mangler Module

Gabriel Vasile
FhG FOKUS

Edited by
Gabriel Vasile

mangler Module

Edited by and Gabriel Vasileand Gabriel Vasile

Copyright © 2003 Fill in here

Revision History

Revision $Revision: 1.1 $ $Date: 2003/07/24 16:35:42 $

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. `contact_flds_separator` (string) 1
1.4. Exported Functions 2
1.4.1. `sdp_mangle_ip(pattern, newip)` 2
1.4.2. `sdp_mangle_port(offset)` 2
1.4.3. `encode_contact(encoding_prefix)` 3
1.4.4. `decode_contact()` 3
1.4.5. `decode_contact_header()` 3
2. Developer's Guide 5
3. Frequently Asked Questions 6

List of Examples

1-1. Set `db_url` parameter 1
1-2. `sdp_mangle_ip` usage 2
1-3. `sdp_mangle_port` usage 2
1-4. `encode_contact` usage 3
1-5. `decode_contact` usage 3
1-6. `decode_contact_header` usage 4

Chapter 1. User's Guide

1.1. Overview

This is a module to help with SDP mangling. Still in testing.

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. `contact_flds_separator` (string)

First char of this parameter is used as separator for encoding/decoding Contact header.

> **Warning**
>
> First char of this field must be set to a value which is not used inside username,password or other fields of contact.Otherwise it is posible for the decoding step to fail/produce wrong results.

Default value is "".*

Example 1-1. Set `db_url` parameter

```
...
modparam("module", "contact_flds_separator", "-")
...
```

then an encoded uri might look sip:user-password-ip-port-protocol@PublicIP

1.4. Exported Functions

1.4.1. `sdp_mangle_ip(pattern, newip)`

Changes IP addresses inside SDP package in lines describing connections like c=IN IP4 Currently in only changes IP4 addresses since IP6 probably will not need to traverse NAT :)

The function returns negative on error, or number of replacements + 1.

Meaning of the parameters is as follows:

- *pattern* - A pair ip/mask used to match IP's located inside SDP package in lines c=IN IP4 ip. This lines will only be mangled if located IP is in the network described by this pattern. Examples of valid patterns are "10.0.0.0/255.0.0.0" or "10.0.0.0/8" etc.
- *newip* - A string representing the new IP to be put inside SDP package if old IP address matches pattern.

Example 1-2. `sdp_mangle_ip` usage

```
...
sdp_mangle_ip("10.0.0.0/8","193.175.135.38");
...
```

1.4.2. `sdp_mangle_port(offset)`

Changes ports inside SDP package in lines describing media like m=audio 13451.

The function returns negative on error, or number of replacements + 1.

Meaning of the parameters is as follows:

- *offset* - A string representing an integer which will be added/substracted from the located port.

Example 1-3. `sdp_mangle_port` usage

```
...
sdp_mangle_port("-12000");
```

```
...
```

1.4.3. encode_contact(encoding_prefix)

This function will encode uri-s inside Contact header in the following manner sip:username:password@ip:port;transport=protocol goes sip:enc_pref*username*ip*port*protocol@public_ip * is the default separator.

The function returns negative on error, 1 on succes.

Meaning of the parameters is as follows:

- *encoding_prefix* - Something to allow us to determine that a contact is encoded publicip--a routable IP,most probably you should put your external IP of your NAT box.

Example 1-4. encode_contact usage

```
...
if (src_ip == 10.0.0.0/8) encode_contact("enc_prefix","193.175.135.38");
...
```

1.4.4. decode_contact()

This function will decode the URI in first line in packets wich come with encoded URI in the following manner sip:enc_pref*username*ip*port*protocol@public_ip goes to sip:username:password@ip:port;transport=protocol It uses the default set parameter for contact encoding separator.

The function returns negative on error, 1 on succes.

Meaning of the parameters is as follows:

Example 1-5. decode_contact usage

```
...
if (uri =~ "^enc*") { decode_contact(); }
...
```

1.4.5. decode_contact_header()

This function will decode URIs inside Contact header in the following manner sip:enc_pref*username*ip*port*protocol@public_ip goes to sip:username:password@ip:port;transport=protocol. It uses the default set parameter for contact encoding separator.

The function returns negative on error, 1 on succes.

Meaning of the parameters is as follows:

Example 1-6. `decode_contact_header` usage

```
...
if (uri =~ "^enc*") { decode_contact_header(); }
...
```

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> and e-mail regarding development versions or CVS snapshots should be send to <serdev@iptel.org>.

If you want to keep the mail private, send it to <serhelp@iptel.org>.

3. How can I report a bug?

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