Command-Line Interface fs_cli

About

The fs_cli program is a Command-Line Interface that allows a user to connect to a running FreeSWITCH™ instance. The fs_cli program can connect to the FreeSWITCH™ process on the local machine or on a remote system. (Network connectivity to the remote system is, of course, required.) The fs_cli program uses FreeSWITCH™ 's event socket to send commands issued by the user and to collect the server responses to send to the display. A C-based socket library, ESL, was developed for use with fs_cli although a programmer could use this library for any C language program that needs to connect to the event socket. With the -x switch (see below) fs_cli can issue a command to the server, get a response, and then disconnect.

The fs_cli program can connect to FreeSWITCH™ whether it is running as a daemon or in console mode, regardless of operating system or if running in screen.

Requirements

Note the fs_cli **requires** the mod_event_socket to be loaded in order to connect to the FreeSWITCH™ server.

The default mod_event_socket configuration binds to ::, which will work on ipv4 or ipv6, this will need to be be changed to be able to connect from remote servers. There are security risks to doing this, so you should also limit access via a firewall and/or ACL, as well as never using the default password.

Install

Server

The FreeSWITCH™ server will build and install the fs_cli client by default.

Client

The client can also be built without needing to build the entire FreeSWITCH™ server.

To build:

```
make current
cd libs/esl
make
```

To run:

```
./fs_cli
```

Usage

Launching fs_cli

The usage for fs_cli is:

command syntax

```
  -?,-h --help                  Usage Information
  -H, --host=hostname           Host to connect (default is 127.0.0.1)
  -P, --port=port               Port to connect (default is "8021")
  -u, --user=user@domain        user@domain
  -p, --password=password       Password (default is "ClueCon")
  -i, --interrupt               Allow Control-c to interrupt
  -x, --execute=command         Execute Command and Exit
  -l, --loglevel=command        Log Level (default is debug)
  -U, --log-uuid                Include UUID in log output
  -S, --log-uuid-short          Include shortened UUID in log output
  -q, --quiet                   Disable logging
  -r, --retry                   Retry connection on failure every two seconds until connected (or until 2
minutes has passed)
  -R, --reconnect               Reconnect if disconnected
  -d, --debug=level             Debug Level (0 - 7)
  -b, --batchmode               Batch mode
  -t, --timeout                 Timeout for API commands (in miliseconds)
  -T, --connect-timeout         Timeout for socket connection (in miliseconds)
  -n, --no-color                Disable color
```

While Connected

While connected the user can issue any command normally available on the FreeSWITCH™ command line. Additionally, there are several commands that can be issued using a forward slash (/) character. The following commands all disconnect from the FreeSWITCH™ command line:

- /quit
- /bye
- /exit

Other "slash" commands correspond to mod_event_socket commands:

- /event - Enable events (More Info)
- /noevents - Disable all events previously enabled with /event
- /nixevent - Enable all but one type of event (More Info)
- /log - set loglevel of the FreeSWITCH™ daemon (e.g. /log info or /log alert) (More Info)
- /nolog - Disable logging (More Info)
- /uuid - Filter logs for a single call uuid
- /filter (More Info)
- /help - List fs_cli commands

Additional command available only in fs_cli

- /logfilter <string> - Displays only lines in /log containing <string>. /logfilter without argument disables logfilter

For command-line editing, see "Command-Line Editing" section of mod_console. The details are the same as of SVN r13964.

Configuration Options

Optional configuration files can be set up. The config file uses a simple INI-style layout and allows for multiple profiles. This allows one to access many FreeSWITCH™ systems from a single workstation.

There is /etc/fs_cli.conf for system-wide settings and ~/.fs_cli_conf for user-specific settings.

fs_cli.conf options

```
host => <hostname>                    Host to connect (default is 127.0.0.1)
port => <port>                        Port to connect (default is "8021")
user => <username>                    user@domain
password => <password>                Password (default is "ClueCon")
interrupt => true | false             Allow Control-c to interrupt
execute => <command>                  Execute Command and Exit
loglevel => console | alert | crit | err | warning | notice | info | debug        Log Level (default is debug)
log-uuid => true | false              Include UUID in log output
log-uuid-short => true | false        Include shortened UUID in log output
quiet => true | false                 Disable logging
retry => true | false                 Retry connection on failure every two seconds until connected (or until
2 minutes has passed)
reconnect => true | false             Reconnect if disconnected
debug => <0 .. 7>                     Debug Level (0 - 7)
batchmode => true | false             Batch mode
timeout => <milliseconds>             Timeout for API commands (in miliseconds)
connect-timeout => <milliseconds>     Timeout for socket connection (in miliseconds)
no-color => true | false              Disable color
```

An example file:

```
[default]
; Put me in /etc/fs_cli.conf or ~/.fs_cli_conf
;overide any default options here
loglevel => 6
log-uuid => false
quiet => false
key_f1 => help
key_f2 => status
key_f3 => show channels
key_f4 => show calls
key_f5 => sofia status
key_f6 => reloadxml
key_f7 => /log console
key_f8 => /log debug
key_f9 => sofia status profile internal
key_f10 => sofia global siptrace on
key_f11 => sofia global siptrace off
key_f12 => version

[profile1]
host     => 192.168.1.10
port     => 8021
password => secret_password
debug    => 7

[profile2]
host     => 192.168.1.11
port     => 8021
password => someother_password
loglevel => info
```

Usage Examples

Simple

`fs_cli`

Launches fs_cli which connects to local machine using default username, password, and debug level.

With Profile

`fs_cli My_profile`

Launches fs_cli using profile named "My_profile" found in .fs_cli_conf file.

Sending a command and then logging off

```
fs_cli -x "sofia status profile internal"
```

Launches fs_cli and sends a command before logging off. The output of the above command looks like this:

```
fs_cli -x "sofia status"
                  Name          Type                                Data    State
=================================================================================================
              external      profile      sip:mod_sofia@71.6.70.229:5080     RUNNING (0)
           example.com      gateway              sip:joeuser@example.com     NOREG
              internal      profile   sip:mod_sofia@192.168.50.229:5060     RUNNING (0)
        192.168.50.229        alias                             internal     ALIASED
          internal-ipv6      profile           sip:mod_sofia@[::1]:5060     RUNNING (0)
               default        alias                             internal     ALIASED
                   nat        alias                             external     ALIASED
              outbound        alias                             external     ALIASED
=================================================================================================
3 profiles 4 aliases
```

Wish List

- Option to see all output to include FS console output.
- Option to see all output from all fs_cli instances connected to the FS box, plus the console.
- Option to connect to more than one FS box.

See Also

- mod_event_socket
- Event Socket Library
- Event list
- Fs_gui