IPv6 Ready Logo Phase 2
Session Initiation Protocol

Test Profile
Registrar Server

Version 2.0.0

IPv6 Forum
Converged Test Specification
IPv6 Logo Committee
IPv6 Promotion Council (Japan)

http://www.ipv6forum.org
http://www.ipv6ready.org

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

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 tests 3
3 Common OBSERVABLE RESULTS 4
3.1 generic_message 4
3.2 generic_make_response 4
3.3 generic_status 5
3.4 generic_www-auth 6
4. Test Profile: Server operation 7
4.1 Registration 7
4.1.1 RG-1-1-1 - SIP Registrar- Successful New Registration 7
4.1.2 RG-1-1-2 - SIP Registrar- Update of Contact List 14
4.1.3 RG-1-1-3 - SIP Registrar- Request for Current Contact List 19
4.1.4 RG-1-1-4 - SIP Registrar- Cancellation of Registration 24
4.1.5 RG-1-1-5 - SIP Registrar- REGISTER without Expires header field 28
4.1.6 RG-1-1-6 - SIP Registrar- processing of REGISTER requests in the correct order. 32
4.1.7 RG-1-1-7 - SIP Registrar- REGISTER with two Contact header fields with separate expires parameters 40
4.1.8 RG-1-2-1 - SIP Registrar- Unsuccessful Registration (Wrong user ID / password) 45
4.1.9 RG-1-2-2 - SIP Registrar- Expiration of REGISTER less than a registrar-configured minimum 50
4.1.10 RG-1-2-3 - SIP Registrar- REGISTER with the equal CSeq value to or lower than the last value of CSeq 54
4.1.11 RG-1-2-4 - SIP Registrar- The case of receiving Contact header field includeing the value "*" 60
4.1.12 RG-2-1-1 - SIP Registrar- Record-Route in REGISTER request 69
4.1.13 RG-2-1-2 - SIP Registrar- REGISTER with the different Call-ID value 74
4.1.14 RG-2-1-3 - SIP Registrar- Sequence Number is not Incremented. 81
4.1.15 RG-2-1-4 - SIP Registrar- URI with user-param in To header field in REGISTER . 87
4.1.16 RG-2-1-5 - SIP Registrar- URI with escaped characters in To header field in REGISTER 94
4.1.17 RG-2-2-1 - SIP Registrar- REGISTER with an invalid AoR 100
4.1.18 RG-2-2-2 - SIP Registrar- Illegal parameter value in Contact header field 103
4.1.19 RG-2-2-3 - SIP Registrar- REGISTER request with two Contact header field values and update failure 107
4.1.20 RG-3-1-1 - SIP Registrar- Forwarded REGISTER request 115
4.1.22 RG-4-1-1 - SIP Registrar- Future extension about new header field 120

4.1.23 RG-4-1-2 - SIP Registrar- Non-supported option-tag in Require header field 125
AUTHORS' LIST 131

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 as 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

NUT	- Node Under Test (applicant implementation)
UA	- SIP User Agent
EP	- SIP Endpoint
B2BUA	- SIP Back to Back User Agent

RG - SIP Registrar Server
PX - SIP Proxy Sever
IF - Interface
UNI - User-Network Interface
NNI - Network-Network Interface

Reference standards

(1) RFC3261: SIP: Session Initiation Protocol (http://www.ietf.org/rfc/rfc3261.txt)
(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.

2.3 Index of BASIC tests

Test rank	Function	Test category	Profile No
BASIC	- Digest authentication (REGISTER) - Registration	Registration	RG-1-1-1 RG-1-1-2 RG-1-1-3 RG-1-1-4 RG-1-1-5 RG-1-1-6 RG-1-1-7 RG-1-2-1 RG-1-2-2 RG-1-2-3 RG-1-2-4 RG-2-1-1 RG-2-1-2 RG-2-1-3 RG-2-1-4 RG-2-1-5 RG-2-2-1 RG-2-2-2 RG-2-2-3 RG-4-1-1 RG-4-1-2

ADVANCED	- Processing of forwarded REGISTER request	Registration	RG-3-1-1

3 Common OBSERVABLE RESULTS

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_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. [RFCC3261.20]
    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]

3.3 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:

* 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]

3.4 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: Server operation

4.1 Registration

4.1.1 RG-1-1-1 - SIP Registrar- Successful New Registration

[NAME]

RG-1-1-1 - SIP Registrar- Successful New Registration

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly processes a REGISTER request for new registration.

[REQUIREMENT]

NONE

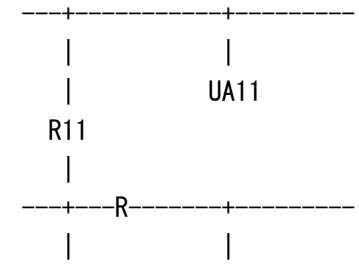
[PARAMETER]

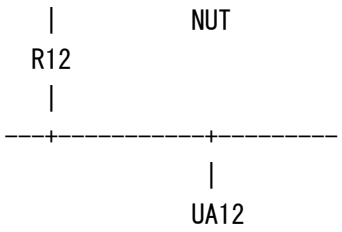
NUT(AOR)	sip:ss.under.test.com
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com
UA12(AOR)	sip:UA12@under.test.com
UA12(Contact)	sip:UA12@node11.under.test.com

[ADDRESS]

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

[TOPOLOGY]

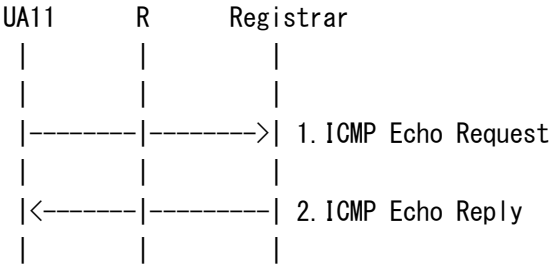




[CONFIGURATION for NUT]

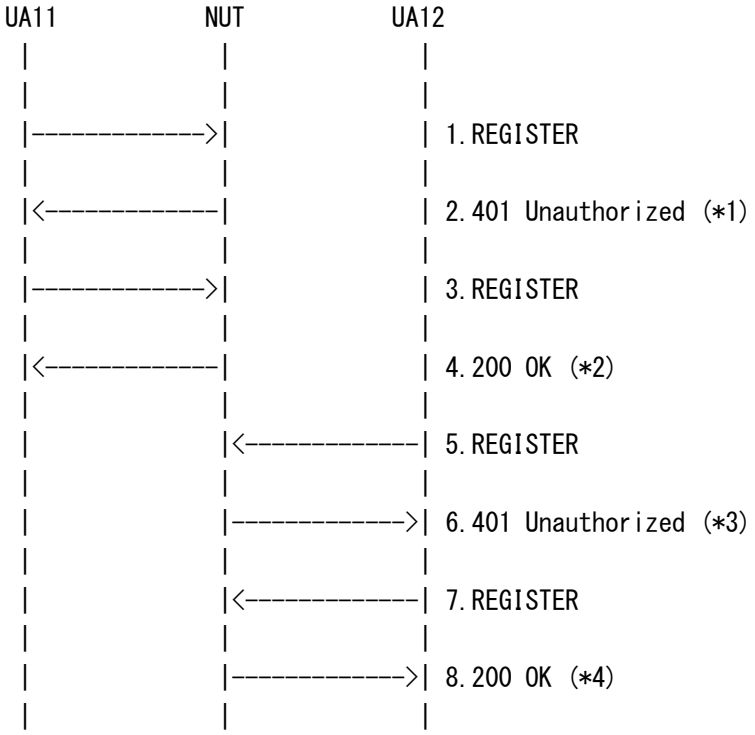
NUT	sip:ss.under.test.com
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized. (*1)
3. UA11 Send REGISTER.
4. UA11 Receive 200 OK. (*2)
5. UA12 Send REGISTER.
6. UA12 Receive 401 Unauthorized. (*3)
7. UA12 Send REGISTER.
8. UA12 Receive 200 OK. (*4)

=== Message example ===

1. REGISTER UA11 -> NUT

```
REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0
```

2. 401 Unauthorized NUT -> UA11

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0
```

3. REGISTER UA11 -> NUT

```
REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
```

From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

5. REGISTER UA12 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnasgh7i
Max-Forwards: 70
From: UA12 <sip:UA12@under.test.com>;tag=a73kszlfl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 9k6hJKItGh4kI8d6@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA12@node11.under.test.com>
Expires: 3600
Content-Length: 0

6. 401 Unauthorized NUT -> UA12

SIP/2.0 401 Unauthorized

Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnasgh7i
 ;received=3ffe:501:ffff:5::X
From: UA12 <sip:UA12@under.test.com>;tag=a73kszlfl
To: UA12 <sip:UA12@under.test.com>;tag=1410948204
Call-ID: 9k6hJKItGh4kI8d6@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="1cec4341ae6cbe5a359ea9c8e88df84f",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

7. REGISTER UA12 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnasgg8k
Max-Forwards: 70
From: UA12 <sip:UA12@under.test.com>;tag=ja743ks76zlflH
To: UA12 <sip:UA12@under.test.com>
Call-ID: 9k6hJKItGh4kI8d6@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA12@node11.under.test.com>
Expires: 3600
Authorization: Digest username="UA12",
 realm="under.test.com",
 nonce="1cec4341ae6cbe5a359ea9c8e88df84f", opaque="",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

8. 200 OK NUT -> UA12

SIP/2.0 200 OK
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnasgg8k
 ;received=3ffe:501:ffff:5::X
From: UA12 <sip:UA12@under.test.com>;tag=ja743ks76zlflH
To: UA12 <sip:UA12@under.test.com>;tag=37GkEhwl6
Call-ID: 9k6hJKItGh4kI8d6@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA12@node11.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:29:00 GMT
Content-Length: 0

[OBSERVABLE RESULTS]

*1:401 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]

*2:200 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 NUT. [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]

*3:401 response from NUT to UA12.
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]

*4:200 response from NUT to UA12.
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".[RFCrfc3261 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 NUT. [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]

RFC3665 SIP Basic Call Flow Examples
See Section 2.1

4.1.2 RG-1-1-2 - SIP Registrar- Update of Contact List

[NAME]

RG-1-1-2 - SIP Registrar- Update of Contact List

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly processes update of a Contact list

[REQUIREMENT]

NONE

[PARAMETER]

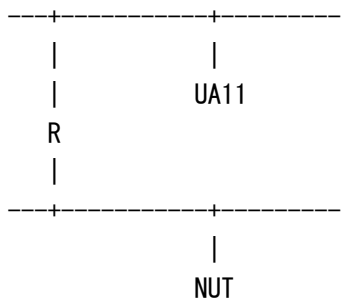
NUT(AOR)	sip:ss.under.test.com
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64

UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64

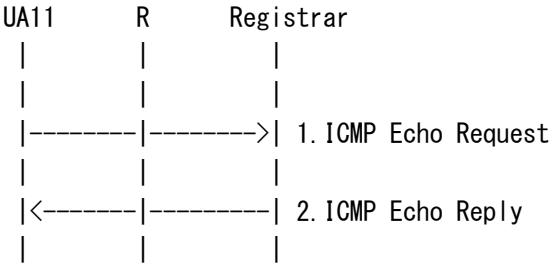
[TOPOLOGY]



[CONFIGURATION for NUT]

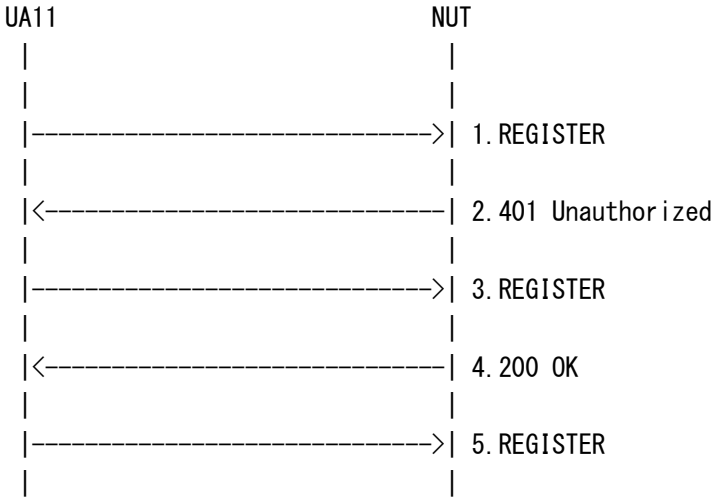
NUT	sip:ss.under.test.com
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



```
    |<-----------------------------| 6. 200 OK(*1)
    |                              |
```

1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized.
3. UA11 Send REGISTER.
4. UA11 Receive 200 OK.
5. UA11 Send REGISTER.
6. UA11 Receive 200 OK. (*1)

=== Message example ===

1. REGISTER UA11 -> NUT

```
REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0
```

2. 401 Unauthorized NUT -> UA11

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0
```

3. REGISTER UA11 -> NUT

```
REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
```

From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359", opaque="",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=30
Date: Sat,13 Nov 2004 23:29:00 GMT
Content-Length: 0

5. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=jS8Mwi2mq
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 3 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359", opaque="",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",

response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

6. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds9
;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=jS8Mwi2mq
To: UA11 <sip:UA11@under.test.com>;tag=34095828jh
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 3 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:29:30 GMT
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_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 NUT. [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]

RFC3665 SIP Basic Call Flow Examples

See Section 2.2

4.1.3 RG-1-1-3 - SIP Registrar- Request for Current Contact List

[NAME]

RG-1-1-3 - SIP Registrar- Request for Current Contact List

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly processes a request for a current contact list.

[REQUIREMENT]

NONE

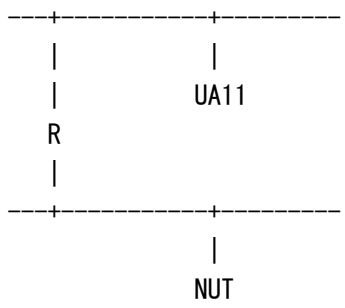
[PARAMETER]

NUT(AOR)	sip:ss.under.test.com
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64

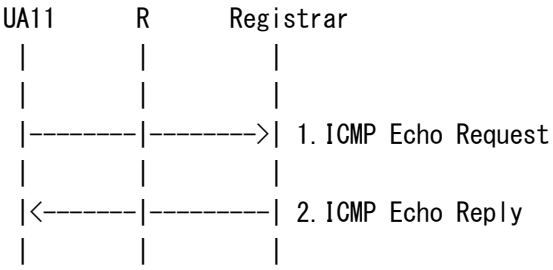
[TOPOLOGY]



[CONFIGURATION for NUT]

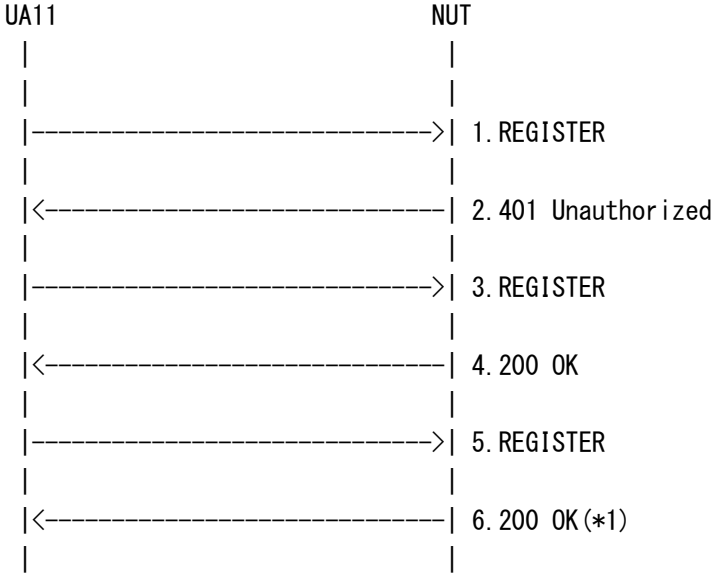
NUT	sip:ss.under.test.com
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64　(IPv6)

[INITIALIZATION]



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

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized.
3. UA11 Send REGISTER.
4. UA11 Receive 200 OK.
5. UA11 Send REGISTER.
6. UA11 Receive 200 OK. (*1)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0

2. 401 Unauthorized NUT -> UA11

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359", opaque="",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"

Content-Length: 0

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:29:00 GMT
Content-Length: 0

5. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=jS8Mwi2mq
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 3 REGISTER
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359", opaque="",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

* Contact is not exist.

6. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds9
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=jS8Mwi2mq
To: UA11 <sip:UA11@under.test.com>;tag=jqoiweu75
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com

CSeq: 3 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3590
Date: Sat,13 Nov 2004 23:29:10 GMT
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_status
Status-Code: Must be "200". [FC3261 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. [RFC3261-10-5-]
contact-param: Must be the Contact address of NUT. [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]

RFC3665 SIP Basic Call Flow Examples
See Section 2.3

4.1.4 RG-1-1-4 - SIP Registrar- Cancellation of Registration

[NAME]

RG-1-1-4 - SIP Registrar- Cancellation of Registration

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly processes a REGISTER for canceling a registration.

[REQUIREMENT]

NONE

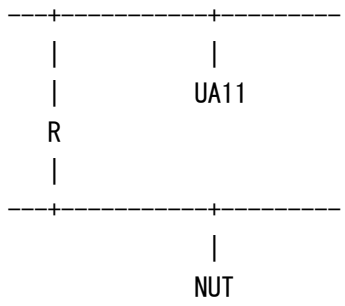
[PARAMETER]

NUT(AOR)	sip:ss.under.test.com
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64

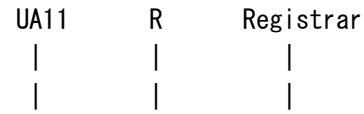
[TOPOLOGY]

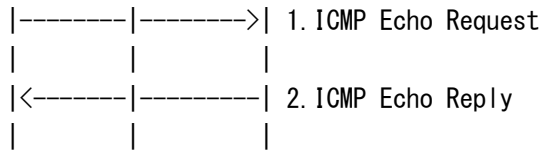


[CONFIGURATION for NUT]

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

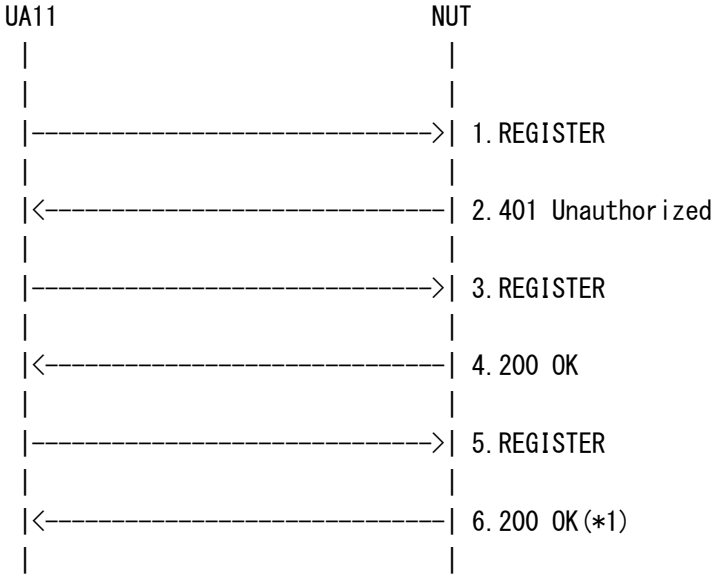
[INITIALIZATION]





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

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized.
3. UA11 Send REGISTER.
4. UA11 Receive 200 OK.
5. UA11 Send REGISTER.
6. UA11 Receive 200 OK. (*1)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>

Expires: 3600
Content-Length: 0

2. 401 Unauthorized NUT -> UA11

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359", opaque="",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6

Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Content-Length: 0

5. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=jS8Mwi2mq
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 3 REGISTER
Expires: 0
Contact: *
Content-Length: 0

6. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds9
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=jS8Mwi2mq
To: UA11 <sip:UA11@under.test.com>;tag=1418nmdsrf
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 3 REGISTER
Date: Sat,13 Nov 2004 23:29:00 GMT
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_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

contact-param: Must not include the specified parameter. [RFC3261-10-50],[RFC3261 10.2.2]

[REFERENCE]

RFC3665 SIP Basic Call Flow Examples

See Section 2.4

4.1.5 RG-1-1-5 - SIP Registrar- REGISTER without Expires header field

[NAME]

RG-1-1-5 - SIP Registrar- REGISTER without Expires header field

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly processes a REGISTER request without an Expires header field.

[REQUIREMENT]

NONE

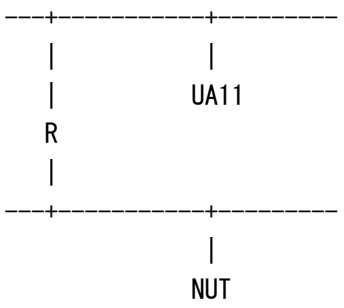
[PARAMETER]

NUT(AOR)	sip:ss.under.test.com;lr
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64

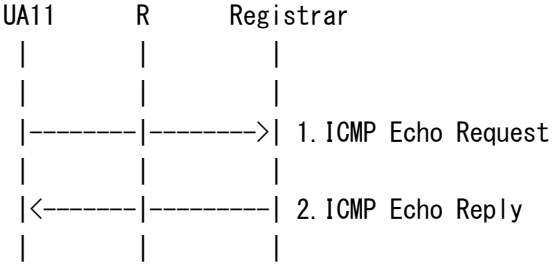
[TOPOLOGY]



[CONFIGURATION for NUT]

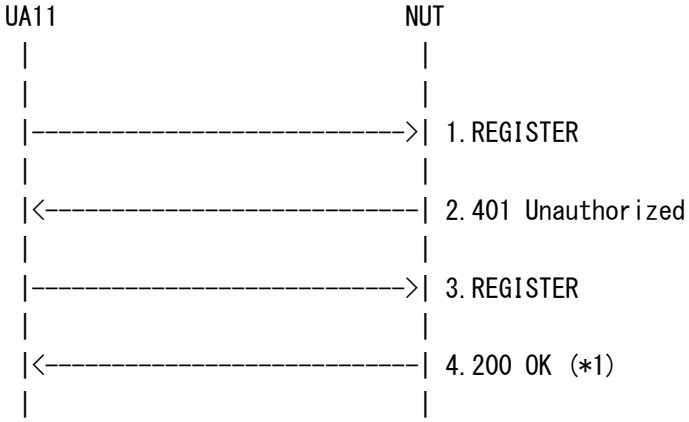
NUT	sip:ss.under.test.com;lr
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



1. UA11 Send REGISTER without Expires header field.
2. UA11 Receive 401 Unauthorized.
3. UA11 Send REGISTER without Expires header field.
4. UA11 Receive 200 OK. (*1)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Content-Length: 0

* No Expires header field exists

2. 401 Unauthorized NUT -> UA11

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",

response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

* No Expires header field exists.

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
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_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]

c-p-expires:
Must exist.[RFC3261-10-40,41,51]
delta-seconds: Must not be "0". [RFC3261 10.2.2]
delta-seconds: Must contain a locally-configured default value.
[RFC3261-10-42]

[REFERENCE]

[RFC3261-10-40,41,42]
10.3 Processing REGISTER Requests

7. The registrar now processes each contact address in the Contact header field in turn. For each address, it determines the expiration interval as follows:

- If the field value has an "expires" parameter, that value MUST be taken as the requested expiration.

- If there is no such parameter, but the request has an Expires header field, that value MUST be taken as the requested expiration.

- If there is neither, a locally-configured default value MUST be taken as the requested expiration.

4.1.6 RG-1-1-6 - SIP Registrar- processing of REGISTER requests in the correct order.

[NAME]

RG-1-1-6 - SIP Registrar- The right order processing REGISTER requests

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly processes each REGISTER request in the correct order.

[REQUIREMENT]

NONE

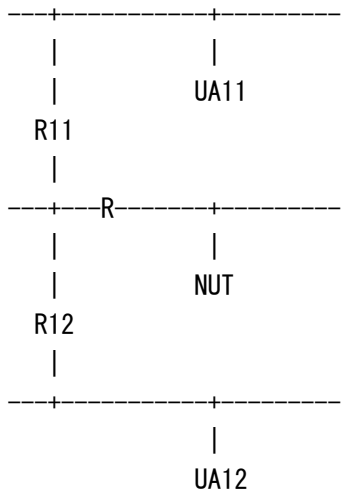
[PARAMETER]

NUT(AOR)	sip:ss.under.test.com;lr
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
UA11	3ffe:501:ffff:1::1/64
UA12	3ffe:501:ffff:2::2/64
R(IPv6)	3ffe:501:ffff:50::1/64

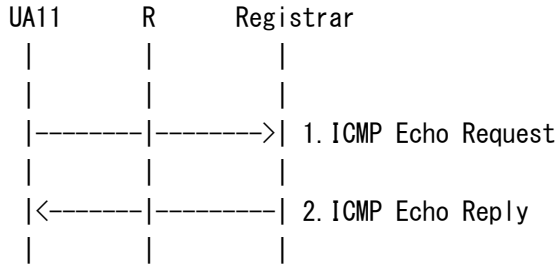
[TOPOLOGY]



[CONFIGURATION for NUT]

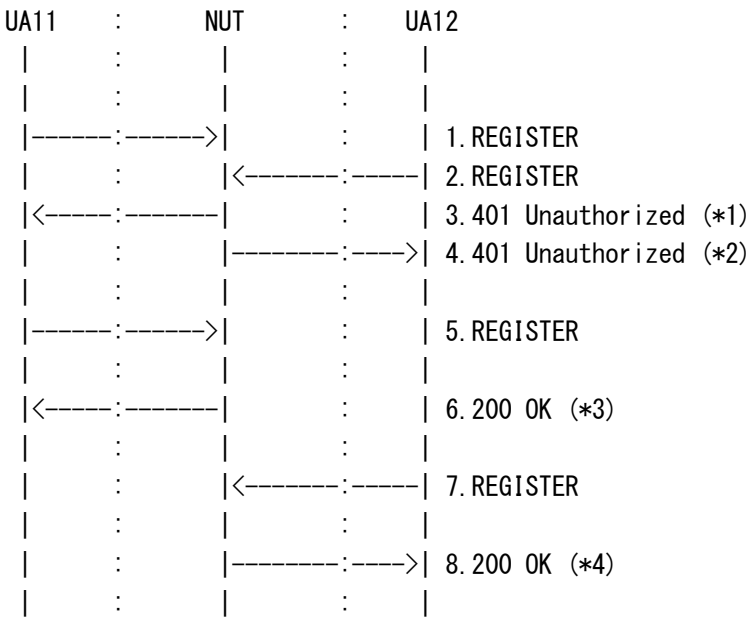
NUT	sip:ss.under.test.com;lr
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA12 Send REGISTER.
3. UA11 Receive 401 Unauthorized. (*1)
4. UA12 Receive 401 Unauthorized. (*2)
5. UA11 Send REGISTER.
6. UA11 Receive 200 OK. (*3)
7. UA12 Send REGISTER.
8. UA12 Receive 200 OK. (*4)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0

2. REGISTER UA12 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnasgh7i
Max-Forwards: 70
From: UA12 <sip:UA12@under.test.com>;tag=a73kszlfl
To: UA12 <sip:UA12@under.test.com>
Call-ID: 9k6hJKItGh4kI8d6@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA12@node11.under.test.com>
Expires: 3600
Content-Length: 0

3. 401 Unauthorized NUT -> UA11

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

4. 401 Unauthorized NUT -> UA12

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnasgh7i
 ;received=3ffe:501:ffff:5::X
From: UA12 <sip:UA12@under.test.com>;tag=a73kszlfl
To: UA12 <sip:UA12@under.test.com>;tag=1410948204
Call-ID: 9k6hJKItGh4kI8d6@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="1cec4341ae6cbe5a359ea9c8e88df84f",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

5. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

6. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

7. REGISTER UA12 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnasgg8k
Max-Forwards: 70
From: UA12 <sip:UA12@under.test.com>;tag=ja743ks76zlflH
To: UA12 <sip:UA12@under.test.com>
Call-ID: 9k6hJKItGh4kI8d6@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA12@node11.under.test.com>
Expires: 3600
Authorization: Digest username="UA12",
 realm="under.test.com",

nonce="1cec4341ae6cbe5a359ea9c8e88df84f", opaque="",
qop=auth, nc=00000002, cnonce="d4e4cec0",
uri="sip:ss.under.test.com",
response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

8. 200 OK NUT -> UA12

SIP/2.0 200 OK
Via: SIP/2.0/UDP node11.under.test.com:5060;branch=z9hG4bKnasgg8k
;received=3ffe:501:ffff:5::X
From: UA12 <sip:UA12@under.test.com>;tag=ja743ks76zlflH
To: UA12 <sip:UA12@under.test.com>;tag=37GkEhwl6
Call-ID: 9k6hJKItGh4kI8d6@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA12@node11.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:29:00 GMT
Content-Length: 0

[OBSERVABLE RESULTS]

*1:401 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]

*2:401 response from NUT to UA12.

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]

*3:200 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]

*4:200 response from NUT to UA12.
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]
[RFC3261-10-24, 25, 26]
10.3 Processing REGISTER Requests

(snip)

A registrar has to know (for example, through configuration) the set of domain(s) for which it maintains bindings. REGISTER requests MUST be processed by a registrar in the order that they are received. REGISTER requests MUST also be processed atomically, meaning that a particular REGISTER request is either processed completely or not at all. Each REGISTER message MUST be processed independently of any other registration or binding changes.

4.1.7 RG-1-1-7 - SIP Registrar- REGISTER with two Contact header fields with separate expires parameters

[NAME]

RG-1-1-7 - SIP Registrar- REGISTER request with two Contact header fields with separate expires parameters

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly processes a REGISTER request with two Contact header fields, one has an expires parameter and the other doesn't.

[REQUIREMENT]

NONE

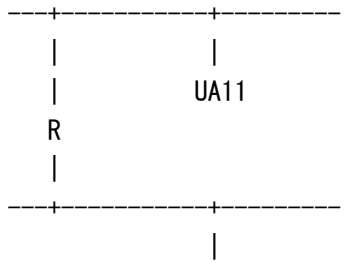
[PARAMETER]

NUT(AOR)	sip:ss.under.test.com;lr
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64

[TOPOLOGY]

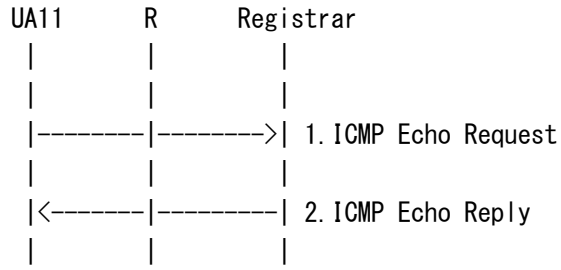


NUT

[CONFIGURATION for NUT]

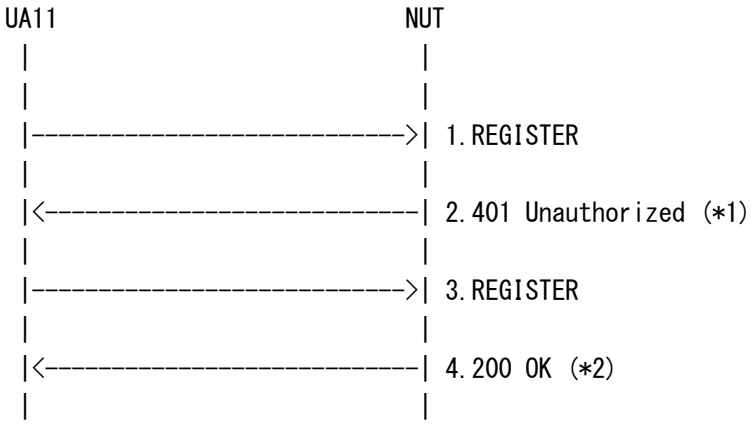
NUT	sip:ss.under.test.com;lr
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64　(IPv6)

[INITIALIZATION]



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

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized. (*1)
3. UA11 Send REGISTER.
4. UA11 Receive 200 OK. (*2)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
Record-Route: <sip:example.under.test.com;lr>

From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=1800
Contact: <sip:11UA11@node.under.test.com>
Expires: 3600
Content-Length: 0

2. 401 Unauthorized NUT -> UA11

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=1800
Contact: <sip:11UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=1800
Contact: <sip:11UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

[OBSERVABLE RESULTS]

*1:401 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]

*2:200 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]

* Record-Route
Must not contain this header field. [RFC3261-10-3, 22, 23]

* 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]
c-p-expires-param: Must be equal to correct parameter. [RFC3261 10.2.1.1]

[REFERENCE]

[RFC3261 10.2.1.1]

10.2.1.1 Setting the Expiration Interval of Contact Addresses

When a client sends a REGISTER request, it MAY suggest an expiration interval that indicates how long the client would like the registration to be valid. (As described in Section 10.3, the registrar selects the actual time interval based on its local policy.)

There are two ways in which a client can suggest an expiration interval for a binding: through an Expires header field or an "expires" Contact header parameter. The latter allows expiration intervals to be suggested on a per-binding basis when more than one

binding is given in a single REGISTER request, whereas the former suggests an expiration interval for all Contact header field values that do not contain the "expires" parameter.
If neither mechanism for expressing a suggested expiration time is present in a REGISTER, the client is indicating its desire for the server to choose.

4.1.8 RG-1-2-1 - SIP Registrar- Unsuccessful Registration (Wrong user ID / password)

[NAME]
RG-1-2-1 - SIP Registrar- Unsuccessful Registration (wrong user ID / password)

[TARGET]
SIP Registrar

[PURPOSE]
Verify that a NUT doesn't register a wrong user ID and a passward.

[REQUIREMENT]
NONE

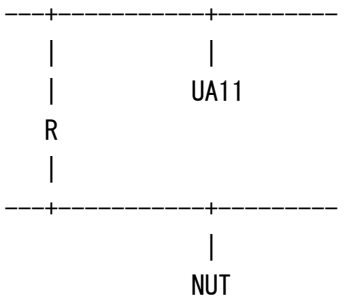
[PARAMETER]

NUT(AOR)	sip:ss.under.test.com
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64

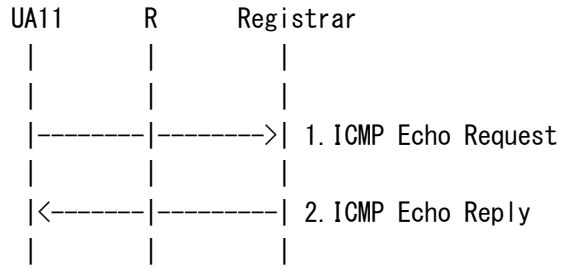
[TOPOLOGY]



[CONFIGURATION for NUT]

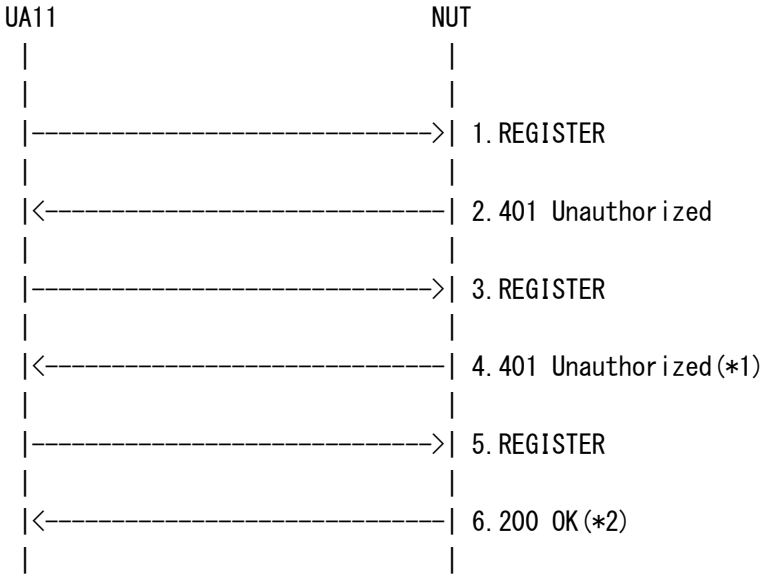
NUT	sip:ss.under.test.com
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64　(IPv6)

[INITIALIZATION]



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

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized.
3. UA11 Send REGISTER.
4. UA11 Receive 401 Unauthorized. (*1)
5. UA11 Send REGISTER.
6. UA11 Receive 200 OK. (*2)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0

Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0

2. 401 Unauthorized NUT -> UA11

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359", opaque="",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="e6774026cfc04b7fd380421ac892639d"
Content-Length: 0

- correct user ID("UA11") and wrong password("dummypassword")

4. 401 Unauthorized NUT -> UA11

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

5. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds9
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 3 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359", opaque="",
 qop=auth, nc=00000003, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

- default correct user ID("UA11") and password("nutsip")

6. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds9
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH

To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 3 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Content-Length: 0

[OBSERVABLE RESULTS]

*1:401 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]

*2:200 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. [RFF3261-10-50]
contact-param: Must be the Contact address of NUT. [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]

RFC3665 SIP Basic Call Flow Examples
See Section 2.5

4.1.9 RG-1-2-2 - SIP Registrar- Expiration of REGISTER less than a registrar-configured minimum

[NAME]

RG-1-2-2 - SIP Registrar- Expiration of REGISTER less than a registrar-configured minimum

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT sends a 423 (Interval Too Brief) response when expiration of a REGISTER request is less than a NUT-configured minimum.

[REQUIREMENT]

NONE

[PARAMETER]

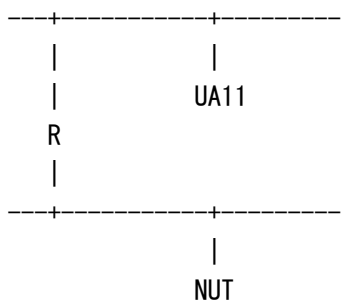
NUT(AOR)	sip:ss.under.test.com;lr
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64

UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64

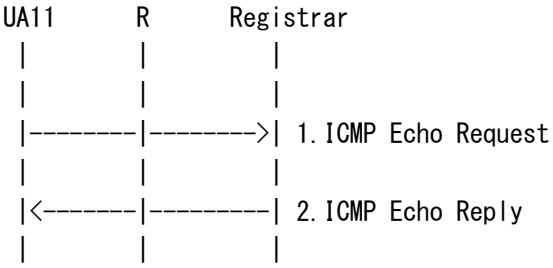
[TOPOLOGY]



[CONFIGURATION for NUT]

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

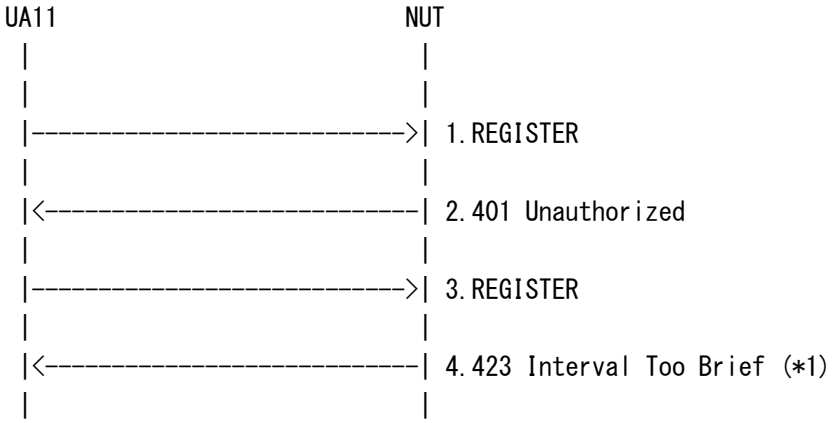
[INITIALIZATION]



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

* NUT is able to return 423 response.

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized .
3. UA11 Send REGISTER.
4. UA11 Receive 423 Interval Too Brief. (*1)

=== Message example ===

1. REGISTER UA11 -> NUT

```
REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 30
Content-Length: 0
```

* Too small Expires header field parameter

2. 401 Unauthorized NUT -> UA11

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0
```

3. REGISTER UA11 -> NUT

```
REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag= a73kszlflH
To: UA11 <sip:UA11@under.test.com>
```

Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 30
Authorization: Digest username="UA11",
realm="under.test.com",
nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
qop=auth, nc=00000002, cnonce="d4e4cec0",
uri="sip:ss.under.test.com",
response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

2. 423 Interval Too Brief NUT -> UA11

SIP/2.0 423 Interval Too Brief
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlflH
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Min-Expires: 3600
Content-Length: 0

[OBSERVABLE RESULTS]

*1:423 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 "423". [RFC3261 10.3.7]

- 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]

* Min-Expires
  Must exist. [RFC3261-10-43]

[REFERENCE]

[RFC3261-10-43]
10.3 Processing REGISTER Requests

> The registrar MAY choose an expiration less than the requested expiration interval. If and only if the requested expiration interval is greater than zero AND smaller than one hour AND less than a registrar-configured minimum, the registrar MAY reject the registration with a response of 423 (Interval Too Brief). This response MUST contain a Min-Expires header field that states the minimum expiration interval the registrar is willing to honor. It then skips the remaining steps.

4.1.10 RG-1-2-3 - SIP Registrar- REGISTER with the equal CSeq value to or lower than the last value of CSeq

[NAME]

RG-1-2-3 - SIP Registrar- REGISTER with the equal CSeq header field value to or lower than the last value of CSeq

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly rejects the REGISTER request with the equal CSeq header field value to the last value of CSeq or lower than the number..

[REQUIREMENT]

NONE

[PARAMETER]

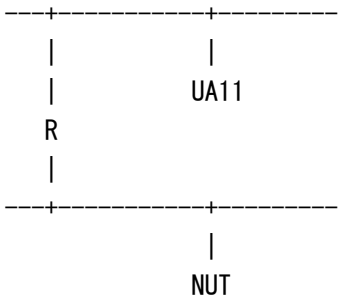
NUT(AOR)	sip:ss.under.test.com;lr
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64

R(IPv6)	3ffe:501:ffff:50::1/64

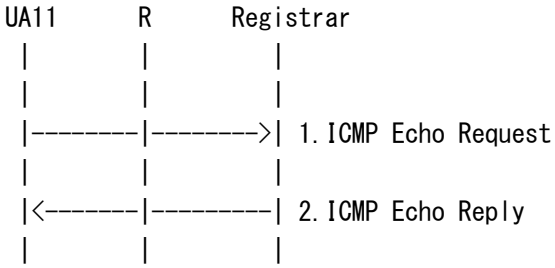
[TOPOLOGY]



[CONFIGURATION for NUT]

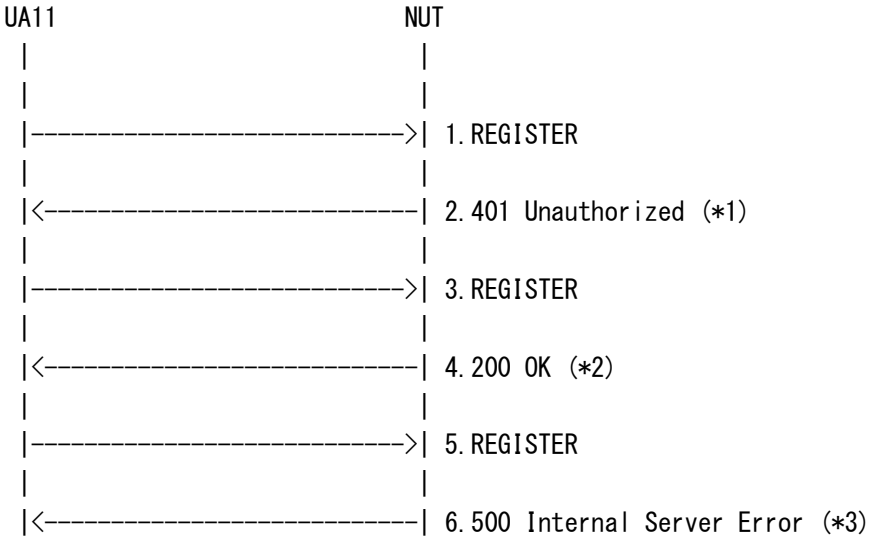
NUT	sip:ss.under.test.com;lr
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



| |

1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized. (*1)
3. UA11 Send REGISTER.
4. UA11 Receive 200 OK. (*2)
5. UA11 Send REGISTER.
6. UA11 Receive 500. (*3)

=== Message example ===

1. REGISTER UA11 -> NUT

```
REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0
```

2. 401 Unauthorized NUT -> UA11

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0
```

3. REGISTER UA11 -> NUT

```
REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
```

To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

5. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"

Content-Length: 0

6. 500 Internal Server Error NUT -> UA11

SIP/2.0 500 Internal Server Error
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

[OBSERVABLE RESULTS]

*1:401 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]

*2:200 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]

*3:500 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 "500". [RFC3261-10-46,48]

[REFERENCE]
[RFC3261-10-45, 46]
10.3 Processing REGISTER Requests

For each address, the registrar then searches the list of current bindings using the URI comparison rules. If the binding does not exist, it is tentatively added. If the binding does exist, the registrar checks the Call-ID value. If the Call-ID value in the existing binding differs from the

Call-ID value in the request, the binding MUST be removed if the expiration time is zero and updated otherwise. If they are the same, the registrar compares the CSeq value. If the value is higher than that of the existing binding, it MUST update or remove the binding as above. If not, the update MUST be aborted and the request fails.

[RFC3261-12-59, 60]
12.2.2 UAS Behavior

If the remote sequence number is empty, it MUST be set to the value of the sequence number in the CSeq header field value in the request. If the remote sequence number was not empty, but the sequence number of the request is lower than the remote sequence number, the request is out of order and MUST be rejected with a 500 (Server Internal Error) response. If the remote sequence number was not empty, and the sequence number of the request is greater than the remote sequence number, the request is in order.

4.1.11 RG-1-2-4 - SIP Registrar- The case of receiving Contact header field includeing the value "*"

[NAME]
RG-1-2-4 - SIP Registrar- The case of receiving Contact header field including the value "*"

[TARGET]
SIP Registrar

[PURPOSE]
Verify that a NUT properly sends a 400 (Bad Request) response when receiving the request that Contact value is “*” and the wrong headers (or the wrong header value).

[REQUIREMENT]
NONE

[PARAMETER]

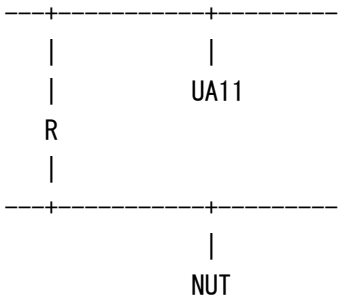
NUT(AOR)	sip:ss.under.test.com;lr
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64

R(IPv6)	3ffe:501:ffff:50::1/64

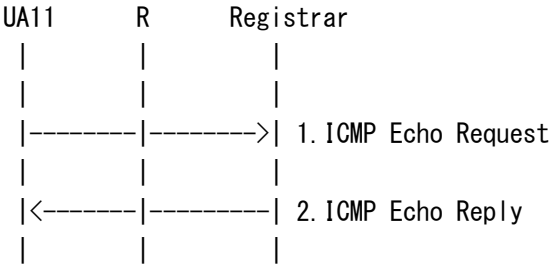
[TOPOLOGY]



[CONFIGURATION for NUT]

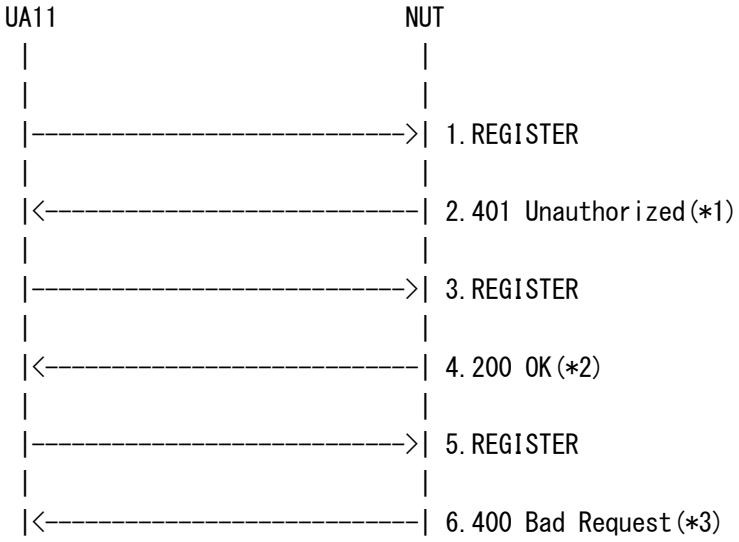
NUT	sip:ss.under.test.com;lr
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

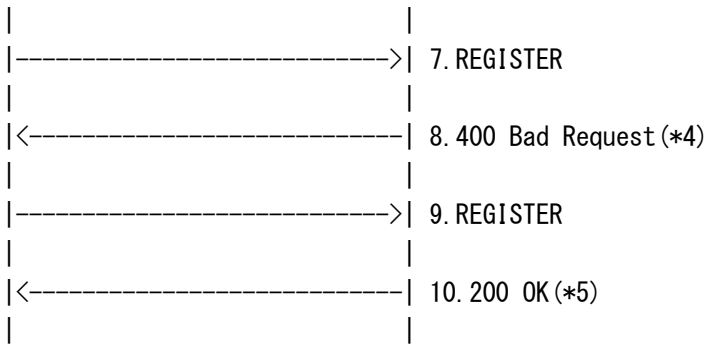
[INITIALIZATION]



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

[PROCEDURE]





1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized. (*1)
3. UA11 Send REGISTER.
4. UA11 Receive 200 OK. (*2)
5. UA11 Send REGISTER.
6. UA11 Receive 400 Bad Request. (*3)
7. UA11 Send REGISTER.
8. UA11 Receive 400 Bad Request. (*4)
9. UA11 Send REGISTER.
10. UA11 Receive 200 OK. (*5)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
Record-Route: <sip:example.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 11 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0

2. 401 Unauthorized NUT -> UA11

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl

To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 11 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
Record-Route: <sip:example.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 12 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 12 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

5. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds9
Max-Forwards: 70
Record-Route: <sip:example.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 13 REGISTER
Contact: *
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

6. 400 Bad Request NUT -> UA11

SIP/2.0 400 Bad Request
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds9
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 13 REGISTER
Content-Length: 0

7. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds9
Max-Forwards: 70
Record-Route: <sip:example.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 14 REGISTER
Contact: *
Expires: 0
Contact: <sip:UA111@node.under.tset.com>
Authorization: Digest username="UA11",

realm="under.test.com",
nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
qop=auth, nc=00000002, cnonce="d4e4cec0",
uri="sip:ss.under.test.com",
response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

8. 400 Bad Request NUT -> UA11

SIP/2.0 400 Bad Request
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds9
;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 14 REGISTER
Content-Length: 0

9. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
Record-Route: <sip:example.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 15 REGISTER
Contact: *
Expires: 0
Authorization: Digest username="UA11",
realm="under.test.com",
nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
qop=auth, nc=00000002, cnonce="d4e4cec0",
uri="sip:ss.under.test.com",
response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

10. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
;received=3ffe:501:ffff:5::X

From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 15 REGISTER
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

[OBSERVABLE RESULTS]

*1:401 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]

*2:200 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]

*3:400 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 "400". [RFC3261-10-36,38,39]

- 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]

*4:400 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 "400". [RFC3261-10-36,38,39]

- 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]

*5:200 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
contact-param: Must not include the specified parameter.
[RFC3261-10-50],[RFC3261 10.2.2]

[REFERENCE]

[RFC3261-10-36, 37, 38, 39]
10.3 Processing REGISTER Requests

6. The registrar checks whether the request contains the Contact header field. If not, it skips to the last step. If the Contact header field is present, the registrar checks if there is one Contact field value that contains the special value "*" and an Expires field. If the request has additional Contact fields or an expiration time other than zero, the request is invalid, and the server MUST return a 400 (Invalid Request) and skip the remaining steps. If not, the registrar checks whether the Call-ID agrees with the value stored for each binding. If not, it MUST remove the binding. If it does agree, it MUST remove the binding only if the CSeq in the request is higher than the value stored for that binding. Otherwise, the update MUST be aborted and the request fails.

4.1.12 RG-2-1-1 - SIP Registrar- Record-Route in REGISTER request

[NAME]

RG-2-1-1 - SIP Registrar- Record-Route in REGISTER request

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly ignores a Record-Route header field in a REGISTER request.

[REQUIREMENT]

NONE

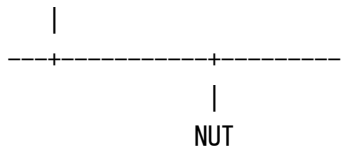
[PARAMETER]

NUT(AOR)	sip:ss.under.test.com;lr
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64

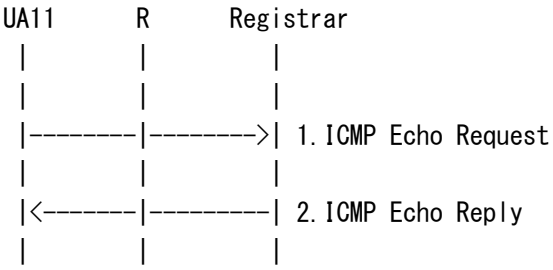
[TOPOLOGY]



[CONFIGURATION for NUT]

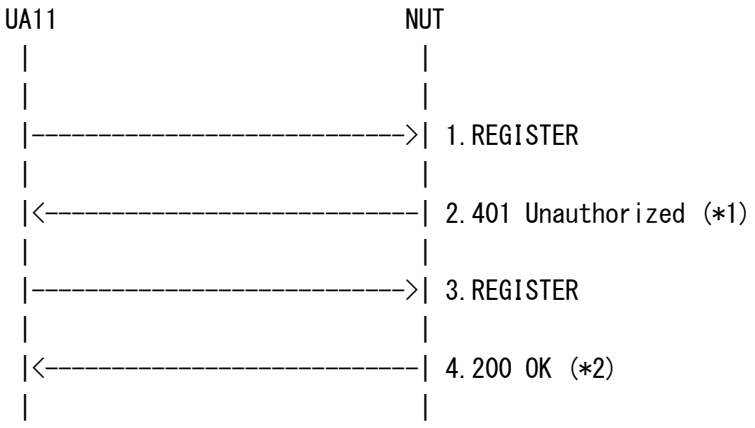
NUT	sip:ss.under.test.com;lr
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized. (*1)
3. UA11 Send REGISTER.
4. UA11 Receive 200 OK. (*2)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0

Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
Record-Route: <sip:example.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0

2. 401 Unauthorized NUT -> UA11

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
Record-Route: <sip:example.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"

Content-Length: 0

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

[OBSERVABLE RESULTS]

*1:401 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]

* Record-Route
Must not contains this header field. [RFC3261-10-3, 22, 23]

*2:200 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]

* Record-Route
Must not contains this header field. [RFC3261-10-3, 22, 23]

* 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]

[RFC3261-10-3]
10.2 Constructing the REGISTER Request

The Record-Route header field has no meaning in
REGISTER requests or responses, and MUST be ignored
if present.

[RFC3261-10-22, 23]
10.3 Processing REGISTER Requests

Registrars MUST ignore the Record-Route header field if it is
included in a REGISTER request. Registrars MUST NOT include a

Record-Route header field in any response to a REGISTER request.

4.1.13 RG-2-1-2 - SIP Registrar- REGISTER with the different Call-ID value

[NAME]

RG-2-1-2 - SIP Registrar- REGISTER with the different Call-ID value

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly removes the binding if the expiration time is zero and updated when the Call-ID value in a REGISTER request is different from the Call-ID value in the existing binding.

[REQUIREMENT]

NONE

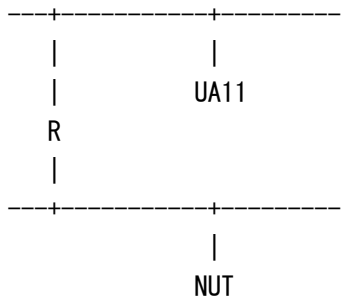
[PARAMETER]

NUT(AOR)	sip:ss.under.test.com;lr
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64

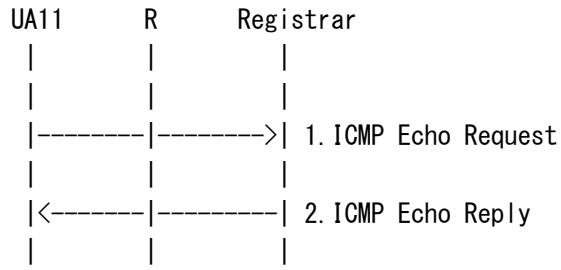
[TOPOLOGY]



[CONFIGURATION for NUT]

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

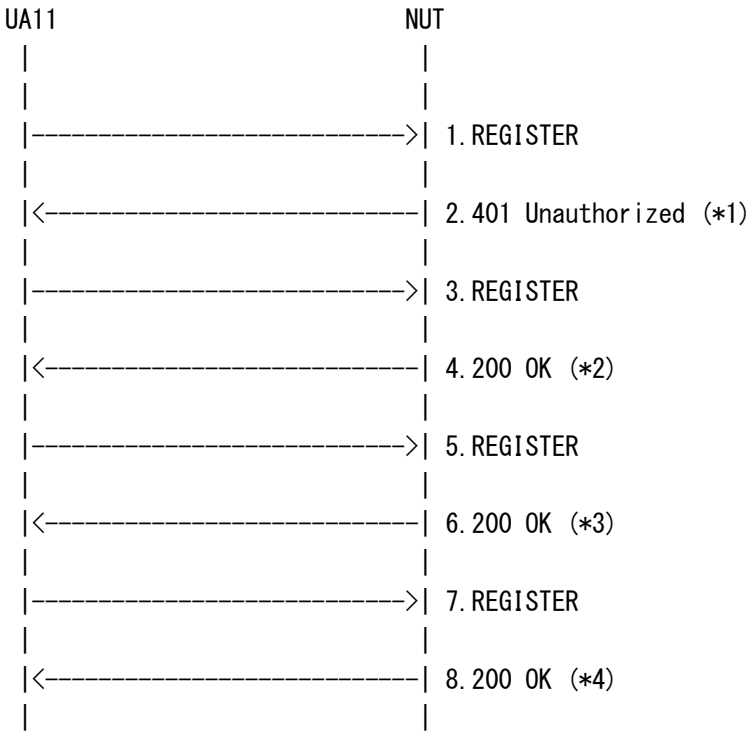
[INITIALIZATION]



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

* NUT is able to return 423 response.

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized. (*1)
3. UA11 Send REGISTER.
4. UA11 Receive 200 OK. (*2)
5. UA11 Send REGISTER.
6. UA11 Receive 200 OK. (*3)
7. UA11 Send REGISTER.

8. UA11 Receive 200 OK. (*4)

=== Message example ===

1. REGISTER UA11 -> NUT

```
REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0
```

2. 401 Unauthorized NUT -> UA11

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0
```

3. REGISTER UA11 -> NUT

```
REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
```

nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
qop=auth, nc=00000002, cnonce="d4e4cec0",
uri="sip:ss.under.test.com",
response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

5. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: j8m4luaNjkh2nGdy@under.test.com
CSeq: 3 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
realm="under.test.com",
nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
qop=auth, nc=00000002, cnonce="d4e4cec0",
uri="sip:ss.under.test.com",
response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

6. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
;received=3ffe:501:ffff:5::X

From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: j8m4luaNjkh2nGdy@under.test.com
CSeq: 3 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

* Call-ID parameter of Message 5 and 6 differ from Message 1-4.

7. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: pe7hS8nkf3kLen7v@under.test.com
CSeq: 4 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 0
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

8. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: pe7hS8nkf3kLen7v@under.test.com
CSeq: 4 REGISTER
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

* Call-ID parameter of Message 7 and 8 differ from Message 1-4 and 5-6.

[OBSERVABLE RESULTS]

*1:401 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]

*2:200 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 NUT. [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". [RFC32261 10.2.2]

*3:200 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]

*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]

[REFERENCE]

[RFC3261-10-44]
10.3 Processing REGISTER Requests

If the Call-ID value in the existing binding differs from the Call-ID value in the request, the binding MUST be removed if the expiration time is zero and updated otherwise.

4.1.14 RG-2-1-3 - SIP Registrar- Sequence Number is not Incremented.

[NAME]

RG-2-1-3 - SIP Registrar-Sequence Number is not Incremented.

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly sends a 500 (Server Internal Error) response when receiving a REGISTER with the equal CSeq header field value to the last value of CSeq or lower than the number and “*” in a Contact header field.

[REQUIREMENT]

NONE

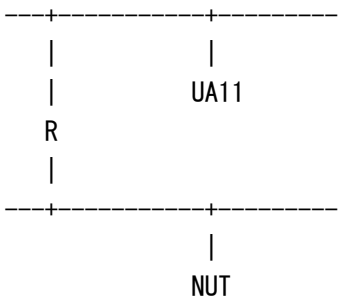
[PARAMETER]

NUT(AOR)	sip:ss.under.test.com;lr
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64

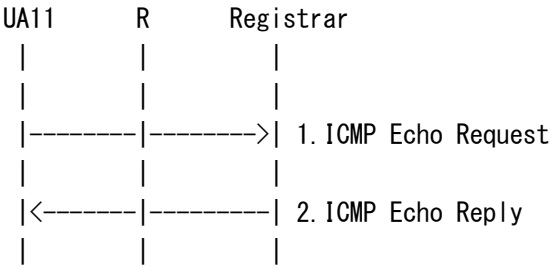
[TOPOLOGY]



[CONFIGURATION for NUT]

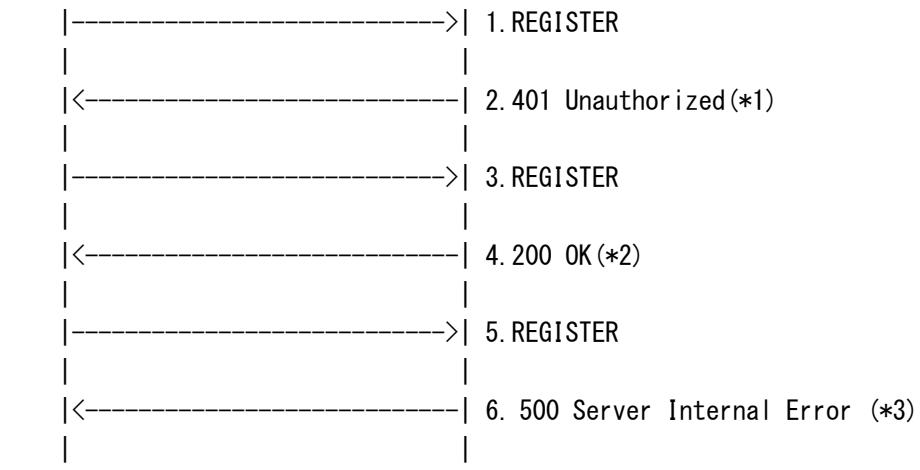
NUT	sip:ss.under.test.com;lr
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized. (*1)
3. UA11 Send REGISTER.
4. UA11 Receive 200 OK. (*2)
5. UA11 Send REGISTER.
6. UA11 Receive 500 Server Internal Error. (*3)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
Record-Route: <sip:example.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0

2. 401 Unauthorized NUT -> UA11

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com

CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
Record-Route: <sip:example.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

5. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds9

Max-Forwards: 70
Record-Route: <sip:example.under.test.com;lr>
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: *
Expires: 0
Authorization: Digest username="UA11",
realm="under.test.com",
nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
qop=auth, nc=00000002, cnonce="d4e4cec0",
uri="sip:ss.under.test.com",
response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

* CSeq number is not incremented.

6. 500 Server Internal Error NUT -> UA11

SIP/2.0 500 Server Internal Error
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

[OBSERVABLE RESULTS]

*1:401 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]

*2:200 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]

*3:500 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 "500". [RFC3261-10-36,38,39]

- 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]

[REFERENCE]

[RFC3261-10-36, 37, 38, 39]

10.3 Processing REGISTER Requests

6. The registrar checks whether the request contains the Contact header field. If not, it skips to the last step. If the Contact header field is present, the registrar checks if there is one Contact field value that contains the special value "*" and an Expires field. If the request has additional Contact fields or an expiration time other than zero, the request is invalid, and the server MUST return a 400 (Invalid Request) and skip the remaining steps. If not, the registrar checks whether the Call-ID agrees with the value stored for each binding. If not, it MUST remove the binding. If it does agree, it MUST remove the binding only if the CSeq in the request is higher than the value stored for that binding. Otherwise, the update MUST be aborted and the request fails.

4.1.15 RG-2-1-4 - SIP Registrar- URI with user-param in To header field in REGISTER

[NAME]

RG-2-1-4 - SIP Registrar- URI with user-param in To header field in REGISTER request

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly processes a URI with a user parameter in a To header field in a REGISTER request.

[REQUIREMENT]

NONE

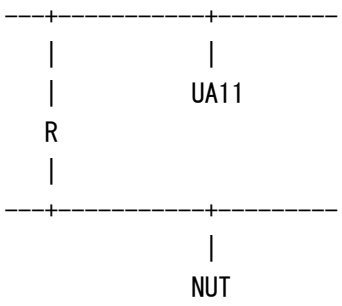
[PARAMETER]

NUT(AOR)	sip:ss.under.test.com;lr
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64

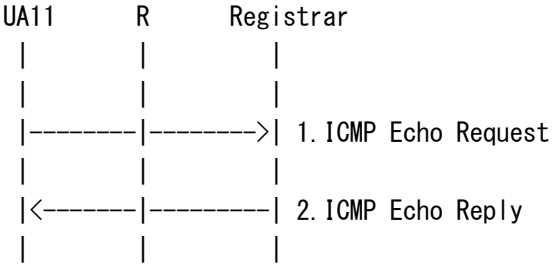
[TOPOLOGY]



[CONFIGURATION for NUT]

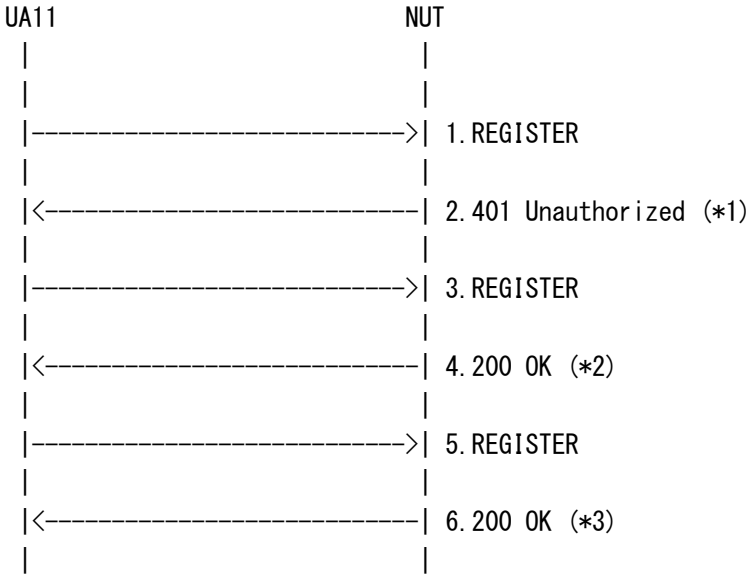
NUT	sip:ss.under.test.com;lr
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized. (*1)
3. UA11 Send REGISTER.
4. UA11 Receive 200 OK. (*2)
5. UA11 Send REGISTER.
6. UA11 Receive 200 OK. (*3)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com;user=phone>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0

* URI in To header field has uri-param "user=phone".

2. 401 Unauthorized NUT -> UA11

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com;user=phone>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

* URI in To header field does not change.

3. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com;user=phone>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

* URI in To header field has uri-param "user=phone".

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com;user=phone>;tag=37GkEhwl6

Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

* URI in To header field does not change.
* As an internal processing, registrar must convert URI in To header field to a canonical form to compare with the binding list that registrar has.

5. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

* Not contain user-param in To header field value.

6. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3599
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

[OBSERVABLE RESULTS]

*1:401 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]

* To

user-param: Must not remove user-param. [RFC3261-10-33, 34]

*2:200 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". [RFC3261-10-33, RFC3261-10-34]

- 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]

* To

user-param: Must not remove user-param. [RFC3261-10-33, 34]

* 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]

*3:200 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". [RFC3261-10-33,34]

- 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]

[RFC3261-10-34]

10.3 Processing REGISTER Requests

5. The registrar extracts the address-of-record from the To header field of the request. If the address-of-record is not valid for the domain in the Request-URI, the registrar MUST send a 404 (Not Found) response and skip the remaining steps. The URI MUST then be converted to a canonical form. To do that, all URI parameters MUST be removed (including the user-param), and any escaped characters MUST be converted to their unescaped form. The result serves as an index into the list of bindings.

4.1.16 RG-2-1-5 - SIP Registrar- URI with escaped characters in To header field in REGISTER

[NAME]

RG-2-1-5 - SIP Registrar- URI with escaped characters in To header field in REGISTER request

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly treats the escaped characters as unescaped characters when URI in a To header field in a REGISTER request contains escaped characters.

[REQUIREMENT]

NONE

[PARAMETER]

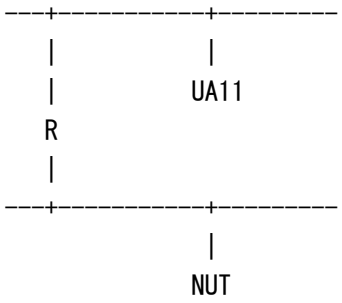
NUT(AOR)	sip:ss.under.test.com;lr
UA11(AOR)	sip:U%4111@under.test.com
UA11(Contact)	sip:U%4111@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64

R(IPv6)	3ffe:501:ffff:50::1/64

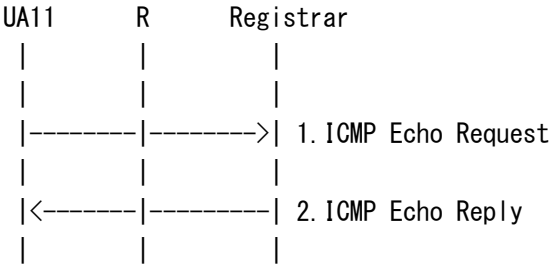
[TOPOLOGY]



[CONFIGURATION for NUT]

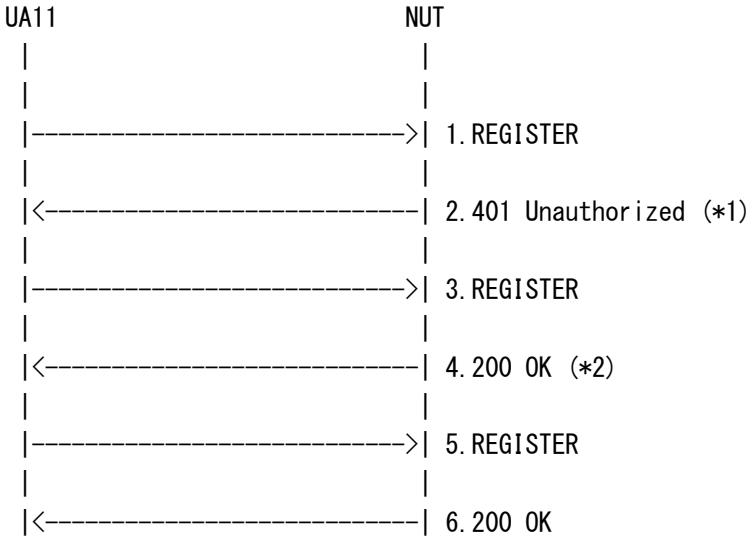
NUT	sip:ss.under.test.com;lr
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



| |

1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized. (*1)
3. UA11 Send REGISTER.
4. UA11 Receive 200 OK. (*2)
5. UA11 Send REGISTER.
6. UA11 Receive 200 OK.

=== Message example ===

1. REGISTER UA11 -> NUT

```
REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:U%4111@under.test.com>;tag=a73kszlfl
To: UA11 <sip:U%4111@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:U%4111@node.under.test.com>
Expires: 3600
Content-Length: 0
```

* userinfo of URI in To header field has escaped character (SP(0x20)).

2. 401 Unauthorized NUT -> UA11

```
SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:U%4111@under.test.com>;tag=a73kszlfl
To: UA11 <sip:U%4111@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0
```

* URI in To header field does not change.

3. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:U%4111@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:U%4111@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:U%4111@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

* userinfo of URI in To header field has escaped character (SP(0x20)).

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:U%4111@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:U%4111@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:U%4111@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

* URI in To header field does not change.
* As an internal processing, registrar must convert URI in To header field to a canonical form to compare with the binding list that registrar has.

5. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70

From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Expires: 3600
Authorization: Digest username="UA11",
realm="under.test.com",
nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
qop=auth, nc=00000002, cnonce="d4e4cec0",
uri="sip:ss.under.test.com",
response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

* Not contain escaped-character in To header field value.

6. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

[OBSERVABLE RESULTS]

*1:401 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]

* To
Must not convert from escaped characters to unescaped characters. [RFC3261-10-33, RFC3261-10-35]

*2:200 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". [RFC3261-10-33, RFC3261-10-35]

- 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]

* To
Must not convert from escaped characters to unescaped characters. [RFC3261-10-33, RFC3261-10-35]

* 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]

[RFC3261-10-33,35]
10.3 Processing REGISTER Requests

5. The registrar extracts the address-of-record from the To header field field of the request. If the address-of-record is not valid for the domain in the Request-URI, the registrar MUST send a 404 (Not Found) response and skip the remaining steps. The URI MUST then be converted to a canonical form. To do that, all URI parameters MUST be removed (including the user-param), and any escaped characters MUST be converted to their unescaped form. The result serves as an index into the list of bindings.

4.1.17 RG-2-2-1 - SIP Registrar- REGISTER with an invalid AoR

[NAME]

RG-2-2-1 - SIP Registrar- REGISTER upon an invalid AoR extracted

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly sends 404 (Not Found) response and skip the remaining steps when the AoR is not valid for the domain in the Request-URI.

[REQUIREMENT]

NONE

[PARAMETER]

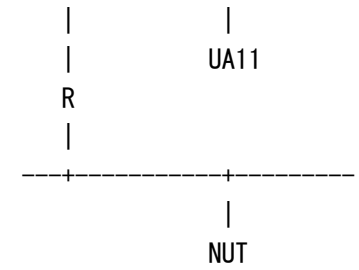
NUT(AOR)	sip:ss.under.test.com;lr
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64

[TOPOLOGY]

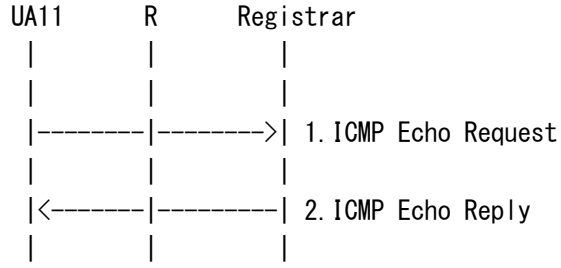
---+----------+---------



[CONFIGURATION for NUT]

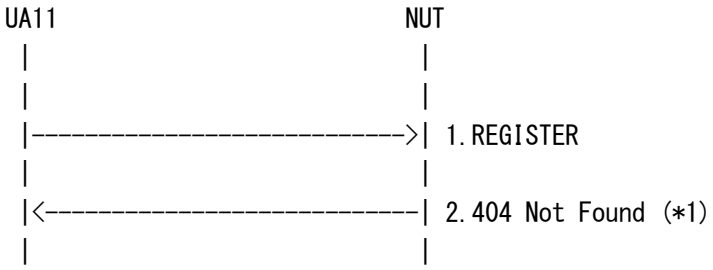
NUT	sip:ss.under.test.com;lr
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA11 Receive 404 Not Found. (*1)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@biloxi.example.com>;tag=a73kszlfl

To: UA11 <sip:UA11@biloxi.example.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0

2. 404 Not Found NUT -> UA11

SIP/2.0 404 Not Found
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@biloxi.example.com>;tag=a73kszlfl
To: UA11 <sip:UA11@biloxi.example.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
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_status
Status-Code: Must be "404". [RRC3261-10-32]

- 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]

[REFERENCE]

[RFC3261-10-32]
10.3 Processing REGISTER Requests

5. The registrar extracts the address-of-record from the To header field

field of the request. If the address-of-record is not valid for the domain in the Request-URI, the registrar MUST send a 404 (Not Found) response and skip the remaining steps.

4.1.18 RG-2-2-2 - SIP Registrar- Illegal parameter value in Contact header field

[NAME]

RG-2-2-2 - SIP Registrar- Illegal parameter in Contact field

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly sends a 400 (Bad Request) response when receiving the request whose Contact header field value is "*" and Expires header field value is other than zero.

[REQUIREMENT]

NONE

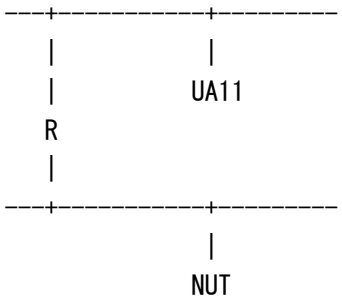
[PARAMETER]

NUT(AOR)	sip:ss.under.test.com;lr
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64

[TOPOLOGY]

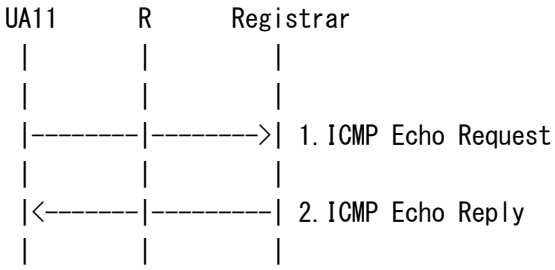


[CONFIGURATION for NUT]

NUT	sip:ss.under.test.com;lr

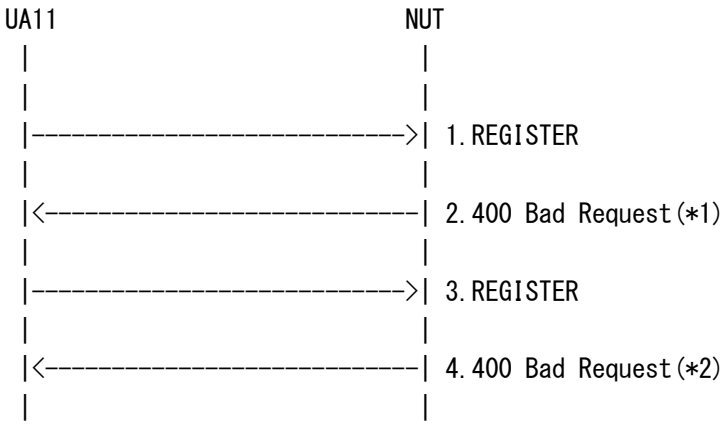
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA11 Receive 400 Bad Request. (*1)
3. UA11 Send REGISTER.
4. UA11 Receive 400 Bad Request. (*2)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: *

Expires: 3600
Content-Length: 0

* Contact header field value is "*" but Expires parameter is not "0".

2. 400 Bad Request NUT -> UA11

SIP/2.0 400 Bad Request
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Content-Length: 0

3. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: hf7Klijndt3Skn3y@under.test.com
CSeq: 2 REGISTER
Contact: *
Contact: <sip:UA11@node.under.test.com>
Expires: 0
Content-Length: 0

* Contact header field values are "*" and <sip:UA11@node.under.test.com>.

4. 400 Bad Request NUT -> UA11

SIP/2.0 400 Bad Request
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=1643728903
Call-ID: hf7Klijndt3Skn3y@under.test.com
CSeq: 2 REGISTER
Content-Length: 0

[OBSERVABLE RESULTS]

*1:400 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 "400". [RRC3261-10-36]

- 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]

*2:400 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 "400". [RRC3261-10-36]

- 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]

[REFERENCE]

[RFC3261-10-36]

10.3 Processing REGISTER Requests

6. The registrar checks whether the request contains the Contact header field. If not, it skips to the last step. If the Contact header field is present, the registrar checks if there is one Contact field value that contains the special value "*" and an Expires field. If the request has additional Contact fields or an expiration time other than zero, the request is invalid, and the server MUST return a 400 (Invalid Request) and skip the remaining steps.

4.1.19 RG-2-2-3 - SIP Registrar- REGISTER request with two Contact header field values and update failure

[NAME]

RG-2-2-3 - SIP Registrar- REGISTER request with two Contact header field value and update failure

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT sends a 500 (Server Error) response when receiving a REGISTER request with two header field values

[REQUIREMENT]

NONE

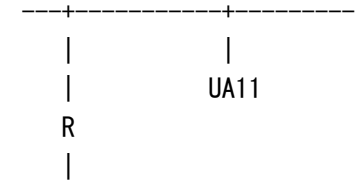
[PARAMETER]

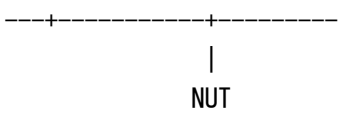
NUT(AOR)	sip:ss.under.test.com;lr
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64

[TOPOLOGY]

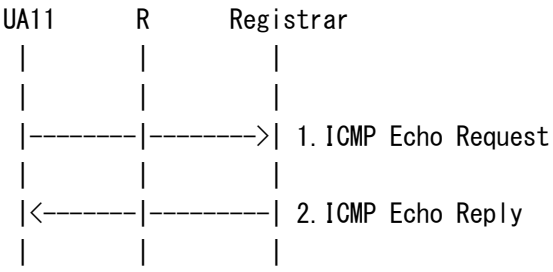




[CONFIGURATION for NUT]

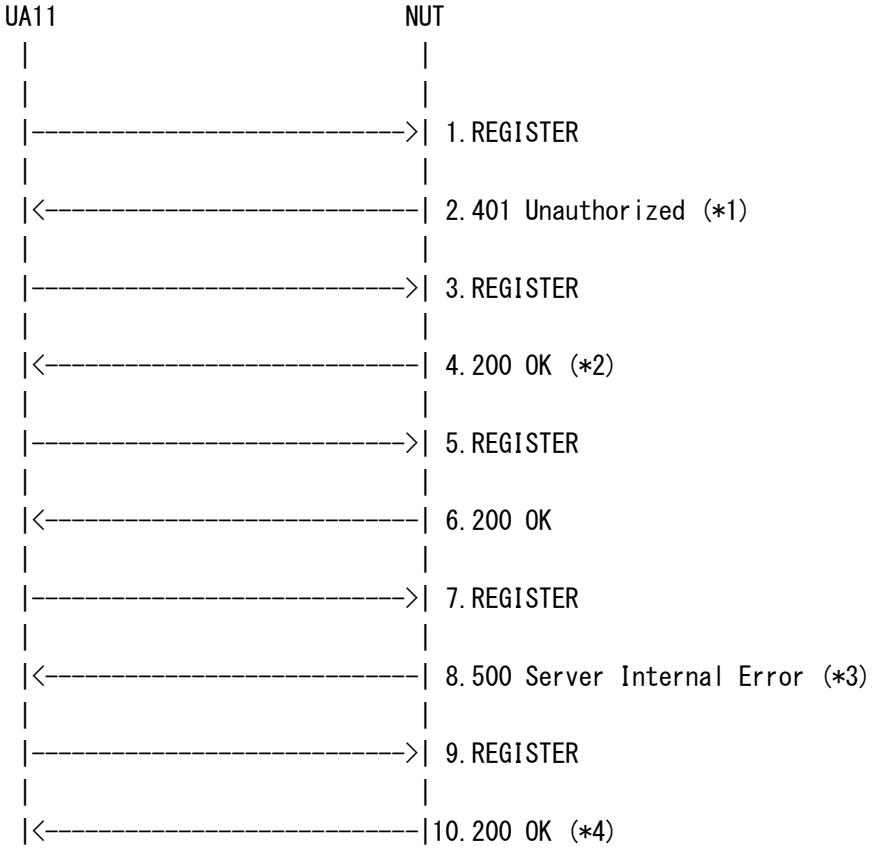
NUT	sip:ss.under.test.com;lr
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



| | |

1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized. (*1)
3. UA11 Send REGISTER.
4. UA11 Receive 200 OK. (*2)
5. UA11 Send REGISTER.
6. UA11 Receive 200 OK.
7. UA11 Send REGISTER.
8. UA11 Receive 500 Server Internal Error. (*3)
9. UA11 Send REGISTER.
10. UA11 Receive 200 OK. (*4)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0

2. 401 Unauthorized NUT -> UA11

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

5. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 3 REGISTER
Contact: <sip:11UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",

realm="under.test.com",
nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
qop=auth, nc=00000002, cnonce="d4e4cec0",
uri="sip:ss.under.test.com",
response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

6. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 3 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Contact: <sip:11UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

7. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 3 REGISTER
Contact: <sip:UA11@node.under.test.com>
Contact: <sip:11UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
realm="under.test.com",
nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
qop=auth, nc=00000002, cnonce="d4e4cec0",
uri="sip:ss.under.test.com",
response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

/* CSeq parameter is the same as 5.REGISTER */

8. 500 Internal Server Error NUT -> UA11

SIP/2.0 500 Internal Server Error
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 3 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Contact: <sip:11UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

9. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 4 REGISTER
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

10. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 4 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600

Contact: <sip:11UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

[OBSERVABLE RESULTS]

*1:401 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]

*2:200 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]

*3:500 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 "500". [RFC3261-10-46,48]

*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]
[RFC3261-10-47, 48, 49]

10.3 Processing REGISTER Requests

7. The binding updates MUST be commited (that is, made visible to the proxy or redirect server) if and only if all binding updates and additions succeed. If any one of them fails (for example, because the back-end database commit failed), the request MUST fail with a 500 (Server Error) response and all tentative binding updates MUST be removed.

4.1.20 RG-3-1-1 - SIP Registrar- Forwarded REGISTER request

[NAME]
RG-3-1-1 - SIP Registrar- Forwarded REGISTER request

[TARGET]
SIP Registrar

[PURPOSE]
Verify that a NUT properly processes a forwarded REGISTER request.

[REQUIREMENT]
Only when the equipment can process forwarded requests.

[PARAMETER]

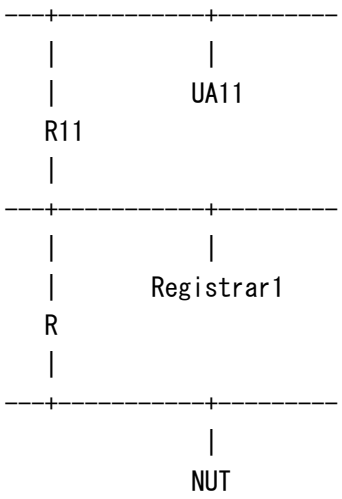
NUT(AOR)	sip:ss.under.test.com;lr

UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com
Registrar1	sip:ss1.biloxi.example.com

[ADDRESS]

NUT (IPv6)	3ffe:501:ffff:50::50/64
UA11(IPv6)	3ffe:501:ffff:1::1/64
R(IPv6)	3ffe:501:ffff:50::1/64
Registrar1(IPv6)	3ffe:501:ffff:20::20/64

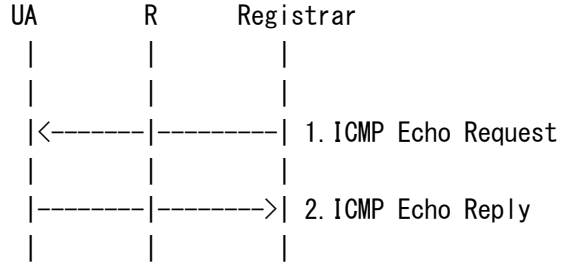
[TOPOLOGY]



[CONFIGURATION for NUT]

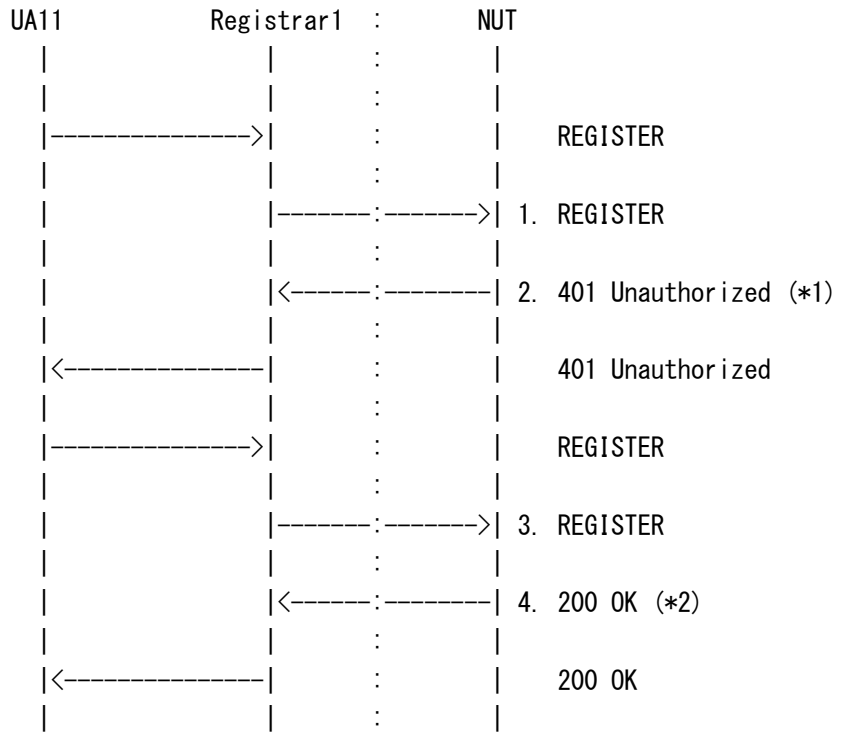
NUT	sip:ss.under.test.com;lr
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



1. Registrar1 Send REGISTER.
2. Registrar1 Receive 401 Unauthorized. (*1)
3. Registrar1 Send REGISTER.
4. Registrar1 Receive 200 OK. (*2)

=== Message example ===

1. REGISTER Registrar1 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss1.biloxi.example.com:5060;branch=z9hG4bKjy8hGfv
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:1::1
Max-Forwards: 69
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0

2. 401 Unauthorized NUT -> Registrar1

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP ss1.biloxi.example.com:5060;branch=z9hG4bKjy8hGfv
 ;received=3ffe:501:ffff:20::20
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
 ;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. REGISTER Registrar1 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP ss1.biloxi.example.com:5060;branch=z9hG4bKjy8hGfz
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:1::1
Max-Forwards: 69
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.biloxi.example.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

4. 200 OK NUT -> Registrar1

SIP/2.0 200 OK
Via: SIP/2.0/UDP ss1.biloxi.example.com:5060;branch=z9hG4bKjy8hGfz
 ;received=3ffe:501:ffff:50::50
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:1::1
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl

To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

[OBSERVABLE RESULTS]

*1:401 response from NUT to Registrar1.

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]

*2:200 response from NUT to Registrar1.

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]

Variation of sequence from RFC3665 Section 3.2.

[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.

4.1.22 RG-4-1-1 - SIP Registrar- Future extension about new header field

[NAME]

RG-4-1-1 - SIP Registrar- Future extension about new header field

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly processes a new header field as normal processing in order to extend that for future.

[REQUIREMENT]

NONE

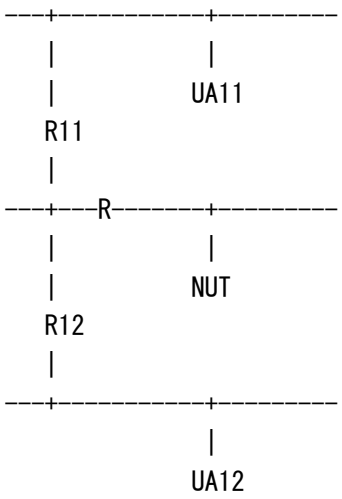
[PARAMETER]

NUT(AOR)	sip:ss.under.test.com;lr
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
UA11(IPv6)	3ffe:501:ffff:1::1/64
UA12(IPv6)	3ffe:501:ffff:2::2/64
R(IPv6)	3ffe:501:ffff:50::1/64

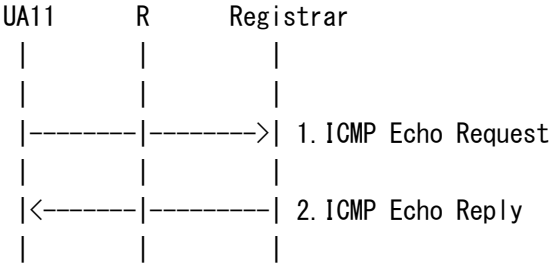
[TOPOLOGY]



[CONFIGURATION for NUT]

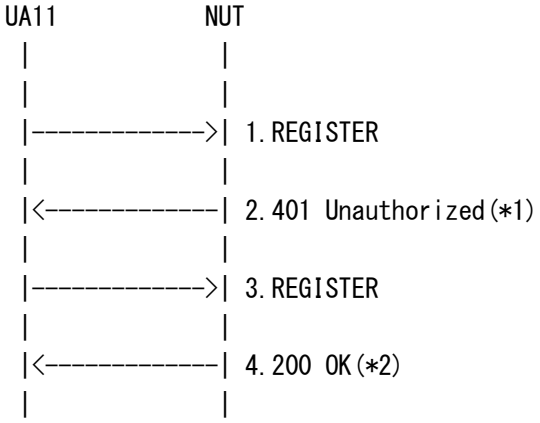
NUT	sip:ss.under.test.com;lr
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA11 Receive 401 Unauthorized. (*1)
3. UA11 Send REGISTER.
4. UA11 Receive 200 OK. (*2)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
NewHeader field: new
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0

2. 401 Unauthorized NUT -> UA11

SIP/2.0 401 Unauthorized
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl

To: UA11 <sip:UA11@under.test.com>;tag=1410948204
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
WWW-Authenticate: Digest realm="under.test.com", qop="auth",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
Max-Forwards: 70
NewHeader field: new
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Authorization: Digest username="UA11",
 realm="under.test.com",
 nonce="ea9c8e88df84f1cec4341ae6cbe5a359",
 qop=auth, nc=00000002, cnonce="d4e4cec0",
 uri="sip:ss.under.test.com",
 response="b7fd380421adc89263e6774026cfc049"
Content-Length: 0

4. 200 OK NUT -> UA11

SIP/2.0 200 OK
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds8
 ;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag=ja743ks76zlflH
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 2 REGISTER
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

[OBSERVABLE RESULTS]

*1:401 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". [RFC3261-8-70,71]

- 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]

*2:200 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". [RFC3261-8-70,71]

- 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 NUT. [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]

[RFC3261-8-70, 71]

8 General User Agent Behavior

8.2 UAS Behavior

8.2.2 Header field Inspection

If a UAS does not understand a header field in a request (that is, the header field is not defined in this specification or in any supported extension), the server MUST ignore that header field and continue processing the message. A UAS SHOULD ignore any malformed header fields that are not necessary for processing requests.

4.1.23 RG-4-1-2 - SIP Registrar- Non-supported option-tag in Require header field

[NAME]

RG-4-1-2 - SIP Registrar- Non-supported option-tag in Require filed

[TARGET]

SIP Registrar

[PURPOSE]

Verify that a NUT properly sends a 420 (Bad Extension) response when receiving an option-tag parameter in a Require header field that isn't supported.

[REQUIREMENT]

NONE

[PARAMETER]

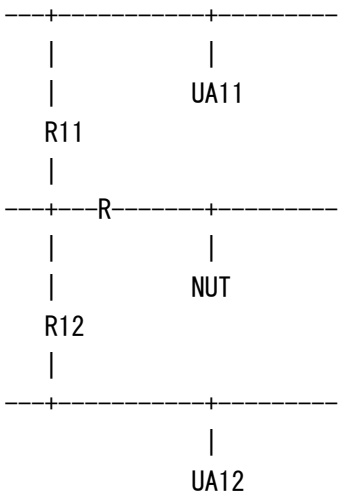
NUT(AOR)	sip:ss.under.test.com;lr
UA11(AOR)	sip:UA11@under.test.com
UA11(Contact)	sip:UA11@node.under.test.com
UA12(AOR)	sip:UA12@under.test.com

UA12(Contact)	sip:UA12@node11.under.test.com

[ADDRESS]

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

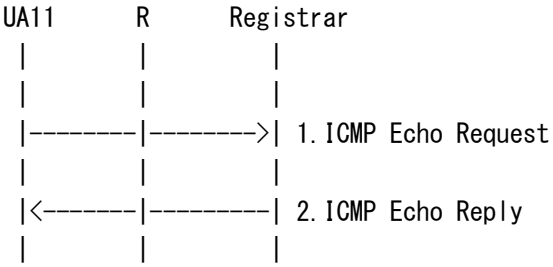
[TOPOLOGY]



[CONFIGURATION for NUT]

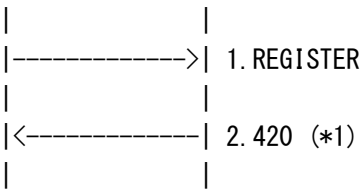
NUT	sip:ss.under.test.com;lr
NUT(IPADDRESS)	3ffe:501:ffff:50::50/64 (IPv6)

[INITIALIZATION]



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

[PROCEDURE]



1. UA11 Send REGISTER.
2. UA11 Receive 420 Bad Extension. (*1)

=== Message example ===

1. REGISTER UA11 -> NUT

REGISTER sip:ss.under.test.com SIP/2.0
Via: SIP/2.0/UDP node.under.test.com:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
From: UA11 <sip:UA11@under.test.com>;tag=a73kszlfl
To: UA11 <sip:UA11@under.test.com>
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Require: 999rel
Contact: <sip:UA11@node.under.test.com>
Expires: 3600
Content-Length: 0

8. 420 Bad Extension NUT -> UA11

SIP/2.0 420 Bad Extension
Via: SIP/2.0/UDP node.under.test.com:5060;branch= z9hG4bKnashds7
;received=3ffe:501:ffff:5::X
From: UA11 <sip:UA11@under.test.com>;tag= a73kszlfl
To: UA11 <sip:UA11@under.test.com>;tag=37GkEhwl6
Call-ID: 1j9FpLxk3uxtm8tn@under.test.com
CSeq: 1 REGISTER
Unsupported:999rel
Contact: <sip:UA11@node.under.test.com>;expires=3600
Date: Sat,13 Nov 2004 23:28:00 GMT
Content-Length: 0

[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". [RFC3261-8-78,79]

- 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. [RFC3261-8-79]
  option-tag: Must include "999rel". [RFC3261-8-78,79]

[REFERENCE]
[RFC3261-8-78, 79]
8.2.2.3 Require

Assuming the UAS decides that it is the proper element to process the request, it examines the Require header field, if present.

The Require header field is used by a UAC to tell a UAS about SIP extensions that the UAC expects the UAS to support in order to process the request properly. Its format is described in Section 20.32. If a UAS does not understand an option-tag listed in a Require header field, it MUST respond by generating a response with status code 420 (Bad Extension). The UAS MUST add an Unsupported header field, and list in it those options it does not understand amongst those in the Require header field of the request.

[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.

****

Copyright (C) 2005-2009 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)