



## EEM SMTP Library Command Extensions

All Simple Mail Transfer Protocol (SMTP) library command extensions belong to the `::cisco::lib` namespace.

To use this library, the user needs to provide an e-mail template file. The template file can include Tcl global variables so that the e-mail service and the e-mail text can be configured through the **event manager environment Cisco IOS** command-line interface (CLI) configuration command. There are commands in this library to substitute the global variables in the e-mail template file and to send the desired e-mail context with the To address, CC address, From address, and Subject line properly configured using the configured e-mail server.

### E-Mail Template

The e-mail template file has the following format:



**Note** Based on RFC 2554, the SMTP e-mail server name--Mailservername-- can be in any one of the following template formats: `username:password@host`, `username@host`, or `host`.

```
Mailservername:<space><the list of candidate SMTP server addresses>
From:<space><the e-mail address of sender>
To:<space><the list of e-mail addresses of recipients>
Cc:<space><the list of e-mail addresses that the e-mail will be copied to>
Sourceaddr:<space><the IP addresses of the recipients>
Subject:<subject line>
<a blank line>
<body>
```



**Note** Note that the template normally includes Tcl global variables for configuration.

In a Tcl policy, the port number can be specified by a "Port" line in the e-mail template. If port is not specified, the default port of 25 is used.

Below is a sample e-mail template file:

```
Mailservername: $_email_server
From: $_email_from
To: $_email_to
Cc: $_email_cc
```

```
Sourceaddr: $_email_ipaddr
Port: <port number>
Subject: From router $routername: Process terminated
process name: $process_name
subsystem: $sub_system
exit status: $exit_status
respawn count: $respawn_count
```

- [smtp\\_send\\_email, on page 2](#)
- [smtp\\_subst, on page 3](#)

## smtp\_send\_email

Given the text of an e-mail template file with all global variables already substituted, sends the e-mail out using Simple Mail Transfer Protocol (SMTP). The e-mail template specifies the candidate mail server addresses, To addresses, CC addresses, From address, subject line, and e-mail body.



**Note** A list of candidate e-mail servers can be provided so that the library will try to connect the servers on the list one by one until it can successfully connect to one of them.

### Syntax

```
smtp_send_email text
```

### Arguments

<code>text</code>	(Mandatory) The text of an e-mail template file with all global variables already substituted.
-------------------	--

### Result String

None

### Set \_cerrno

- Wrong 1st line format--Mailservername:list of server names.
- Wrong 2nd line format--From:from-address.
- Wrong 3rd line format--To:list of to-addresses.
- Wrong 4th line format--CC:list of cc-addresses.
- Error connecting to mail server:--\$sock closed by remote server (where \$sock is the name of the socket opened to the mail server).
- Error connecting to mail server:--\$sock reply code is \$k instead of the service ready greeting (where \$sock is the name of the socket opened to the mail server; \$k is the reply code of \$sock).
- Error connecting to mail server:--cannot connect to all the candidate mail servers.
- Error disconnecting from mail server:--\$sock closed by remote server (where \$sock is the name of the socket opened to the mail server).

## Sample Scripts

After all needed global variables in the e-mail template are defined:

```
if [catch {smtp_subst [file join $tcl_library email_template_sm]} result] {
    puts stderr $result
    exit 1
}
if [catch {smtp_send_email $result} result] {
    puts stderr $result
    exit 1
}
```

# smtp\_subst

Given an e-mail template file `e-mail_template`, substitutes each global variable in the file by its user-defined value. Returns the text of the file after substitution.

## Syntax

```
smtp_subst e-mail_template
```

## Arguments

e-mail_template	(Mandatory) Name of an e-mail template file in which global variables need to be substituted by a user-defined value. An example filename could be <code>/disk0://example.template</code> which represents a file named <code>example.template</code> in a top-level directory on an ATA flash disk in slot 0.
-----------------	--

## Result String

The text of the e-mail template file with all the global variables substituted.

## Set \_cerrno

- cannot open e-mail template file
- cannot close e-mail template file

