2.0 503 Service Unavailable
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKbf9f45
:received=3ffe:501:ffff:50::50
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA21 <sip:UA21@biloxi.example.com>;tag=53flq1Q3
Call-ID: 2xTb9xxSit55UX7p8@under.test.com
Retry: 3600
CSeq: 1 INVITE
Content-Length: 0

[OBSERVABLE RESULTS]
** precondition for testing this scenario:
   - NUT must resolve PX2 to two IP addresses.

*1:500 response from NUT to PX2.

destination IP address Must be different from that in 1.INVITE.
   As a SIP Message,
      See generic_message
      The destination address should be equal to 3ffe:501:ffff:50::51.
      [RFC3261-21-27]

As a SIP Message,
   See generic_message

As a SIP response,

   · Status-Line:
      See generic_forward_from-UA12
      Status-Code: Must be "500". [RFC3261-16-118,119]

   · Header fields:
      See generic_forward_from-UA12
      See generic_forward_response
      * Via
         via-received: Must be added if the host portion of the "sent-by" parameter
                     contains a domain name. [RFC3261-18-27]
         via-received: Must contain the source address from which the packet was
                       received. [RFC3261-18-28]

[REFERENCE]
21.5.4 503 Service Unavailable

A client (proxy or UAC) receiving a 503 (Service Unavailable) SHOULD attempt to forward the request to an alternate server. It SHOULD NOT forward any other requests to that server for the duration specified in the Retry-After header field, if present.

4.3.17 FW-3-1-1 - SIP Proxy- Session Establishment Through Two Proxies with a strict router in separate domains (Callee)

(NAME)
FW-3-1-1 - SIP Proxy- Session Establishment Through Two Proxies with a strict router in separate domains (Callee)

(TARGET)
SIP Proxy

(PURPOSE)
Verify that a NUT properly processes when a session through two proxies with a strict router in separate domains is established and a BYE request is forwarded from the strict router.

(REQUIREMENT)
Only when a proxy supports 2-proxy architecture and strict routing.
Set up registrar server to use location service, if necessary.

(PARAMETER)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:ss.under.test.com:lr</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA21(Contact)</td>
<td>sip:<a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>PX2</td>
<td>sip:ss2.biloxi.example.com</td>
</tr>
</tbody>
</table>

(ADDRESS)

<table>
<thead>
<tr>
<th>Address</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff::1/164</td>
</tr>
<tr>
<td>UA21 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
</tbody>
</table>
[TOPOLOGY]

---+-----------+---------
    |           |
    |          UA11
R11
    |
---+---R-------+-----------+---------
    |           |           |
    |         NUT       Registrar
R12
    |
---+-----------+---------
    |           |
    |          PX2
R13
    |
---+----------+
    |           |
    |          UA21

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td>sip: ss.under.test.com:lr</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NUT(IPADDRESS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>---</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1. ICMP Echo Request</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. ICMP Echo Reply</td>
</tr>
<tr>
<td>&lt;-----</td>
<td>---</td>
<td>---------------------</td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.
1. Send REGISTER Request.
2. Receive 200 OK response.

**[PROCEDURE]**

|---------|--------|1. REGISTER |
|         |         |<---------|
|         |         |2. 200 OK |

1. Send REGISTER Request.
2. Receive 200 OK response.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>---------</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>&lt;--------</td>
</tr>
</tbody>
</table>
1. PX2 Send INVITE.
2. UA11 Receive INVITE.
3. PX2 Receive 100 Trying.
4. UA11 Send 180 Ringing.
5. PX2 Receive 180 Ringing.
6. UA11 Send 200 OK.
7. PX2 Receive 200 OK.
8. PX2 Send ACK.
9. UA11 Receive ACK.
10. PX2 Send BYE.
11. UA11 Receive BYE. (*1)
12. UA11 Send 200 OK.
13. PX2 Receive 200 OK.

=== Message example ===

1.INVITE Proxy 2 -> NUT

INVITE sip:UA11@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501::ff:2::2
Max-Forwards: 69
Record-Route: <sip:ss2.biloxi.example.com>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 2 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501::ff:2::2
s=-
c=IN IP6 3ffe:501::ff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

2.INVITE NUT -> UA11

INVITE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
Max-Forwards: 68
Record-Route: <sip:ss.under.test.com;lr>,<sip:ss2.biloxi.example.com>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fced76sl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 2 INVITE
Contact: <sip:UA21@client.biloxi.example.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA21 2890844526 2890844526 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

3.100 Trying NUT -> Proxy 2

SIP/2.0 100 Trying
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 2 INVITE
Content-Length: 0

4.180 Ringing UA11 -> NUT

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
5.180 Ringing NUT -> Proxy 2

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:fff::50
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:fff::2
Record-Route: <sip:ss.under.test.com;lr>,<sip:ss2.biloxi.example.com>
From: UA21 <sip:UA21@biloxi.example.com>:tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>:tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
Contact: <sip:UA11@node.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

6.200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK21e418c4.1
:received=3ffe:501:fff::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:fff::50
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:fff::2
Record-Route: <sip:ss.under.test.com;lr>,<sip:ss2.biloxi.example.com>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA11 2890844527 2890844527 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

7.200 OK NUT -> Proxy 2

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>,
<sip:ss2.biloxi.example.com>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA11 2890844527 2890844527 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
8. ACK Proxy 2 -> NUT

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds87
  :received=3ffe:501:fffe:1::1
Max-Forwards: 69
Route: <UA11@node.under.test.com>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 2 ACK
Content-Length: 0

9. ACK NUT -> UA11

ACK sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
  :received=3ffe:501:fffe:1::1
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds87
  :received=3ffe:501:fffe:1::1
Max-Forwards: 69
Route: <UA11@node.under.test.com>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 2 ACK
Content-Length: 0

/* RTP streams are established between UA11 and UA21 */
/* UA21 Hangs Up with UA11. */

10. BYE Proxy2 -> NUT

BYE sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
  :received=3ffe:501:fffe:1::1
Max-Forwards: 69
Route: <UA11@node.under.test.com>
Record-Route: <sip:ss2.biloxi.example.com>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 3 BYE
Content-Length: 0

11.BYE NUT -> UA11

BYE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKhgtuBft3
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
  ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
  ;received=3ffe:501:ffff:1::1
Max-Forwards: 68
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 3 BYE
Content-Length: 0

12.200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKhgtuBft3
  ;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP ss2.biloxi.example.com:5060;branch=z9hG4bK721e418c4.1
  ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
  ;received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com>
From: UA21 <sip:UA21@biloxi.example.com>
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 3 BYE
Content-Length: 0

13.200 OK NUT -> Proxy2
SIP/2.0 200 OK
Via: SIP/2.0/UDP ss2.under.test.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>, <sip:ss2.under.test.com>
From: UA21 <sip:UA21@biloxi.example.com>;tag=9fxced76sl
To: UA11 <sip:UA11@under.test.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 3 BYE
Content-Length: 0

[OBSERVABLE RESULTS]
*1: BYE request from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP request,
- Request-Line:
  See generic_forward_from-UA11
Request-URI: Must be the URI of first URI from the route set,
  because that does not contain the lr parameter. [RFC3261-8-33]

- Header fields:
  - outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

  * Route: Must not exist. [RFC3261-16-22]

  - Bodies:
    See generic_forward_from-UA11

[REFERENCE]
[RFC3261-16-22, 23]

16.4 Route Information Preprocessing

If the Request-URI of the request contains a value this proxy previously
placed into a Record-Route header field (see Section 16.6 item 4),
the proxy MUST replace the Request-URI in the request with the last value from the Route header field, and remove that value from the Route header field. The proxy MUST then proceed as if it received this modified request.

4.3.18 FW-3-1-2 - SIP Proxy- Session Establishment Through Two Proxies with a strict router in separate domains (Caller)

[NAME]
FW-3-1-2 · SIP Proxy- Session Establishment Through Two Proxies with a strict router in separate domains (Caller)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when a session through two proxies with a strict router in separate domains is established and a BYE request is forwarded to the strict router.

[REQUIREMENT]
Only when a proxy supports 2-proxy architecture and strict routing.
Set up registrar server to use location service, if necessary.

(PARAMETER)

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA21(Contact)</td>
<td>sip:<a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>PX2</td>
<td>sip:ss2.biloxi.example.com</td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501::fff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501::fff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501::fff:1::1/64</td>
</tr>
<tr>
<td>UA21(IPv6)</td>
<td>3ffe:501::fff:2::2/64</td>
</tr>
<tr>
<td>PX2(IPv6)</td>
<td>3ffe:501::fff:20::20/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501::fff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]
[CONFIGURATION for NUT]

NUT sip:ss.under.test.com;lr
NUT(IPADDRESS) 3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. REGISTER
2. 200 OK
1. Send REGISTER Request.
2. Receive 200 OK response.

**[PROCEDURE]**

<table>
<thead>
<tr>
<th>UA21</th>
<th>NUT</th>
<th>PX2</th>
<th>UA11</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>407</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ACK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3. 100</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4. 100</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>5. 180</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>6. 180</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>7. 200</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>8. 200</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>9. ACK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>10. ACK (*1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>Both Way RTP Media</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>11. BYE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>12. BYE (*2)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>13. 200</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>14. 200</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. UA21 Send INVITE.
2. PX2 Receive INVITE.
3. UA21 Receive 100 Trying.
4. PX2 Send 100 Trying.
5. PX2 Send 180 Ringing.
6. UA21 Receive 180 Ringing.
7. PX2 Send 200 OK.
8. UA21 Receive 200 OK.
9. UA21 Send ACK.
10. PX2 Receive ACK. (*1)
11. UA21 Send BYE.
12. PX2 Receive BYE. (*2)
13. PX2 Send 200 OK.
14. UA21 Receive 200 OK.

=== Message example ===

11.BYE UA21 -> NUT

BYE sip:UA21@biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
Route: <sip:ss.under.test.com;lr>, <sip:ss2.biloxi.example.com>
From: UA21 <sip:UA21@under.test.com>;tag=9fxced76sl
To: UA11 <sip:UA11@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 3 BYE
Content-Length: 0

12.BYE NUT -> Proxy 2

BYE sip:ss2.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKhgtuBft3
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Route: <UA11@biloxi.example.com>
Record-Route: <sip:ss.under.test.com;lr>
From: UA21 <sip:UA21@under.test.com>;tag=9fxced76sl
To: UA11 <sip:UA11@biloxi.example.com>;tag=314159
Call-ID: 3848276298220188511@biloxi.example.com
CSeq: 3 BYE
Content-Length: 0

[OBSERVABLE RESULTS]

*1:ACK request from NUT to PX2.
As a SIP Message,
  See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain
  Request-URI: Must equal the last value of Route header field.
  [RFC3261-16-70, RFC3261-16-71]

- Header fields:
  - outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

  * Route: Must equal the Request-URI in "10.ACK".
    [RFC3261-16-70,72]

- Bodies:
  See generic_forward_from-UA11

*2:BYE request from NUT to UA11.

As a SIP Message,
  See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA11
  Request-URI: Must equal the last value of Route header field.
  [RFC3261-16-70, 71]
  Must be the URI of first URI from the route set, because that does not contain the lr parameter. [RFC3261-8-33]

- Header fields:
  - outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request
* Route: Must equal the Request-URI in "10.ACK". [RFC3261-16-70, RFC3261-16-72]

- Bodies:
  See generic_forward_from-UA11

[REFERENCE]
[RFC3261-16-69, 70, 71, 72]
16.6 Request Forwarding

If the copy contains a Route header field, the proxy MUST inspect the URI in its first value. If that URI does not contain an lr parameter, the proxy MUST modify the copy as follows:

- The proxy MUST place the Request-URI into the Route header field as the last value.

- The proxy MUST then place the first Route header field value into the Request-URI and remove that value from the Route header field.

4.3.19 FW-4-1-1 - SIP Proxy- Unsuccessful No Answer with Proxy-Require [CANCEL]

[NAME]
FW-4-1-1 · Unsuccessful No Answer with Proxy-Require [CANCEL]

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT ignores a Proxy-Require header field in the CANCEL request when a UA receives no response and sends a CANCEL request.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]
<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
</tbody>
</table>
UA12(AOR) | sip:UA12@under.test.com
UA12(Contact) | sip:UA12@node11.under.test.com

[ADDRESS]

| NUT (IPv6)       | 3ffe:501:ffff:50::50/64 |
| Registrar (IPv6) | 3ffe:501:ffff:50::60/64 |
| UA11(IPv6)       | 3ffe:501:ffff:1::1/64   |
| UA12(IPv6)       | 3ffe:501:ffff:2::2/64   |
| R(IPv6)          | 3ffe:501:ffff:50::1/64  |

[TOPOLOGY]

---+-----------+---------
<p>| | |
|           |         |
| 1.        | UA11    |
| UA11      |         |
|-----------|---------|---------|
|           |         | NUT     |
|           |         | Registrar|
| 2.        |         |</p>
<table>
<thead>
<tr>
<th>R12</th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>R12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>---------</td>
<td>---------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>UA12</td>
</tr>
</tbody>
</table>

[CONFIGURATION for NUT]

| NUT | sip:ss.under.test.com |
| NUT(IPADDRESS) | 3ffe:501:ffff:50::50/64 (IPv6) |

[INITIALIZATION]

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.
### Phase 2 Test Specification

**SIP IPv6**

<table>
<thead>
<tr>
<th>UA11</th>
<th>UA12</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>---</td>
<td>-----------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>---</td>
<td>-----------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2. 200 OK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>---</td>
<td>-----------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>---</td>
<td>-----------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4. 200 OK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

### [PROCEDURE]

<table>
<thead>
<tr>
<th>UA11 : NUT : UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>------:------:---</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>------:------:---</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>------:------:---</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>------:------:---</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>------:------:---</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>------:------:---</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>------:------:---</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>------:------:---</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>------:------:---</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>------:------:---</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>------:------:---</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>------:------:---</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>------:------:---</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>:      :</td>
</tr>
<tr>
<td>:      :</td>
</tr>
</tbody>
</table>
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA11 Send CANCEL.
10. UA11 Receive 200 OK. (*1)
11. UA12 Receive CANCEL. (*2)
12. UA12 Send 200 OK.
13. UA12 Send 487 Request Terminated.
14. UA12 Receive ACK.
15. UA11 Receive 487 Request Terminated.
16. UA11 Send ACK.

==== Message example ====

9. CANCEL UA11 -> NUT

CANCEL sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Route: <sip:ss.under.test.com;lr>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 CANCEL
Proxy-Require: 100rel
Content-Length: 0

* Contained Proxy-Require header field.

10. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 CANCEL
Content-Length: 0

* Not contained Proxy-Require header field

11. CANCEL NUT -> UA12

CANCEL sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 CANCEL
Content-Length: 0

* Not contained Proxy-Require header field.

[OBSERVABLE RESULTS]
* 1:200 response from NUT to UA11.
   As a SIP Message,
   See generic_message

   As a SIP response,

   · Status-Line:
     See generic_make_response-200_for-CANCEL
     Status-Code: Must be "200". [RFC3261 16.10]

   · Header fields:
     See generic_make_response-200_for-CANCEL
     Must not contain Proxy-Require header field. [RFC3261-8-81]

   * Via
     via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
     via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]
*2:CANCEL request from NUT to UA12.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_make_CANCEL

- Header fields:
  - outside of a dialog
  See generic_make_CANCEL
  Must not contain Proxy-Require header field. [RFC3261-8·81]

- Bodies:
  See generic_make_CANCEL

[REFERENCE]
[RFC3261-8·80, 81]

8 General User Agent Behavior

8.2 UAS Behavior

8.2.2 Header field Inspection

8.2.2.3 Require

Note that Require and Proxy-Require MUST NOT be used in a SIP CANCEL request, or in an ACK request sent for a non-2xx response. These header fields MUST be ignored if they are present in these requests.

4.3.20 FW-4-1-2 - SIP Proxy- processing of CANCEL upon no provisional response

[NAME]
FW-4-1-2 - SIP Proxy- processing of CANCEL upon no provisional response

[TARGET]
SIP Proxy
[PURPOSE]
Verify that a NUT doesn’t send a CANCEL request when no provisional response has been received.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

| NUT(AOR)   | sip:ss.under.test.com |
| Registrar(AOR) | sip:reg.under.test.com |
| UA11(AOR) | sip:UA11@under.test.com |
| UA11(Contact) | sip:UA11@node.under.test.com |
| UA12(AOR) | sip:UA12@under.test.com |
| UA12(Contact) | sip:UA12@node11.under.test.com |

[ADDRESS]

| NUT (IPv6) | 3ffe:501:ffff:50::50/64 |
| Registrar (IPv6) | 3ffe:501:ffff:50::60/64 |
| UA11(IPv6) | 3ffe:501:ffff:1::1/64 |
| UA12(IPv6) | 3ffe:501:ffff:2::2/64 |
| R(IPv6) | 3ffe:501:ffff:50::1/64 |

[TOPOLOGY]

```
+------------+-----------+---------
|           |           |
|           |          UA11 |
|         R11 |           |
|           |           |
+------------+-----------+---------
|           |           |
|         R  |       NUT  | Registrar |
|           |           |
|         R12 |           |
|           |           |
+------------+-----------+---------
|           |           |
|           |          UA12 |
```

[CONFIGURATION for NUT]

| NUT       | sip:ss.under.test.com |
| NUT(IPADDRESS) | 3ffe:501:ffff:50::50/64 (IPv6) |

[INITIALIZATION]
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

<table>
<thead>
<tr>
<th>UA11 :  NUT :  UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td>:      :      :</td>
</tr>
<tr>
<td>:      :      :</td>
</tr>
<tr>
<td>:      :      :</td>
</tr>
<tr>
<td>:      :      :</td>
</tr>
<tr>
<td>:      :      :</td>
</tr>
<tr>
<td>:      :      :</td>
</tr>
<tr>
<td>:      :      :</td>
</tr>
<tr>
<td>:      :      :</td>
</tr>
<tr>
<td>:      :      :</td>
</tr>
<tr>
<td>:      :      :</td>
</tr>
<tr>
<td>:      :      :</td>
</tr>
</tbody>
</table>

1. INVITE
2. 407
3. ACK
4. INVITE
5. INVITE
6. 100
7. CANCEL
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA11 Send CANCEL.
8. UA11 Receive 200 OK. (*1)
9. UA11 Receive 487 Request Terminated.
10. UA11 Send ACK.

--- Message example ---

7. CANCEL UA11 -> NUT

CANCEL sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Route: <sip:ss.under.test.com;lr>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 CANCEL
Content-Length: 0

8. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:f1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 CANCEL
Content-Length: 0
[OBSERVABLE RESULTS]

*1: Must not send CANCEL request. [RFC3261-9-7,8,9]

[REFERENCE]

[RFC3261-9-7, 8, 9, 10, 11]

9 Canceling a Request

9.1 Client Behavior

Once the CANCEL is constructed, the client SHOULD check whether it has received any response (provisional or final) for the request being cancelled (herein referred to as the "original request").

If no provisional response has been received, the CANCEL request MUST NOT be sent; rather, the client MUST wait for the arrival of a provisional response before sending the request. If the original request has generated a final response, the CANCEL SHOULD NOT be sent, as it is an effective no-op, since CANCEL has no effect on requests that have already generated a final response. When the client decides to send the CANCEL, it creates a client transaction for the CANCEL and passes it the CANCEL request along with the destination address, port, and transport. The destination address, port, and transport for the CANCEL MUST be identical to those used to send the original request.

4.4 Forwarding Request

4.4.1 RQ-1-1-1 - SIP Proxy- Receipt of OPTIONS from UAC

[NAME]
RQ-1-1-1 - SIP Proxy- Receipt of OPTIONS from UAC

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT returns a 200 response when receiving an OPTIONS request from a UAC.

[REQUIREMENT]
Only when a proxy understands a OPTIONS request.
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT (AOR)</th>
<th>sip:.ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11 (AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11 (Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12 (AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12 (Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

---+-----------+---------
|           |          |
|           | UA11     |
| R11       |           |
---+---R-------+-----------+---------
|           |           |
|          NUT       Registrar |
| R12       |           |
---+-----------+---------
|           |          |
|           | UA12     |

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip: ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1. ICMP Echo Request</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>----</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

```
|----------|----------| 2. ICMP Echo Reply
 |          |          |
```

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

```
[PROCEDURE]
UA11  :  NUT  :  UA12
| : : |
| : : |
<--------| ------| 1. OPTIONS
| : : |
| : : |
| : : |
<--------| ------| 2. 200 (*1)
| : : |
```

1. UA11 Send OPTIONS.
2. UA11 Receive 200 OK. (*1)

```== Message example ===
1.OPTIONS UA11 -> NUT

OPTIONS sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
```
To: UA12 <sip:UA12@node.under.test.com>
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 1 OPTIONS
Contact: <sip:UA11@node.under.test.com>
Accept: application/sdp
Content-Length: 0

2.200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@node.under.test.com>;tag=314159
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 1 OPTIONS
Accept: application/sdp
Accept-Encoding: gzip
Accept-Language: en
Supported: timer

[OBSERVABLE RESULTS]
*1:200 response from NUT to UA11.
   As a SIP Message,
      See generic_message

   As a SIP response,
      · Status-Line:
         See generic_make_response
         Status-Code: Must be "200". [RFC3261 11.2]

      · Header fields:
         See generic_make_response
         See generic_proxy-auth
      * Via
         via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
         via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

      * Record-Route
Must copy all Record-Route header field values from the request into the response. [RFC3261-12-2]
rec-route: Must maintain the order of Record-Route header field values. [RFC3261-12-3]

* Allow
  Should not exist. [RFC3261-11-5]

* Accept
  Should exist. [RFC3261-11-4]

* Accept-Encoding
  Should exist. [RFC3261-11-4]

* Accept-Language
  Should exist. [RFC3261-11-4]

* Supported
  Should exist. [RFC3261-11-4]

[REFERENCE]
[RFC3261-11-4, 5]

11.2 Processing of OPTIONS Request

If the response to an OPTIONS is generated by a proxy server, the proxy returns a 200 (OK), listing the capabilities of the server. The response does not contain a message body.

Allow, Accept, Accept-Encoding, Accept-Language, and Supported header field fields SHOULD be present in a 200 (OK) response to an OPTIONS request. If the response is generated by a proxy, the Allow header field field SHOULD be omitted as it is ambiguous since a proxy is method agnostic. Contact header fields MAY be present in a 200 (OK) response and have the same semantics as in a 3xx response. That is, they may list a set of alternative names and methods of reaching the user. A Warning header field MAY be present.

4.4.2 RQ-1-1-2 - SIP Proxy- Receipt of OPTIONS when a UAS is ready to accept a call

[NAME]
RQ-1-1-2 - SIP Proxy- Receipt of OPTIONS when a UAS is ready to accept a call
[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT copies the header fields in an OPTIONS request from UAC and doesn’t reorder the ordering of them.

[REQUIREMENT]
Only when a proxy understands a OPTIONS request.
Set up registrar server to use location service, if necessary.

[PARAMETER]
| NUT(AOR)       | sip:ss.under.test.com:lr |
| Registrar(AOR) | sip:reg.under.test.com  |
| UA11(AOR)      | sip:UA11@under.test.com  |
| UA11(Contact)  | sip:UA11@node.under.test.com |
| UA12(AOR)      | sip:UA12@under.test.com  |
| UA12(Contact)  | sip:UA12@node11.under.test.com |

[ADDRESS]
| NUT (IPv6)      | 3ffe:501:ffff:50::50/64 |
| Registrar (IPv6)| 3ffe:501:ffff:50::60/64 |
| UA11(IPv6)      | 3ffe:501:ffff:1::1/64  |
| UA12(IPv6)      | 3ffe:501:ffff:2::2/64  |
| R(IPv6)         | 3ffe:501:ffff:50::1/64 |

[TOPOLOGY]
```
---+-----------+---------
|           |
|          UA11 |
R11        |
---+---R-------+-----------+---------
|           |           |
|         NUT       Registrar |
R12        |
---+-----------+---------
|           |
|          UA12 |
```
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

1. OPTIONS
   2. OPTIONS
   3. 200
   4. 200

IPv6 FORUM TECHNICAL DOCUMENT
IPv6 Ready Logo Program
Phase 2 Test Specification
SIP IPv6
1. UA11 Send OPTIONS.
2. UA12 Receive OPTIONS. (*1)
3. UA12 Send 200 OK.
4. UA11 Receive 200 OK. (*2)

=== Message example ===

1.OPTIONS UA11 -> NUT

```plaintext
OPTIONS sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@atlanta.example.com
CSeq: 1 OPTIONS
Contact: <sip:UA12@node11.under.test.com>
Accept: application/sdp
Content-Length: 0
```

3.200 OK UA12 -> NUT

```plaintext
SIP/2.0 200 OK
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@atlanta.example.com
Contact: <sip:UA12@node11.under.test.com>
Allow: INVITE, ACK, CANCEL, OPTIONS, BYE
Accept: application/sdp
Accept-Encoding: gzip
Accept-Language: en
Supported: timer
Content-Type: application/sdp
Content-Length: 147

v=0
c=IN IP6 3ffe:501:ffff:2::2
```

---

IPv6 FORUM TECHNICAL DOCUMENT
IPv6 Ready Logo Program
Phase 2 Test Specification
SIP IPv6
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

[OBSERVABLE RESULTS]
*1: OPTIONS request from NUT to UA12

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain

- Header fields:
  - outside of a dialog
  See generic_forward_from-UA11
  See generic_forward_request

- Bodies:
  See generic_forward_from-UA11

*2: 200 response from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA12
  Status-Code: Must be "200". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA12
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
received. [RFC3261-18-28]

* Accept
  Must equal as that in the message from UA12. [RFC3261-16-43]

* Allow
  Must equal as that in the message from UA12. [RFC3261-16-43]

* Accept-Encoding
  Must equal as that in the message from UA12. [RFC3261-16-43]

* Accept-Language
  Must equal as that in the message from UA12. [RFC3261-16-43]

* Supported
  Must equal as that in the message from UA12. [RFC3261-16-43]

* Contact
  Must equal as that in the message from UA12. [RFC3261-16-43]

* Bodies:
  See generic_forward_from-UA11

[REFERENCE]
[RFC3261-16-42, 43, 44, 45, 46]
16.6 Request Forwarding

1. Copy request

   The proxy starts with a copy of the received request. The copy
   MUST initially contain all of the header fields from the
   received request. Fields not detailed in the processing
   described below MUST NOT be removed. The copy SHOULD maintain
   the ordering of the header fields as in the received request.
   The proxy MUST NOT reorder field values with a common field
   name (See Section 7.3.1). The proxy MUST NOT add to, modify,
   or remove the message body.

   An actual implementation need not perform a copy; the primary
   requirement is that the processing for each next hop begin with
   the same request.
4.4.3 RQ-2-1-1 - SIP Proxy- Future extension about new header fields

[NAME]
RQ-2-1-1 - SIP Proxy- Future extension about new header fields

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT doesn’t refuse to forward a new header field in a request.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com;lrr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>-----</td>
<td>1. ICMP Echo Request</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. ICMP Echo Reply</td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

<table>
<thead>
<tr>
<th>UA11</th>
<th>UA12</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>-------</td>
<td>----</td>
<td>1. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2. 200 OK</td>
</tr>
<tr>
<td>&lt;------</td>
<td>-------</td>
<td>----</td>
<td>3. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4. 200 OK</td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
|-------:------->|       :        | 4. INVITE
|       :        |-------:------->| 5. INVITE(*1)
|<------:--------|       :        | 6. 100
|       :        |<------:--------| 7. 180
|<------:--------|       :        | 8. 180
|       :        |<------:--------| 9. 200
|<------:--------|       :        |10. 200
|       :        |       :        |
|-------:------->|       :        |11. ACK
|       :        |-------:------->|12. ACK
|       :        |       :        |
|<===============================>| Both Way RTP Media
|       :        |<------:--------|13. BYE
|<------:--------|       :        |14. BYE
|       :        |       :        |
|-------:------->|       :        |15. 200
|       :        |-------:------->|16. 200
|       :        |       :        |

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE. (*1)
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK.
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA12 Send BYE.
14. UA11 Receive BYE.
15. UA11 Send 200.
16. UA12 Receive 200.

== Message example ==

2. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
:received=3ffe:501:ffff:1::1
5. INVITE NUT -> UA12

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
NewHeader field: new
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

[OBSERVABLE RESULTS]
*1:INVITE request from NUT to UA12.

As a SIP Message,
See generic_message
Must not refuse to forward this message. [RFC3261-16-14]

As a SIP request,
- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain

- Header fields:
  - outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

  * NewHeader field
    Should equal that of original message. [RFC3261-16-13]

- Bodies:
  See generic_forward_from-UA11

**[REFERENCE]**
[RFC3261-16-12, 13, 14]

16.3 Request Validation

1. Reasonable syntax check

The request MUST be well-formed enough to be handled with a server transaction. Any components involved in the remainder of these Request Validation steps or the Request Forwarding section MUST be well-formed. Any other components, well-formed or not, SHOULD be ignored and remain unchanged when the message is forwarded.

(snip)

This protocol is designed to be extended. Future extensions may define new methods and header fields at any time. An element MUST NOT refuse to proxy a request because it contains a method or header field it does not know about.

**4.4.4 RQ-2-1-2 - SIP Proxy- Request without a tag in From field**

**[NAME]**
RQ-2-1-2 - SIP Proxy- Request without a tag in From field

**[TARGET]**
SIP Proxy
[PURPOSE]
Verify that a NUT properly processes a request without a tag in a From header field.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node11.under.test.com">UA11@node11.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
  +-----------+---------
  |           |
  |          UA11
  R11       |
  +---R----+-NUT Registrar
  |       |
  |           |
  |          R12
  |       |
  |           |
  |          UA12
```

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

UA11  R  NUT
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11 : NUT : UA12

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------:------:</td>
<td></td>
<td>1. INVITE</td>
</tr>
<tr>
<td>&lt;-------:-------:</td>
<td></td>
<td>2. 407 (*1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------:------:</td>
<td></td>
<td>3. ACK</td>
</tr>
<tr>
<td>&lt;-------:-------:</td>
<td></td>
<td>4. INVITE</td>
</tr>
<tr>
<td></td>
<td>&lt;------:--------:</td>
<td></td>
</tr>
<tr>
<td>&lt;-------:-------:</td>
<td></td>
<td>6. 100 (*3)</td>
</tr>
<tr>
<td>&lt;-------:-------:</td>
<td>&lt;------:--------:</td>
<td></td>
</tr>
<tr>
<td>&lt;-------:-------:</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-------:-------:</td>
<td></td>
<td>&lt;------:--------:</td>
</tr>
</tbody>
</table>
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required. (*1)
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE. (*2)
6. UA11 Receive 100 Trying. (*3)
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing. (*4)
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK. (*5)
11. UA11 Send ACK.
12. UA12 Receive ACK. (*6)
13. UA12 Send BYE.
14. UA11 Receive BYE. (*7)
15. UA11 Send 200 OK.
16. UA12 Receive 200 OK. (*8)

--- Message example ---

1. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
(snip)
From: UA11 <sip:UA11@under.test.com>
To: UA12 <sip:UA12@under.test.com>
(snip)

* No From tag.

2. 407 Proxy Authentication Required NUT -> UA11
SIP/2.0 407 Proxy Authentication Required
(snip)
From: UA11 <sip:UA11@under.test.com>
To: UA12 <sip:UA12@under.test.com>;tag=314159
(snip)

* No From tag.

4.INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
(snip)
From: UA11 <sip:UA11@under.test.com>
To: UA12 <sip:UA12@under.test.com>
(snip)

* No From tag.

5.INVITE NUT -> UA12

INVITE sip:UA12@under.test.com SIP/2.0
(snip)
From: UA11 <sip:UA11@under.test.com>
To: UA12 <UA12@under.test.com>
(snip)

* No From tag.

6.100 Trying NUT -> UA11

SIP/2.0 100 Trying
(snip)
From: UA11 <sip:UA11@under.test.com>
To: UA12 <sip:UA12@under.test.com>;tag=314159
(snip)

* No From tag.

7.180 Ringing UA12 -> NUT

SIP/2.0 180 Ringing
(snip)
From: UA11 <sip:UA11@under.test.com>
To: UA12 <sip:UA12@under.test.com>;tag=314159
(snip)

* No From tag.

8.180 Ringing NUT -> UA11

SIP/2.0 180 Ringing
(snip)
From: UA11 <sip:UA11@under.test.com>
To: UA12 <sip:UA12@under.test.com>;tag=314159
(snip)

* No From tag.

9.200 OK UA12 -> NUT

SIP/2.0 200 OK
(snip)
From: UA11 <sip:UA11@under.test.com>
To: UA12 <sip:UA12@under.test.com>;tag=314159
(snip)

* No From tag.

10.200 OK NUT -> UA11

SIP/2.0 200 OK
(snip)
From: UA11 <sip:UA11@under.test.com>
To: UA12 <sip:UA12@under.test.com>;tag=314159
(snip)

* No From tag.

11. ACK UA11 -> NUT

ACK sip:UA12@under.test.com SIP/2.0
(snip)
From: UA11 <sip:UA11@under.test.com>
To: UA12 <sip:UA12@under.test.com>;tag=314159
(snip)
12. ACK NUT -> UA12

ACK sip:UA12@node11.under.test.com SIP/2.0
(snip)
From: UA11 <sip:UA11@under.test.com>
To: UA12 <sip:UA12@under.test.com>;tag=314159
(snip)

* No From tag.

13. BYE UA12 -> NUT

BYE sip:UA12@node.under.test.com SIP/2.0
(snip)
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>
(snip)

* No To tag.

14. BYE NUT -> UA11

BYE sip:UA11@node.under.test.com SIP/2.0
(snip)
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>
(snip)

* No To tag.

15. 200 OK UA11 -> NUT

SIP/2.0 200 OK
(snip)
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>
(snip)

* No To tag.

16. 200 OK NUT -> UA12
SIP/2.0 200 OK

From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>

* No To tag.

[OBSERVABLE RESULTS]

*1:407 response from NUT to UA11.
   As a SIP Message,
       See generic_message

   As a SIP response,
       · Status-Line:
           See generic_make_response
           Status-Code: Must be "407". [RFC3261 22.3]
       · Header fields:
           See generic_make_response
           * Via
               via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
               via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

       * From
           tag-param: Must be omitted from the From header field. [RFC3261-12-17]

*2:INVITE request from NUT to UA12.

   As a SIP Message,
       See generic_message

   As a SIP request,
       · Request-Line:
           See generic_forward_from-UA11
           See generic_forward_R-URI_non-responsible-domain

       · Header fields:
· outside of a dialog
  See generic_forward_from-UA11
  See generic_forward_request
  * From
    tag-param: Must be omitted from the From header field. [RFC3261-12-17]

· Bodies:
  See generic_forward_from-UA11

*3:100 response from NUT to UA11. (Optional)
  As a SIP Message,
  See generic_message

  As a SIP response,

  · Status-Line:
    See generic_make_response
    Status-Code: Must be "100" [RFC3261 4].

  · Header fields:
    See generic_make_response
    * Via
      via-received: Must be added if the host portion of the "sent-by" parameter
      contains a domain name. [RFC3261-18-27]
      via-received: Must contain the source address from which the packet was
      received. [RFC3261-18-28]

* From
  tag-param: Must be omitted from the From header field. [RFC3261-12-17]

*4:180 response from NUT to UA11.
  As a SIP Message,
  See generic_message

  As a SIP response,

  · Status-Line:
    See generic_forward_from-UA12
    Status-Code: Must be "180". [RFC3261-16-104]

  · Header fields:
    See generic_forward_from-UA12
See generic_forward_response

* Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

* From
  tag-param: Must be omitted from the From header field. [RFC3261-12-17]

- Bodies:
  See generic_forward_from-UA12

*5:200 response from NUT to UA11.
  As a SIP Message,
  See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA12
  Status-Code: Must be "200". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA12
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

* From
  tag-param: Must be omitted from the From header field. [RFC3261-12-17]

- Bodies:
  See generic_forward_from-UA12

*6:ACK request from NUT to UA12.
  As a SIP Message,
  See generic_message
As a SIP request,

- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain

- Header fields:
  - outside of a dialog
  See generic_forward_from-UA11
  See generic_forward_request

  * From
tag-param: Must be omitted from the From header field. [RFC3261-12-17]

- Bodies:
  See generic_forward_from-UA11

*7:BYE request from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA12

- Header fields:
  - outside of a dialog
  See generic_forward_from-UA12
  See generic_forward_request

  * To
tag-param: Must be omitted from the To header field.

- Bodies:
  See generic_forward_from-UA12

*8:200 response from NUT to UA12.

As a SIP Message,
See generic_message
As a SIP response,

- **Status-Line:**
  
  See generic_forward_from-UA11

  Status-Code: Must be "200". [RFC3261-16-104]

- **Header fields:**
  
  See generic_forward_from-UA11

  See generic_forward_response

  * **To**

    tag-param: Must be omitted from the To header field. [RFC3261-16-121, RFC3261-16-122]

- **Bodies:**
  
  See generic_forward_from-UA12

### [REFERENCE]

[RFC3261-12-12, 13, 14, 15, 16, 17]

#### 12.1.1 UAS behavior

The remote sequence number MUST be set to the value of the sequence number in the CSeq header field of the request. The local sequence number MUST be empty. The call identifier component of the dialog ID MUST be set to the value of the Call-ID in the request. The local tag component of the dialog ID MUST be set to the tag in the To field in the response to the request (which always includes a tag), and the remote tag component of the dialog ID MUST be set to the tag from the From field in the request. A UAS MUST be prepared to receive a request without a tag in the From field, in which case the tag is considered to have a value of null.

[RFC3261-16-12, 13]

### 16.3 Request Validation

1. **Reasonable syntax check**

   The request MUST be well-formed enough to be handled with a server transaction. Any components involved in the remainder of these Request Validation steps or the Request Forwarding section MUST be well-formed. Any other components, well-formed or not, SHOULD be ignored and remain unchanged when the message is forwarded. For
instance, an element would not reject a request because of a malformed Date header field. Likewise, a proxy would not remove a malformed Date header field before forwarding a request.

[RFC3261-16-121, 122]

16.7 Response Processing

6. Choosing the best response

1xx and 2xx responses may be involved in the establishment of dialogs. When a request does not contain a To tag, the To tag in the response is used by the UAC to distinguish multiple responses to a dialog creating request. A proxy MUST NOT insert a tag into the To header field of a 1xx or 2xx response if the request did not contain one. A proxy MUST NOT modify the tag in the To header field of a 1xx or 2xx response.

4.4.5 RQ-2-1-3 - SIP Proxy- Response without a tag in To field

[NAME]
RQ-2-1-3 - SIP Proxy- Response without a tag in the To field

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes a response without a tag in a To header field.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
</tbody>
</table>
**Phase 2 Test Specification**

**SIP IPv6**

**UA11(IPv6) 3ffe:501:ffff:1::1/64**

**UA12(IPv6) 3ffe:501:ffff:2::2/64**

**R(IPv6) 3ffe:501:ffff:50::1/64**

---

**[TOPOLOGY]**

```
                     +-----------+---------+
                     |           |         |
                     |          UA11
                     R11
                     +---------R----+
                    |           |         |
                    |         NUT   Registrar
                    R12
                     +---------R----+
                     |           |         |
                     |          UA12
```

---

**[CONFIGURATION for NUT]**

```
NUT sip: ss.under.test.com:lr
NUT(IPADDRESS) 3ffe:501:ffff:50::50/64 (IPv6)
```

---

**[INITIALIZATION]**

```
UA11          R       NUT
   |           |         |
   |           |         |
   |----------|---|-------> 1. ICMP Echo Request
   |           |         |
   |<-------|---|---------| 2. ICMP Echo Reply
   |           |         |
```

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

<table>
<thead>
<tr>
<th>UA11 :</th>
<th>NUT :</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------:------&gt;</td>
<td>:</td>
<td>1. INVITE</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>2. 407</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>3. ACK</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>4. INVITE</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>5. INVITE</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>6. 100</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>7. 180</td>
</tr>
<tr>
<td>&lt;------:-------</td>
<td>:</td>
<td>8. 180 (*1)</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>9. 200</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>10. 200 (*2)</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>11. ACK</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>12. ACK (*3)</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>13. BYE</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>14. BYE (*4)</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td>15. 200</td>
</tr>
<tr>
<td>&lt;--------:------</td>
<td>:</td>
<td>16. 200 (*5)</td>
</tr>
<tr>
<td>:</td>
<td>:</td>
<td></td>
</tr>
</tbody>
</table>

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing. (*1)
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK. (*2)
11. UA11 Send ACK.
12. UA12 Receive ACK. (*3)
13. UA11 Send BYE.
14. UA12 Receive BYE. (*4)
15. UA12 Send 200 OK.
16. UA11 Receive 200 OK. (*5)

=== Message example ===

7. 180 Ringing UA12 -> NUT

SIP/2.0 180 Ringing
(snip)
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
(snip)

* No To tag.

8. 180 Ringing NUT -> UA11

SIP/2.0 180 Ringing
(snip)
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
(snip)

* No To tag.

9. 200 OK UA12 -> NUT

SIP/2.0 200 OK
(snip)
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
(snip)

* No To tag.
10. 200 OK NUT -> UA11

SIP/2.0 200 OK
(snip)
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
(snip)

* No To tag.

11. ACK UA11 -> NUT

ACK sip:UA12@node11.under.test.com SIP/2.0
(snip)
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
(snip)

* No To tag.

12. ACK NUT -> UA12

ACK sip:UA12@node11.under.test.com SIP/2.0
(snip)
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
(snip)

* No To tag.

13. BYE UA11 -> NUT

BYE sip:UA12@node11.under.test.com SIP/2.0
(snip)
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
(snip)

* No To tag.

14. BYE NUT -> UA12
BYE sip:UA12@node11.under.test.com SIP/2.0
(snip)
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
(snip)

* No To tag.

15.200 OK UA12 -> NUT

SIP/2.0 200 OK
(snip)
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
(snip)

* No To tag.

16.200 OK NUT -> UA11

SIP/2.0 200 OK
(snip)
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
(snip)

* No To tag.

[OBSERVABLE RESULTS]

*1:180 response from NUT to UA11.
   As a SIP Message,
      See generic_message

   As a SIP response,

   · Status-Line:
      See generic_forward_from-UA12
      Status-Code: Must be "180".[RFC3261-16-104]

   · Header fields:
      See generic_forward_from-UA12
      See generic_forward_response
      * Via
via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]

via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

* To
tag-param: Must be omitted from the To header field. [RFC3261-16-121, RFC3261-16-122]

- Bodies:
  See generic_forward_from-UA12

*2:200 response from NUT to UA11.
  As a SIP Message,
  See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA12
  Status-Code: Must be "200". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA12
  See generic_forward_response
* Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

* To
tag-param: Must be omitted from the To header field. [RFC3261-16-121, RFC3261-16-122]

- Bodies:
  See generic_forward_from-UA12

*3:ACK request from NUT to UA12.
  As a SIP Message,
  See generic_message
As a SIP request,

- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain

- Header fields:
  - outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

  * From
    tag-param: Must equal that contained in the From header field of "1.INVITE" request. [RFC3261-12-37]

  * To
    tag-param: Must be omitted from the To header field. [RFC3261-16-121, RFC3261-16-122]

- Bodies:
  See generic_forward_from-UA11

*4:BYE request from NUT to UA12.

As a SIP Message,
  See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA12

- Header fields:
  - outside of a dialog
    See generic_forward_from-UA12
    See generic_forward_request

  * To
    tag-param: Must be omitted from the To header field. [RFC3261-16-121, RFC3261-16-122]

- Bodies:
  See generic_forward_from-UA12
5:200 response from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA11
  Status-Code: Must be "200". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA11
  See generic_forward_response

* To
  tag-param: Must be omitted from the To header field.
  [RFC3261-16-121, RFC3261-16-122]

- Bodies:
  See generic_forward_from-UA12

[REFERENCE]
[RFC3261-16-121, 122]

16.7 Response Processing

6. Choosing the best response

1xx and 2xx responses may be involved in the establishment of dialogs. When a request does not contain a To tag, the To tag in the response is used by the UAC to distinguish multiple responses to a dialog creating request. A proxy MUST NOT insert a tag into the To header field of a 1xx or 2xx response if the request did not contain one. A proxy MUST NOT modify the tag in the To header field of a 1xx or 2xx response.

4.4.6 RQ-2-1-4 - SIP Proxy- Unrecognized type of body

[NAME]
RQ-2-1-4 · SIP Proxy- Unrecognized type of body

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly copies the received request and forwards the request when receiving an unrecognized media type of body.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

PARAMETER

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip: ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip: reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip: <a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip: <a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip: <a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip: <a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

ADDRESS

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501::FFFF:50::60/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501::FFFF:60::64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501::FFFF:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501::FFFF:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501::FFFF:50::1/64</td>
</tr>
</tbody>
</table>

TOPOLOGY

```
---+-----------+---------
|           |          |
|          UA11 |
R11       |
---+---R-------+-----------+---------
|           |           |
|         NUT       Registrar |
R12       |
---+-----------+---------
|           |          |
|          UA12 |
```

CONFIGURATION for NUT

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip: ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501::FFFF:50::60/64 (IPv6)</td>
</tr>
</tbody>
</table>
**INITIALIZATION**

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>---&gt;</td>
<td>1. ICMP Echo Request</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;--------</td>
<td>2. ICMP Echo Reply</td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

**PROCEDURE**

<table>
<thead>
<tr>
<th>UA11</th>
<th>UA12</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;----</td>
<td>------</td>
<td>----</td>
<td>-----------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2. 200 OK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;----</td>
<td>------</td>
<td>----</td>
<td>-----------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4. 200 OK</td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required. (*1)
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE. (*2)
6. UA11 Receive 100 Trying.
7. UA12 Send 415 Unsupported Media Type.
8. UA12 Receive ACK.
9. UA11 Receive 415 Unsupported Media Type. (*3)
10. UA11 Send ACK.

=== Message example ===

4. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch;z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Disposition: session;handling=required
Content-Type: unknown
Content-Length: XXX

5. INVITE NUT -> UA12

INVITE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKghnF9tJ9
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Disposition: session;handling=required
Content-Type: unknown
Content-Length: XXX

unknownunknownunknownunknownunknownunknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknownunknownunknownunknownunknownunknown

7. 415 Unsupported Media Type UA12 -> NUT

SIP/2.0 415 Unsupported Media Type
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKghnF9tJ9
received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Accept: application/sdp
Content-Length: 0

9. 415 Unsupported Media Type NUT -> UA11

SIP/2.0 415 Unsupported Media Type
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Accept: application/sdp
Content-Length: 0

[OBSERVABLE RESULTS]
*1:407 response from NUT to UA11.
   As a SIP Message,
   See generic_message
   
   As a SIP response,
   
     - Status-Line:
     See generic_make_response
     Status-Code: Must be "407". [RFC3261 22.3]
   
     - Header fields:
     See generic_make_response
     * Via
       via-received: Must be added if the host portion of the "sent-by" parameter
       contains a domain name. [RFC3261-18-27]
       via-received: Must contain the source address from which the packet was
       received. [RFC3261-18-28]
   
*2:INVITE request from NUT to UA12.

   As a SIP Message,
   See generic_message

   As a SIP request,

     - Request-Line:
     See generic_forward_from-UA11
     See generic_forward_R-URI_non-responsible-domain

     - Header fields:
       - outside of a dialog
         See generic_forward_from-UA11
         See generic_forward_request

       - Bodies:
See generic_forward_from-UA11

*3:415 response from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response
  Status-Code: Must be "415".

- Header fields:
  See generic_make_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter
    contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
    received. [RFC3261-18-28]

* Record-Route
  Must exist.
  MUST copy all Record-Route header field values from the request into the
  response. [RFC3261-12-2]
  rec-route: MUST maintain the order of Record-Route header field values.
  [RFC3261-12-3]

* Accept
  Must equal as that in the message from UA12. [RFC3261-16-43]

[REFERENCE]
[RFC3261-16-42, 43, 44, 45, 46]
16.6 Request Forwarding

1. Copy request

The proxy starts with a copy of the received request. The copy
MUST initially contain all of the header fields from the
received request. Fields not detailed in the processing
described below MUST NOT be removed. The copy SHOULD maintain
the ordering of the header fields as in the received request.
The proxy MUST NOT reorder field values with a common field
name (See Section 7.3.1). The proxy MUST NOT add to, modify, or remove the message body.

An actual implementation need not perform a copy; the primary requirement is that the processing for each next hop begin with the same request.

4.4.7 RQ-2-1-5 - SIP Proxy- Unrecognized encoding of body

[NAME]
RQ-2-1-5 · SIP Proxy- Unrecognized encoding of body

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes an unrecognized encoding of body.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
---+-----------+---------
|           |
|          UA11 |
|           |
|           |
|            |
|            |
| R11 |
```
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501::fff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
[PROCEDURE]

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td>1.</td>
<td>INVITE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. 407 (*1)</td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>------</td>
</tr>
<tr>
<td>3. ACK</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4.</td>
<td>INVITE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>5. INVITE (*2)</td>
<td></td>
</tr>
<tr>
<td>6. 100</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>7. 415</td>
<td></td>
</tr>
<tr>
<td>8. ACK</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>9. 415 (*3)</td>
<td></td>
</tr>
<tr>
<td>10. ACK</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required. (*1)
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE. (*2)
6. UA11 Receive 100 Trying.
7. UA12 Send 415 Unsupported Media Type.
8. UA12 Receive ACK.
9. UA11 Receive 415 Unsupported Media Type. (*3)
10. UA11 Send ACK.

--- Message example ---

4.INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Disposition: session;handling=required
Content-Encoding: unknownEncoding
Content-Length: XXX
5. INVITE NUT -> UA12

INVITE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKghnF9tJ9
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
    received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Disposition: session;handling=required
Content-Encoding: unknownEncoding
Content-Length: XXX

7. 415 Unsupported Media Type UA12 -> NUT

SIP/2.0 415 Unsupported Media Type
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKghnF9tJ9
    received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
    received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Accept-Encoding: gzip
Content-Length: 0

9. 415 Unsupported Media Type NUT -> UA11

SIP/2.0 415 Unsupported Media Type
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Accept-Encoding: gzip
Content-Length: 0

[OBSERVABLE RESULTS]
*1:407 response from NUT to UA11.
   As a SIP Message,
      See generic_message

   As a SIP response,
      · Status-Line:
         See generic_make_response
         Status-Code: Must be "407". [RFC3261 22.3]

      · Header fields:
         See generic_make_response
         * Via
            via-received: Must be added if the host portion of the "sent-by" parameter
            contains a domain name. [RFC3261-18-27]
            via-received: Must contain the source address from which the packet was
            received. [RFC3261-18-28]

*2:INVITE request from NUT to UA12.

   As a SIP Message,
      See generic_message
As a SIP request,

- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain

- Header fields:
  - outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

- Bodies:
  See generic_forward_from-UA11

*3:415 response from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_response
  Status-Code: Must be "415".
- Header fields:
  See generic_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

* Record-Route
  Must exist.
  MUST copy all Record-Route header field values from the request into the response. [RFC3261-12-2]
  rec-route: MUST maintain the order of Record-Route header field values. [RFC3261-12-3]

* Accept-Encoding
  Must equal as that in the message from UA12. [RFC3261-16-43]

[REFERENCE]
1. Copy request

The proxy starts with a copy of the received request. The copy MUST initially contain all of the header fields from the received request. Fields not detailed in the processing described below MUST NOT be removed. The copy SHOULD maintain the ordering of the header fields as in the received request. The proxy MUST NOT reorder field values with a common field name (See Section 7.3.1). The proxy MUST NOT add to, modify, or remove the message body.

An actual implementation need not perform a copy; the primary requirement is that the processing for each next hop begin with the same request.

4.4.8 RQ-2-1-6 - SIP Proxy- Unrecognized language of body

[NAMES]
RQ-2-1-6 - SIP Proxy- Unrecognized language of body

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes a message body in unrecognized language

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

| NUT (IPv6) | 3ffe:501::ffff:50::50/64 |


**Phase 2 Test Specification**

### IPv6

<table>
<thead>
<tr>
<th>Registrar (IPv6)</th>
<th>3ffe:501:ffff:50::60/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

**[TOPOLOGY]**

```
---+-----------+---------
|           |          |
|          UA11 |
R11        |
---+---R-------+---------
|           |          |
|         NUT       Registrar |
R12        |
---+-----------+---------
|           |          |
|          UA12 |
```

**[CONFIGURATION for NUT]**

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64  (IPv6)</td>
</tr>
</tbody>
</table>

**[INITIALIZATION]**

1. **Send ICMP Echo Request.**
2. **Receive ICMP Echo Reply.**

```
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.
```

```
UA11     UA12    R     Registrar
<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>--------</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td></td>
</tr>
</tbody>
</table>
```

```
<--------|---------| 1. REGISTER
|        |         |        |
```

```
<--------|---------| 2. 200 OK
|        |         |        |
```
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

| UA11 | : | NUT | : | UA12 |
|------| : | :   | : |------|
|      | : | :   | : |------|
|<------| : | 1. INVITE | : |------|
|      | : | :   | : |------|
|      | : | 2. 407 (*1) | : |------|
|      | : | :   | : |------|
|      | : | 3. ACK | : |------|
|      | : | :   | : |------|
|      | : | 4. INVITE | : |------|
|      | : | :   | : |------|
|      | : | 5. INVITE (*2) | : |------|
|<------| : | 6. 100 | : |------|
|      | : | :   | : |------|
|      | : | 7. 415 | : |------|
|      | : | :   | : |------|
|      | : | 8. ACK | : |------|
|      | : | :   | : |------|
|      | : | 9. 415 (*3) | : |------|
|      | : | :   | : |------|
|      | : | 10. ACK | : |------|
|      | : | :   | : |------|

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required. (*1)
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE. (*2)
6. UA11 Receive 100 Trying.
7. UA12 Send 415 Unsupported Media Type.
8. UA12 Receive ACK.
9. UA11 Receive 415 Unsupported Media Type. (*3)
10. UA11 Send ACK.

***Message example***

4. INVITE UA11 -> NUT
INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 69
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Disposition: session;handling=required
Content-Language: unknownLanguage
Content-Length: XXX

unknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknown

5. INVITE NUT -> UA12

INVITE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKghnF9tJ9
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
Max-Forwards: 69
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Disposition: session;handling=required
Content-Language: unknownLanguage
Content-Length: XXX

unknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknown
unknownunknownunknownunknownunknownunknown

7. 415 Unsupported Media Type UA12 -> NUT
SIP/2.0 415 Unsupported Media Type
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKghnF9tJ9
  ;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
  ;received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 384827629820188511@under.test.com
CSeq: 2 INVITE
Accept-Language: en
Content-Length: 0

9. 415 Unsupported Media Type NUT -> UA11

SIP/2.0 415 Unsupported Media Type
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
  ;received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 384827629820188511@under.test.com
CSeq: 2 INVITE
Accept-Language: en
Content-Length: 0

[OBSERVABLE RESULTS]
*1:407 response from NUT to UA11.
  As a SIP Message,
    See generic_message
  As a SIP response,
    · Status-Line:
      See generic_make_response
      Status-Code: Must be "407", [RFC3261 22.3]
    · Header fields:
      See generic_make_response
      * Via
        via-received: Must be added if the host portion of the "sent-by" parameter
contains a domain name. [RFC3261-18-27]

via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

*2: INVITE request from NUT to UA12.

As a SIP Message,
See generic_message

As a SIP request,

· Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain

· Header fields:
  · outside of a dialog
  See generic_forward_from-UA11
  See generic_forward_request

· Bodies:
  See generic_forward_from-UA11

*3: 415 response from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP response,

· Status-Line:
  See generic_response
  Status-Code: Must be "415".

· Header fields:
  See generic_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]
* Record-Route
   Must exist.
   MUST copy all Record-Route header field values from the request into the
   response. [RFC3261-12-2]
   rec-route: MUST maintain the order of Record-Route header field values.
   [RFC3261-12-3]

* Accept-Language
   Must equal as that in the message from UA12. [RFC3261-16-43]

[REFERENCE]
[RFC3261-16-42, 43, 44, 45, 46]
16.6 Request Forwarding

1. Copy request

   The proxy starts with a copy of the received request. The copy
   MUST initially contain all of the header fields from the
   received request. Fields not detailed in the processing
described below MUST NOT be removed. The copy SHOULD maintain
the ordering of the header fields as in the received request.
The proxy MUST NOT reorder field values with a common field
name (See Section 7.3.1). The proxy MUST NOT add to, modify,
or remove the message body.

   An actual implementation need not perform a copy; the primary
requirement is that the processing for each next hop begin with
the same request.

4.4.9 RQ-3-1-1 - SIP Proxy- Receipt of BYE with an unacceptable header
   field

[NAME]
RQ-3-1-1 · SIP Proxy- Receipt of BYE with an unacceptable header field

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT ignores and forwards the BYE request, remaining unchanging that
header field when receiving a BYE request with an unacceptable header field.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

**[PARAMETER]**

<table>
<thead>
<tr>
<th>NUT (ProxyServer1/Registrar)</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>UA11 (AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11 (Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12 (AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12 (Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

**[ADDRESS]**

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1:1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501:ffff:2:2/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

**[TOPOLOGY]**

```
---+-----------+---------
     |           |
     |          UA11
R11
---+---R-------+-----------+---------
     |           |           |
     |         NUT       Registrar
R12
---+-----------+---------
     |           |
     |          UA12
```

**[CONFIGURATION for NUT]**

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

**[INITIALIZATION]**

```
UA11  R  NUT
|   |   |   |
|   |   |   |
|   |   |   |
|   |   |   | 1. ICMP Echo Request
|   |   |   |
|<---|---|   | 2. ICMP Echo Reply
```
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

<table>
<thead>
<tr>
<th>UA11</th>
<th>UA12</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------------------</td>
<td>---------</td>
<td>1. REGISTER</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------------------</td>
<td>---------</td>
<td>2. 200 OK</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------------------</td>
<td>---------</td>
<td>3. REGISTER</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------------------</td>
<td>---------</td>
<td>4. 200 OK</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

UA11 : NUT : UA12

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td>:</td>
<td>1. INVITE</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td>:</td>
<td>2. 407</td>
</tr>
<tr>
<td>------:-------&gt;</td>
<td>:</td>
<td>3. ACK</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td>:</td>
<td>4. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>------:-------&gt;</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td>:</td>
<td>6. 100</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td>:</td>
<td>7. 180</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td>:</td>
<td>8. 180</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td>:</td>
<td>9. 200</td>
</tr>
<tr>
<td></td>
<td></td>
<td>------:-------&gt;</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td>:</td>
<td>11. ACK</td>
</tr>
<tr>
<td></td>
<td></td>
<td>------:-------&gt;</td>
</tr>
<tr>
<td>&lt;------------------</td>
<td>---------</td>
<td>Both Way RTP Media</td>
</tr>
</tbody>
</table>
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK.
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA12 Send BYE.
14. UA11 Receive BYE. (*1)
15. UA11 Send 200.
16. UA12 Receive 200.

=== Message example ===

13. BYE UA12 -> NUT

BYE sip:UA11@node.under.test.com SIP/2.0
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
Route: <sip:ss.under.test.com;lr>
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 1 BYE
Content-Length: 0

* Contact header field is "Not applicable" in BYE

[OBSERVABLE RESULTS]

*1: BYE request from NUT to UA11.
As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA12

- Header fields:
  - outside of a dialog
  See generic_forward_from-UA12
  See generic_forward_request
  * Contact
    Should be the same as "13.BYE".[RFC3261-7-12, RFC3261-16-13]

- Bodies:
  See generic_forward_from-UA12

[REFERENCE]
[RFC3261-16-12, 13]

16.3 Request Validation

1. Reasonable syntax check

   The request MUST be well-formed enough to be handled with a server transaction. Any components involved in the remainder of these Request Validation steps or the Request Forwarding section MUST be well-formed. Any other components, well-formed or not, SHOULD be ignored and remain unchanged when the message is forwarded. For instance, an element would not reject a request because of a malformed Date header field. Likewise, a proxy would not remove a malformed Date header field before forwarding a request.

[RFC3261-20-4, 5, 6, 7, 8, 9, 10, 11]

20 Header field Fields

"Optional" means that an element MAY include the header field in a request or response, and a UA MAY ignore the header field if present in the request or response (The exception to this rule is the Require header field discussed in 20.32). A "mandatory" header field MUST be present in a request, and MUST be understood by the UAS receiving the request. A mandatory response header field MUST be present in the
response, and the header field MUST be understood by the UAC processing the response. "Not applicable" means that the header field field MUST NOT be present in a request. If one is placed in a request by mistake, it MUST be ignored by the UAS receiving the request. Similarly, a header field labeled "not applicable" for a response means that the UAS MUST NOT place the header field in the response, and the UAC MUST ignore the header field in the response.

4.4.10 RQ-3-1-2 - SIP Proxy- BYE not matching an existing dialog

[NAME]
RQ·3-1-2 · SIP Proxy· BYE not matching an existing dialog

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when a BYE request doesn’t match an existing dialog.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:60::64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
---+-----------+---------
       |          |
       |          |
       | UA11     |
       |          |
R11   |          |
```
[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
[PROCEDURE]

UA11 : NUT : UA12

|       |       |       |
|-------:------->|       :        | 1. INVITE
|       |       |       |
|<--------:------|       :        | 2. 407
|       |       |       |
|-------:------->|       :        | 3. ACK
|       |       |       |
|-------:------->|       :        | 4. INVITE
|       |       |       |
|<-------------------:--------|       :        | 5. INVITE
|       |       |       |
|<--------:------|       :        | 6. 100
|       |       |       |
|<--------:------|       :        | 7. 180
|       |       |       |
|<--------:------|       :        | 8. 180
|       |       |       |
|<--------:------|       :        | 9. 200
|       |       |       |
|<--------:------|       :        | 10. 200
|       |       |       |
|-------:------->|       :        | 11. ACK
|       |       |       |
|<--------:------|       :        | 12. ACK
|       |       |       |
|<=====================================================================>| Both Way RTP Media
|       |       |       |
|-------:------->|       :        | 13. BYE with another Call-ID
|       |       |       |
|<-------------------:--------|       :        | 14. BYE with another Call-ID (*1)
|       |       |       |
|       |       |       |
|<--------:------|       :        | 15. 481
|       |       |       |
|<--------:------|       :        | 16. ACK
|       |       |       |
|<--------:------|       :        | 17. 481 (*2)
|       |       |       |
|       |       |       |
|<--------:------|       :        | 18. ACK
|       |       |       |
|-------:------->|       :        | 19. BYE without From tag
|       |       |       |
|<-------------------:--------|       :        | 20. BYE without From tag (*3)
|       |       |       |
|       |       |       |
|<--------:------|       :        | 21. 481
|       |       |       |
|<--------:------|       :        | 22. ACK
|       |       |       |
|<--------:------|       :        | 23. 481 (*4)
|       |       |       |
|-------:------->|       :        | 24. ACK
|       |       |       |
|-------:------->|       :        | 25. BYE without To tag
|       |       |       |
|<-------------------:--------|       :        | 26. BYE without To tag (*5)
|       |       |       |
|       |       |       |
|<--------:------|       :        | 27. 481
|       |       |       |
|<--------:------|       :        | 28. ACK
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK.
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA11 Send BYE with another Call-ID.
14. UA12 Receive BYE with another Call-ID. (*1)
15. UA12 Send 481 Call/Transaction Does Not Exist.
16. UA12 Receive ACK.
17. UA11 Receive 481 Call/Transaction Does Not Exist. (*2)
18. UA11 Send ACK.
19. UA11 Send BYE without From tag.
20. UA12 Receive BYE without From tag. (*3)
21. UA12 Send 481 Call/Transaction Does Not Exist.
22. UA12 Receive ACK.
23. UA11 Receive 481 Call/Transaction Does Not Exist. (*4)
24. UA11 Send ACK.
25. UA11 Send BYE without To tag.
26. UA12 Receive BYE without To tag. (*5)
27. UA12 Send 481 Call/Transaction Does Not Exist.
28. UA12 Receive ACK.
29. UA11 Receive 481 Call/Transaction Does Not Exist. (*6)
30. UA11 Send ACK.
31. UA11 Send BYE.
32. UA12 Receive BYE. (*7)
33. UA12 Send 200.
34. UA11 Receive 200. (**8)**

==== Message example ====

4. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1ce41e6cbe5aea9c8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA12@under.test.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501::ffff:1::1
s=
 c=IN IP6 3ffe:501::ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

9. 200 OK UA12 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
 :received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501::ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA12 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

13. BYE UA11 -> NUT

BYE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
Route: <sip:ss.under.test.com;lro>
From: UA11 <sip:UA11@under.test.com>;tag=9fced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: LMjiweinxonYxbq@under.test.com
CSeq: 2 BYE
Content-Length: 0

* Call-ID header field value is different from that in Initial-INVITE.

19. BYE UA11 -> NUT

BYE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
Route: <sip:ss.under.test.com;lro>
From: UA11 <sip:UA11@under.test.com>
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 3 BYE
Content-Length: 0

* No tag in the From header field.

25. BYE UA11 -> NUT

BYE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
Route: <sip:ss.under.test.com;l r>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 4 BYE
Content-Length: 0

* No tag in the To header field.

31. BYE UA12 -> NUT

BYE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under .test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
Route: <sip:ss.under.test.com;l r>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 5 BYE
Content-Length: 0

* This is the genuine BYE request with correct dialog parameters (Call-ID, From tag, To tag).

[OBSERVABLE RESULTS]
*1,*3,*5,*7:BYE request from NUT to UA12.

As a SIP Message,
  See generic_message

As a SIP request,
  - Request-Line:
    See generic_forward_from-UA11

  - Header fields:
    - inside of a dialog
      See generic_forward_from-UA11
      See generic_forward_request

  - Bodies:
    See generic_forward_from-UA11
*2,*4,*6: *481 response from NUT to UA11.
   As a SIP Message,
   See generic_message

   As a SIP response,
   · Status-Line:
     See generic_forward_from-UA12
     Status-Code: Must be "481". [RFC3261 16.7.6], [RFC3261 21.4.19]
   · Header fields:
     See generic_forward_from-UA12
     See generic_forward_response
   · Bodies:
     See generic_forward_from-UA12

*8: *200 response from NUT to UA11.
   As a SIP Message,
   See generic_message

   As a SIP response,
   · Status-Line:
     See generic_forward_from-UA12
     Status-Code: Must be "200" [RFC3261-16-104].
   · Header fields:
     See generic_forward_from-UA12
     See generic_forward_response
   · Bodies:
     See generic_forward_from-UA12

[REFERENCE]
[RFC3261-16-42, 43, 44, 45, 46]
16.6 Request Forwarding

   1. Copy request
The proxy starts with a copy of the received request. The copy MUST initially contain all of the header fields from the received request. Fields not detailed in the processing described below MUST NOT be removed. The copy SHOULD maintain the ordering of the header fields as in the received request. The proxy MUST NOT reorder field values with a common field name (See Section 7.3.1). The proxy MUST NOT add to, modify, or remove the message body.

An actual implementation need not perform a copy; the primary requirement is that the processing for each next hop begin with the same request.

4.4.11 RQ-3-1-3 - SIP Proxy- BYE with a lower CSeq

[NAME]
RQ-3-1-3 - SIP Proxy- BYE with a lower CSeq

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes a BYE request with lower value in a CSeq header field.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
</tbody>
</table>
[TOPOLOGY]

---+-----------+---------
|           |          |
|          UA11 |        |
R11        |
---+---R-------+-----------+---------
|           |           |           |
|         NUT       Registrar |
R12        |
---+-----------+---------
|           |          |
|           |          |
|           |          |
|           |          |
|           |          |

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

Pu11      R      NUT
|        |         |         |
|        |         |         |
|--------|--------|         |
|        |         | 1. ICMP Echo Request |
|        |         |         |
|<-------|---------|         |
|         |         | 2. ICMP Echo Reply |
|         |         |         |

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11   UA12   R   Registrar
|       |       |         |
|       |       |         |
|--------|--------|         |
|--------|--------|         |
|--------|--------|         |
|        |         | 1. REGISTER |
|        |         |         |
|<--------|--------|----------|
|        |         | 2. 200 OK |
|        |         |         |
|        |         | 3. REGISTER |
|        |         |         |
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11 : NUT : UA12

<--------|---------|4. 200 OK

|         |         |
|         |         |
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11 : NUT : UA12

<--------|---------|4. 200 OK

|         |         |
|         |         |
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK.
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA11 Send BYE.
14. UA12 Receive BYE. (*1)
15. UA12 Send 500 Server Internal Error.
16. UA11 Receive 500 Server Internal Error. (*2)
17. UA12 Send BYE.
18. UA11 Receive BYE.
19. UA11 Send 200 OK.
20. UA12 Receive 200 OK.

**[OBSERVABLE RESULTS]**

*1: BYE request from NUT to UA12.

   As a SIP Message,
   See generic_message

   As a SIP request,

   - Request-Line:
     See generic_forward_from-UA11

   - Header fields:
     - inside of a dialog
       See generic_forward_from-UA11
       See generic_forward_request

   - Bodies:
     See generic_forward_from-UA11

*2: 500 response from NUT to UA11.

   As a SIP Message,
   See generic_message
As a SIP response,

- Status-Line:
  See generic_response
  Status-Code: Must be "500". [RFC3261-12-61]

- Header fields:
  See generic_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

  * Record-Route
    Must exist.
    MUST copy all Record-Route header field values from the request into the response. [RFC3261-12-2]
    rec-route: MUST maintain the order of Record-Route header field values. [RFC3261-12-3]

[REFERENCE]
[RFC3261-16-12, 13]
16.3 Request Validation

1. Reasonable syntax check

The request MUST be well-formed enough to be handled with a server transaction. Any components involved in the remainder of these Request Validation steps or the Request Forwarding section MUST be well-formed. Any other components, well-formed or not, SHOULD be ignored and remain unchanged when the message is forwarded. For instance, an element would not reject a request because of a malformed Date header field. Likewise, a proxy would not remove a malformed Date header field before forwarding a request.

4.4.12 RQ-4-1-1 - SIP Proxy- receipt of CANCEL with an unacceptable header field

(NAME)
RQ-4-1-1 - SIP Proxy- receipt of CANCEL with an unacceptable header field
[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT ignores and forwards the CANCEL request, remaining unchanging that header field when receiving a CANCEL request with an unacceptable header field.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:ss.under.test.com:lr</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
---+-----------+---------
   |           |
   |          UA11
   |         R11
---+---R-------+-----------+---------
   |           |           |
   |         NUT       Registrar
   |         R12
---+-----------+---------
   |           |
   |          UA12
```

[CONFIGURATION for NUT]

| NUT                | sip:ss.under.test.com:lr |
NUT(IPADDRESS) 3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]

UA11  R  NUT

| | | |
| | | |
|--------|----> 1. ICMP Echo Request |
| | | |
|<--------|---- 2. ICMP Echo Reply |

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11  UA12  R  Registrar

| | | | |
| | | | |
|------------------------> 1. REGISTER |
| | | |
|<---------------------- 2. 200 OK |
| | | |
|------------> 3. REGISTER |
| | | |
|<-------- 4. 200 OK |

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11  :  NUT  :  UA12

| : | | |
| : | | |
|--------:---> 1. INVITE |
|<--------:--- 2. 407 |
| : | | |
|--------:---> 3. ACK |
| : | | |
|--------:---> 4. INVITE |
| : |--------:---> 5. INVITE |
|<--------:--------| : | 6. 100 |
| : |<--------:--------| : | 7. 180 |
|<--------:--------| : | 8. 180 |
| : | |
| |<--------:--------> | : | 9. CANCEL |
| |<--------:--------> | : | 10. 200 |
| |<--------:--------> | : | |
| | |<--------:--------> | : | 11. CANCEL (*1) |
| |<--------:--------> | : | 12. 200 |
| |<--------:--------> | : | 13. 487 |
|<--------:--------> | : | 14. ACK |
|<--------:--------> | : | 15. 487 |
|<--------:--------> | : | 16. ACK |

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA11 Send CANCEL.
10. UA11 Receive 200 OK.
11. UA12 Receive CANCEL. (*1)
12. UA12 Send 200 OK.
13. UA12 Send 487 Request Terminated.
14. UA12 Receive ACK.
15. UA11 Receive 487 Request Terminated.
16. UA11 Send ACK.

***Message example***

9. CANCEL UA11 -> NUT

CANCEL sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Route: <sip:ss.under.test.com;l r>
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
Contact: <UA11@node.under.test.com>
CSeq: 2 CANCEL
Content-Length: 0

* Contact header field is "Not applicable" in CANCEL

[OBSERVABLE RESULTS]
*1:CANCEL request from NUT to UA12.

As a SIP Message,
  See generic_message

As a SIP request,

  - Request-Line:
    See generic_make_CANCEL

  - Header fields:
    - outside of a dialog
    See generic_make_CANCEL

    * Contact
    Should be the same as "9.CANCEL".[RFC3261-7-12, RFC3261-16-13]

  - Bodies:
    See generic_make_CANCEL

[REFERENCE]
[RFC3261-16-12, 13]

16.3 Request Validation

1. Reasonable syntax check

  The request MUST be well-formed enough to be handled with a server
  transaction. Any components involved in the remainder of these
  Request Validation steps or the Request Forwarding section MUST be
  well-formed. Any other components, well-formed or not, SHOULD be
  ignored and remain unchanged when the message is forwarded. For
instance, an element would not reject a request because of a malformed Date header field. Likewise, a proxy would not remove a malformed Date header field before forwarding a request.

[RFC3261-20-4, 5, 6, 7, 8, 9, 10, 11]
20 Header field Fields

"Optional" means that an element MAY include the header field in a request or response, and a UA MAY ignore the header field if present in the request or response (The exception to this rule is the Require header field discussed in 20.32). A "mandatory" header field MUST be present in a request, and MUST be understood by the UAS receiving the request. A mandatory response header field MUST be present in the response, and the header field MUST be understood by the UAC processing the response. "Not applicable" means that the header field MUST NOT be present in a request. If one is placed in a request by mistake, it MUST be ignored by the UAS receiving the request. Similarly, a header field labeled "not applicable" for a response means that the UAS MUST NOT place the header field in the response, and the UAC MUST ignore the header field in the response.

4.4.13 RQ-4-1-2 - SIP Registrar - Forwarding REGISTER request

[NAME]
RQ-4-1-2 · SIP Registrar/Proxy · Forwarding REGISTER request

[TARGET]
SIP Registrar/Proxy

[PURPOSE]
Verify that a registrar/proxy properly forwards a REGISTER request to an alternate registrar server.

[REQUIREMENT]
Only when the equipment can forward REGISTER requests.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@biloxi.example.com">UA11@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@client.biloxi.example.com">UA11@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>Registrar2</td>
<td>sip:ss2.biloxi.example.com</td>
</tr>
</tbody>
</table>
[ADDRESS]

<table>
<thead>
<tr>
<th>ADDRESS</th>
<th>IPv6 Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
<tr>
<td>Registrar2(IPv6)</td>
<td>3ffe:501:ffff:20::20/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

---+-----------+---------
  |           |        |
  |          UA11 |
  |           |
  R11       R
  |           |
  |          NUT |
  |           |
  R12       R
  |           |
  ---+-----------+---------
  |
  Registrar2

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

UA R NUT
|   |   |   |
|   |   |   |
|   |   |   |
|   |   |   |
|<-----|-------| 1. ICMP Echo Request |
|     |       |   |
|     |------|------> 2. ICMP Echo Reply |
|     |   |   |

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

* NUT is able to forward REGISTER request.

[PROCEDURE]

UA11 : NUT : Registrar2
|   |   |   |
|   |   |   |
|   |   |   |
1. UA11 Send REGISTER.
2. Registrar Receive REGISTER. (⋆1)
3. Registrar Send 401 Unauthorized.
4. UA11 Receive 401 Unauthorized. (⋆2)
5. UA11 Send REGISTER.
6. Registrar Receive REGISTER. (⋆3)
7. Registrar Send 200 OK.
8. UA11 Receive 200 OK. (⋆4)

--- Message example ---

1. REGISTER UA11 -> NUT

**REGISTER** sip:ss2.biloxi.example.com SIP/2.0
**Via:** SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
**Max-Forwards:** 70
**From:** UA11 <sip:UA11@biloxi.example.com>;tag=a73kszfl
**To:** UA11 <sip:UA11@biloxi.example.com>
**Call-ID:** 1j9FpLxk3uxtm8tn@biloxi.example.com
**CSeq:** 1 REGISTER
**Contact:** <sip:UA11@client.biloxi.example.com>
**Expires:** 3600
**Content-Length:** 0

2. REGISTER NUT -> Registrar2

**REGISTER** sip:ss2.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKjy8hGfv
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
From: UA11 <sip:UA11@biloxi.example.com>;tag=a73kszlfl
To: UA11 <sip:UA11@biloxi.example.com>
Call-ID: 1j9FpLxk3uxtm8tn@biloxi.example.com
CSeq: 1 REGISTER
Contact: <sip:UA11@client.biloxi.example.com>
Expires: 3600
Content-Length: 0

3. 401 Unauthorized Registrar2 -> NUT

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKjy8hGfv
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@biloxi.example.com>;tag=a73kszlfl
To: UA11 <sip:UA11@biloxi.example.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@biloxi.example.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="biloxi.example.com", qop="auth",
 nonce="ea9c8e88df84f1ecc4341ae6be5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

4. 401 Unauthorized NUT -> UA11

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds7
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@biloxi.example.com>;tag=a73kszlfl
To: UA11 <sip:UA11@biloxi.example.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@biloxi.example.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="biloxi.example.com", qop="auth",
 nonce="ea9c8e88df84f1ecc4341ae6be5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

5. REGISTER UA11 -> NUT
REGISTER sip:ss2.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@biloxi.example.com>;tag=a73kszlfl
To: UA11 <sip:UA11@biloxi.example.com>
Call-ID: 1j9FpLx3uxtm8tn@biloxi.example.com
CSeq: 2 REGISTER
Contact: <sip:UA11@client.biloxi.example.com>
Expires: 3600
Authorization: Digest username="UA11",
realm="biloxi.example.com",
nonce="ea9c8e88df84f1ce4341ae6cbe5a359",
qop=auth, nc=00000002, cnonce="d4e4ced0",
uri="sip:ss2.biloxi.example.com",
response="b7fd380421ad89263e6774026cfc049"
Content-Length: 0

6. REGISTER NUT -> Registrar2

REGISTER sip:ss2.biloxi.example.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKjy8hGfz
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bKnashds8
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
From: UA11 <sip:UA11@biloxi.example.com>;tag=a73kszlfl
To: UA11 <sip:UA11@biloxi.example.com>
Call-ID: 1j9FpLx3uxtm8tn@biloxi.example.com
CSeq: 2 REGISTER
Contact: <sip:UA11@client.biloxi.example.com>
Expires: 3600
Authorization: Digest username="UA11",
realm="biloxi.example.com",
nonce="ea9c8e88df84f1ce4341ae6cbe5a359",
qop=auth, nc=00000002, cnonce="d4e4ced0",
uri="sip:ss2.biloxi.example.com",
response="b7fd380421ad89263e6774026cfc049"
Content-Length: 0

7. 200 OK Registrar2 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKjy8hGfz
8. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP client.biloxi.example.com:5060;branch=z9hG4bK1ashds8
:received=3ffe:501:ffff:50::50
From: UA11 <sip:UA11@biloxi.example.com>;tag=a73kszlfl
To: UA11 <sip:UA11@biloxi.example.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@biloxi.example.com
CSeq: 2 REGISTER
Contact: <sip:UA11@client.biloxi.example.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

[OBSERVABLE RESULTS]
*1:REGISTER request from NUT to Registrar.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain

- Header fields:
- outside of a dialog
  See generic_forward_from-UA11
  See generic_forward_request

- Bodies:
  See generic_forward_from-UA11
*2: 401 response from NUT to UA11.
   As a SIP Message,
       See generic_message
   As a SIP response,
   · Status-Line:
       See generic_status
       Status-Code: Must be "401". [RFC3261 22.2]
   · Header fields:
       See generic_status
       See generic.www-auth
       * Via
       via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
       via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

*3: REGISTER request from NUT to Registrar.
   As a SIP Message,
       See generic_message
       Should Forward this request to Registrar2. [RFC3261-10-27]
   As a SIP request,
   · Request-Line:
       See generic_forward_from-UA11
       See generic_forward_R-URI_non-responsible-domain
   · Header fields:
       · outside of a dialog
       See generic_forward_from-UA11
       See generic_forward_request
   · Bodies:
       See generic_forward_from-UA11

*4: 200 response from NUT to UA11.
   As a SIP Message,
       See generic_message
As a SIP response,

- **Status-Line:**
  See generic_status
  Status-Code: Must be "200". [RFC3261 4]

- **Header fields:**
  See generic_status
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

  * Date
    Should exist.[RFC3261-10-52]
    rfc1123-date: Must finish with "GMT". [rfc3261 20.17, rfc2616 3.3.1]

  * Contact
    Must exist. [RFC3261-10-50]
    contact-param: Must be the specified parameter. [RFC3261-10-50]
    contact-param: Must be the Contact address of UA11. [RFC3261-10-50]
    contact-param: "*" MUST NOT be used unless the Expires header field is present with a value of "0". [RFC3261-10-15]
    c-p-expires:
    Must exist.[RFC3261-10-51]
    delta-seconds: Must not be "0". [RFC3261 10.2.2]

[REFERENCE]
[RFC3261-10-27]

10.3 Processing REGISTER Requests

1. The registrar inspects the Request-URI to determine whether it has access to bindings for the domain identified in the Request-URI. If not, and if the server also acts as a proxy server, the server SHOULD forward the request to the addressed domain, following the general behavior for proxying messages described in Section 16.
4.5 Forwarding Response

4.5.1 RS-1-1-1 - SIP Proxy - Unrecognized response code (2xx)

[NAME]
RS-1-1-1 · SIP Proxy · Unrecognized response code (2xx)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly forwards an unrecognized response code (2xx) immediately.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
---------+-----------+---------
|           |           |
|          UA11 |
R11
|           |
---------+-----------+---------
|           |           |
|          NUT   Registrar |
R12
|           |
```
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:fff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

**[INITIALIZATION]**

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

1. INVITE
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 299 OK.
10. UA11 Receive 299 OK. (*1)
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA12 Send BYE.
14. UA11 Receive BYE.
15. UA11 Send 200.
16. UA12 Receive 200.

--- Message example ---

9. 299 OK UA12 -> NUT
SIP/2.0 299 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA12 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

10. 299 OK NUT -> UA11

SIP/2.0 299 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA12 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
**[OBSERVABLE RESULTS]**

*1:299 response from NUT to UA11.
   As a SIP Message,
   Must be sent. [RFC3261-16-107]
   See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA12
  Status-Code: Must be "299". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA12
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter
    contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
    received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from-UA12

**[REFERENCE]**

Sequence from RFC3665 Section 3.2.

[RFC3261-16-104]
16.7 Response Processing

5. Check response for forwarding

   Until a final response has been sent on the server transaction,
   the following responses MUST be forwarded immediately:

   - Any provisional response other than 100 (Trying)

   - Any 2xx response

4.5.2 RS-1-1-2 - SIP Proxy- Unrecognized response code (4xx)

**[NAME]**
RS-1-1-2 · SIP Proxy· Unrecognized response code (4xx)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes an unrecognized response code (4xx) as 400 response.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th></th>
<th>sip:<a href="mailto:ua11@under.test.com">ua11@under.test.com</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th></th>
<th>3ffe:501::ff::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPv6)</td>
<td>3ffe:501::ff::50/64</td>
</tr>
<tr>
<td>Registrar(IPv6)</td>
<td>3ffe:501::ff::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501::ff::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501::ff::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501::ff::50/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

---+-----------+---------
 |           |
 |          UA11
 |          |
 R11       
 ---+---R-------+-----------+---------
 |           |           |
 |         NUT       Registrar |
 |          |
 |          |
 R12       
 ---+-----------+---------
 |           |
 |          UA12

[CONFIGURATION for NUT]
[INITIALIZATION]

UA11 R NUT
| | |
| | |
| | |
|<-------- |--------> 1. ICMP Echo Request
| | |
|<-------- |--------> 2. ICMP Echo Reply
| | |

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

[PROCEDURE]

UA11 : NUT : UA12
| : | : |
| | | |
|<-----:--------> 1. REGISTER
| | |
|<-----:--------> 2. 200 OK
| | |
|<-----:--------> 3. REGISTER
| | |
|<-----:--------> 4. 200 OK
| | |

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 499 response.
8. UA12 Receive ACK. (*1)
9. UA11 Receive 499 response. (*2)
10. UA11 Send ACK.

== Message example ==

7. 499 response UA12 -> NUT

SIP/2.0 499 Error
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under .test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

8. ACK NUT -> UA12

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

9. 499 response NUT -> UA11

SIP/2.0 499 Error
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

10. ACK UA11 -> NUT

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

[OBSERVABLE RESULTS]
*1: ACK request from NUT to UA12.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_make_ACK_for-non2XX

- Header fields:
  - outside of a dialog
  See generic_make_ACK_for-non2XX

* To
tag-param: Should equal as that in the original message. [RFC3261-16-123]
- Bodies:
  See generic_make_ACK_for-non2XX

*2:499 response from NUT to UA11.
  As a SIP Message,
  See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response
  Status-Code: Must be "499". [RFC3261 16.7.6]

- Header fields:
  See generic_make_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter
    contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
    received. [RFC3261-18-28]

[REFERENCE]
  Sequence from RFC3665 Section 3.9.

[RFC3261-16-123]
[rfc3261]
16.7 Response Processing

6. Choosing the best response

  3·6xx responses are delivered hop-by-hop. When issuing a 3·6xx
  response, the element is effectively acting as a UAS, issuing
  its own response, usually based on the responses received from
  downstream elements. An element SHOULD preserve the To tag
  when simply forwarding a 3·6xx response to a request that did
  not contain a To tag.

[RFC3261-13-15]
13.2.2.3 4xx, 5xx and 6xx Responses

  A single non·2xx final response may be received for the INVITE. 4xx,
  5xx and 6xx responses may contain a Contact header field value
indicating the location where additional information about the error can be found. Subsequent final responses (which would only arrive under error conditions) MUST be ignored.

All early dialogs are considered terminated upon reception of the non-2xx final response.

After having received the non-2xx final response the UAC core considers the INVITE transaction completed. The INVITE client transaction handles the generation of ACKs for the response (see Section 17).

4.5.3 RS-1-1-3 - SIP Proxy- Unrecognized response code (5xx)

[NAME]
RS-1-1-3 - SIP Proxy- Unrecognized response code (5xx)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes an unrecognized response code (5xx) as 500 response.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1:1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2:2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50:1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 599 response.
8. UA12 Receive ACK. (*1)
9. UA11 Receive 599 response. (*2)
10. UA11 Send ACK.

--- Message example ---

7. 599 response UA12 -> NUT

SIP/2.0 599 Error
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
;received=3ffe:501::50:50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

9. 599 response NUT -> UA11

SIP/2.0 599 Error
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

10. ACK UA11 -> NUT

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

[OBSERVABLE RESULTS]
*1: ACK request from NUT to UA12.
As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_make_ACK_for-non2XX

- Header fields:
  - outside of a dialog
  See generic_make_ACK_for-non2XX

  * To
    tag-param: Should equal as that in the original message. [RFC3261-16-123]

- Bodies:
  See generic_make_ACK_for-non2XX

*2:599 response from NUT to UA11.
As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response
  Status-Code: Must be "599". [RFC3261 16.7.6]

- Header fields:
  See generic_make_response

  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
Sequence from RFC3665 Section 3.9.

[RFC3261·16·123]
16.7 Response Processing
6. Choosing the best response

3-6xx responses are delivered hop-by-hop. When issuing a 3-6xx response, the element is effectively acting as a UAS, issuing its own response, usually based on the responses received from downstream elements. An element SHOULD preserve the To tag when simply forwarding a 3-6xx response to a request that did not contain a To tag.

[RFC3261-13-15]

13.2.2.3 4xx, 5xx and 6xx Responses

A single non-2xx final response may be received for the INVITE. 4xx, 5xx and 6xx responses may contain a Contact header field value indicating the location where additional information about the error can be found. Subsequent final responses (which would only arrive under error conditions) MUST be ignored.

All early dialogs are considered terminated upon reception of the non-2xx final response.

After having received the non-2xx final response the UAC core considers the INVITE transaction completed. The INVITE client transaction handles the generation of ACKs for the response (see Section 17).

4.5.4 RS-1-1-4 - SIP Proxy- Unrecognized response code (6xx)

[NAME]
RS-1-1-4 - SIP Proxy- Unrecognized response code (6xx)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes an unrecognized response code (6xx) as 600 response.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
</tbody>
</table>
[ADDRESS]

<table>
<thead>
<tr>
<th>Address</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
   ---+-----------+---------
   |           |         |
   |          UA11
   R11
   |         |
   ---+---------------
   |       |           |
   |       |           |
   |       NUT Registrar
   R12
   |         |
   ---+-----------+---------
   |           |         |
   |          UA12
```

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>Address</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT</td>
<td>sip:ss.under.test.com:lr</td>
</tr>
<tr>
<td>NUT (IPADDRESS)</td>
<td>3ffe:501:ffff:50::64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

```
UA11 R NUT
<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>1. ICMP Echo Request</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>2. ICMP Echo Reply</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
```

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 699 response.
8. UA12 Receive ACK. (*1)
9. UA11 Receive 699 response. (*2)
10. UA11 Send ACK.

==== Message example ====

7. 699 response UA12 -> NUT

SIP/2.0 699 Error
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
 :received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0

8. ACK NUT -> UA12

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

9. 699 response NUT -> UA11

SIP/2.0 699 Error
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 INVITE
Content-Length: 0
10. ACK UA11 -> NUT

ACK sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@under.test.com
CSeq: 2 ACK
Content-Length: 0

[OBSERVABLE RESULTS]

*1:ACK request from NUT to UA12.

As a SIP Message,
   See generic_message

As a SIP request,

   · Request-Line:
     See generic_make_ACK_for-non2XX

   · Header fields:
     · outside of a dialog
     See generic_make_ACK_for-non2XX

     * To
     tag-param: Should equal as that in the original message. [RFC3261-16-123]

   · Bodies:
     See generic_make_ACK_for-non2XX

*2:699 response from NUT to UA11.

As a SIP Message,
   See generic_message

As a SIP response,

   · Status-Line:
     See generic_make_response
     Status-Code: Must be "699". [RFC3261 16.7.6]
· Header fields:
  See generic_make_response
* Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
Sequence from RFC3665 Section 3.9.

[RFC3261-16-123]

16.7 Response Processing

6. Choosing the best response

3-6xx responses are delivered hop-by-hop. When issuing a 3-6xx response, the element is effectively acting as a UAS, issuing its own response, usually based on the responses received from downstream elements. An element SHOULD preserve the To tag when simply forwarding a 3-6xx response to a request that did not contain a To tag.

[RFC3261-13-15]

13.2.2.3 4xx, 5xx and 6xx Responses

A single non-2xx final response may be received for the INVITE. 4xx, 5xx and 6xx responses may contain a Contact header field value indicating the location where additional information about the error can be found. Subsequent final responses (which would only arrive under error conditions) MUST be ignored.

All early dialogs are considered terminated upon reception of the non-2xx final response.

After having received the non-2xx final response the UAC core considers the INVITE transaction completed. The INVITE client transaction handles the generation of ACKs for the response (see Section 17).

4.5.5 RS-1-1-5 - SIP Proxy- Provisional response other than 100 response
[NAME]
RS-1-1-5 - SIP Proxy - Provisional response other than a 100 response

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT forwards a provisional response other than 100 response immediately.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

| NUT(AOR)     | sip:ss.under.test.com:lr |
|Registrar(AOR)| sip:reg.under.test.com |
|UA11(AOR)     | sip:UA11@under.test.com |
|UA11(Contact) | sip:UA11@node.under.test.com |
|UA12(AOR)     | sip:UA12@under.test.com |
|UA12(Contact) | sip:UA12@node11.under.test.com |

[ADDRESS]

| NUT (IPv6)     | 3ffe:501:ffff:50::50/64 |
|Registrar (IPv6) | 3ffe:501:ffff:50::60/64 |
|UA11(IPv6)      | 3ffe:501:ffff:1::1/64  |
|UA12(IPv6)      | 3ffe:501:ffff:2::2/64  |
|R(IPv6)         | 3ffe:501:ffff:50::1/64 |

[TOPOLOGY]

```
---+-----------+---------
|           |          |
|          UA11 |
R11        |
---+---R-------+-----------+---------
|           |          |
|         NUT       Registrar |
R12        |
---+-----------+---------
|           |          |
|          UA12 |
```
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

UA11  R  NUT

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
</table>
|<--------|        | 1.ICMP Echo Request
|        |        |        |
|        |        | 2.ICMP Echo Reply
|        |        |<--------|

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11  UA12  R  Registrar

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
</table>
|        |        | 1. REGISTER
|-------|-------|--------|--------|
|        |<------| 2. 200 OK
|-------|-------|--------|--------|
|        |       | 3. REGISTER
|        |<------| 4. 200 OK
|        |       |        |        |

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11  :  NUT  :  UA12

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>--------</td>
<td>-------</td>
<td>-------</td>
</tr>
</tbody>
</table>
|        | 1. INVITE
|-------|-------|-------|
|<------| 2. 407
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
</tbody>
</table>
|        | 3. ACK
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>-------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>--------</td>
<td>-------</td>
<td>-------</td>
</tr>
</tbody>
</table>
|        | 4. INVITE
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 199 response. (*1)
8. UA11 Send 199 response. (*1)
9. UA12 Send 180 Ringing.
10. UA11 Receive 180 Ringing.
11. UA12 Send 200 OK.
12. UA11 Receive 200 OK.
13. UA11 Send ACK.
14. UA12 Receive ACK.
15. UA12 Send BYE.
16. UA11 Receive BYE.
17. UA11 Send 200.
18. UA12 Receive 200.

--- Message example ---

7. 199 response UA12 -> NUT
SIP/2.0 199 Going on
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
 :received=3ffe:501:ffff:50:50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

8. 199 response NUT -> UA11

SIP/2.0 199 Going on
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

[OBSERVABLE RESULTS]

*1:199 response from NUT to UA11.

As a SIP Message,
Must be sent. [RFC3261-16-104]
See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA12
  Status-Code: Must be "199". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA12
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter
contains a domain name. [RFC3261-18-27]

via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from-UA12

[REFERENCE]
Sequence from RFC3665 Section 3.2.

[RFC3261-16-104]
16.7 Response Processing

5. Check response for forwarding

Until a final response has been sent on the server transaction, the following responses MUST be forwarded immediately:

- Any provisional response other than 100 (Trying)
- Any 2xx response

4.5.6 RS-1-1-6 - SIP Proxy- Receipt of 200 with an unacceptable header field

[NAME]
RS-1-1-6 - SIP Proxy - Receipt of 200 with an unacceptable header field

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when receiving a 200 response with an acceptable header field.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:ss.under.test.com:lr</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
</tbody>
</table>
[ADDRESS]

<table>
<thead>
<tr>
<th>Address</th>
<th>IPv6 Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe::501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe::501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe::501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe::501:ffff:2::2/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe::501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

---+-----------+---------
|           |
|          UA11
|           |
R11
---+---R-------+-----------+---------
|           |           |
|         NUT       Registrar
|           |
R12
---+-----------+---------
|           |
|           |
|           |
UA12

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>Configuration</th>
<th>IPv6 Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT</td>
<td>sip:ss.under.test.com:lr</td>
</tr>
<tr>
<td>NUT (IPADDRESS)</td>
<td>3ffe::501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>-----</td>
<td>-----</td>
</tr>
<tr>
<td>&lt;-----</td>
<td>-----</td>
<td>1. ICMP Echo Request</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;---</td>
<td>2. ICMP Echo Reply</td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

```
<table>
<thead>
<tr>
<th align="right">UA11 :</th>
<th align="right">NUT :</th>
<th align="right">UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td align="right">:</td>
<td align="right">:</td>
<td align="right">:</td>
</tr>
<tr>
<td align="right">:</td>
<td align="right">:</td>
<td align="right">:</td>
</tr>
<tr>
<td align="right">:</td>
<td align="right">-------:</td>
<td align="right">-------&gt;</td>
</tr>
<tr>
<td align="right">:</td>
<td align="right">&lt;------:</td>
<td align="right"></td>
</tr>
<tr>
<td align="right">-------:</td>
<td align="right">-------:</td>
<td align="right">------&gt; 3. ACK</td>
</tr>
<tr>
<td align="right">:</td>
<td align="right">:</td>
<td align="right">:</td>
</tr>
<tr>
<td align="right">-------:</td>
<td align="right">------&gt; 4. INVITE</td>
<td align="right"></td>
</tr>
<tr>
<td align="right">:</td>
<td align="right">------:</td>
<td align="right">-------&gt;</td>
</tr>
<tr>
<td align="right">:</td>
<td align="right">-------:</td>
<td align="right">------&gt; 6. 100</td>
</tr>
<tr>
<td align="right">:</td>
<td align="right">&lt;-------:</td>
<td align="right"></td>
</tr>
<tr>
<td align="right">-------:</td>
<td align="right">-------:</td>
<td align="right">------&gt; 8. 180</td>
</tr>
<tr>
<td align="right">:</td>
<td align="right">&lt;-------:</td>
<td align="right"></td>
</tr>
<tr>
<td align="right">-------:</td>
<td align="right">------&gt; 10. 200</td>
<td align="right"></td>
</tr>
<tr>
<td align="right">:</td>
<td align="right">:</td>
<td align="right">:</td>
</tr>
<tr>
<td align="right">-------:</td>
<td align="right">------&gt; 11. ACK</td>
<td align="right"></td>
</tr>
<tr>
<td align="right">:</td>
<td align="right">------:</td>
<td align="right">-------&gt;</td>
</tr>
</tbody>
</table>
<------------------------------------------------------------------------> Both Way RTP Media
```

```
|       : |<------:|       | 13. BYE |
|-------:|-------:|------> 14. BYE |
|       : |       : |      : |
|-------:|------> 15. 200 |
|       : |------:|------->| 16. 200 (*1) |
```
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK.
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA12 Send BYE.
14. UA11 Receive BYE.
15. UA11 Send 200.
16. UA12 Receive 200. (*1)

=== Message example ===

15. 200 OK UA11 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK74b43
 :received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnashds7
 :received=3ffe:501:ffff:2::2
From: UA12 <sip:UA12@under.test.com>;tag=314159
To: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA11@node.under.test.com>
CSeq: 1 BYE
Content-Length: 0

* Contact header field is "Not applicable" in 2xx response for BYE

[OBSERVABLE RESULTS]

*1: 200 response from NUT to UA12.
   As a SIP Message,
   See generic_message

   As a SIP response,
- Status-Line:
  See generic_forward_from-UA11
Status-Code: Must be "200". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA11
  See generic_forward_response
  * Contact
    Should be the same as "9.CANCEL". [RFC3261-7-12, RFC3261-16-13]

- Bodies:
  See generic_forward_from-UA12

[REFERENCE]
[RFC3261-16-12, 13]
16.3 Request Validation

1. Reasonable syntax check

The request MUST be well-formed enough to be handled with a server transaction. Any components involved in the remainder of these Request Validation steps or the Request Forwarding section MUST be well-formed. Any other components, well-formed or not, SHOULD be ignored and remain unchanged when the message is forwarded. For instance, an element would not reject a request because of a malformed Date header field. Likewise, a proxy would not remove a malformed Date header field before forwarding a request.

[RFC3261-20-4, 5, 6, 7, 8, 9, 10, 11]
20 Header field Fields

"Optional" means that an element MAY include the header field in a request or response, and a UA MAY ignore the header field if present in the request or response (The exception to this rule is the Require header field discussed in 20.32). A "mandatory" header field MUST be present in a request, and MUST be understood by the UAS receiving the request. A mandatory response header field MUST be present in the response, and the header field MUST be understood by the UAC processing the response. "Not applicable" means that the header field field MUST NOT be present in a request. If one is placed in a request by mistake, it MUST be ignored by the UAS receiving the request. Similarly, a header field labeled "not applicable" for a response means that the UAS MUST NOT place the header field in the
response, and the UAC MUST ignore the header field in the response.

4.6 Forking

4.6.1 FK-1-1-1 - SIP Proxy - Forked request with different Via branch parameters

[NAME]
FK-1-1-1 - SIP Proxy - forked request with different Via branch parameters

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly forks requests with each Via branch parameter.

[REQUIREMENT]
Only when a proxy supports the function of forking.
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
<tr>
<td>UA13(Contact)</td>
<td>sip:<a href="mailto:UA13@node12.under.test.com">UA13@node12.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT(IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>UA13(IPv6)</td>
<td>3ffe:501:ffff:3::3/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

---+-----------+---------
|           |
|          UA11|
|          R11 |
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

* UA12@under.test.com is associated with following two contact URI:
  UA12@node11.under.test.com
  UA13@node12.under.test.com
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
5. Send REGISTER Request.

**[PROCEDURE]**

<table>
<thead>
<tr>
<th align="right">UA11 : NUT : UA12 : UA13</th>
</tr>
</thead>
<tbody>
<tr>
<td align="right">-------------------------</td>
</tr>
<tr>
<td align="right">1. INVITE</td>
</tr>
<tr>
<td align="right">2. 407</td>
</tr>
<tr>
<td align="right">3. ACK</td>
</tr>
<tr>
<td align="right">4. INVITE (*)1</td>
</tr>
<tr>
<td align="right">5. INVITE (*)1</td>
</tr>
<tr>
<td align="right">6. INVITE (*)2</td>
</tr>
<tr>
<td align="right">7. 100</td>
</tr>
<tr>
<td align="right">8. 180</td>
</tr>
<tr>
<td align="right">9. 180 (*)3</td>
</tr>
<tr>
<td align="right">10. 180</td>
</tr>
<tr>
<td align="right">11. 180 (*)4</td>
</tr>
<tr>
<td align="right">12. 200</td>
</tr>
<tr>
<td align="right">13. 200 (*)5</td>
</tr>
<tr>
<td align="right">14. CANCEL (*)6</td>
</tr>
</tbody>
</table>
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE. (*1)
6. UA13 Receive INVITE. (*2)
7. UA11 Receive 100 Trying.
8. UA12 Send 180 Ringing.
9. UA11 Receive 180 Ringing. (*3)
10. UA13 Send 180 Ringing.
11. UA11 Receive 180 Ringing. (*4)
12. UA12 Send 200 OK.
13. UA11 Receive 200 OK. (*5)
14. UA13 Receive CANCEL. (*6)
15. UA13 Send 200 OK.
16. UA13 Send 487 Request Terminated.
17. UA13 Receive ACK.
18. UA11 Send ACK.
19. UA12 Receive ACK.
20. UA12 Send BYE.
21. UA11 Receive BYE.
22. UA11 Send 200 OK.
23. UA12 Receive 200 OK.

=== Message example ===
5. INVITE NUT -> UA12

INVITE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lrc>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

6. INVITE NUT -> UA13

INVITE sip:UA13@node12.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKf9juth0Ighq
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lrc>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

* Via branch parameter is different from that in "5.INVITE".

14. CANCEL NUT -> UA13

CANCEL sip:UA13@node12.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKf9juth0Ighq
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 CANCEL
Content-Type: application/sdp
Content-Length: 0

[OBSERVABLE RESULTS]

*1:INVITE request from NUT to UA12.

As a SIP Message,
See generic_message

As a SIP request,

- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_non-responsible-domain

- Header fields:
  - outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

- Bodies:
  See generic_forward_from-UA11

*2:INVITE request from NUT to UA13.
As a SIP Message,
   See generic_message

As a SIP request,

    · Request-Line:
       See generic_forward_from-UA11
       See generic_forward_R-URI_non-responsible-domain

    · Header fields:
       · outside of a dialog
          See generic_forward_from-UA11
          See generic_forward_request

       * Via
          via-branch: Must be different from that in "5.INVITE". [RFC3261-16-80]

    · Bodies:
       See generic_forward_from-UA11

*3:180 response from NUT to UA11.
   Must be forwarded immediately this response. [RFC3261-16-104]

As a SIP Message,
   See generic_message

As a SIP response,

    · Status-Line:
       See generic_forward_from-UA12
       Status-Code: Must be "180". [RFC3261-16-104]

    · Header fields:
       See generic_forward_from-UA12
       See generic_forward_response

       * Via
          via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
          via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]
- Bodies:
  See generic_forward_from-UA12

*4:180 response from NUT to UA11.
  Must be forwarded immediately this response. [RFC3261-16-104]

As a SIP Message,
  See generic_message

As a SIP response,

  - Status-Line:
    See generic_forward_from-UA12
    Status-Code: Must be "180". [RFC3261-16-104]

  - Header fields:
    See generic_forward_from-UA12
    See generic_forward_response
    * Via
      via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
      via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from-UA12

*5:200 response from NUT to UA11.
  Must be forwarded immediately this response. [RFC3261-16-104]

As a SIP Message,
  See generic_message

As a SIP response,

  - Status-Line:
    See generic_forward_from-UA12
    Status-Code: Must be "200". [RFCF3261-16-104]

  - Header fields:
    See generic_forward_from-UA12
See generic_forward_response
* Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from-UA12

*6:CANCEL request from NUT to UA13.

As a SIP Message,
  Must generate this request. [RFC3261-16-137]
  See generic_message

As a SIP request,

  - Request-Line:
    See generic_make_CANCEL

  - Header fields:
    - outside of a dialog
      See generic_make_CANCEL
      Must be the same dialog ID of original request. [RFC3261-16-2]

  - Bodies:
    See generic_make_CANCEL

[REFERENCE]
[RFC3261-16-80]
16.6 Request Forwarding

  Since each attempt uses a new client transaction, it represents a new branch. Thus, the branch parameter provided with the Via header field inserted in step 8 MUST be different for each attempt.

[RFC3261-16-104, 137]
16.7 Response Processing

  5. Check response for forwarding
Until a final response has been sent on the server transaction, the following responses MUST be forwarded immediately:

- Any provisional response other than 100 (Trying)
- Any 2xx response

10. Generate CANCELS

If the forwarded response was a final response, the proxy MUST generate a CANCEL request for all pending client transactions associated with this response context.

4.6.2 FK-1-1-2 - SIP Proxy- Choice of response to forked request (6xx response)

[NAME]
FK-1-1-2 - SIP Proxy- Choice of response to forked request (6xx response)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly chooses a response in the 6xx class when any response exists.

[REQUIREMENT]
Only when a proxy supports the function of forking.
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:ss.under.test.com;l=lr</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
<tr>
<td>UA13(AOR)</td>
<td>sip:<a href="mailto:UA13@under.test.com">UA13@under.test.com</a></td>
</tr>
<tr>
<td>UA13(Contact)</td>
<td>sip:<a href="mailto:UA13@node12.under.test.com">UA13@node12.under.test.com</a></td>
</tr>
<tr>
<td>UA14(AOR)</td>
<td>sip:<a href="mailto:UA14@under.test.com">UA14@under.test.com</a></td>
</tr>
<tr>
<td>UA14(Contact)</td>
<td>sip:<a href="mailto:UA14@node13.under.test.com">UA14@node13.under.test.com</a></td>
</tr>
</tbody>
</table>
**ADDRESS**

<table>
<thead>
<tr>
<th>Address</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>UA13 (IPv6)</td>
<td>3ffe:501:ffff:3::3/64</td>
</tr>
<tr>
<td>UA14 (IPv6)</td>
<td>3ffe:501:ffff:4::4/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

**TOPOLOGY**

```
---+-----------+---------
 |           |
 |          UA11 |
 |           |
---+---R-------+-----------+---------
 |           |           |
 | RUT       Registrar |
 |           |
---+-----------+---------
 |           |
 |          UA12 |
 |           |
---+-----------+---------
 |           |
 |          UA13 |
 |           |
---+-----------+---------
 |           |
 |          UA14 |
```

**CONFIGURATION for NUT**

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip: ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

**INITIALIZATION**

```
UA11  R  NUT
|   |   |   |
```
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
5. Send REGISTER Request.
7. Send REGISTER Request.

[PROCEDURE]

UA11 : NUT : UA12 UA13 UA14
| : | : | : |
| : | : | : |
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA13 Receive INVITE.
7. UA14 Receive INVITE.
8. UA11 Receive 100 Trying.
9. UA12 Send 500 Server Internal Error.
10. UA14 Send 500 Server Internal Error.
11. UA13 Send 603 Decline.
12. UA13 Receive ACK.
13. 603 (*1)
14. ACK
13. UA11 Receive 603 Decline. (*1)
14. UA11 Send ACK.

=== Message example ===

5. INVITE NUT -> UA12

INVITE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

6. INVITE NUT -> UA13

INVITE sip:UA13@node12.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKjg8UhtEro04mp
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

7. INVITE NUT -> UA14

INVITE sip:UA14@node13.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK7ht6gBvdpqd3h
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

13. 603 Decline NUT -> UA11

SIP/2.0 603 Decline
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

[OBSERVABLE RESULTS]
*1:603 response from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP response,

· Status-Line:
  See generic_forward_from-UA12
  Status-Code: Must be "603". [RFC3261-16-111,112,114,115]

· Header fields:
  See generic_forward_from-UA12
  See generic_forward_response
  Must be the same dialog ID of original request. [RFC3261-16-2]

* Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
[RFC3261-16-114, 115]
16.7 Response Processing

Otherwise, the proxy MUST forward a response from the responses stored in the response context. It MUST choose from the 6xx class responses if any exist in the context. If no 6xx class responses are present, the proxy SHOULD choose from the lowest response class stored in the response context. The proxy MAY select any response within that chosen class.

4.6.3 FK-1-1-3 - SIP Proxy- Choice of response to forked request (The lowest response class)
[NAME]
FK-1-1-3 - SIP Proxy: Choice of response to forked request (The lowest response class)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly chooses a response from the lowest response class when 6xx class responses are not present.

[REQUIREMENT]
Only when a proxy supports the function of forking.
Set up registrar server to use location service, if necessary.

[PARAMETER]
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:ss.under.test.com;l</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
<tr>
<td>UA13(AOR)</td>
<td>sip:<a href="mailto:UA13@under.test.com">UA13@under.test.com</a></td>
</tr>
<tr>
<td>UA13(Contact)</td>
<td>sip:<a href="mailto:UA13@node12.under.test.com">UA13@node12.under.test.com</a></td>
</tr>
<tr>
<td>UA14(AOR)</td>
<td>sip:<a href="mailto:UA14@under.test.com">UA14@under.test.com</a></td>
</tr>
<tr>
<td>UA14(Contact)</td>
<td>sip:<a href="mailto:UA14@node13.under.test.com">UA14@node13.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]
<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>UA13(IPv6)</td>
<td>3ffe:501:ffff:3::3/64</td>
</tr>
<tr>
<td>UA14(IPv6)</td>
<td>3ffe:501:ffff:4::4/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]
```
+----------+
|          |
|          |
|          |
|
R11
```

IPv6 FORUM TECHNICAL DOCUMENT
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
5. Send REGISTER Request.
7. Send REGISTER Request.

[PROCEDURE]

UA11 : NUT : UA12 UA13 UA14

|-----------------------------|--------|3. REGISTER
|<--------------------------------|---------|4. 200 OK
|-----------------------------|--|5. REGISTER
|<--------------------------------|---------|6. 200 OK
|-----------------------------|--|7. REGISTER
|<--------------------------------|---------|8. 200 OK

1. INVITE
2. 407
3. ACK
4. INVITE
5. INVITE
6. INVITE
7. INVITE
8. 100
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA13 Receive INVITE.
7. UA14 Receive INVITE.
8. UA11 Receive 100 Trying.
9. UA13 Send 503 Service Unavailable.
10. UA12 Send 480 Temporarily Unavailable.
11. UA12 receive ACK
12. UA14 Send 486 Busy Here.
13. UA14 receive ACK.
14. UA11 Receive 480 Temporarily Unavailable. (*1)
15. UA11 Send ACK.

=== Message example ===

5. INVITE NUT -> UA12

INVITE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9;
    :received=3ffe:501::ff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

6. INVITE NUT -> UA13

INVITE sip:UA13@node12.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKjg8UhtEro04mp
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

7. INVITE NUT -> UA14

INVITE sip:UA14@node13.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK7ht6gBvdpqd3h
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
I Pv6 FORUM TECHNICAL DOCUMENT

IPv6 Ready Logo Program
Phase 2 Test Specification
SIP IPv6

:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Type: application/sdp
Content-Length: 151

v=0
o=UA11 2890845426 2890845426 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

9. 503 Service Unavailable UA13 -> NUT

SIP/2.0 503 Service Unavailable
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKjg8UhtEro04mp
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=1301
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

10. 480 Temporarily Unavailable UA12 -> NUT

SIP/2.0 480 Temporarily Unavailable
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9f谏ed76sl
To: UA12 <sip:UA12@under.test.com>;tag=1201
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

12. 486 Busy Here UA14 -> NUT

SIP/2.0 486 Busy Here
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK7ht6gBvdpqd3h
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9f谏ed76sl
To: UA12 <sip:UA12@under.test.com>;tag=1401
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

14. 480 Temporarily Unavailable NUT -> UA11

SIP/2.0 480 Temporarily Unavailable
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9f谏ed76sl
To: UA12 <sip:UA12@under.test.com>;tag=1201
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

[OBSERVABLE RESULTS]
*1:4xx response from NUT to UA11.

As a SIP Message,
See generic_message
As a SIP response,

- Status-Line:
  See generic_forward_from-UA12
  Status-Code: Should be "4xx". [RFC3261-16-111,112,116]

- Header fields:
  See generic_forward_from-UA12
  See generic_forward_response
  Must be the same dialog ID of original request. [RFC3261-16-2]

* Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
[RFC3261-116]
16.7 Response Processing

Otherwise, the proxy MUST forward a response from the responses stored in the response context. It MUST choose from the 6xx class responses if any exist in the context. If no 6xx class responses are present, the proxy SHOULD choose from the lowest response class stored in the response context. The proxy MAY select any response within that chosen class.

4.6.4 FK-1-1-4 - SIP Proxy- Stateful proxy receiving a CANCEL request

[NAME]
FK-1-1-4 - SIP Proxy- Stateful proxy receiving a CANCEL request

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly generates CANCEL requests for forking and terminates any pending transactions when receiving a CANCEL request.

[REQUIREMENT]
Only when a proxy supports the function of forking.
Set up registrar server to use location service, if necessary.

**[PARAMETER]**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:ss.under.test.com:lr</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
<tr>
<td>UA13(AOR)</td>
<td>sip:<a href="mailto:UA13@under.test.com">UA13@under.test.com</a></td>
</tr>
<tr>
<td>UA13(Contact)</td>
<td>sip:<a href="mailto:UA13@node12.under.test.com">UA13@node12.under.test.com</a></td>
</tr>
<tr>
<td>UA14(AOR)</td>
<td>sip:<a href="mailto:UA14@under.test.com">UA14@under.test.com</a></td>
</tr>
<tr>
<td>UA14(Contact)</td>
<td>sip:<a href="mailto:UA14@node13.under.test.com">UA14@node13.under.test.com</a></td>
</tr>
</tbody>
</table>

**[ADDRESS]**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:fff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:fff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:fff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:fff:2::2/64</td>
</tr>
<tr>
<td>UA13(IPv6)</td>
<td>3ffe:501:fff:3::3/64</td>
</tr>
<tr>
<td>UA14(IPv6)</td>
<td>3ffe:501:fff:4::4/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:fff:50::1/64</td>
</tr>
</tbody>
</table>

**[TOPOLOGY]**

```
          UA11
         /|
        / |
       UA1

_______________________________________
| | |
| NUT Registrar |
| |
| UA12 |

_______________________________________
| | |
| UA12 |

_______________________________________
| | |
| UA13 |
```
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;---</td>
<td>--</td>
<td>-----</td>
</tr>
<tr>
<td>1.</td>
<td>ICMP Echo Request</td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. ICMP Echo Reply</td>
<td></td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

<table>
<thead>
<tr>
<th>UA11</th>
<th>UA12</th>
<th>UA13</th>
<th>UA14</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>1.</td>
<td>REGISTER</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>2. 200 OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>3. REGISTER</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>4. 200 OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>5. REGISTER</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>6. 200 OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>7. REGISTER</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>8. 200 OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
5. Send REGISTER Request.
7. Send REGISTER Request.

[PROCEDURE]

UA11 : NUT : UA12 UA13 UA14

|       |       |       |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|       :        |       :        |       |       |
|       :        |       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
|-------:------->|       :        |       |       |
|       :        |       :        |       |       |
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA13 Receive INVITE.
7. UA14 Receive INVITE.
8. UA11 Receive 100 Trying.
9. UA12 Send 180 Ringing.
10. UA11 Receive 180 Ringing.
11. UA13 Send 180 Ringing.
12. UA11 Receive 180 Ringing.
13. UA14 Send 180 Ringing.
14. UA11 Receive 180 Ringing.
15. UA11 Send CANCEL.
16. UA11 Receive 200 OK. (*1)
17. UA12 Receive CANCEL. (*2)
18. UA13 Receive CANCEL. (*3)
19. UA14 Receive CANCEL. (*4)
20. UA12 Send 200 OK.
21. UA13 Send 200 OK.
22. UA14 Send 200 OK.
23. UA12 Send 487 Request Terminated.
24. UA12 Receive ACK.
25. UA13 Send 487 Request Terminated.
26. UA13 Receive ACK.
27. UA14 Send 487 Request Terminated.
28. UA14 Receive ACK.
29. UA11 Receive 487 Request Terminated. (*5)
30. UA11 Send ACK.

=== Message example ===
16. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 CANCEL
Contact: <sip:UA12@node11.under.test.com>
Content-Type: application/sdp
Content-Length: 0

17. CANCEL NUT -> UA12

CANCEL sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKf9juth0Ighq
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 CANCEL
Content-Type: application/sdp
18. CANCEL NUT -> UA13

CANCEL sip:UA13@node12.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKjgi9pWqk4fg
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 CANCEL
Content-Type: application/sdp
Content-Length: 0

19. CANCEL NUT -> UA12

CANCEL sip:UA14@node13.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKyUhg9lf7yr
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 CANCEL
Content-Type: application/sdp
Content-Length: 0

[OBSERVABLE RESULTS]

*1:200 response from NUT to UA11.
   As a SIP Message,
   Must immediately return this message. [RFC3261-16-145]
   See generic_message

   As a SIP response,

   · Status-Line:
     Status-Code: Must be "200". [RFC3261-16-104]

   · Header fields:
     * Via
       via-received: Must be added if the host portion of the "sent-by" parameter
       contains a domain name. [RFC3261-18-27]
via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

*2-*4:CANCEL request from NUT to UA12, UA13, UA14.

As a SIP Message,
Must generate for all pending client transaction. [RFC3261-16-144, RFC3261-16-146]
See generic_message

As a SIP request,

- Request-Line:
  See generic_make_CANCEL

- Header fields:
  - outside of a dialog
    See generic_make_CANCEL

- Bodies:
  See generic_make_CANCEL

*5:487 response from NUT to UA11
As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response
  Status-Code: Must be "487". [RFC3261-9-15]

- Header fields:
  See generic_make_response
  Must be the same dialog ID of original request. [RFC3261-16-2]

* Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]
16.10 CANCEL Processing

A stateful proxy MAY generate a CANCEL to any other request it has generated at any time (subject to receiving a provisional response to that request as described in section 9.1). A proxy MUST cancel any pending client transactions associated with a response context when it receives a matching CANCEL request.

(snip)

While a CANCEL request is handled in a stateful proxy by its own server transaction, a new response context is not created for it. Instead, the proxy layer searches its existing response contexts for the server transaction handling the request associated with this CANCEL. If a matching response context is found, the element MUST immediately return a 200 (OK) response to the CANCEL request. In this case, the element is acting as a user agent server as defined in Section 8.2. Furthermore, the element MUST generate CANCEL requests for all pending client transactions in the context as described in Section 16.7 step 10.

4.6.5 FK-1-1-5 - SIP Proxy- Forked request

[NAME]
FK-1-1-5 · SIP Proxy- Forked request

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly generates CANCEL requests for the other forked requests when receiving a 603 (Decline) response.

[REQUIREMENT]
Only when a proxy supports the function of forking.
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11 (Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12 (AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12 (Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
<tr>
<td>UA13 (AOR)</td>
<td>sip:<a href="mailto:UA13@under.test.com">UA13@under.test.com</a></td>
</tr>
<tr>
<td>UA13 (Contact)</td>
<td>sip:<a href="mailto:UA13@node12.under.test.com">UA13@node12.under.test.com</a></td>
</tr>
<tr>
<td>UA14 (AOR)</td>
<td>sip:<a href="mailto:UA14@under.test.com">UA14@under.test.com</a></td>
</tr>
<tr>
<td>UA14 (Contact)</td>
<td>sip:<a href="mailto:UA14@node13.under.test.com">UA14@node13.under.test.com</a></td>
</tr>
</tbody>
</table>

**[ADDRESS]**

| NUT (IPv6)       | 3ffe:501:ffff:50::50/64 |
| Registrar (IPv6) | 3ffe:501:ffff:50::60/64 |
| UA11 (IPv6)      | 3ffe:501:ffff:1::1/64   |
| UA12 (IPv6)      | 3ffe:501:ffff:2::2/64   |
| UA13 (IPv6)      | 3ffe:501:ffff:3::3/64   |
| UA14 (IPv6)      | 3ffe:501:ffff:4::4/64   |
| R (IPv6)         | 3ffe:501:ffff:50::1/64  |

**[TOPOLOGY]**

```
---+-----------+---------
|           |
|          UA11
R11   |
---+---R-------+-----------+---------
|           |           |
|         NUT       Registrar
R12   |
---+-----------+---------
|           |
|          UA12
R13   |
---+-----------+---------
|           |
|          UA13
R14   |
---+-----------+---------
|           |
|          UA14
```
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
5. Send REGISTER Request.
7. Send REGISTER Request.

[PROCEDURE]

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
<th>UA13</th>
<th>UA14</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------:-------&gt;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------:-------&gt;</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----:--------</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
|<-----:-------->|      |      |      |      | 1. INVITE
|<-----:--------|      |      |      |      |
|<-----:--------|      |      |      |      | 6. INVITE
|<-----:--------|      |      |      |      | 7. INVITE
|<-----:--------|      |      |      |      | 8. 100
|<-----:--------|      |      |      |      | 9. 603
|      |     |      |      |      | 10. ACK (*1)
|<-----:--------|      |      |      |      | 11. CANCEL (*2)
|<-----:--------|      |      |      |      | 12. CANCEL (*3)
|<-----:--------|      |      |      |      | 13. 200
|<-----:--------|      |      |      |      | 14. 200
|<-----:--------|      |      |      |      | 15. 487
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA13 Receive INVITE.
7. UA14 Receive INVITE.
8. UA11 Receive 100 Trying.
9. UA13 Send 603 Decline.
10. UA13 Receive ACK. (*1)
11. UA12 Receive CANCEL. (*2)
12. UA14 Receive CANCEL. (*3)
13. UA12 Send 200 OK.
14. UA14 Send 200 OK.
15. UA12 Send 487 Request Terminated.
16. UA12 Receive ACK.
17. UA14 Send 487 Request Terminated.
18. UA14 Receive ACK.
19. UA11 Receive 603 Decline. (*4)
20. UA11 Send ACK.

--- Message example ---

11. CANCEL NUT -> UA12

CANCEL sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKf9juth0Ighq
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 CANCEL
12. CANCEL NUT -> UA14

CANCEL sip:UA14@node13.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bKjg9Uie7y67
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=9fxc676sl
To: UA12 <sip:UA14@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 CANCEL
Content-Type: application/sdp
Content-Length: 0

19. 603 Decline NUT -> UA11

SIP/2.0 603 Decline
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxc676sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Length: 0

[OBSERVABLE RESULTS]
*1 After ACK request from NUT to UA13
Should not immediately forwarded 603 response. [RFC3261-16-105]

*2:CANCEL request from NUT to UA12.
As a SIP Message,
Should generate this request. [RFC3261-16-105, RFC3261-16-138]
See generic_message

As a SIP request,
- Request-Line:
See generic_make_CANCEL

Content-Type: application/sdp
Content-Length: 0
- Header fields:
  - outside of a dialog
    See generic_make_CANCELL

* Via
  via-branch: Must not create any new branches. [RFC3261-16-106]

- Bodies:
  See generic_make_CANCELL

*3: CANCEL request from NUT to UA14.

As a SIP Message,
  Should generate this request. [RFC3261-16-105, RFC3261-16-138]
  See generic_message

As a SIP request,

- Request-Line:
  See generic_make_CANCELL

- Header fields:
  - outside of a dialog
    See generic_make_CANCELL

* Via
  via-branch: Must not create any new branches. [RFC3261-16-106]

- Bodies:
  See generic_make_CANCELL

*4: 603 response from NUT to UA11.

As a SIP Message,
  Should not immediately forward this message. [RFC3261-16-108]
  See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-UA12
  Status-Code: Must be "603". [RFC3261 16.7.5]
- Header fields:
  See generic_forward_from-UA12
  See generic_forward_response
  Must be the same dialog ID of original request. [RFC3261-16-2]

* Via
  via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
  via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
[RFC3261-16-80]
16.6 Request Forwarding

Since each attempt uses a new client transaction, it represents a new branch. Thus, the branch parameter provided with the Via header field inserted in step 8 MUST be different for each attempt.

16.7 Response Processing

5. Check response for forwarding

If a 6xx response is received, it is not immediately forwarded, but the stateful proxy SHOULD cancel all client pending transactions as described in Section 10, and it MUST NOT create any new branches in this context.

(snip)

A stateful proxy MUST NOT immediately forward any other responses. In particular, a stateful proxy MUST NOT forward any 100 (Trying) response. Those responses that are candidates for forwarding later as the "best" response have been gathered as described in step "Add Response to Context".

(snip)

10. Generate CANCELs
If the forwarded response was a final response, the proxy MUST generate a CANCEL request for all pending client transactions associated with this response context. A proxy SHOULD also generate a CANCEL request for all pending client transactions associated with this response context when it receives a 6xx response.

4.7 Transaction

4.7.1 TS-1-1-1 - SIP Proxy- INVITE client transaction (Stop of retransmission of INVITE upon Timer B fired)

[NAME]
TS-1-1-1 - SIP Proxy- INVITE client transaction (Stop of retransmission of INVITE upon Timer B fired)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT stops retransmitting and informs the UAC of failure of forwarding the request.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::30/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
</tbody>
</table>
[TOPOLOGY]

---+-----------+---------
|           |          |
|          UA11 |          |
|           |          |
---+---R-------+---------
|           |          |
|         NUT       Registrar |          |
|           |          |
---+-----------+---------
|           |          |
|          |          |
---+-----------+---------
|           |          |
|          |          |
---+-----------+---------
|           |          |
|          |          |

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

UA11  R  NUT
|    |    |    |
|    |    |    |
|----|----|--->
|    |    |    | 1. ICMP Echo Request
|    |    |    |
|<---|---|    | 2. ICMP Echo Reply
|    |    |    |

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11  UA12  R  Registrar
|    |    |    |    |
|    |    |    |    |
|----|----|----|--->
|    |    |    | 1. REGISTER
|    |    |    |    |
|<---|---|----| 2. 200 OK
|    |    |    |    |
|----|----|----|--->
|    |    |    | 3. REGISTER
|    |    |    |    |
|<---|---|----| 4. 200 OK
|    |    |    |    |
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

Note: This sequence is an example.
The number of retransmission changes depending on a case.

UA11 : NUT : UA12

|        :       |       :        |
|        :       |       :        |
|--------:------>|       :        | 1. INVITE
|        :       |-------:------->| 2. INVITE — Timer B started
|        :       |       :        |
|        :       |-------:------->| 3. 100
|        :       |       :        |
|        :       |-------:------->| 4. INVITE (*1)
|        :       |       :        |
|        :       |-------:------->| 5. INVITE (*2)
|        :       |       :        |
|        :       |-------:------->| 6. INVITE (*3)
|        :       |       :        |
|        :       |-------:------->| 7. INVITE (*4)
|        :       |       :        |
|        :       |-------:------->| 8. INVITE (*5)
|        :       |       :        |
|        :       |-------:------->| 9. INVITE (*6)
|        :       |       :        |
|        :       |-------:------->| Timer B fired
|        :       |       :        |
|<-------:-------|       :        | 10. 480
|        :       |       :        |
|        :       |-------:------->| 11. ACK
|        :       |       :        |

1. UA11 Send INVITE.
2. UA12 Receive INVITE.
3. UA11 Receive 100 Trying.
4. UA12 Receive INVITE. (*1)
5. UA12 Receive INVITE. (*2)
6. UA12 Receive INVITE. (*3)
7. UA12 Receive INVITE. (*4)
8. UA12 Receive INVITE. (*5)
9. UA12 Receive INVITE. (*6)
(*7)
10. UA11 Receive 480 Temporary Unavailable.
11. UA11 Send ACK.

**OBSERVABLE RESULTS**

** this scenario checks only timing (message format is not checked)

*1: INVITE request from NUT to UA12.

Must be retransmitted after Timer A (= T1 sec.) fired.

[RFC3261-17-4][RFC3261-17-7]
Recommended not to be retransmitted with intervals that is shorter than 500msec.

[RFC3261-17-12]

*2: INVITE request from NUT to UA12.

Must be retransmitted with intervals that double after each transmission(2*T1).

[RFC3261-17-8][RFC3261-17-9][RFC3261-17-10][RFC3261-17-14]

*3: INVITE request from NUT to UA12.

Must be retransmitted with intervals that double after each transmission(2*T1).

[RFC3261-17-8][RFC3261-17-9][RFC3261-17-10][RFC3261-17-14]

*4: INVITE request from NUT to UA12.

Must be retransmitted with intervals that double after each transmission(2*T1).

[RFC3261-17-8][RFC3261-17-9][RFC3261-17-10][RFC3261-17-14]

*5: INVITE request from NUT to UA12.

Must be retransmitted with intervals that double after each transmission(2*T1).

[RFC3261-17-8][RFC3261-17-9][RFC3261-17-10][RFC3261-17-14]

*6: INVITE request from NUT to UA12.
Must be retransmitted with intervals that double after each transmission (2*T1).

[RFC3261-17-8][RFC3261-17-9][RFC3261-17-10][RFC3261-17-14]

*7:* INVITE request from NUT to UA12.

INVITE request Must not be retransmitted after Timer B (64*T1) fired.

[RFC3261-17-6][RFC3261-17-11]

Must not send ACK request. [RFC3261-17-16]

[REFERENCE]

[RFC3261-12-56, 57]

12.2.1.2 Processing the Responses

The UAC will receive responses to the request from the transaction layer. If the client transaction returns a timeout, this is treated as a 408 (Request Timeout) response.

If the response for a request within a dialog is a 481 (Call/Transaction Does Not Exist) or a 408 (Request Timeout), the UAC SHOULD terminate the dialog. A UAC SHOULD also terminate a dialog if no response at all is received for the request (the client transaction would inform the TU about the timeout.)

For INVITE initiated dialogs, terminating the dialog consists of sending a BYE.

[RFC3261-17-4, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16]

17.1.1.2 Formal Description

*snip* If an unreliable transport is being used, the client transaction MUST start timer A with a value of T1.

*snip* For any transport, the client transaction MUST start timer B with a value of 64*T1 seconds (Timer B controls transaction timeouts).

When timer A fires, the client transaction MUST retransmit the request by passing it to the transport layer, and MUST reset the timer with a value of 2*T1. The formal definition of retransmit within the context of the transaction layer is to take the message previously sent to the transport layer and pass it to the transport layer once more.
When timer A fires 2*T1 seconds later, the request MUST be retransmitted again (assuming the client transaction is still in this state). This process MUST continue so that the request is retransmitted with intervals that double after each transmission. These retransmissions SHOULD only be done while the client transaction is in the "calling" state.

*snip*

The default value for T1 is 500 ms. T1 is an estimate of the RTT between the client and server transactions. Elements MAY (though it is NOT RECOMMENDED) use smaller values of T1 within closed, private networks that do not permit general Internet connection. T1 MAY be chosen larger, and this is RECOMMENDED if it is known in advance (such as on high latency access links) that the RTT is larger. Whatever the value of T1, the exponential backoffs on retransmissions described in this section MUST be used.

If the client transaction is still in the "Calling" state when timer B fires, the client transaction SHOULD inform the TU that a timeout has occurred. The client transaction MUST NOT generate an ACK. The value of 64*T1 is equal to the amount of time required to send seven requests in the case of an unreliable transport.

4.7.2 TS-1-1-2 - SIP Proxy- INVITE client transaction (Stop of retransmission upon Receipt of 180 Ringing)

[NAME]
TS-1-1-2 · SIP Proxy- INVITE client transaction (Stop of retransmission upon Receipt of 180 Ringing)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT stops retransmitting when receiving a 180 Ringing.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

| NUT(AOR) | sip:ss.under.test.com:lr |

_______________________________________________________________________________________
**Phase 2 Test Specification**

### SIP IPv6

<table>
<thead>
<tr>
<th>Registrar(AOR)</th>
<th>sip:reg.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

### ADDRESS

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

### TOPOLOGY

```
+----------------+-----------------+---------------------+---------------------+
|                | NUT              | Registrar           |
|                | UA11             | R11                 |
|                | UA11             | R12                 |
|                | R12              | UA12                |
```

### CONFIGURATION for NUT

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com;lre</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

### INITIALIZATION

```
UA11  R  NUT
|     |    |    |
|     |    |    |
|     | <---|    |
```

1. Send ICMP Echo Request.

---

IPv6 FORUM TECHNICAL DOCUMENT
IPv6 Ready Logo Program
Phase 2 Test Specification
SIP IPv6
2. Receive ICMP Echo Reply.

```
2. Receive ICMP Echo Reply.

<table>
<thead>
<tr>
<th>UA11</th>
<th>UA12</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>------</td>
<td>-----</td>
<td>----------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&lt;------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2. 200 OK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>&lt;------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4. 200 OK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
```

[PROCEDURE]

Note: This sequence is an example.
The number of retransmission changes depending on a case.

```
<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>--&lt;--</td>
<td>------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>1. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. INVITE --Timer B started</td>
</tr>
<tr>
<td>&lt;-----</td>
<td>--&lt;--</td>
<td>------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. 100</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;-----</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;-----</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7. 180</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>&lt;-----</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8. 180</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. UA11 Send INVITE.
```
2. UA12 Receive INVITE.
3. UA11 Receive 100 Trying.
4. UA12 Receive INVITE.
5. UA12 Receive INVITE.
6. UA12 Receive INVITE.
7. UA12 Send 180 Ringing.

(*)
8. UA11 Receive 180 Ringing.

**[OBSERVABLE RESULTS]**

** this scenario checks only timing (message format is not checked)

*1:after 180 response from UA12 to NUT.
Should not retransmit INVITE request. [RFC3261-17-11][RFC3261-17-17]

**[REFERENCE]**

[RFC3261-17-11, 12, 17]
17.1.1.2 Formal Description

*snip*
These retransmissions SHOULD only be done while the client transaction is in the "calling" state.

If the client transaction receives a provisional response while in the "Calling" state, it transitions to the "Proceeding" state. In the "Proceeding" state, the client transaction SHOULD NOT retransmit the request any longer. *snip*

**4.7.3 TS-1-1-3 - SIP Proxy- INVITE client transaction (Stop of ACK upon Timer D fired)**

**[NAME]**
TS-1-1-3 - SIP Proxy- INVITE client transaction (Stop of ACK upon Timer D fired)

**[TARGET]**
SIP Proxy

**[PURPOSE]**
Verify that a NUT stops sending an ACK for 300-699 response when Timer D is fired.

**[REQUIREMENT]**
Set up registrar server to use location service, if necessary.

**[PARAMETER]**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:ss.under.test.com:lr</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

**[ADDRESS]**

<table>
<thead>
<tr>
<th>Address</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501::50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501::50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501::1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501::2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501::50::1/64</td>
</tr>
</tbody>
</table>

**[TOPOLOGY]**

```
---+-----------+---------
|           |         |
|          UA11|
R11       |
---+---R-------+-----------+---------
|           |         |
|         NUT       Registrar|
R12       |
---+-----------+---------
|           |         |
|           |
UA12       
```

**[CONFIGURATION for NUT]**

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501::50::50/64 IPv6</td>
</tr>
</tbody>
</table>

**[INITIALIZATION]**

```
UA11      R      NUT
|        |         |         |
|        |         |         |
|--------|-------->| 1.ICMP Echo Request
|        |         |         |
```

IPv6 FORUM TECHNICAL DOCUMENT  IPv6 Ready Logo Program
Phase 2 Test Specification  SIP IPv6
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]
Note: This sequence is an example.
The number of retransmission changes depending on a case.

1. INVITE
2. INVITE — Timer B started
3. 100
4. INVITE
5. INVITE
6. INVITE
7. 486
8. ACK — Timer D has started
1. UA11 Send INVITE.
2. UA12 Receive INVITE.
3. UA11 Receive 100 Trying.
4. UA12 Receive INVITE.
5. UA12 Receive INVITE.
6. UA12 Receive INVITE.
7. UA12 Send 486 Busy Here.
8. UA12 Receive ACK.
9. UA11 Receive 486 Busy Here.
10. UA11 Send ACK.
11. UA12 Send 486 Busy Here.
12. UA12 Receive ACK. (*1)
13. UA12 Send 486 Busy Here. (*2)

[OBSERVABLE RESULTS]
** this scenario checks only timing (message format is not checked)
** response(486 is used below) may be 300-699 response.

*1:ACK

ACK request Must be retransmitted before Timer D(>=32sec.) fires.
[RFC3261-17-24][RFC3261-17-25]

*2:after 486 response(Timer D is already fired) from UA12.

ACK request Must not be retransmitted. [RFC3261-17-27]
17.1.1.2 Formal Description

When in either the "Calling" or "Proceeding" states, reception of a response with status code from 300-699 MUST cause the client transaction to transition to "Completed". *snip* The client transaction SHOULD start timer D when it enters the "Completed" state, with a value of at least 32 seconds for unreliable transports, and a value of zero seconds for reliable transports. *snip*

When in either the "Calling" or "Proceeding" states, reception of a 2xx response MUST cause the client transaction to enter the "Terminated" state, and *snip*

If timer D fires while the client transaction is in the "Completed" state, the client transaction MUST move to the terminated state.

4.7.4 TS-2-1-1 - SIP Proxy- Non-INVITE Client Transaction (Stop of retransmission of CANCEL upon Timer F fired)

[NAME]
TS-2-1-1 - SIP Proxy- Non-INVITE Client Transaction (Stop of retransmission of CANCEL upon Timer F fired)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT stops retransmitting a CANCEL request and informs the UAC of time out of the request when Timer F is fired.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip:ss.under.test.com</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>
**TOPOLOGY**

```
---+-----------+---------
 |           |
 |          UA11
 R11 ---+---R-------+-----------+---------
 |           |           |
 |         NUT       Registrar
 R12 ---+-----------+---------
 |           |
 |           |
 |           |
 |           |
 |           |
 |           |
 UA12
```

**CONFIGURATION for NUT**

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

**INITIALIZATION**

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

```
UA11  R  NUT
 |    |   |
 |    |   |
 |----|----| 1. ICMP Echo Request |
 |    |   |   |
 <----|----| 2. ICMP Echo Reply |
 |    |   |   |
```

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

```
UA11  UA12  R  Registrar
 |    |    |   |
 |    |    |   |
 |    |    |   |
 |-------------------|--------|1. REGISTER |
```

---
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11     :       NUT       :      UA12
|       :        |        :       |
|       :        |        :       |
|-------:------->|        :       | 1. INVITE
|       :        |--------:------>| 2. INVITE
|<------:--------|        :       | 3. 100
|       :        |<-------:-------| 4. 180
|<------:--------|        :       | 5. 180
|       :        |--------:------>| 6. CANCEL --Timer F started
|<------:--------|        :       | 7. 200
|       :        |--------:------>| 8. CANCEL (*1)
|       :        |--------:------>| 9. CANCEL (*2)
|       :        |--------:------>| 10. CANCEL (*3)
|       :        |--------:------>| 11. CANCEL (*4)
|       :        |--------:------>| 12. CANCEL (*5)
|       :        |--------:------>| 13. CANCEL (*6)
|       :        |--------:------>| 14. CANCEL (*7)
|       :        |--------:------>| 15. CANCEL (*8)
|       :        |--------:------>| 16. CANCEL (*9)
1. UA11 Send INVITE.
2. UA12 Receive INVITE.
3. UA11 Receive 100 Trying.
4. UA12 Send 180 Ringing.
5. UA11 Receive 180 Ringing.
6. UA11 Send CANCEL.
7. UA11 Receive 100 Trying.
8. UA12 Receive CANCEL. (*1)
9. UA12 Receive CANCEL. (*2)
10. UA12 Receive CANCEL. (*3)
11. UA12 Receive CANCEL. (*4)
12. UA12 Receive CANCEL. (*5)
13. UA12 Receive CANCEL. (*6)
14. UA12 Receive CANCEL. (*7)
15. UA12 Receive CANCEL. (*8)
16. UA12 Receive CANCEL. (*9)
17. UA12 Receive CANCEL. (*10)(*11)
18. UA11 Receive 408 Request Timeout.
19. UA11 Send ACK.

** [OBSERVABLE RESULTS]**

** this scenario checks only timing (message format is not checked)

*1: CANCEL request from NUT to UA12.

Must be retransmitted after Timer E(= T1 sec.) fired.
[RFC3261-17-40][RFC3261-17-41][RFC3261-17-42]

*2: CANCEL request from NUT to UA12.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]
*3:CANCEL request from NUT to UA12.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*4:CANCEL request from NUT to UA12.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*5:CANCEL request from NUT to UA12.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*6:CANCEL request from NUT to UA12.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*7:CANCEL request from NUT to UA12.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*8:CANCEL request from NUT to UA12.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*9:CANCEL request from NUT to UA12.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*10:CANCEL request from NUT to UA12.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*11:before 200 response(Timer F(=64*T1 sec.) is already fired) from Proxy.
The "Trying" state is entered when the TU initiates a new client transaction with a request. When entering this state, the client transaction SHOULD set timer F to fire in 64*T1 seconds. The request MUST be passed to the transport layer for transmission. If an unreliable transport is in use, the client transaction MUST set timer E to fire in T1 seconds. If timer E fires while still in this state, the timer is reset, but this time with a value of MIN(2*T1, T2). When the timer fires again, it is reset to a MIN(4*T1, T2). This process continues so that retransmissions occur with an exponentially increasing interval that caps at T2. The default value of T2 is 4s, and it represents the amount of time a non-INVITE server transaction will take to respond to a request, if it does not respond immediately. For the default values of T1 and T2, this results in intervals of 500 ms, 1 s, 2 s, 4 s, 4 s, 4 s, etc.

If Timer F fires while the client transaction is still in the "Trying" state, the client transaction SHOULD inform the TU about the timeout, and then it SHOULD enter the "Terminated" state. If a provisional response is received while in the "Trying" state,

*snip* If a final response (status codes 200-699) is received while in the "Trying" state, the response MUST be passed to the TU, and the client transaction MUST transition to the "Completed" state.

*snip* If timer F fires while in the "Proceeding" state, the TU MUST be informed of a timeout, and the client transaction MUST transition to the terminated state. If a final response (status codes 200-699) is received while in the "Proceeding" state, the response MUST be passed to the TU, and the client transaction MUST transition to the "Completed" state.

*snip* The "Completed" state exists to buffer any additional response retransmissions that may be received (which is why the client transaction remains there only for
unreliable transports). T4 represents the amount of time the network will take to clear messages between client and server transactions. The default value of T4 is 5s. A response is a retransmission when it matches the same transaction, using the rules specified in Section 17.1.3. If Timer K fires while in this state, the client transaction MUST transition to the "Terminated" state.

Once the transaction is in the terminated state, it MUST be destroyed immediately.

[RFC3261-10-10] 10.2 Constructing the REGISTER Request
[RFC3261 Page 58 Paragraph 5]

UAs MUST NOT send a new registration (that is, containing new Contact header field values, as opposed to a retransmission) until they have received a final response from the registrar for the previous one or the previous REGISTER request has timed out.

[RFC3261-10-20] 10.2.7 Transmitting a Request
[RFC3261 Page 63 Paragraph 1]

If the transaction layer returns a timeout error because the REGISTER yielded no response, the UAC SHOULD NOT immediately re-attempt a registration to the same registrar.

An immediate re-attempt is likely to also timeout. Waiting some reasonable time interval for the conditions causing the timeout to be corrected reduces unnecessary load on the network. No specific interval is mandated.

[RFC3261-16-113] 16.7 Response Processing

If there are no final responses in the context, the proxy MUST send a 408 (Request Timeout) response to the server transaction.

4.7.5 TS-2-1-2 - SIP Proxy- Non-INVITE Client Transaction (Stop of retransmission of BYE upon Timer F fired)
TS-2.1-2 · SIP Proxy- Non-INVITE Client Transaction (Stop of retransmission of BYE upon Timer F fired)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT doesn’t forwards a BYE request when Timer F is fired.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501::50/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
---+-----------+---------
|           |          |
|          | UA11    |
|           |         |
| R11       | R-------+
|           | R       |
|           |         |
|           |         |
|           |         |
|           | Registrar|
|           |         |
| R12       |         |
|           |         |
|           |         |
|           |         |
|           |         |
|           |         |
|           |         |
|           |         |
|           |         |
|           |         |
|           |         |
|           |         |
|         UA12|
```
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501::ff050::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

3. Send REGISTER Request.
4. Receive 200 OK response.

PROCEDURE

Note: This sequence is an example.

The number of retransmission changes depending on a case.

1. Invite
2. Invite

1. UA11 Send INVITE.
2. UA12 Receive INVITE.
3. UA11 Receive 100 Trying.
4. UA12 Send 180 Ringing.
5. UA11 Receive 180 Ringing.
6. UA12 Send 200 OK.
7. UA11 Receive 200 OK.
8. UA11 Send ACK.
9. UA12 Receive ACK.
10. UA11 Send BYE.
11. UA12 Receive BYE.
12. UA12 Receive BYE. (*1)
13. UA11 Send BYE.
14. UA12 Receive BYE. (*2)
15. UA12 Receive BYE. (*3)
16. UA11 Send BYE.
17. UA12 Receive BYE. (*4)
18. UA11 Send BYE.
19. UA12 Receive BYE. (*5)
20. UA12 Receive BYE. (*6)
21. UA12 Receive BYE. (*7)
22. UA12 Receive BYE. (*8)
23. UA11 Send BYE.
24. UA12 Receive BYE. (*9)
25. UA11 Send BYE. (*10)

* Break off BYE request from UA11 at NUT.
* BYE request from NUT retransmitting based retransmit interval.

** this scenario checks only timing (message format is not checked)

*1:BYE request from NUT.

Must be retransmitted after Timer E(= T1 sec.) fired.
[RFc3261-17-40][RFc3261-17-41][RFc3261-17-42]

*2:BYE request from NUT.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*3:BYE request from NUT.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]
*4: BYE request from NUT.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*5: BYE request from NUT.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*6: BYE request from NUT.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*7: BYE request from NUT.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*8: BYE request from NUT.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*9: BYE request from NUT.

Must be retransmitted after MIN(2*T1, T2) sec. [RFC3261 17.1.2.2]

*10: before 200 response (Timer F(=64*T1 sec.) is already fired) from Proxy.

BYE request Must not be retransmitted. [RFC3261-17-43,44]

[REFERENCE]
[RFC3261-17-40, 41, 42, 43, 44, 47, 48, 51, 52, 53, 54, 56, 57]

17.1.2.2 Formal Description

.....

The "Trying" state is entered when the TU initiates a new client transaction with a request. When entering this state, the client
transaction SHOULD set timer F to fire in 64*T1 seconds. The request MUST be passed to the transport layer for transmission. If an unreliable transport is in use, the client transaction MUST set timer E to fire in T1 seconds. If timer E fires while still in this state, the timer is reset, but this time with a value of MIN(2*T1, T2). When the timer fires again, it is reset to a MIN(4*T1, T2). This process continues so that retransmissions occur with an exponentially increasing interval that caps at T2. The default value of T2 is 4s, and it represents the amount of time a non-INVITE server transaction will take to respond to a request, if it does not respond immediately. For the default values of T1 and T2, this results in intervals of 500 ms, 1 s, 2 s, 4 s, 4 s, 4 s, etc.

If Timer F fires while the client transaction is still in the "Trying" state, the client transaction SHOULD inform the TU about the timeout, and then it SHOULD enter the "Terminated" state. If a provisional response is received while in the "Trying" state, *snip* If a final response (status codes 200-699) is received while in the "Trying" state, the response MUST be passed to the TU, and the client transaction MUST transition to the "Completed" state.

*snip* If timer F fires while in the "Proceeding" state, the TU MUST be informed of a timeout, and the client transaction MUST transition to the terminated state. If a final response (status codes 200-699) is received while in the "Proceeding" state, the response MUST be passed to the TU, and the client transaction MUST transition to the "Completed" state.

*snip* The "Completed" state exists to buffer any additional response retransmissions that may be received (which is why the client transaction remains there only for unreliable transports). T4 represents the amount of time the network will take to clear messages between client and server transactions. The default value of T4 is 5s. A response is a retransmission when it matches the same transaction, using the rules specified in Section 17.1.3. If Timer K fires while in this state, the client transaction MUST transition to the "Terminated" state.

Once the transaction is in the terminated state, it MUST be destroyed immediately.

[ RFC3261-10-10 ]
10.2 Constructing the REGISTER Request
[RFC3261 Page 58 Paragraph 5]

UAs MUST NOT send a new registration (that is, containing new Contact header field values, as opposed to a retransmission) until they have received a final response from the registrar for the previous one or the previous REGISTER request has timed out.

[RFC3261-10-20]
10.2.7 Transmitting a Request
[RFC3261 Page 63 Paragraph 1]

If the transaction layer returns a timeout error because the REGISTER yielded no response, the UAC SHOULD NOT immediately re-attempt a registration to the same registrar.

An immediate re-attempt is likely to also timeout. Waiting some reasonable time interval for the conditions causing the timeout to be corrected reduces unnecessary load on the network. No specific interval is mandated.

4.7.6 TS-2-1-3 - Non-INVITE Client Transaction (Receipt of 100 response to CANCEL and reset of Timer E with T2 )

[NAME]
TS·2·1·3 · SIP Proxy· Non-INVITE Client Transaction (receipt of 100 response to CANCEL and reset of Timer E with T2 )

[TARGET] SIP Proxy

[PURPOSE] Verify that a NUT resets Timer E with T2 when receiving a 100 response to previous CANCEL request.

[REQUIREMENT] Set up registrar server to use location service, if necessary.

[PARAMETER]
<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
</tbody>
</table>
Phase 2 Test Specification

**SIP IPv6**

<table>
<thead>
<tr>
<th>UA21(AOR)</th>
<th>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></th>
</tr>
</thead>
<tbody>
<tr>
<td>UA21(Contact)</td>
<td>sip:<a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>PX2</td>
<td>sip:ss2.biloxi.example.com</td>
</tr>
</tbody>
</table>

**[ADDRESS]**

<table>
<thead>
<tr>
<th>Location</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA21 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>PX2 (IPv6)</td>
<td>3ffe:501:ffff:20::20/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

**[TOPOLOGY]**

```
---+-----------+---------
|           |
|          UA11 |
R11
|           |
---+---R-------+---
|           |           |
|         NUT       Registrar |
R12
|           |
---+-----------+---------
|           |
|          PX2 |
R13
|           |
---+-----------+---------
|           |
|          UA21 |
```

**[CONFIGURATION for NUT]**

<table>
<thead>
<tr>
<th>Location</th>
<th>Address</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT</td>
<td>sip:ss.under.test.com</td>
</tr>
<tr>
<td>NUT (IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

**[INITIALIZATION]**

```
UA11  R  NUT
|   |   |   |
|   |   |   |
|-----|-----| 1. ICMP Echo Request |
|   |   |   |
```
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11 R Registrar

|        |         |         |
|        |         |         |
|        |         |         |
|        |         |<--------|         |

1. REGISTER
2. 200 OK

1. Send REGISTER Request.
2. Receive 200 OK response.

[PROCEDURE]

UA11 : NUT : PX2 : UA21

|        |        |        |        |
|        |        |        |        |
|        |        |        |        |
|        |        |        |        |
|        |        |        |        |
|        |        |        |        |
|        |        |        |        |
|        |        |        |        |
|        |        |        |        |
|        |        |        |        |
|        |        |        |        |
|        |        |        |        |
|        |        |        |        |
|        |        |        |        |

1. INVITE
2. INVITE
180
3. 180
4. 180
5. CANCEL
6. 200
7. CANCEL
8. CANCEL (*1)
9. CANCEL (*2)
10. 100 (for CANCEL)
11. CANCEL (*3)
12. CANCEL
1. UA11 Send INVITE.
2. PX2 Receive INVITE.
3. PX2 Send 180 Ringing.
4. UA11 Receive 180 Ringing.
5. UA11 Send CANCEL.
6. UA11 Receive 200 OK.
7. PX2 Receive CANCEL.
8. PX2 Receive CANCEL. (*1)
9. PX2 Receive CANCEL. (*2)
10. PX2 Send 100 Trying.
11. PX2 Receive CANCEL. (*3)
12. PX2 Receive CANCEL.
13. PX2 Send 200 OK.
14. PX2 Send 487 Request Terminated.
15. PX2 Receive ACK.
16. UA11 Receive 487 Request Terminated.
17. UA11 Send ACK.

**OBSERVABLE RESULTS**

** this scenario checks only timing (message format is not checked)

*1:CANCEL request from NUT.

Must be retransmitted after Timer E(= T1 sec.) fired. [RFC3261-17-42]
*2:* CANCEL request from NUT.

Further retransmit Must be MIN(2*T1,T2) sec., but not equal T2.

[RFc3261 17.1.2.2]

*3:* after 100 response from Proxy.

Should be retransmitted after T2 sec. [RFC3261-17-49][RFC3261-17-50]

[REFERENCE]

[RFc3261-17-45, 46, 49, 50]

17.1.2.2 Formal Description

.....

*snip* If a provisional response is received while in the "Trying" state, the response MUST be passed to the TU, and then the client transaction SHOULD move to the "Proceeding" state. *snip*

.....

If Timer E fires while in the "Proceeding" state, the request MUST be passed to the transport layer for retransmission, and Timer E MUST be reset with a value of T2 seconds.

**4.7.7 TS-2-1-4 - SIP Proxy- Non-INVITE Client Transaction (Receipt of 100 response to BYE and reset of Timer E with T2)**

[NAME]

TS-2-1-4 - SIP Proxy- Non-INVITE Client Transaction (receipt of 100 response to BYE and reset of Timer E with T2)

[TARGET]

SIP Proxy

[PURPOSE]

Verify that a NUT resets Timer E with T2 when receiving a 100 response to previous BYE request.
[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA21(AOR)</td>
<td>sip:<a href="mailto:UA21@biloxi.example.com">UA21@biloxi.example.com</a></td>
</tr>
<tr>
<td>UA21(Contact)</td>
<td>sip:<a href="mailto:UA21@client.biloxi.example.com">UA21@client.biloxi.example.com</a></td>
</tr>
<tr>
<td>PX2</td>
<td>sip:ss2.biloxi.example.com</td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA21(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>PX2(IPv6)</td>
<td>3ffe:501:ffff:20::20/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

---+-----------+---------
|           |          |
|          UA11 |
| R11       |
---+---R-------+-----------+---------
|           |           |
|         NUT           Registrar |
| R12       |
---+-----------+---------
|           |          |
|          PX2 |
| R13       |
---+-----------+---------
|           |
|          UA21 |

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
</table>
[INITIALIZATION]

UA11  R  NUT
|     |     |     |
|     |     |     |
|<------|--------| 2. ICMP Echo Reply
|     |     |     |

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11  R  Registrar
|     |     |     |
|     |     |     |
|<------|--------| 2. 200 OK
|     |     |     |

1. Send REGISTER Request.
2. Receive 200 OK response.

[PROCEDURE]

Note: This sequence is an example.
The number of retransmission changes depending on a case.

UA11 : NUT : PX2 : UA21
|     :     :     :     |
|     :     :     :     |
|<------|--------|     | 1. INVITE
|     :     :     : 2. INVITE
|<------|--------|     | 3. 100
|     :     :     : 4. 100
|     :     :     : 5. 180
|<------|--------|     | 6. 180
|     :     :     : 7. 200
|<------|--------|     | 8. 200
1. UA11 Send INVITE.
2. PX2 Receive INVITE.
3. UA11 Receive 100 Trying.
4. PX2 Send 100 Trying.
5. PX2 Send 180 Ringing.
6. UA11 Receive 180 Ringing.
7. PX2 Send 200 OK.
8. UA11 Receive 200 OK.
9. UA11 Send ACK.
10. PX2 Receive ACK.
11. UA11 Send BYE.
12. PX2 Receive BYE.
13. PX2 Receive BYE. (*1)
14. UA11 Send BYE.
15. PX2 Receive BYE. (*2)
16. PX2 Send 100 Trying.
17. PX2 Receive BYE.
18. UA11 Send BYE.
(*3)
19. PX2 Receive BYE.
20. PX2 Send 200 OK.
21. UA11 Receive 200 OK.

[OBSERVABLE RESULTS]
** this scenario checks only timing (message format is not checked)

*1: BYE request from NUT.

Must be retransmitted after Timer E(= T1 sec.) fired. [RFC3261-17-42]

*2: BYE request from NUT.

Further retransmit Must be MIN(2*T1,T2) sec., but not equal T2.

[RFC3261 17.1.2.2]

*3: after 100 response from Proxy.

Should be retransmitted after T2 sec. [RFC3261-17-49][RFC3261-17-50]

[REFERENCE]
[RFC3261-17-45, 46, 49, 50]
17.1.2.2 Formal Description

.....

*snip* If a provisional response is received while in the "Trying" state, the response MUST be passed to the TU, and then the client transaction SHOULD move to the "Proceeding" state. *snip*

.....

If Timer E fires while in the "Proceeding" state, the request MUST be passed to the transport layer for retransmission, and Timer E MUST be
reset with a value of T2 seconds.

4.7.8 TS-3-1-1 - SIP Proxy- INVITE Server Transaction (Response after Timer H fired)

[NAME]
TS-3-1-1 - INVITE Server Transaction (Response after Timer H fired)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT doesn’t forward 4xx-6xx responses to the requests that has the same to-tag as one of response after Timer H is fired. Also, verify that the proxy retransmits 4xx-6xx response to the requests, when the to-tag of that request is different from that before time-out of Timer H, after Timer H is fired.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(AOR)</td>
<td>sip: ss.under.test.com</td>
</tr>
<tr>
<td>Registrar(AOR)</td>
<td>sip: reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>Address</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
+---------------+        +---------------+        +---------------+
|               |        |               |        |               |
| UA11          |        | UA11          |        | UA11          |
|               |        |               |        |               |
| R11           |        | R             |        | R             |
```

I P v 6 F O R U M T E C H N I C A L D O C U M E N T
IP v 6 R e a d y L o g o P r o g r a m

Phase 2 Test Specification

SIP IPv6

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT</td>
<td>Registrar</td>
</tr>
<tr>
<td>R12</td>
<td></td>
</tr>
<tr>
<td>-----------</td>
<td>-----------</td>
</tr>
<tr>
<td>UA12</td>
<td></td>
</tr>
</tbody>
</table>

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

[INITIALIZATION]

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>-----&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>---</td>
<td>-----&gt;</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

<table>
<thead>
<tr>
<th>UA11</th>
<th>UA12</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>---</td>
<td>-----------</td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>---</td>
<td>-----------</td>
</tr>
<tr>
<td>-------</td>
<td>-----</td>
<td>---</td>
<td>-----------</td>
</tr>
<tr>
<td>-------</td>
<td>-----</td>
<td>---</td>
<td>-----------</td>
</tr>
<tr>
<td>-------</td>
<td>-----</td>
<td>---</td>
<td>-----------</td>
</tr>
<tr>
<td>&lt;-------</td>
<td>-----</td>
<td>---</td>
<td>-----------</td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]
Note: This sequence is an example.

The number of retransmission changes depending on a case.

1. UA11 Send INVITE.
2. UA12 Receive INVITE.
3. UA11 Send INVITE.
4. UA12 Receive INVITE.
5. UA11 Send INVITE.
6. UA12 Receive INVITE.
7. UA11 Send INVITE.
8. UA12 Receive INVITE.
9. UA11 Send INVITE.
10. UA12 Send 4xx-6xx.
11. UA12 Receive ACK.

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>1. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>2. INVITE --Timer B,C started</td>
</tr>
<tr>
<td></td>
<td></td>
<td>3. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>6. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>7. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>9. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>10. 4xx-6xx</td>
</tr>
<tr>
<td></td>
<td></td>
<td>11. ACK</td>
</tr>
<tr>
<td></td>
<td></td>
<td>12. 4xx-6xx --Timer H started</td>
</tr>
<tr>
<td></td>
<td></td>
<td>13. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>14. 4xx-6xx(*1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>15. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>16. 4xx-6xx(*2)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>17. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td>18. 4xx-6xx(*3–2)</td>
</tr>
</tbody>
</table>

---

IPv6 FORUM TECHNICAL DOCUMENT
IPv6 Ready Logo Program
Phase 2 Test Specification
SIP IPv6
12. UA11 Receive 4xx-6xx.
13. UA11 Send INVITE.
14. UA11 Receive 4xx-6xx. (*1)
15. UA11 Send INVITE.
16. UA11 Receive 4xx-6xx. (*2)
17. UA11 Send INVITE. (*3-1)
18. UA11 Receive 4xx-6xx. (*3-2)

[OBSERVABLE RESULTS]
** this scenario checks only timing (message format is not checked, except 1xx response)

*1: 4xx-6xx response from NUT to UA11.

Must be retransmitted after Timer G(= T1 sec.) fired.
[RFC3261-17-67, RFC3261-17-68]

*2: 4xx-6xx response from NUT to UA11.

Further retransmit Must be MIN(2*T1, T2) sec.
[RFC3261 17.1.2.2]

*3-1: after Timer H(=64*T1 sec.) fired.

4xx-6xx response Must not be retransmitted. [RFC3261-17-69, RFC3261-17-72, RFC3261-17-73]

*3-2: 4xx-6xx response from NUT to UA11.

To tag Must be different from retransmitted 4xx-6xx response before Timer H fires.
[RFC3261 17.1.2.2]

[REFERENCE]
[RFC3261-17-67, 68, 69, 70, 72, 73]
17.2.1 INVITE Server Transaction

.....

While in the "Proceeding" state, if the TU passes a response with
status code from 300 to 699 to the server transaction, the response MUST be passed to the transport layer for transmission, and the state machine MUST enter the "Completed" state. For unreliable transports, timer G is set to fire in T1 seconds, and is not set to fire for reliable transports.

This is a change from RFC 2543, where responses were always retransmitted, even over reliable transports.

When the "Completed" state is entered, timer H MUST be set to fire in 64*T1 seconds for all transports. Timer H determines when the server transaction abandons retransmitting the response. Its value is chosen to equal Timer B, the amount of time a client transaction will continue to retry sending a request. If timer G fires, the response is passed to the transport layer once more for retransmission, and timer G is set to fire in MIN(2*T1, T2) seconds. From then on, when timer G fires, the response is passed to the transport again for transmission, and timer G is reset with a value that doubles, unless that value exceeds T2, in which case it is reset with the value of T2. This is identical to the retransmit behavior for requests in the "Trying" state of the non-INVITE client transaction. Furthermore, while in the "Completed" state, if a request retransmission is received, the server SHOULD pass the response to the transport for retransmission.

If timer H fires while in the "Completed" state, it implies that the ACK was never received. In this case, the server transaction MUST transition to the "Terminated" state, and MUST indicate to the TU that a transaction failure has occurred.

4.7.9 TS-3-1-2 - SIP Proxy- INVITE Server Transaction (Stop of retransmission after Timer H fired)

[NAME]
TS-3-1-2 · SIP Proxy- INVITE Server Transaction (Stop retransmission after Timer H fired)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT stops retransmitting a 4xx-6xx response and after Timer H is fired.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

| NUT(AOR)          | sip:ss.under.test.com |
| Registrar(AOR)    | sip:reg.under.test.com |
| UA11(AOR)         | sip:UA11@under.test.com |
| UA11(Contact)     | sip:UA11@node.under.test.com |
| UA12(AOR)         | sip:UA12@under.test.com |
| UA12(Contact)     | sip:UA12@node11.under.test.com |

[ADDRESS]

| NUT (IPv6)         | 3ffe:501:ffff:50::50/64 |
| Registrar (IPv6)   | 3ffe:501:ffff:50::60/64 |
| UA11(IPv6)         | 3ffe:501:ffff:1::1/64  |
| UA12(IPv6)         | 3ffe:501:ffff:2::2/64  |
| R(IPv6)            | 3ffe:501:ffff:50::1/64 |

[TOPOLOGY]

---+-----------+---------
|           |          |
|          UA11 |         |
| R11       |          |
---+---R-------+-----------+---------
|           |          |
|         NUT       Registrar |
| R12       |          |
---+-----------+---------
|           |          |
|          UA12 |         |

[CONFIGURATION for NUT]

| NUT           | sip:ss.under.test.com |
| NUT(IPADDRESS) | 3ffe:501:ffff:50::50/64 (IPv6) |

[INITIALIZATION]

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
</table>
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11       UA12       R       Registrar
|       |       |       |
|-------|-------|-----|--
|       |       |-----| 1. REGISTER
|       |       |-----| 2. 200 OK
|       |       |-----| 3. REGISTER
|       |       |-----| 4. 200 OK

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11     NUT     UA12
|       |       |       |
|-------|-------|-----|--
|<-----:-----|       | 1. INVITE
|       |-----:-----| 2. INVITE
|       |<-------:----| 3. 4xx-6xx
|       |-----|-----| 4. ACK
|<------:-----|       | 5. 4xx-6xx —Timer H started
|<------:-----|       | 6. 4xx-6xx (*1)
|<------:-----|       | 7. 4xx-6xx (*2)
1. UA11 Send INVITE.
2. UA12 Receive INVITE.
3. UA12 Send 4xx-6xx.
4. UA12 Receive ACK.
5. UA11 Receive 4xx-6xx.
6. UA11 Receive 4xx-6xx. (*1)
7. UA11 Receive 4xx-6xx. (*2)
8. UA11 Receive 4xx-6xx. (*3)
9. UA11 Receive 4xx-6xx. (*4)
10. UA11 Receive 4xx-6xx. (*5)
11. UA11 Receive 4xx-6xx. (*6)
12. UA11 Receive 4xx-6xx. (*7)
13. UA11 Receive 4xx-6xx. (*8)
14. UA11 Receive 4xx-6xx. (*9)
15. UA12 Send 4xx-6xx. (*10)

[OBSERVABLE RESULTS]

** this scenario checks only timing (message format is not checked)
** Target receives INVITE with To-tag, and Tester expects 481 response.
** But target may do any way that like for sending 4xx-6xx response.

*1: 4xx-6xx response from NUT.
Must be retransmitted after Timer G(= T1 sec.) fired. [RFC3261-17.67, RFC3261-17.68]

*2: INVITE request from NUT.
   Must be retransmitted after MIN(2*T1,T2) sec. [RFC3261 17.1.2.2]

*3: INVITE request from NUT.
   Must be retransmitted after MIN(2*T1,T2) sec. [RFC3261 17.1.2.2]

*4: INVITE request from NUT.
   Must be retransmitted after MIN(2*T1,T2) sec. [RFC3261 17.1.2.2]

*5: INVITE request from NUT.
   Must be retransmitted after MIN(2*T1,T2) sec. [RFC3261 17.1.2.2]

*6: INVITE request from NUT.
   Must be retransmitted after MIN(2*T1,T2) sec. [RFC3261 17.1.2.2]

*7: INVITE request from NUT.
   Must be retransmitted after MIN(2*T1,T2) sec. [RFC3261 17.1.2.2]

*8: INVITE request from NUT.
   Must be retransmitted after MIN(2*T1,T2) sec. [RFC3261 17.1.2.2]

*9: INVITE request from NUT.
   Must be retransmitted after MIN(2*T1,T2) sec. [RFC3261 17.1.2.2]

*10: after Timer H(=64*T1 sec.) fired.
4xx-6xx response Must not be retransmitted. [RFC3261-17-69, RFC3261-17-72, RFC3261-17-73]

**[REFERENCE]**

[RFC3261-17-67, 68, 69, 70, 72, 73]

17.2.1 INVITE Server Transaction

.....

While in the "Proceeding" state, if the TU passes a response with status code from 300 to 699 to the server transaction, the response MUST be passed to the transport layer for transmission, and the state machine MUST enter the "Completed" state. For unreliable transports, timer G is set to fire in T1 seconds, and is not set to fire for reliable transports.

This is a change from RFC 2543, where responses were always retransmitted, even over reliable transports.

When the "Completed" state is entered, timer H MUST be set to fire in 64*T1 seconds for all transports. Timer H determines when the server transaction abandons retransmitting the response. Its value is chosen to equal Timer B, the amount of time a client transaction will continue to retry sending a request. If timer G fires, the response is passed to the transport layer once more for retransmission, and timer G is set to fire in MIN(2*T1, T2) seconds. From then on, when timer G fires, the response is passed to the transport again for transmission, and timer G is reset with a value that doubles, unless that value exceeds T2, in which case it is reset with the value of T2. This is identical to the retransmit behavior for requests in the "Trying" state of the non-INVITE client transaction. Furthermore, while in the "Completed" state, if a request retransmission is received, the server SHOULD pass the response to the transport for retransmission.

.....

If timer H fires while in the "Completed" state, it implies that the ACK was never received. In this case, the server transaction MUST transition to the "Terminated" state, and MUST indicate to the TU that a transaction failure has occurred.
4.7.10 TS-3-1-4 - SIP Proxy- INVITE Server Transaction (Stop of retransmission of 4xx-6xx response upon receipt of ACK)

[NAME]
TS-3-1-4 · SIP Proxy- INVITE Server Transaction (Stop of retransmission of 4xx-6xx response upon receipt of ACK)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly stops retransmitting a 4xx-6xx response when receiving an ACK.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]
| NUT(AOR)          | sip:ss.under.test.com |
| Registrar(AOR)   | sip:reg.under.test.com |
| UA11(AOR)        | sip:UA11@under.test.com |
| UA11(Contact)    | sip:UA11@node.under.test.com |
| UA12(AOR)        | sip:UA12@under.test.com |
| UA12(Contact)    | sip:UA12@node11.under.test.com |

[ADDRESS]
| NUT (IPv6)          | 3ffe:501:ffff:50::50/64 |
| Registrar (IPv6)   | 3ffe:501:ffff:50::60/64 |
| UA11(IPv6)         | 3ffe:501:ffff:1::1/64 |
| UA12(IPv6)         | 3ffe:501:ffff:2::2/64 |
| R(IPv6)            | 3ffe:501:ffff:50::1/64 |

[TOPOLOGY]

```
---------------
|              |
|              |
|          UA11|
|        R11  |
---------------

---------------
|              |
|              |
|              |
|              |
|        NUT   |
|        Registrar|
|   R12        |
```
[INITIALIZATION]

UA11  R  NUT

|        |        |        |
|        |        |        |
|<--------|---------| 2. ICMP Echo Reply
|        |        |        |

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11  UA12  R  Registrar

|        |        |        |
|        |        |        |
|<------------------|---------|2. 200 OK
|        |        |        |

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11  NUT  UA12

|        |        |        |
|        |        |        |

IPv6 FORUM TECHNICAL DOCUMENT
IPv6 Ready Logo Program
Phase 2 Test Specification
SIP IPv6
1. UA11 Send INVITE.
2. UA12 Receive INVITE.
3. UA12 Send 4xx-6xx.
4. UA12 Receive ACK.
5. UA11 Receive 4xx-6xx.
6. UA11 Receive 4xx-6xx.
7. UA11 Send ACK.

(*1)

[OBSERVABLE RESULTS]
** this scenario checks only timing (message format is not checked)
** Target receives INVITE with To-tag, and Tester expects 481 response.
** But target may do any way that like for sending 4xx-6xx response.

*1:after ACK request from UA11.

4xx-6xx response Must not be retransmitted. [RFC3261-17-71]

[REFERENCE]
[RFC3261-17-67, 68, 71]
17.2.1 INVITE Server Transaction

.....

While in the "Proceeding" state, if the TU passes a response with
status code from 300 to 699 to the server transaction, the response
MUST be passed to the transport layer for transmission, and the state
machine MUST enter the "Completed" state. For unreliable transports, 
timer G is set to fire in T1 seconds, and is not set to fire for 
reliable transports.

.....

If an ACK is received while the server transaction is in the 
"Completed" state, the server transaction MUST transition to the 
"Confirmed" state. As Timer G is ignored in this state, any 
retransmissions of the response will cease.

4.7.11 TS-3-1-5 - SIP Proxy- 487 to CANCEL for INVITE request after 
64*T1 fired

[NAME]  
TS-3-1-5 - SIP Proxy- 487 to CANCEL for INVITE request after 64*T1 fired

[TARGET]  
SIP Proxy

[PURPOSE]  
Verify that a NUT doesn't sends any response to that downstream when receiving a 487 
response to previous CANCEL request for INVITE request after 64*T1 is fired

[REQUIREMENT]  
Set up registrar server to use location service, if necessary.

[PARAMETER]  

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]  

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IpV6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IpV6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>
[TOPOLOGY]

---+-----------+---------
|           |
|          UA11 |
R11        |
---+---R-------+-----------+---------
|           |           |
|         NUT       Registrar |
R12        |
---+-----------+---------
|           |
|           |
|           |
|           |
UA12

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

UA11  R  NUT
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11  UA12  R  Registrar
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |
| | | |

1. REGISTER
2. 200 OK
3. REGISTER
4. 200 OK
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>[---:-----]</td>
<td>:</td>
<td>1. INVITE</td>
</tr>
<tr>
<td>[---:-----]</td>
<td>:</td>
<td>2. INVITE</td>
</tr>
</tbody>
</table>

---

64*T1 count down started

| [---:-----] | :       | 3. 100   |
| [---:-----] | :       | 4. 180   |
| [---:-----] | :       | 5. 180   |
| [---:-----] | :       | 6. CANCEL|
| [---:-----] | :       | 7. 200   |
| [---:-----] | :       | 8. CANCEL|
| [---:-----] | :       | 9. 200   |

---

after 64*T1 sec passage

| [---:-----] | :       | 10. 408  |
| [---:-----] | :       | 11. ACK  |
| [---:-----] | :       | 12. 487  |
| [---:-----] | :       | (no ack) |

(*1)

1. UA11 Send INVITE.
2. UA12 Receive INVITE.
3. UA11 Receive 100 Trying.
4. UA12 Send 180 Ringing.
5. UA11 Receive 180 Ringing.
6. UA11 Send CANCEL.
7. UA11 Receive 200 OK.
8. UA12 Receive CANCEL.
9. UA12 Send 200 OK.
10. UA11 Receive 408 Request Timeout.
11. UA11 Send 200 OK.
12. UA12 Send 487 Request Terminated.
(*1)

[OBSERVABLE RESULTS]

** this scenario checks only timing (message format is not checked)

*1:after 487 response from UA12.

* Should not send ACK request. [RFC3261-9-12][RFC3261-9-13]

[REFERENCE]
[RFC3261-9-12, 13]
9.1 Client Behavior
[RFC3261 Page 54 Paragraph 8]

Note that both the transaction corresponding to the original request and the CANCEL transaction will complete independently. However, a UAC canceling a request cannot rely on receiving a 487 (Request Terminated) response for the original request, as an RFC 2543-compliant UAS will not generate such a response. If there is no final response for the original request in 64*T1 seconds (T1 is defined in Section 17.1.1.1), the client SHOULD then consider the original transaction cancelled and SHOULD destroy the client transaction handling the original request.

4.7.12 TS-4-1-1 - SIP Proxy- Non-INVITE Server Transaction (Response for CANCEL after Timer J fired)

[NAME]
TS-4-1-1 - SIP Proxy- Non-INVITE Server Transaction (response for CANCEL after Timer J fired)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly sends an error response for a CANCEL request after Timer J is fired.
[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
---+-----------+---------
|           |
|          UA11 |
R11
| R-----------+---------
|           |
|         NUT       Registrar |
R12
|           |
---+-----------+---------
|          |
| UA12      |
```

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

```
UA11  R   NUT
|    |    |    |
|    |    |    |
```
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11      UA12        R     Registrar
|       |          |       |
|<-------|---------| 2. 200 OK
|       |          |       |
|       |<--------| 4. 200 OK
|       |          |       |
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]
UA11 :       NUT :      UA12
|       |          |       |
|-------:--------|       |<-------:-------| 5. 180
|       |          |<--------:-------| 7. 200
|       |          |<-------:-------| 9. 200

1. INVITE
2. INVITE
3. 100
4. 180
5. 180
6. CANCEL   --Timer J start
7. 200
8. CANCEL
9. 200
1. UA11 Send INVITE.
2. UA12 Receive INVITE.
3. UA11 Receive 100 Trying.
4. UA12 Send 180 Ringing.
5. UA11 Receive 180 Ringing.
6. UA11 Send CANCEL.
7. UA11 receive 200 OK. (*1)
8. UA12 receive CANCEL.
9. UA12 send 200 OK.
10. UA11 send CANCEL.
11. UA11 receive 200 OK. (*2)
12. UA11 Send CANCEL.

**[OBSERVABLE RESULTS]**

** this scenario checks only timing (message format is not checked, except 1xx response)

** response(481 is used below) may be 300-699 response.

*1:after 200 response from NUT.

Must not retransmit further 200 response. [RFC3261-17-83, 84, 85, 86]

*2:200 response from NUT. (CANCEL is sent every 2 seconds)
Must send 200 response. [RFC3261-17-80][RFC3261-17-81][RFC3261-17-82]
Must send response before Timer J (=64*T1 sec.) fires. [RFC3261-17-83]

*3: after Timer J fired. (CANCEL is sent every 2 seconds)

Must send 481 response. [RFC3261-17-85][RFC3261-17-86]

[REFERENCE]
[RFC3261-17-82, 83, 84, 85, 86]

17.2.2 Non-INVITE Server Transaction

When the server transaction enters the "Completed" state, it MUST set Timer J to fire in 64*T1 seconds for unreliable transports, and zero seconds for reliable transports. While in the "Completed" state, the server transaction MUST pass the final response to the transport layer for retransmission whenever a retransmission of the request is received. Any other final responses passed by the TU to the server transaction MUST be discarded while in the "Completed" state. The server transaction remains in this state until Timer J fires, at which point it MUST transition to the "Terminated" state.

The server transaction MUST be destroyed the instant it enters the "Terminated" state.

4.7.13 TS-4-1-2 - SIP Proxy- Non-INVITE Server Transaction (Forwarding of BYE upon Timer J fired)

[NAME]
TS-4-1-2 - SIP Proxy- Non-INVITE Server Transaction (forwarding of BYE upon Timer J fired)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT forwards a BYE request to that upstream after Timer J is fired.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]
Phase 2 Test Specification

SIP IPv6

NUT (AOR) sip: ss.under.test.com
Registrar (AOR) sip: reg.under.test.com
UA11 (AOR) sip: UA11@under.test.com
UA11 (Contact) sip: UA11@node.under.test.com
UA12 (AOR) sip: UA12@under.test.com
UA12 (Contact) sip: UA12@node11.under.test.com

[ADDRESS]

| NUT (IPv6) | 3ffe:501:ffff:50::50/64 |
| Registrar (IPv6) | 3ffe:501:ffff:50::60/64 |
| UA11 (IPv6) | 3ffe:501:ffff:1::1/64 |
| UA12 (IPv6) | 3ffe:501:ffff:2::2/64 |
| R (IPv6) | 3ffe:501:ffff:50::1/64 |

[TOPOLOGY]

---+-----------+---------
|           |          |
|          UA11 |
R11       |
---+---R-------+-----------+---------
|           |           |
|         NUT       Registrar |
R12       |
---+-----------+---------
|           |          |
|          UA12 |

[CONFIGURATION for NUT]

NUT sip: ss.under.test.com
NUT (IPADDRESS) 3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]

UA11 R NUT
| | | |
| | | |
|--------| | 1. ICMP Echo Request |
| | | |
|<--------| | 2. ICMP Echo Reply |
| | | |
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

<table>
<thead>
<tr>
<th>UA11</th>
<th>UA12</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2. 200 OK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4. 200 OK</td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------:------:</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Both Way RTP Media</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. INVITE
2. INVITE
3. 100
4. 180
5. 180
6. 200
7. 200
8. ACK
9. ACK
10. BYE —Timer J started
11. BYE
12. 200
13. 200 (*1)
1. UA11 Send INVITE.
2. UA12 Receive INVITE.
3. UA11 Receive 100 Trying.
4. UA12 Send 180 Ringing.
5. UA11 Receive 180 Ringing.
6. UA12 Send 200 OK.
7. UA11 Receive 200 OK.
8. UA11 Send ACK.
9. UA12 Receive ACK.
10. UA11 Send BYE.
11. UA12 Receive BYE.
12. UA12 Send 200 OK.
13. UA11 Receive 200 OK. (*1)
14. UA11 Send BYE.
15. UA11 Receive 200 OK. (*2)
16. UA11 Send BYE. (*2)
17. UA12 Receive BYE.
18. UA12 Send 481 Call/Transaction Does Not Exist.
19. UA11 Receive 481 Call/Transaction Does Not Exist. (*3)

[OBSERVABLE RESULTS]
** this scenario checks only timing (message format is not checked, except 1xx response)
** response(481 is used below) may be 300-699 response.

*1:* after 200 response from NUT.

Must not retransmit further 200 response. [RFC3261-17-83, 84, 85, 86]

*2:* 200 response from NUT. (BYE is sent every 2 seconds)

Must send 200 response. [RFC3261-17-80][RFC3261-17-81][RFC3261-17-82]
Must send response before Timer J(=64*T1 sec.) fires. [RFC3261-17-83]

*3:* after Timer J fired. (BYE is sent every 2 seconds)

Must send 481 response. [RFC3261-17-85][RFC3261-17-86]

[REFERENCE]
[RFC3261-17-82, 83, 84, 85, 86]
17.2.2 Non-INVITE Server Transaction

When the server transaction enters the "Completed" state, it MUST set Timer J to fire in 64*T1 seconds for unreliable transports, and zero seconds for reliable transports. While in the "Completed" state, the server transaction MUST pass the final response to the transport layer for retransmission whenever a retransmission of the request is received. Any other final responses passed by the TU to the server transaction MUST be discarded while in the "Completed" state. The server transaction remains in this state until Timer J fires, at which point it MUST transition to the "Terminated" state.

The server transaction MUST be destroyed the instant it enters the "Terminated" state.

4.7.14 TS-5-1-1 - SIP Proxy- Session Establishment Through One Proxy without Provisional Response in the same domain

[NAME]
TS-5-1-1 - SIP Proxy- Session Establishment Through One Proxy without Provisional Response in the same domain

[TARGET]
SIP Proxy
[PURPOSE]
Verify that a NUT properly processes when a session through a NUT is established without a provisional response in the same domain.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

| NUT(AOR)          | sip:ss.under.test.com |
| Registrar(AOR)    | sip:reg.under.test.com |
| UA11(AOR)         | sip:UA11@under.test.com |
| UA11(Contact)     | sip:UA11@node.under.test.com |
| UA12(AOR)         | sip:UA12@under.test.com |
| UA12(Contact)     | sip:UA12@node11.under.test.com |

[ADDRESS]

| NUT (IPv6)          | 3ffe:501:ffff:50::50/64 |
| Registrar (IPv6)    | 3ffe:501:ffff:50::60/64 |
| UA11(IPv6)          | 3ffe:501:ffff:1::1/64  |
| UA12(IPv6)          | 3ffe:501:ffff:2::2/64  |
| R(IPv6)             | 3ffe:501:ffff:50::1/64 |

[TOPOLOGY]

```
---+-----------+---------
 |           |
 |          UA11
R11
---+---R-------+-----------+---------
 |           |           |
 |         NUT       Registrar
R12
---+-----------+---------
 |           |
 |          UA12
```

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>
[INITIALIZE]

UA11 | R | NUT

----------> 1. ICMP Echo Request
<---------- 2. ICMP Echo Reply

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11 | UA12 | R | Registrar

----------> 1. REGISTER
<---------- 2. 200 OK
----------> 3. REGISTER
<---------- 4. 200 OK

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11 : NUT : UA12

----------> 1. INVITE
<---------- 2. 407
----------> 3. ACK
<---------- 4. INVITE
----------> 5. INVITE
<---------- 6. 100
----------> 7. 200
<---------- 8. 200 (*1)
Phase 2 Test Specification

SIP IPv6

|       :        |       :        | 9. ACK
|-------:------->|       :        | 10. ACK (*2)
|       :        |-------:------->| 11. ACK
|<===============================>| Both Way RTP Media
|       :        |       :        |
|<--------:------<|       :        |

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 200 OK.
8. UA11 Receive 200 OK. (*1)
9. UA11 Send ACK.
10. UA12 Receive ACK. (*2)
11. UA12 Send BYE.
12. UA11 Receive BYE.
13. UA11 Send 200.

=== Message example ===

4. INVITE UA11 -> NUT

INVITE sip-UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under .test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1cecc41e6cbe5aeb9c8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA12@under.test.com",
response="b51e504e73af54829e4f2bd7f8d6c4654"
From: UA11 <sip:UA11@under.test.com>;tag=9fxcce7d6sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE  
Content: <sip:UA11@node.under.test.com>  
Content-Type: application/sdp  
Content-Length: 151  

v=0  
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1  
s=  
c=IN IP6 3ffe:501:ffff:1::1  
t=0 0  
m=audio 49172 RTP/AVP 0  
a=rtpmap:0 PCMU/8000  

/* Proxy(NUT) accepts the credentials and forwards the INVITE to UA12  
2. Client for UA11 prepares to receive data on port 49172 from the network. */  

5. INVITE NUT -> UA12  

INVITE sip:UA12@node11.under.test.com SIP/2.0  
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1  
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9;received=3ffe:501:ffff:1::1  
Max-Forwards: 69  
Record-Route: <sip:ss.under.test.com;lr>  
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl  
To: UA12 <sip:UA12@under.test.com>  
Call-ID: 3848276298220188511@under.test.com  
CSeq: 2 INVITE  
Contact: <sip:UA11@node.under.test.com>  
Content-Type: application/sdp  
Content-Length: 151  

v=0  
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1  
s=  
c=IN IP6 3ffe:501:ffff:1::1  
t=0 0  
m=audio 49172 RTP/AVP 0  
a=rtpmap:0 PCMU/8000  

7. 200 OK UA12 -> NUT
SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
  ;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
  ;received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 384827629822018511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA12 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=*
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

8. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
  ;received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 384827629822018511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA12 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=*
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
9. ACK UA11 -> NUT

ACK sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b76
Max-Forwards: 70
Proxy-Authorization: Digest username="UA11",
realm="under.test.com",
nonce="f84f1ce41e66be5ae9e8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",
uri="sip:UA12@under.test.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
Route: <sip:ss.under.test.com:lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 ACK
Content-Length: 0

10. ACK NUT -> UA12

ACK sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK721e418c4.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b76
;received=fffc:501:ffff:1::1
Max-Forwards: 69
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 ACK
Content-Length: 0

[OBSERVABLE RESULTS]

*1:200 response from NUT to UA11.
As a SIP Message,
See generic_message

As a SIP response,
Must forward this response. [RFC3261-17-28]

- Status-Line:
  See generic_forward_from-UA12
Status-Code: Must be "200". [RFC3261-16-104]

- Header fields:
  See generic_forward_from-UA12
  See generic_forward_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from-UA12

*2:ACK request from NUT to UA12.

As a SIP Message,
  See generic_message

As a SIP request,
  Must forward this request. [RFC3261-17-28]

- Request-Line:
  See generic_forward_from-UA11
  See generic_forward_R-URI_responsible-domain

- Header fields:
  - outside of a dialog
    See generic_forward_from-UA11
    See generic_forward_request

- Bodies:
  See generic_forward_from-UA11

[REFERENCE]
[RFC3261-17-28, 29]

17 Transactions

17.1 Client Transaction

17.1.1 INVITE Client Transaction
17.1.1.2 Formal Description

When in either the "Calling" or "Proceeding" states, reception of a 2xx response MUST cause the client transaction to enter the "Terminated" state, and the response MUST be passed up to the TU. The handling of this response depends on whether the TU is a proxy core or a UAC core. A UAC core will handle generation of the ACK for this response, while a proxy core will always forward the 200 (OK) upstream. The differing treatment of 200 (OK) between proxy and UAC is the reason that handling of it does not take place in the transaction layer.

4.7.15 TS-5-1-2 - SIP Proxy- Session Establishment Through One Proxy with Multiple Provisional Responses in the same domain

(NAME)
TS-5-1-2 - SIP Proxy- Session Establishment Through One Proxy with Multiple Provisional Responses in the same domain

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when a session through a NUT is established with multiple provisional responses in the same domain.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501::fff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501::fff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501::fff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501::fff:2::2/64</td>
</tr>
</tbody>
</table>
**Phase 2 Test Specification**

**SIP IPv6**

---

| R(IPv6) | 3ffe:501:ffff:50::1/64 |
---|---|

**[TOPOLOGY]**

```
  +-----------+---------
  |           |         |
  |          UA11 |
  +---R-------+---------
  |           |         |
  |         NUT       Registrar |
  +-----------+---------
  |           |         |
  |         NUT       Registrar |
  +-----------+---------
  |           |         |
  |         UA12 |
```

**[CONFIGURATION for NUT]**

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

**[INITIALIZATION]**

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------:------&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>------:------&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------:-------&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------:-------&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------:-------&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------:-------&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------:-------&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;===============================&gt;</td>
<td>Both Way RTP Media</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;================================&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td></td>
</tr>
<tr>
<td>------:-------&gt;</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 180 Ringing.
10. UA11 Receive 180 Ringing. (*1)
11. UA12 Send 180 Ringing.
12. UA11 Receive 180 Ringing. (*2)
13. UA12 Send 200 OK.
14. UA11 Receive 200 OK. (*3)
15. UA11 Send ACK.
16. UA12 Receive ACK.
17. UA12 Send BYE.
18. UA11 Receive BYE.
19. UA11 Send 200.
20. UA12 Receive 200.

=== Message example ===

7.9.11. 180 Ringing UA12 -> NUT

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

8.10.12 180 Ringing NUT -> UA11

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

[OBSERVABLE RESULTS]
*1 2:180 response from NUT to UA11.
   As a SIP Message,
   See generic_message

   As a SIP response,
   Must forward this response. [RFC3261-17-63]

   · Status-Line:
     See generic_forward_from-UA12
     Status-Code: Must be "180". [RFC3261-16-104]

   · Header fields:
     See generic_forward_from-UA12
     See generic_forward_response
     * Via
       via-received: Must be added if the host portion of the "sent-by" parameter
       contains a domain name. [RFC3261-18-27]
       via-received: Must contain the source address from which the packet was
       received. [RFC3261-18-28]

   · Bodies:
     See generic_forward_from-UA12

*3:200 response from NUT to UA11.
   As a SIP Message,
   See generic_message

   As a SIP response,

   · Status-Line:
     See generic_forward_from-UA12
     Status-Code: Must be "200". [RFC3261-16-104]

   · Header fields:
     See generic_forward_from-UA12
     See generic_forward_response
     * Via
via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]

via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

- Bodies:
  See generic_forward_from-UA12

[REFERENCE]
[RFC3261-17-63]
17 Transactions

17.2 Server Transaction

17.2.1 INVITE Server Transaction

The TU passes any number of provisional responses to the server transaction. So long as the server transaction is in the "Proceeding" state, each of these MUST be passed to the transport layer for transmission. They are not sent reliably by the transaction layer (they are not retransmitted by it) and do not cause a change in the state of the server transaction.

4.7.16 TS-5-1-3 - SIP Proxy- Retransmission of INVITE Request Before Session Establishment Through One Proxy

[NAME]
TS-5-1-3 - SIP Proxy- Retransmission of INVITE Request Before Session Establishment Through One Proxy

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT sends a 180 response to that downstream when receiveing an INVITE request before establishing a session through a NUT.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip-reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>-------------------</td>
<td>-------------------------------------</td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:fffe:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:fffe:60::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:fffe:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:fffe:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:fffe:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

---+-----------+---------
|           |          |
|          UA11
R11
---+---R-------+-----------+---------
|           |           |
|         NUT       Registrar
R12
---+-----------+---------
|           |          |
|           |          |
|           |          |
|           |          |

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:fffe:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>&lt;--------&gt; 1. ICMP Echo Request</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.
<table>
<thead>
<tr>
<th>UA11</th>
<th>UA12</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>1. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>2. 200 OK</td>
</tr>
<tr>
<td>&lt;-----</td>
<td>------</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>3. REGISTER</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>4. 200 OK</td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;=================================</td>
<td>Both Way RTP Media</td>
<td></td>
</tr>
</tbody>
</table>
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA11 Send INVITE.
10. UA11 Receive 180 Ringing. (*1)
11. UA12 Send 200 OK.
12. UA11 Receive 200 OK. (*2)
13. UA11 Send ACK.
14. UA12 Receive ACK.
15. UA12 Send BYE.
16. UA11 Receive BYE.
17. UA11 Send 200.
18. UA12 Receive 200.

=== Message example ===

9. INVITE UA11 -> NUT

INVITE sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
\:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>:tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 384827629820188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 151
10. 180 Ringing NUT -> UA11

SIP/2.0 180 Ringing
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

12. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Type: application/sdp
Content-Length: 147

v=0
o=UA12 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
[OBSERVABLE RESULTS]

*1: 180 response from NUT to UA11.
   As a SIP Message,
   See generic_message
   
   As a SIP response,
   Must send 180 response. [RFC3261-17-64]
   
   - Status-Line:
     See generic_forward_from-UA12
     Status-Code: Must be "180". [RFC3261-16-104]
   
   - Header fields:
     See generic_forward_from-UA12
     See generic_forward_response
     * Via
     via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
     via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]
   
   - Bodies:
     See generic_forward_from-UA12

*2: 200 response from NUT to UA11.
   As a SIP Message,
   See generic_message
   
   As a SIP response,
   
   - Status-Line:
     See generic_forward_from-UA12
     Status-Code: Must be "200". [RFC3261-16-104]
   
   - Header fields:
     See generic_forward_from-UA12
     See generic_forward_response
     * Via
     via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
     via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]
- Bodies:
  See generic_forward_from-UA12

[REFERENCE]
[RFC3261-17-64]

17 Transactions

17.2 Server Transaction

17.2.1 INVITE Server Transaction

If a request retransmission is received while in the "Proceeding" state, the most recent provisional response that was received from the TU MUST be passed to the transport layer for retransmission. A request is a retransmission if it matches the same server transaction based on the rules of Section 17.2.3.

4.8 Session Progress

4.8.1 PG-1-1-1 - SIP Proxy- Session Progress response

[NAME]
PG-1-1-1 - SIP Proxy- Session Progress response

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes a 183 (Session Progress) response.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

PARAMETER

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]
**Phase 2 Test Specification**

**SIP IPv6**

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

**TOPOLOGY**

```
---+-----------+---------
       |           |
       |          UA11
R11    |
---+---R-------+-----------+---------
       |           |           |
       |         NUT       Registrar
R12    |
---+-----------+---------
       |           |
       |          UA12
```

**CONFIGURATION for NUT**

- NUT sip:ss.under.test.com:lr
- NUT(IPADDRESS) 3ffe:501:ffff:50::50/64 (IPv6)

**INITIALIZATION**

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

```
UA11  R  NUT
    |    |   |
    |    |   |
    |-------> 1. ICMP Echo Request
    |    |   |
    |<------ 2. ICMP Echo Reply
    |    |   |
```

```
UA11  UA12  R  Registrar
    |    |    |
    |    |    |
    |-------> 1. REGISTER
```
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td>:</td>
<td>1. INVITE</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>2. 407</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>3. ACK</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>4. INVITE</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>5. INVITE</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>6. 100</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>7. 183</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>8. 183 (*1)</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>9. 180</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>10. 180</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>11. 200</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>12. 200</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>13. ACK</td>
</tr>
<tr>
<td>&lt;------:--------</td>
<td></td>
<td>14. ACK</td>
</tr>
<tr>
<td>&lt;-------------------------&gt;</td>
<td>Both Way RTP Media</td>
<td></td>
</tr>
<tr>
<td>&lt;-------------------------&gt;</td>
<td>15. BYE</td>
<td></td>
</tr>
<tr>
<td>&lt;-------------------------&gt;</td>
<td>16. BYE</td>
<td></td>
</tr>
<tr>
<td>&lt;-------------------------&gt;</td>
<td>17. 200</td>
<td></td>
</tr>
<tr>
<td>&lt;-------------------------&gt;</td>
<td>18. 200</td>
<td></td>
</tr>
</tbody>
</table>
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 183 Session Progress.
8. UA11 Receive 183 Session Progress. (*1)
9. UA12 Send 180 Ringing.
10. UA11 Receive 180 Ringing.
11. UA12 Send 200 OK.
12. UA11 Receive 200 OK.
13. UA11 Send ACK.
14. UA12 Receive ACK.
15. UA12 Send BYE.
16. UA11 Receive BYE.
17. UA11 Send 200.
18. UA12 Receive 200.

--- Message example ---

7. 183 Session Progress UA12 -> NUT

SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
 :received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 384827629820188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

8. 183 Session Progress NUT -> UA11

SIP/2.0 183 Session Progress
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
 :received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
Contact: <sip:UA12@node11.under.test.com>
CSeq: 2 INVITE
Content-Length: 0

[OBSERVABLE RESULTS]
*1:183 response from NUT to UA11.
   As a SIP Message,
   Must be sent. [RFC3261-16-104]
   See generic_message

   As a SIP response,

   · Status-Line:
     See generic_forward_from-UA12
     Status-Code: Must be "183". [RFC3261-16-104]

   · Header fields:
     See generic_forward_from-UA12
     See generic_forward_response
     * Via
       via-received: Must be added if the host portion of the "sent-by" parameter
                    contains a domain name. [RFC3261-18-27]
       via-received: Must contain the source address from which the packet was
                    received. [RFC3261-18-28]

   · Bodies:
     See generic_forward_from-UA12

[REFERENCE]
   Sequence from RFC3665 Section 3.2.

[RFC3261-16-104]
16.7 Response Processing

   5. Check response for forwarding

   Until a final response has been sent on the server transaction,
   the following responses MUST be forwarded immediately:

   · Any provisional response other than 100 (Trying)

   · Any 2xx response
4.8.2 PG-1-1-2 - SIP Proxy- INVITE Client Transaction (Extension of Timer C)

[NAME]
PG-1-1-2 - SIP Proxy- INVITE Client Transaction (extension of Timer C)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT resets Timer C when the provisional responses with status codes 101 to 199.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
---+-----------+---------
|           |          |
|          UA11
R11
|   |       |
---+---R--+-+---+-----+-----+-+---+-----|
|           |          |
|         NUT       Registrar
R12
```
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:fff:50::50/64   (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

3. Send REGISTER Request.
4. Receive 200 OK response.

5. Send REGISTER Request.

[PROCEDURE]

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
|-------:------->|        :       | 1. INVITE
|       :        |--------:------>| 2. INVITE -- Timer B, C started
|<------:--------|        :       | 3. 100
|       :        |<-------:-------| 4. 180
---------------------------------------Timer C extended
|<------:--------|        :       | 5. 180
---------------------------------------Timer B fired
| :        :       |        :       | (*1)
| :        :       |        :       | (before timer C fires)
| :        <--------:--------| 6. 183
|       :        :       |
---------------------------------------Timer C extended
| :        :       |        :       | (*2)
| :        :       :       |
| :        :       :       | (before timer C fires)
| :        <--------:--------| 8. 183
|<------:--------|        :       | 9. 183
|       :        :       |
---------------------------------------Timer C extended
| :        :       |        :       | (*3)
| :        :       :       |

1. UA11 Send INVITE.
2. UA12 Receive INVITE.
3. UA11 Receive 100 Trying.
4. UA12 Send 180 Ringing.
5. UA11 Receive 180 Ringing.
(*1)
6. UA12 Send 183 Session Progress.
7. UA11 Receive 183 Session Progress.
(*2)
8. UA12 Send 183 Session Progress.
9. UA11 Receive 183 Session Progress.
(*3)

**[OBSERVABLE RESULTS]**

** this scenario checks only timing (message format is not checked)
** We assume that default value of Timer C as 3 minutes.
** We don't support resetting timer after Timer C fired.

*1: before timer C fires.
Must not send any message to other node. [RFC3261-16-90,91]

*2:after 183 response from UA12 to NUT.

Must not cancel the client transaction (that is, no CANCEL request sent from NUT). [RFC3261-16-90,95]

*3:after 183 response from UA12 to NUT.

Must not cancel the client transaction (that is, no CANCEL request sent from NUT). [RFC3261-16-90,95]

[REFERENCE]
[RFC3261-16-90, 91]

16.6 Request Forwarding

1. Set timer C

In order to handle the case where an INVITE request never generates a final response, the TU uses a timer which is called timer C. Timer C MUST be set for each client transaction when an INVITE request is proxied. The timer MUST be larger than 3 minutes. Section 16.7 bullet 2 discusses how this timer is updated with provisional responses, and Section 16.8 discusses processing when it fires.

[RFC3261-16-95, 96]

16.7 Response Processing

2. Update timer C for provisional responses

For an INVITE transaction, if the response is a provisional response with status codes 101 to 199 inclusive (i.e., anything but 100), the proxy MUST reset timer C for that client transaction. The timer MAY be reset to a different value, but this value MUST be greater than 3 minutes.

4.8.3 PG-1-2-1 - SIP Proxy- INVITE Client Transaction (Reset of Timer C)

[NAME]
PG-1-2-1 - SIP Proxy- INVITE Client Transaction (reset of Timer C)
[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly resets Timer C when Timer C is fired.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

| NUT (AOR) | sip:ss.under.test.com:lr |
|Registrar (AOR) | sip:reg.under.test.com |
|UA11 (AOR) | sip:UA11@under.test.com |
|UA11 (Contact) | sip:UA11@node.under.test.com |
|UA12 (AOR) | sip:UA12@under.test.com |
|UA12 (Contact) | sip:UA12@node11.under.test.com |

[ADDRESS]

| NUT (IPv6) | 3ffe:501:ffff:50::50/64 |
|Registrar (IPv6) | 3ffe:501:ffff:50::60/64 |
|UA11 (IPv6) | 3ffe:501:ffff:1:1/64 |
|UA12 (IPv6) | 3ffe:501:ffff:2:2/64 |
|R (IPv6) | 3ffe:501:ffff:50:1/64 |

[TOPOLOGY]

```
    |   |   |
    |   |   |
    |   |   |
    |   |   |
    |   |   |
    |   |   |
    |   |   |
    |   |   |
    |   |   |
```

[CONFIGURATION for NUT]

| NUT | sip:ss.under.test.com:lr |
|NUT (IPADDRESS) | 3ffe:501:ffff:50::50/64 (IPv6) |
[INITIALIZATION]

UA11  R  NUT
|   |   |   |
|   |   |   |
|<----|<----| 1. ICMP Echo Request
|   |   |   |
|   |   |   |
|   |   |   |

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11  UA12  R  Registrar
|   |   |   |   |
|   |   |   |   |
|<----|<----| 1. REGISTER
|   |   |   |
|   |   |   |
|   |   |   |
|<----|<----| 4. 200 OK
|   |   |   |
|   |   |   |

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11 : NUT : UA12
|   |   |   |
|   |   |   |
|   |   |   |
|<----|<----| 1. INVITE
|   |   |   |
|   |   |   |
|   |<----|<----| 2. INVITE -- Timer B, C started
|   |   |   |
|<----|<----| 3. 100
|   |   |   |
|   |<----|<----| 4. 180
|   |   |   |
|   |   |   |
|<----|<----| Timer B fired
|   |   |   |
|   |   |   |
|   |   |   |
|   |   |   |
|   |   |   |

1. Send INVITE Request.
2. Receive 200 OK response.
3. Send INVITE Request.
4. Receive 200 OK response.
5. Timer B fired
I Pv6 FORUM TECHNICAL DOCUMENT  IPv6 Ready Logo Program  Phase 2 Test Specification  SIP IPv6

1. UA11 Send INVITE.
2. UA12 Receive INVITE.
3. UA11 Receive 100 Trying.
4. UA12 Send 180 Ringing.
   (*1)
5. UA11 Receive 180 Ringing.
   (*1)
6. UA12 Receive CANCEL.  (*2)
7. UA12 Send 200 OK.
8. UA12 Send 487 Request Terminated.
9. UA12 Receive ACK.
10. UA11 Receive 487 Request Terminated.
11. UA11 Send ACK.

[OBSERVABLE RESULTS]
** this scenario checks only timing (message format is not checked)
** We assume that default value of Timer C as 3 minutes.
** We don’t support resetting timer after Timer C fired.

*1:before Timer C fires.

Must not send any message to other node. [RFC3261-16-90,91]

*2:CANCEL request from NUT to UA12.

Must be sent. [RFC3261-16-90,140,141]
16.6 Request Forwarding

11. Set timer C

In order to handle the case where an INVITE request never generates a final response, the TU uses a timer which is called timer C. Timer C MUST be set for each client transaction when an INVITE request is proxied. The timer MUST be larger than 3 minutes. Section 16.7 bullet 2 discusses how this timer is updated with provisional responses, and Section 16.8 discusses processing when it fires.

16.8 Processing Timer C

If timer C should fire, the proxy MUST either reset the timer with any value it chooses, or terminate the client transaction. If the client transaction has received a provisional response, the proxy MUST generate a CANCEL request matching that transaction. If the client transaction has not received a provisional response, the proxy MUST behave as if the transaction received a 408 (Request Timeout) response.

4.8.4 PG-1-2-2 - SIP Proxy- INVITE Client Transaction (No provisional response after Timer C fired)

[NAME]
PG-1-2-2 - SIP Proxy- INVITE Client Transaction (no provisional response after Timer C fired)

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT sends a 408 (Request Timeout) response when not having received any provisional response after Timer C fires.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.
**[PARAMETER]**

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip: ss.under.test.com:\lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip: reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip: <a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip: <a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip: <a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip: <a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

**[ADDRESS]**

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

**[TOPOLOGY]**

---+-----------+---------
| \_ |           | \_     |
|    |          UA11 |       |
| R11 |           |       |
---+---+-----------+---------
|   |           |           |
|   |         NUT     Registrar   |
| R12 |      |       |
|     |     |       |
---+-----------+---------
| \_ |           | \_UA12 |

**[CONFIGURATION for NUT]**

<table>
<thead>
<tr>
<th>NUT sip: ss.under.test.com:\lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS) 3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

**[INITIALIZATION]**

UA11 R NUT

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>----------</td>
<td>----------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>----------</td>
<td>----------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>----------</td>
<td>----------</td>
<td>----------</td>
</tr>
</tbody>
</table>

1. ICMP Echo Request
2. ICMP Echo Reply
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

1. UA11 Send INVITE.
2. UA12 Receive INVITE.
3. UA11 Receive 100 Trying.
4. UA11 Receive 480 Temporarily Unavailable. (*2)
5. UA11 Send ACK.

**[OBSERVABLE RESULTS]**

** this scenario checks only timing (message format is not checked)
** We assume that default value of Timer C as 3 minutes.
** We don't support resetting timer after Timer C fired.

*1:before timer C fires.

Must not send any message to other node. [RFC3261-16-90,91]

*2:408 response from NUT to UA11.

Must be sent. [RFC3261-16-90,140,141]

**[REFERENCE]**

[RFC3261-90,91]

16.6 Request Forwarding

11. Set timer C

In order to handle the case where an INVITE request never generates a final response, the TU uses a timer which is called timer C. Timer C MUST be set for each client transaction when an INVITE request is proxied. The timer MUST be larger than 3 minutes. Section 16.7 bullet 2 discusses how this timer is updated with provisional responses, and Section 16.8 discusses processing when it fires.

[RFC3261-16-139,140,141]

16.8 Processing Timer C

If timer C should fire, the proxy MUST either reset the timer with any value it chooses, or terminate the client transaction. If the client transaction has received a provisional response, the proxy MUST generate a CANCEL request matching that transaction. If the client transaction has not received a provisional response, the proxy MUST behave as if the transaction received a 408 (Request Timeout) response.
4.9 Transport

4.9.1 TP-1-1-1 - SIP Proxy- Receiving INVITE with additional bytes in a transport packet

[NAME]
TP-1-1-1 - SIP Proxy- Receiving INVITE with additional bytes in a transport packet

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly processes when receiving an INVITE request with additional bytes in a transport packet.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
+----------+       +----------+
|          |       |          |
|          |       |  UA11    |
| R11      |       |          |
+----------+       +----------+
```

---
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11 : NUT : UA12
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing. (*1)
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK.
11. UA11 Send ACK.
12. UA12 Receive ACK. (*2)
13. UA11 Send BYE.
14. UA12 Receive BYE.
15. UA12 Send 200 OK.
16. UA11 Receive 200 OK.

== Message example ==
5. INVITE NUT -> UA12

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Content-Type: application/sdp
Content-Length: 0

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1:1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

* Content-Length: 0

12. ACK NUT -> UA12

ACK sip:UA12@node11.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf76
:received=3ffe:501:ffff:1::1
Max-Forwards: 69
Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 ACK
Content-Length: 151

v=0
o=UA1 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
60
m=audio 49170 RTP/AVP 0
a=rtpmap:0 PCMU/8000

* Answer is included.
The port number in the m= line is different from 1.INVITE's.

[OBSERVABLE RESULTS]
*1:180 response from NUT to UA11.
   As a SIP Message,
   See generic_message

   As a SIP response,
   · Status-Line:
     See generic_forward_from-UA12
     Status-Code: Must be "180". [RFC3261-16-104]
   · Header fields:
     See generic_forward_from-UA12
     See generic_forward_response
     * Via
       via-received: Must be added if the host portion of the "sent-by" parameter
       contains a domain name. [RFC3261-18-27]
       via-received: Must contain the source address from which the packet was
       received. [RFC3261-18-28]

   · Bodies:
     See generic_forward_from-UA12

*2:ACK request from NUT to UA12.

   As a SIP Message,
   See generic_message

   As a SIP request,
   · Request-Line:
     See generic_forward_from-UA11
     See generic_forward_R-URI_non-responsible-domain
   · Header fields:
- outside of a dialog
  See generic_forward_from-UA11
  See generic_forward_request

- Bodies:
  See generic_forward_from-UA11

[REFERENCE]
[RFC3261-16-42, 43, 44, 45, 46]

16.6 Request Forwarding

1. Copy request

   The proxy starts with a copy of the received request. The copy
   MUST initially contain all of the header fields from the
   received request. Fields not detailed in the processing
   described below MUST NOT be removed. The copy SHOULD maintain
   the ordering of the header fields as in the received request.
   The proxy MUST NOT reorder field values with a common field
   name (See Section 7.3.1). The proxy MUST NOT add to, modify,
   or remove the message body.

   An actual implementation need not perform a copy: the primary
   requirement is that the processing for each next hop begin with
   the same request.

4.9.2 TP-1-2-1 - SIP Proxy- Transport packet of response ending before
the end of the message body

[NAME]
TP-1-2-1 - SIP Proxy- Transport packet of response ending before the end of the message
body

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT doesn't forwards a transport packet of a response that ends before the
end of a message body.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.
[PARAMETER]

<table>
<thead>
<tr>
<th>parameter</th>
<th>value</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (AOR)</td>
<td>sip:ss.under.test.com;lr</td>
</tr>
<tr>
<td>Registrar (AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11 (AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11 (Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12 (AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12 (Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>address type</th>
<th>address</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::50/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

---+-----------+---------
 |           |          |
 |          | UA11     |
 | R11      |           |
---+---R-------+-----------+---------
 |           |           |
 |         NUT       Registrar |
 | R12      |           |
---+-----------+---------
 |           |          |
 |        |           |
 | UA12    |               |

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT (IPADDRESS)</th>
<th>address</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT</td>
<td>sip:ss.under.test.com:lr</td>
</tr>
<tr>
<td></td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>---</td>
<td>-----</td>
</tr>
<tr>
<td>&lt;-----</td>
<td>---</td>
<td>-----</td>
</tr>
</tbody>
</table>

1. ICMP Echo Request

2. ICMP Echo Reply
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11     UA12        R     Registrar
|        |         |        |
|        |         |        |
|-------------------|-------->|1. REGISTER
|        |         |        |
|                   |         |        |
|                   |         |        |
|-------------------|-------->|2. 200 OK
|        |         |        |
|        |         |        |
|        |         |        |
|-------------------|-------->|3. REGISTER
|        |         |        |
|        |         |        |
|<--------|---------|4. 200 OK
|        |         |        |

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11 : NUT : UA12
| :     :     |
|       :     |
|       :     |
|-------:------->|       :     | 1. INVITE
|<------:--------|       :     | 2. 407
|-------:------->|       :     | 3. ACK
|       :     |
|-------:------->|       :     | 4. INVITE
|       :     |-------:------->| 5. INVITE
|<------:--------|       :     | 6. 100
|       :     |<------:--------| 7. 180
|<------:--------|       :     | 8. 180
|       :     |<------:--------| 9. 200
|<- - - X- - - --|       :     |(*1)
|       :     |
|       :     |<------:--------|10. 200
|<------:--------|       :     |11. 200
|       :     |
|-------:------->|       :     |12. ACK
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 200 OK with invalid Content-Length header field. (*1)
10. UA12 Send 200 OK.
11. UA11 Receive 200 OK.
12. UA11 Send ACK.
13. UA12 Receive ACK.
14. UA12 Send BYE.
15. UA11 Receive BYE.
16. UA11 Send 200.
17. UA12 Receive 200.

=== Message example ===

9. 200 OK UA12 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:fff5:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:fff1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501::50:50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:1::1
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Content-Type: application/sdp
Content-Length: 148

v=0
o=UA12 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

* Larger Content-Length value.
   Actual body size is equal to 148 bytes.

10. 200 OK UA12 -> NUT

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss.under.test.com:5060;branch=z9hG4bK2d4790.1
:received=3ffe:501:ffff:2::2
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
:received=3ffe:501:ffff:2::2
Record-Route: <sip:ss.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>;tag=314159
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA12@node11.under.test.com>
Content-Type: application/sdp
Content-Length: 350

v=0
o=UA12 2890844527 2890844527 IN IP6 3ffe:501:ffff:2::2
s=-
c=IN IP6 3ffe:501:ffff:2::2
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

[OBSERVABLE RESULTS]
*1:after 200 response from UA12 to NUT.

Must not forward this message to UA11, because Content-Length value is larger.
than size of body. [RFC3261-18-39]

[REFERENCE]
[RFC3261-18-38, 39]

18.3 Framing

In the case of message-oriented transports (such as UDP), if the message has a Content-Length header field, the message body is assumed to contain that many bytes. If there are additional bytes in the transport packet beyond the end of the body, they MUST be discarded. If the transport packet ends before the end of the message body, this is considered an error. If the message is a response, it MUST be discarded. If the message is a request, the element SHOULD generate a 400 (Bad Request) response. If the message has no Content-Length header field, the message body is assumed to end at the end of the transport packet.

4.9.3 TP-1-2-2 - SIP Proxy - Transport packet of request ending before the end of the message body

[NAME]
TP-1-2-2 - SIP Proxy - Transport packet of request ending before the end of the message body

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT properly generate a 400 (Bad Request) response when receiving a transport packet of a request that ends before the end of a message body.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip: ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip: reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip: <a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip: <a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip: <a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip: <a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]
NUT (IPv6) 3ffe:501:ffff:50::50/64  
Registrar (IPv6) 3ffe:501:ffff:50::60/64  
UA11(IPv6) 3ffe:501:ffff:1::1/64  
UA12(IPv6) 3ffe:501:ffff:2::2/64  
R(IPv6) 3ffe:501:ffff:50::1/64

[TOPOLOGY]

---+-----------+---------
|           |         |
|          UA11 |        |
R11
---+---R-------+---------
|           |         |
|         NUT       Registrar |
R12
---+-----------+---------
|           |         |
|         |        |
|          |
UA12

[CONFIGURATION for NUT]

| NUT sip:ss.under.test.com:lr |
| NUT(IPADDRESS) 3ffe:501:ffff:50::64 (IPv6) |

[INITIALIZATION]

UA11       R       NUT
|         |         |         |
|         |         |         |
|--------|--------|--------| 1. ICMP Echo Request
|         |         |         |
|<-------|---------| 2. ICMP Echo Reply |
|         |         |         |

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11       UA12       R       Registrar
|         |         |         |
|         |         |         |
|--------|--------|--------| 1. REGISTER
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

<table>
<thead>
<tr>
<th>UA11</th>
<th>NUT</th>
<th>UA12</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>------</td>
<td>1. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>------</td>
<td>2. 407</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>------</td>
<td>3. ACK</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>------</td>
<td>4. INVITE</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>------</td>
<td>5. 400 (*1)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;------</td>
<td>------</td>
<td>6. ACK</td>
</tr>
</tbody>
</table>

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA11 Receive 400 Bad Request. (*1)
6. UA11 Send ACK.

--- Message example ---

4. INVITE UA11 -> NUT

INVITE sip:UA12@under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Proxy-Authentication: Digest username="UA11",
realm="under.test.com",
nonce="f84f1cecc416cbe5ae9c8e88d359", opaque="",
qop=auth, nc=00000004, cnonce="6f54a149",

--- End of message example ---
uri="sip:UA12@under.test.com",
response="b51e504e73af54829e4f2bd7f8dc4654"
From: UA11 <sip:UA11@under.test.com>;tag=9fxced76sl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 3848276298220188511@under.test.com
CSeq: 2 INVITE
Contact: <sip:UA11@node.under.test.com>
Content-Type: application/sdp
Content-Length: 350

v=0
o=UA11 2890844526 2890844526 IN IP6 3ffe:501:ffff:1::1
s=-
c=IN IP6 3ffe:501:ffff:1::1
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

* Larger Content-Length value.
   Actual body size is equal to 148 bytes.

[OBSERVABLE RESULTS]
* 1:400 response from NUT to UA11.
   As a SIP Message,
      See generic_message
   As a SIP response,

   · Status-Line:
      See generic_make_response
      Status-Code: Must be "400", because Content-Length value is larger than size of
      body. [RFC3261-18-39,40,42]

   · Header fields:
      See generic_make_response
      * Via
         via-received: Must be added if the host portion of the "sent-by" parameter
         contains a domain name. [RFC3261-18-27]
         via-received: Must contain the source address from which the packet was
         received. [RFC3261-18-28]

[REFERENCE]
   [RFC3261-18-38,39]
18.3 Framing

In the case of message-oriented transports (such as UDP), if the message has a Content-Length header field, the message body is assumed to contain that many bytes. If there are additional bytes in the transport packet beyond the end of the body, they MUST be discarded. If the transport packet ends before the end of the message body, this is considered an error. If the message is a response, it MUST be discarded. If the message is a request, the element SHOULD generate a 400 (Bad Request) response. If the message has no Content-Length header field, the message body is assumed to end at the end of the transport packet.

4.9.4 TP-2-1-1 - SIP Proxy- Receipt of “ICMP time exceeded” for a sent request

[NAME]
TP-2-1-1 - SIP Proxy- Receiving “ICMP time exceeded” for a sent request

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT retransmits the request when receiving a “ICMP time exceeded” for previously a sent request.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501::ff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501::ff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501::ff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501::ff:2::2/64</td>
</tr>
</tbody>
</table>
[TOPOLOGY]

---+-----------+---------
|           |         |
|          UA11 |         |
R11
|           |         |
---+---R-------+-----------+---------
|           |           |
|         NUT       Registrar |         |
R12
|           |         |
---+-----------+---------
|           |         |
|         UA12 |         |

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:fff:50::50/64</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

UA11  R  NUT

|                   |                     |
|                   |                     |
|                   |                     |
|<------------------|---------|-------------------|1. REGISTER|
|                   |                     |
|                   |                     |
|-------|-------->|3. REGISTER|

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11  UA12  R  Registrar

|                   |                     |
|                   |                     |
|                   |                     |
|-------------------|--------|--|-------------------|1. REGISTER|
|                   |                     |
|                   |                     |
|-------------------|--------|--|-------------------|3. REGISTER|
|                   |                     |
1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

**[PROCEDURE]**

```
UA11  :  NUT  :  R  UA12
|   :   |   :   |
|   :   |   :   |
|-------:------->|   :   |
|<------:--------|   :   |
|-------:------->|   :   |
|   :   |   :   |
|-------:------->|   :   |
|<------:--------|   :   |
|-------:------->|   :   |
|   :   |   :   |
|-------:------->|   :   |
|<------:--------|   :   |
|-------:------->|   :   |
|   :   |   :   |
|-------:------->|   :   |
|<------:--------|   :   |
|-------:------->|   :   |
|   :   |   :   |
```

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. R Receive INVITE.
6. UA11 Receive 100 Trying.
7. R Send ICMP Time Exceeded.
8. UA12 Receive INVITE. (*1)
9. UA12 Receive INVITE. (*2)
10. UA11 Receive 408 Request Timeout.
11. UA11 Send ACK.

[OBSERVABLE RESULTS]
*1: INVITE request from NUT to UA12.
   Should be the same (retransmitted) INVITE. [RFC3261-18-43]

*2 INVITE request from NUT to UA12.
   Must be retransmitted with intervals that double after each transmission(2*T1).
   [RFC3261-8-41, RFC3261-17-8,9,10,14, RFC3261-18-43]

[REFERENCE]
[RFC3261-18-43]
18.4 Error Handling
   If the transport user asks for a message to be sent over an
   unreliable transport, and the result is an ICMP error, the behavior
   depends on the type of ICMP error. Host, network, port or protocol
   unreachable errors, or parameter problem errors SHOULD cause the
   transport layer to inform the transport user of a failure in sending.
   Source quench and TTL exceeded ICMP errors SHOULD be ignored.

4.9.5 TP-2-1-2 - SIP Proxy- Receipt of “ICMP time exceeded” for a sent response

[NAME]
TP-2-1-2 - SIP Proxy- Receipt of “ICMP time exceeded” for a sent response

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT retransmits the response when receiving a “ICMP time exceeded”
message for previously a sent response.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]
<table>
<thead>
<tr>
<th>NUT (AOR)</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11 (AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11 (Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12 (AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12 (Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

**ADDRESS**

<table>
<thead>
<tr>
<th>Address</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA11 (IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>UA12 (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
<tr>
<td>R (IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

**TOPOLOGY**

```
+-----------+---------+
|           |         |
|           | UA11    |
|           |         |
|-----------+---------|
|           |         |
| R11       |         |
|           | R       |
|           |         |
|-----------+---------|
|           |         |
|           | NUT     |
|           | Registrar|
|           |         |
|-----------+---------|
|           |         |
|           | R12     |
|           |         |
|-----------+---------|
|           |         |
|           |         |
|-----------+---------|
|           |         |
|           | UA12    |
```

**CONFIGURATION for NUT**

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT (IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

**INITIALIZATION**

```
<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-------</td>
<td>---</td>
<td>-----</td>
</tr>
<tr>
<td>&lt;-----</td>
<td>---</td>
<td>1.ICMP Echo Request</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>&lt;-----</td>
<td>---</td>
<td>2.ICMP Echo Reply</td>
</tr>
</tbody>
</table>
```
1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

```
UA11   UA12   R   Registrar
|       |       |       |
|-------------------|-------->|1.REGISTER
|       |       |       |
|<------------------|---------|2.200 OK|
|       |       |       |
|       |       |       |
|       |       |       |
|       |       |       |
|       |       |       |
|       |       |       |
|       |       |       |
|       |       |       |
```

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]
```
UA11    R    :      NUT      :      UA12
|       |       |       |       |
|       |       |       |       |
|-------:-------->     |       |       | 1. INVITE
|       |       |       |       |
|<--------:----------|     |       | 2. 407
|       |       |       |       |
|-------:-------->     |       |       | 3. ACK
|       |       |       |       |
|-------:-------->     |       |       | 4. INVITE
|       |       |       |       |
|<--------:----------|     |       | 5. INVITE
|       |       |       |       |
|-------:-------->     |       |       | 6. 100
|       |       |       |       |
|<--------:----------|     |       | 7. 180
|       |       |       |       |
|-------:-------->     |       |       | 8. 180
|       |       |       |       |
|       |<-------:-------     |       |       | 9. 200
|       |<-------:-------     |       |       |
|       |<-------:-------     |       |       | 10. 200
|       |<-------:-------     |       |       | 11. ICMP Time Exceeded
|       |<-------:-------     |       |       |       (500ms from 9. 200)
|       |<-------:-------     |       |       |       |
|<--------:----------     |       |       | 12. 200 (*1)
|       |       |       |       |
```
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 200 OK.
10. R Receive 200 OK.
11. R Send ICMP Time Exceeded.
12. UA11 Receive 200 OK. (*1)
13. UA11 Send ACK.
14. UA12 Receive ACK.
15. UA12 Send BYE.
16. UA11 Receive BYE.
17. UA11 Send 200 OK.
18. UA12 Receive 200 OK.

[OBSERVABLE RESULTS]
*1: 200 response from NUT to UA11.

Should be the same (retransmitted) 200 response. [RFC3261-18-43]

[REFERENCE]
[RFC3261-18-42, 43]
18.4 Error Handling

If the transport user asks for a message to be sent over an unreliable transport, and the result is an ICMP error, the behavior depends on the type of ICMP error. Host, network, port or protocol.
unreachable errors, or parameter problem errors SHOULD cause the transport layer to inform the transport user of a failure in sending. Source quench and TTL exceeded ICMP errors SHOULD be ignored.

4.9.6 TP-2-2-1 - SIP Proxy- Receipt of the “ICMP destination unreachable” message for a sent request

[NAME]
TP-2-2-1 - SIP Proxy- Receiving the “ICMP destination unreachable” message for a sent request

[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT sends a 500 (Internal Server Error) to the UAC when receiving the “ICMP destination unreachable” message.

[REQUIREMENT]
Set up registrar server to use location service, if necessary.

[PARAMETER]

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip:reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip:<a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip:<a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip:<a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip:<a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

[ADDRESS]

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

```
---+-----------+---------
|           |          |
|          UA11 |
|          R11 |
```
R-------+-----------+---------
|           |           |
| NUT       Registrar |
| R12       |
---+-----------+---------
|
---+-----------+---------
| UA12      |

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com:lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:ffff:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

<table>
<thead>
<tr>
<th>UA11</th>
<th>R</th>
<th>NUT</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>---</td>
<td>-----</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

<table>
<thead>
<tr>
<th>UA11</th>
<th>UA12</th>
<th>R</th>
<th>Registrar</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>------</td>
<td>------</td>
<td>---</td>
<td>-----------</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.
[PROCEDURE]

UA11 : NUT : R11 : UA12
| : | : |
| : | : |
|------:------->| : | 1. INVITE |
|<------:--------| : | 2. 407 |
|-------:------->| : | 3. ACK |
|------:------->| : | 4. INVITE |
| : |-------:--| | 5. INVITE |
|-------:------->| : | 6. 100 |
| : |<------:---| | 7. ICMP Destination Unreachable |
|<------:--------| | 8. 500 (*1) |
|------:------->| | 9. ACK |

1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. R11 Receive INVITE.
6. UA11 Receive 100 Trying.
7. R11 Send ICMP Destination Unreachable.
8. UA11 Receive 500 Internal Server Error. (*1)
9. UA11 Send ACK.

[OBSERVABLE RESULTS]

*1: 500 response from NUT to UA11.

As a SIP Message,
See generic_message

As a SIP response,

- Status-Line:
  See generic_forward_from-PX2
  Status-Code: Must be "500". [RFC3261 16.7.6]

- Header fields:
  See generic_forward_from-PX2
  See generic_forward_response
* Via
via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]

via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
[RFC3261-16-118, 119]
16.7 Response Processing

6. Choosing the best response

A proxy which receives a 503 (Service Unavailable) response SHOULD NOT forward it upstream unless it can determine that any subsequent requests it might proxy will also generate a 503. In other words, forwarding a 503 means that the proxy knows it cannot service any requests, not just the one for the Request-URI in the request which generated the 503. If the only response that was received is a 503, the proxy SHOULD generate a 500 response and forward that upstream.

[RFC3261-16-142]
16.9 Handling Transport Errors

If the transport layer notifies a proxy of an error when it tries to forward a request (see Section 18.4), the proxy MUST behave as if the forwarded request received a 503 (Service Unavailable) response.

[RFC3261-18-42, 43]
18.4 Error Handling

If the transport user asks for a message to be sent over an unreliable transport, and the result is an ICMP error, the behavior depends on the type of ICMP error. Host, network, port or protocol unreachable errors, or parameter problem errors SHOULD cause the transport layer to inform the transport user of a failure in sending. Source quench and TTL exceeded ICMP errors SHOULD be ignored.

4.10 Authentication

4.10.1 AU-1-1-1 - SIP Proxy- BYE request with user authentication

[NAME]
AU-1-1-1 - SIP Proxy- BYE request with user authentication
[TARGET]
SIP Proxy

[PURPOSE]
Verify that a NUT sends a 407 (Proxy Authentication Required) response when receiving a BYE request with user authentication.

[REQUIREMENT]
Only when a proxy supports authentication for BYE request.
Set up registrar server to use location service, if necessary.

PARAMETER

<table>
<thead>
<tr>
<th>NUT(AOR)</th>
<th>sip: ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar(AOR)</td>
<td>sip: reg.under.test.com</td>
</tr>
<tr>
<td>UA11(AOR)</td>
<td>sip: <a href="mailto:UA11@under.test.com">UA11@under.test.com</a></td>
</tr>
<tr>
<td>UA11(Contact)</td>
<td>sip: <a href="mailto:UA11@node.under.test.com">UA11@node.under.test.com</a></td>
</tr>
<tr>
<td>UA12(AOR)</td>
<td>sip: <a href="mailto:UA12@under.test.com">UA12@under.test.com</a></td>
</tr>
<tr>
<td>UA12(Contact)</td>
<td>sip: <a href="mailto:UA12@node11.under.test.com">UA12@node11.under.test.com</a></td>
</tr>
</tbody>
</table>

ADDRESS

<table>
<thead>
<tr>
<th>NUT (IPv6)</th>
<th>3ffe:501:ffff:50::50/64</th>
</tr>
</thead>
<tbody>
<tr>
<td>Registrar (IPv6)</td>
<td>3ffe:501:ffff:50::60/64</td>
</tr>
<tr>
<td>UA11(IPv6)</td>
<td>3ffe:501:ffff:1::1/64</td>
</tr>
<tr>
<td>UA12(IPv6)</td>
<td>3ffe:501:ffff:2::2/64</td>
</tr>
<tr>
<td>R(IPv6)</td>
<td>3ffe:501:ffff:50::1/64</td>
</tr>
</tbody>
</table>

[TOPOLOGY]

---+-----------+---------
|           |          |
|          UA11 |         |
R11

---+---R-------+-----------+---------
|           |           |           |
|         NUT       Registrar |
R12

---+-----------+---------
|           |          |
|          UA12 |

[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com;lrd</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:fff:50:50/64 IPv6</td>
</tr>
</tbody>
</table>

[INITIALIZATION]


1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.


1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]


1. INVITE
2. 407
3. ACK
4. INVITE
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK.
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA11 Send BYE.
14. UA11 Receive 407 Proxy Authentication Required. (*1)
15. UA11 Send BYE.
16. UA12 Receive BYE.
17. UA12 Send 200 OK.
18. UA11 Receive 200 OK.

--- Message example ---
14. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxc676sl
To: UA12 <sip:UA12@under.test.com>;tag=3flal12sf
Call-ID: 3848276298220188511@under.test.com
CSeq: 3 BYE
Proxy-Authenticate: Digest realm="under.test.com", qop="auth",
nonce="f84f1ce41e6cbe5aa9c8e8d359",
opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

/* Proxy(NUT) challenges UA11 for authentication to BYE request */

[OBSERVABLE RESULTS]

** precondition for testing this scenario:
- NUT must be able to send an authentication challenge to BYE

*1:407 response from NUT to UA11.
  As a SIP Message,
  See generic_message

  As a SIP response,

  - Status-Line:
    See generic_make_response
    Status-Code: Must be "407". [RFC3261 22.3]

  - Header fields:
    See generic_make_response
    See generic_proxy-auth
    * Via
    via-received: Must be added if the host portion of the "sent-by" parameter
    contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was
    received. [RFC3261-18-28]

[REFERENCE]
NONE
4.10.2 AU-1-1-2 - SIP Proxy- re-INVITE with user authentication

**[NAME]**
AU-1-1-2 - SIP Proxy- re-INVITE with user authentication

**[TARGET]**
SIP Proxy

**[PURPOSE]**
Verify that a NUT sends a 407 (Proxy Authentication Required) response when receiving a re-INVITE request with user authentication.

**[REQUIREMENT]**
Only when a proxy supports authentication for re-INVITE request.
Set up registrar server to use location service, if necessary.

**[PARAMETER]**

| NUT(AOR)          | sip: ss.under.test.com;lر |
| Registrar(AOR)   | sip: reg.under.test.com   |
| UA11(AOR)        | sip: UA11@under.test.com  |
| UA11(Contact)    | sip: UA11@node.under.test.com |
| UA12(AOR)        | sip: UA12@under.test.com  |
| UA12(Contact)    | sip: UA12@node11.under.test.com |

**[ADDRESS]**

| NUT (IPv6)       | 3ffe:501:ffff:50::50/64 |
| Registrar (IPv6) | 3ffe:501:ffff:50::60/64 |
| UA11(IPv6)       | 3ffe:501:ffff:1::1/64   |
| UA12(IPv6)       | 3ffe:501:ffff:2::2/64   |
| R(IPv6)          | 3ffe:501:ffff:50::1/64  |

**[TOPOLOGY]**

```
---+-----------+---------
    |           |
    |          UA11
    |  R11     |
    |           |
---+---R-------+-----------+---------
    |           |           |
    |         NUT       Registrar
    |  R12     |
    |           |
```

IPv6 FORUM TECHNICAL DOCUMENT
IPv6 Ready Logo Program
Phase 2 Test Specification
SIP IPv6
[CONFIGURATION for NUT]

<table>
<thead>
<tr>
<th>NUT</th>
<th>sip:ss.under.test.com;lr</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUT(IPADDRESS)</td>
<td>3ffe:501:fff5:50::50/64 (IPv6)</td>
</tr>
</tbody>
</table>

[INITIALIZATION]

UA11  R  NUT

1. Send ICMP Echo Request.
2. Receive ICMP Echo Reply.

UA11  UA12  R  Registrar

1. Send REGISTER Request.
2. Receive 200 OK response.
3. Send REGISTER Request.
4. Receive 200 OK response.

[PROCEDURE]

UA11  :  NUT  :  UA12

1. INVITE
2. 407
Phase 2 Test Specification

SIP IPv6

| ------:-----> |       :      | 3. ACK |
|      :      |       :      |
| ------:-----> |       :      | 4. INVITE |
|      :      |       :      |
| <-----:------|       :      | 5. INVITE |
|      :      |       :      |
| <-----:------|       :      | 6. 100 Trying |
|      :      |       :      |
| <-----:------|       :      | 7. 180 Ringing |
|      :      |       :      |
| <-----:------|       :      | 8. 180 Ringing |
|      :      |       :      |
| <-----:------|       :      | 9. 200 OK |
|      :      |       :      |
| <-----:------|       :      | 10. 200 OK |
|      :      |       :      |
| ------:-----> |       :      | 11. ACK |
|      :      |       :      |
| ------:-----> |       :      | 12. ACK |
|      :      |       :      |
| <==========================> | Both Way RTP Media Established |
|      :      |       :      |
|      :      |       :      |
|      :      |       :      |
| ------:-----> |       :      | 13. INVITE |
|      :      |       :      |
| <-----:------|       :      | 14. 407 (*1) |
|      :      |       :      |
| ------:-----> |       :      | 15. ACK |
|      :      |       :      |
| ------:-----> |       :      | 16. INVITE |
|      :      |       :      |
|      :      |       :      |
|      :      |       :      |
| ------:-----> |       :      | 17. INVITE |
|      :      |       :      |
| <-----:------|       :      | 18. 200 OK |
|      :      |       :      |
| ------:-----> |       :      | 19. 200 OK |
|      :      |       :      |
| ------:-----> |       :      | 20. ACK |
|      :      |       :      |
| ------:-----> |       :      | 21. ACK |
|      :      |       :      |
|      :      |       :      |
|      :      |       :      |
| ------:-----> |       :      | HOLD |
|      :      |       :      |
|      :      |       :      |
| ------:-----> |       :      | 22. INVITE |
|      :      |       :      |
| <-----:------|       :      | 23. 407 (*2) |
|      :      |       :      |
| ------:-----> |       :      | 24. ACK |
|      :      |       :      |
| ------:-----> |       :      | 25. INVITE |
|      :      |       :      |
|      :      |       :      |
|      :      |       :      |
| ------:-----> |       :      | 26. INVITE |
|      :      |       :      |
| <-----:------|       :      | 27. 200 OK |
|      :      |       :      |
| ------:-----> |       :      | 28. 200 OK |
|      :      |       :      |
| ------:-----> |       :      | 29. ACK |
|      :      |       :      |
| ------:-----> |       :      | 30. ACK |
|      :      |       :      |
| <==========================> | New RTP Media Stream |
|      :      |       :      |
1. UA11 Send INVITE.
2. UA11 Receive 407 Proxy Authentication Required.
3. UA11 Send ACK.
4. UA11 Send INVITE.
5. UA12 Receive INVITE.
6. UA11 Receive 100 Trying.
7. UA12 Send 180 Ringing.
8. UA11 Receive 180 Ringing.
9. UA12 Send 200 OK.
10. UA11 Receive 200 OK.
11. UA11 Send ACK.
12. UA12 Receive ACK.
13. UA11 Send INVITE.
14. UA11 Receive 407 Proxy Authentication Required. (1)
15. UA11 Send ACK.
16. UA11 Send INVITE.
17. UA12 Receive INVITE
18. UA12 Send 200 OK.
19. UA11 Receive 200 OK.
20. UA11 Send ACK.
21. UA12 Receive ACK.
22. UA11 Send INVITE.
23. UA11 Receive 407 Proxy Authentication Required. (2)
24. UA11 Send ACK.
25. UA11 Send INVITE.
26. UA12 Receive INVITE.
27. UA12 Send 200 OK.
28. UA11 Receive 200 OK.
29. UA11 Send ACK.
30. UA12 Receive ACK.
31. UA11 Send BYE.
32. UA12 Receive BYE.
33. UA12 Send 200 OK.
34. UA11 Receive 200 OK.

--- Message example ---
14. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK74b43
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxc6d76sl
To: UA12 <sip:UA12@under.test.com>;tag=3fla12sf
Call-ID: 3848276298220188511@under.test.com
CSeq: 6 INVITE
Proxy-Authenticate: Digest realm="under.test.com", qop="auth",
nonce="thkr983j76h5gut77meka93ol43m2",
opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

/* Proxy(NUT) challenges UA11 for authentication to re-INVITE */

23. 407 Proxy Authentication Required NUT -> UA11

SIP/2.0 407 Proxy Authentication Required
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bK87jyHtgh8
:received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=9fxc6d76sl
To: UA12 <sip:UA12@under.test.com>;tag=3fla12sf
Call-ID: 3848276298220188511@under.test.com
CSeq: 7 INVITE
Proxy-Authenticate: Digest realm="under.test.com", qop="auth",
nonce="f84f1ce41e6c8e88d359",
opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

/* Proxy(NUT) challenges UA11 for authentication to re-INVITE */

** OBSERVABLE RESULTS **
precondition for testing this scenario:
- NUT must be able to send an authentication challenge to re-INVITE

*1:407 response from NUT to UA11.
As a SIP Message,
See generic_message

As a SIP response,
- Status-Line:
  See generic_make_response
Status-Code: Must be "407". [RFC3261 22.3]

- Header fields:
  See generic_make_response
  See generic_proxy-auth
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

*2:407 response from NUT to UA11.
As a SIP Message,
  See generic_message

As a SIP response,

- Status-Line:
  See generic_make_response
  See generic_proxy-auth
  Status-Code: Must be "407". [RFC3261 22.3]

- Header fields:
  See generic_make_response
  * Via
    via-received: Must be added if the host portion of the "sent-by" parameter contains a domain name. [RFC3261-18-27]
    via-received: Must contain the source address from which the packet was received. [RFC3261-18-28]

[REFERENCE]
  NONE
Copyright (C) 2005-2010 IPv6 Forum. All Rights Reserved.

This original documentation is produced by SIP IPv6 SWG members of Certification WG in the IPv6 Promotion Council. The SWG members currently include Nippon Telegraph and Telephone Corporation (NTT), Yokogawa Electric Corporation, University of New Hampshire InterOperability Laboratory (UNH-IOL), and NTT Advanced Technology Corporation (NTT-AT).

No part of this documentation may be reproduced for any purpose without prior permission.
AUTHORS’ LIST

Hiroshi Miyata (Yokogawa Electric Corporation)
Yukiyo Akisada (Yokogawa Electric Corporation)
Timothy Winters(UNH-IOL)
James Swan(UNH-IOL)
Yoshio Yoshida (NTT-AT)
Kenzo Kodama (NTT-AT)
Yoshihiro Inoue (NTT-AT)
Naomi Orimo(NTT-AT)