

nathelper Module

Maxim Sobolev
PortaOne

Edited by
Maxim Sobolev

nathelper Module

Edited by and Maxim Sobolevand Maxim Sobolev

Copyright © 2003 Porta Software Ltd.

Revision History

Revision \$Revision: 1.1 \$ \$Date: 2003/07/16 15:59:44 \$

Table of Contents

1. User's Guide	1
1.1. Overview	1
1.2. Dependencies	1
1.2.1. SER Modules	1
1.2.2. External Libraries or Applications.....	1
1.3. Exported Parameters.....	1
1.3.1. natping_interval (integer)	1
1.3.2. ping_nated_only (integer)	2
1.4. Exported Functions	2
1.4.1. fix_nated_contact()	2
1.5. Exported Functions	2
1.5.1. fix_nated_sdp(mode)	2
1.5.2. force_rtp_proxy()	3
1.5.3. force_rtp_proxy(ip_address)	3
1.5.4. nat_uac_test(mode)	3
2. Developer's Guide	4
3. Frequently Asked Questions	5

List of Examples

1-1. Set natping_interval parameter.....	1
1-2. Set ping_nated_only parameter.....	2
1-3. fix_nated_contact usage	2
1-4. fix_nated_sdp usage.....	2
1-5. force_rtp_proxy usage.....	3
1-6. force_rtp_proxy usage.....	3

Chapter 1. User's Guide

1.1. Overview

This is a module to help with NAT traversal. In particular, it helps symmetric UAs that don't advertise they are symmetric and are not able to determine their public address. `fix_nated_contact` rewrites Contact header field with request's source address:port pair. `fix_nated_sdp` adds the active direction indication to SDP (flag 0x01) and updates source IP address too (flag 0x02).

Known devices that get along over NATs with `nathelper` are ATAs (as clients) and Cisco Gateways (since 12.2(T)) as servers. See

http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a0080110bf9.html>
(http://www.cisco.com/en/US/products/sw/iosswrel/ps1839/products_feature_guide09186a0080110bf9.html)

1.2. Dependencies

1.2.1. SER Modules

The following modules must be loaded before this module:

- *No dependencies on other SER modules.*

1.2.2. External Libraries or Applications

The following libraries or applications must be installed before running SER with this module loaded:

- *None.*

1.3. Exported Parameters

1.3.1. `natping_interval` (integer)

Period of time in seconds between sending short UDP packets to all currently registered UAs to keep their NAT bindings alive. Value of 0 disables this functionality.

Default value is 0.

Example 1-1. Set natping_interval parameter

```
...
modparam("nathelper", "natping_interval", 10)
...
```

1.3.2. ping_nated_only (integer)

If this variable is set then only contacts that have “behind_NAT” flag in user location database set set will get ping.

Default value is 0.

Example 1-2. Set ping_nated_only parameter

```
...
modparam("nathelper", "ping_nated_only", 1)
...
```

1.4. Exported Functions**1.4.1. fix_nated_contact()**

Rewrites Contact HF to contain request's source address:port.

Example 1-3. fix_nated_contact usage

```
...
if (search("User-Agent: Cisco ATA.*") {fix_nated_contact();}
...
```

1.5. Exported Functions**1.5.1. fix_nated_sdp(mode)**

Rewrites Contact HF to contain request's source address:port.

Meaning of the parameters is as follows:

- *mode* - 0x01 (add direction=active), 0x02 (rewrite media IP address with source address of the message).

Example 1-4. fix_nated_sdp usage

```
...
if (search("User-Agent: Cisco ATA.*") {fix_nated_sdp("3");};
...
```

1.5.2. force_rtp_proxy()

Rewrites SDP body to ensure that media is passed through an RTP proxy.

Example 1-5. force_rtp_proxy usage

```
...
if (search("User-Agent: Cisco ATA.*") {force_rtp_proxy();};
...
```

1.5.3. force_rtp_proxy(ip_address)

Rewrites SDP body with given IP address to ensure that media is passed through an RTP proxy.

Meaning of the parameters is as follows:

- *ip_address* - new SDP IP address.

Example 1-6. force_rtp_proxy usage

```
...
if (search("User-Agent: Cisco ATA.*") {force_rtp_proxy("1.2.3.4");};
...
```

1.5.4. nat_uac_test(mode)

Tries to guess if client's request originated behind a nat. The parameter determines what heuristics is used. If flag 1 is set, the "received" test is used -- address in Via is compared against source IP address of signaling. If flag 2 is set, Contact header field is searched for occurrence of RFC1918 addresses. Both flags can be bitwise combined, the test returns true if any of the tests identified a NAT.

Chapter 2. Developer's Guide

The module does not provide any sort of API to use in other SER modules.

Chapter 3. Frequently Asked Questions

1. Where can I find more about SER?

Take a look at <http://iptel.org/ser>.

2. Where can I post a question about this module?

First at all check if your question was already answered on one of our mailing lists:

- <http://mail.iptel.org/mailman/listinfo/serusers>
- <http://mail.iptel.org/mailman/listinfo/serdev>

E-mails regarding any stable version should be sent to [<serusers@iptel.org>](mailto:serusers@iptel.org) and e-mail regarding development versions or CVS snapshots should be sent to [<serdev@iptel.org>](mailto:serdev@iptel.org).

If you want to keep the mail private, send it to [<serhelp@iptel.org>](mailto:serhelp@iptel.org).

3. How can I report a bug?

Please follow the guidelines provided at: <http://iptel.org/ser/bugs>