Client Configuration Guide

Linux/Mac v2.3

This document is designed to quickly get you up and running on Linux or Mac using a customized client configuration.

TELESPLOIT

November 28, 2017

Client Configuration Guide

Linux/Mac v2.3

Contents

telesploit

exploitation at a distance

www.telesploit.com

Overview

The Telesploit solution consists of three distinct parts: the Telesploit server, the Telesploit relay, and an SSH capable client.

Telesploit Server

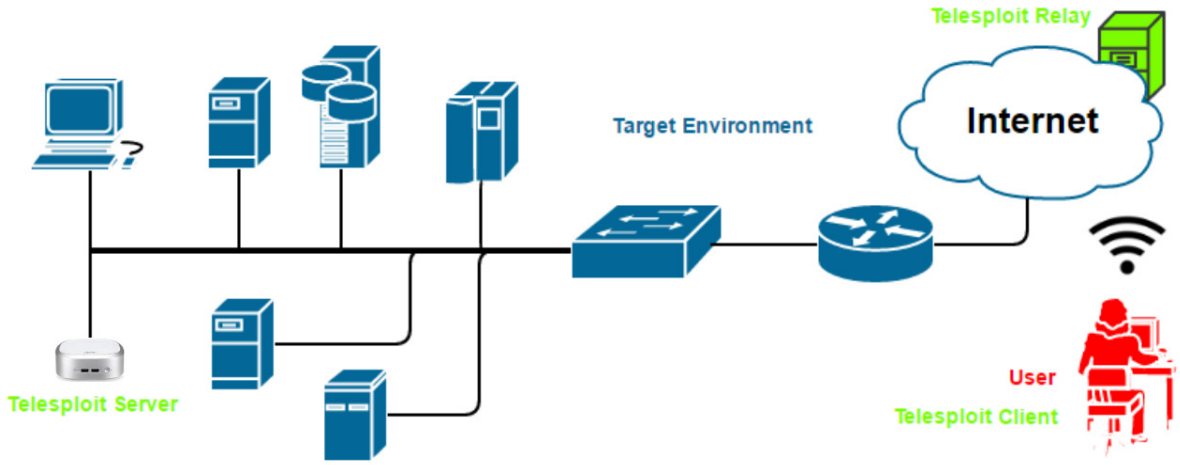
The server runs a customized version of Kali Linux and is deployed within the target environment. Once network connectivity and power have been applied to the device, it will automatically connect to the relay server and create TLS encapsulated reverse SSH tunnels in its default configuration. These connections provide access to a command line interface (SSH), remote desktop (VNC), web proxy (Squid), and many other applications on the Telesploit server.

Telesploit Relay

The relay runs in the cloud and provides secure access to the Telesploit server from Internet-connected clients using SSH key-based authentication. The relay includes pre-configured IRC and Mattermost servers for team-based communication and collaboration.

Client

The client connects to the Telesploit server via the relay. Penetration testing tools, such as Metasploit, can then be run directly from the server within the target environment or proxied through the established connections.



Client Setup

Telesploit will provide a URL to download the customized scripts for connecting to your dedicated relay and server.

Example: https://relay-d015.telesploit.com/8b85b0fbb32c0575bc3cb21cc1af7db4eb167eed0b0d2de101bc7572363415bc/telesploit-d015-client.tar.gz

Download and Extract Client

Download the archive file. In this example we are accessing the Telesploit relay/server with the designation d015. Change the commands to reflect your assigned environment.

curl https://relay-d015.telesploit.com/8b85b0fbb32c0575bc3cb21cc1af7db4eb167eed0b0d2de101bc7572363415bc/telesploit-d015-client.tar.gz -o telesploit-d015-client.tar.gz

```
support@telesploit:~/demo$ curl https://relay-d015.telesploit.com/8b85b0fbb32c0575bc3cb21cc1af7db4eb167eed0b0d2de101bc7572363415bc/telesploit
-d015-client.tar.gz -o telesploit-d015-client.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  4410  100  4410    0     0  13466      0 --:--:-- --:--:-- --:--:-- 13445
```

The integrity may be validated by performing a sha256sum on the file. The value should match the subdirectory name in the URL.

sha256sum telesploit-d015-client.tar.gz

```
support@telesploit:~/demo$ curl https://relay-d015.telesploit.com/8b85b0fbb32c0575bc3cb21cc1af7db4eb167eed0b0d2de101bc7572363415bc/telesploit
-d015-client.tar.gz -o telesploit-d015-client.tar.gz
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100  4410  100  4410    0     0  13466      0 --:--:-- --:--:-- --:--:-- 13445
support@telesploit:~/demo$
support@telesploit:~/demo$ sha256sum telesploit-d015-client.tar.gz
8b85b0fbb32c0575bc3cb21cc1af7db4eb167eed0b0d2de101bc7572363415bc  telesploit-d015-client.tar.gz
```

If the checksum matches then extract the archive file and change into the newly created directory.

tar -zxvf telesploit-d015-client.tar.gz && cd telesploit-client/telesploit-d015

```
support@telesploit:~/demo$ tar -zxvf telesploit-d015-client.tar.gz && cd telesploit-client/telesploit-d015
telesploit-client/telesploit-d015/
telesploit-client/telesploit-d015/client-configs/
telesploit-client/telesploit-d015/client-configs/client.cfg
telesploit-client/telesploit-d015/console.sh
telesploit-client/telesploit-d015/setup_client.sh
telesploit-client/telesploit-d015/update-server/
telesploit-client/telesploit-d015/update-server/network.sh
telesploit-client/telesploit-d015/update-server/server.cfg
telesploit-client/telesploit-d015/update-server/encrypted-configs/
telesploit-client/telesploit-d015/update-server/keys.sh
telesploit-client/telesploit-d015/update-server/connection.sh
telesploit-client/telesploit-d015/create_tunnels.sh
telesploit-client/telesploit-d015/kill_tunnels.sh
support@telesploit:~/demo/telesploit-client/telesploit-d015$
```

You should see the following files and directories:

```
support@telesploit:~/demo/telesploit-client/telesploit-d015$ ls
client-configs  console.sh  create_tunnels.sh  kill_tunnels.sh  readme.txt  setup_client.sh  update-server
```

The readme.txt file contains the server, relay, and port assignments for your Telesploit deployment. These should be used to replace the examples given in the subsequent sections.

```
./setup_client.sh
./create_tunnels.sh
./console.sh

For detailed instructions download the Linux/Mac Configuration Guide from https://www.telesploit.com.

Telesploit Server: telesploit-d015

Telesploit Relay: relay-d015.telesploit.com

Assigned Ports:

SSH: 13015
VNC: 23015
Web Proxy: 33015
SOCKS Proxy: 43015
IRC: 53015
Collaboration: 63015
```

Configure Client

The first time you setup your client you will need to run the script setup_client.sh. If you are in an environment that allows outbound SSH then it is recommended that you select that option. If you will be tunneling the SSH connections through TLS then ncat (part of the nmap suite) will be required on your system. If your environment requires using a proxy for outbound TLS connections then both ncat and proxytunnels are required. The proxy information will also need to be either entered in the configuration file client-configs/client.cfg or interactively from within the setup script.

./setup_client.sh

```
support@telesploit:~/demo/telesploit-client/telesploit-d015$ ./setup_client.sh
This script will configure the telesploit client

After completing this script, run ./create_tunnels.sh to setup the tunnels used to connect to the telesploit relay

After the tunnels have been established:
1) The script ./server_console.sh may be run to obtain a command line on the telesploit server
2) A VNC desktop on the telesploit server can be accessed at 127.0.0.1:23015
3) Web applications can be configured to use the squid proxy on the telesploit server by setting the upstream proxy to 127.0.0.1:33015
4) SOCKS applications can be configured to use the SOCKS proxy on the telesploit server by configuring the SSH config file on the server and
setting the application's upstream proxy to 127.0.0.1:43015
5) The IRC server on the telesploit relay can be accessed at 127.0.0.1:53015
6) The Mattermost server on the telesploit relay can be accessed at 127.0.0.1:63015

Review the following values. If they are incorrect then change the entries in ./client-configs/client.cfg and rerun the script
platform=telesploit
If configuring the telesploit client for an SSH connection or direct TLS outbound connection then no other information is required to be vali
dated

If configuring the telesploit client to use an outbound proxy server then validate that the following values are correct
proxy_server= Note: If no proxy server is defined then it will be requested interactively
proxy_port= Note: If no proxy port is defined then it will be requested interactively

If configuring the telesploit client for a simple proxy (no authentication) then no other information is required to be validated

If configuring the telesploit client to use an authenticated proxy server then validate that the following values are correct
proxy_username= Note: If no proxy username is defined then it will be requested interactively
proxy_password= Note: If no proxy password is defined then it will be requested interactively

If configuring the telesploit client for a proxy employing BASIC authentication, then no other information is required to be validated

proxy_domain= Note: If configuring the telesploit client for a proxy employing NTLM and no proxy domain is configured then it will be request
ed interactively

If all values are correct then proceed
Press any key to continue or Ctrl+C to exit...
```

Enter the path to the private key corresponding to the public key previously provided to Telesploit. Tab complete is enabled on this field.

```
Enter the full path to the private key being used to access the telesploit server, e.g. /home/user/.ssh/user.id_rsa, followed by [ENTER]: ~/d
emo/keys/telesploit-d015
```

Choose your connection type. SSH is recommended. Direct (TLS) requires ncat to be installed while all proxy connection types require both ncat and proxytunnels.

```
What type of connection should the telesploit client use?
[S]SH - SSH connection (recommended if SSH is allowed out).
[D]irect - TLS, no proxy required (must have ncat installed).
[P]lain - Simple proxy. No password required (must have proxytunnels intalled).
[B]asic - Proxy uses BASIC authentication (must have proxytunnels intalled).
[N]TLM - Proxy uses NTLM authentication (must have proxytunnels intalled)
Choose the telesploit client connection type [S/D/P/B/N]: S
```

The script will then pull a file from the relay containing the SSH server's known fingerprint and compare it to a locally generated version. Matching files indicate that an active Man-in-the-Middle attack is not being performed against your connection.

```
Choose the telesploit client connection type [S/D/P/B/N]: S

saving connection status to ./client-configs/connection.cfg
creating ssh config file and saving to ./client-configs/config

retrieving trusted fingerprint from https://relay-d015.telesploit.com/trusted
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left  Speed
100   407  100   407    0     0   1067      0 --:--:-- --:--:-- --:--:--  1068
retrieving ssh fingerprint from relay-d015.telesploit.com
# relay-d015.telesploit.com:22 SSH-2.0-OpenSSH_7.2p2 Ubuntu-4ubuntu2.2
running diff against trusted and tested
Files ./client-configs/trusted and ./client-configs/tested are identical
identical files indicate a secure connection
non-matching files may indicate an active man-in-the-middle attack, review the files 'trusted' and 'tested' before continuing
Press any key to continue or Ctrl+C to exit...
```

Enter the password for the private key corresponding to the public key previously provided to Telesploit. If you are not prompted for your password then verify the location and permissions on your private key and that you have ncat installed if using TLS or proxy connections then re-run setup_client.sh.

```
enter the password for your SSH key for d015 at the prompt
Enter passphrase for key '/home/support/demo/keys/telesploit-d015':
```

You should then be returned to the command prompt.

Establish SSH Tunnels and Connect to the Server

Once the client has been configured for your environment, verify that you can create SSH tunnels to the relay by running the script create_tunnels.sh. You will once again be prompted for the password to your SSH private key.

./create_tunnels.sh

```
support@telesploit:~/demo/telesploit-client/telesploit-d015$ ./create_tunnels.sh
Enter passphrase for key '/home/support/demo/keys/telesploit-d015':
support@telesploit:~/demo/telesploit-client/telesploit-d015$
```

Once the tunnels have been established you can verify connectivity by connecting to the Telesploit server using the script console.sh. You will once again be prompted for the password to your SSH private key. If everything has been configured properly you should see a Kali Linux command prompt.

./console.sh

```
support@telesploit:~/demo/telesploit-client/telesploit-d015$ ./console.sh
Enter passphrase for key '/home/support/demo/keys/telesploit-d015':

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

root@telesploit-d015:~#
```

Disconnecting from Server and Closing Tunnels

Typing 'exit' at the server command prompt will return you to your local shell.

exit

```
root@telesploit-d015:~# exit
logout
Connection to localhost closed.
support@telesploit:~/demo/telesploit-client/telesploit-d015$
```

When not performing testing, you may teardown the tunnels by running the script kill_tunnels.sh. You should immediately be returned to the command prompt.

./kill_tunnels.sh

```
support@telesploit:~/demo/telesploit-client/telesploit-d015$ ./kill_tunnels.sh
support@telesploit:~/demo/telesploit-client/telesploit-d015$
```

The next time you wish to connect to the Telesploit server it is not necessary to re-run the setup_client.sh script, just use the create_tunnels.sh script to bring the tunnels back up.

Common Tool Configurations

Please note that the SSH, VNC, Squid, and PostgreSQL services provided on the Telesploit server have been configured to only listen on localhost. If you install any additional services, such as Nessus, and do not want them to be exposed to the testing environment then restrict their access as well.

The following sections assume that you have configured the Telesploit client and established the required SSH tunnels.

Command Line Interface

Your SSH client of choice may be used by configuring it with the following values. Adjust the port number to match your Telesploit deployment.

Example SSH Configuration

Host: localhost (127.0.0.1)
Username: root
Password: N/A
Private Key: Your SSH private key
Port: 13015
Note: As with any remote console, Telesploit recommends using a detachable session, such as screen, for long running processes.

This example uses the script console.sh included with the Telesploit client.

./console.sh

```
support@telesploit:~/demo/telesploit-client/telesploit-d015$ ./console.sh
Enter passphrase for key '/home/support/demo/keys/telesploit-d015':

The programs included with the Kali GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Kali GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.

root@telesploit-d015:~#
```

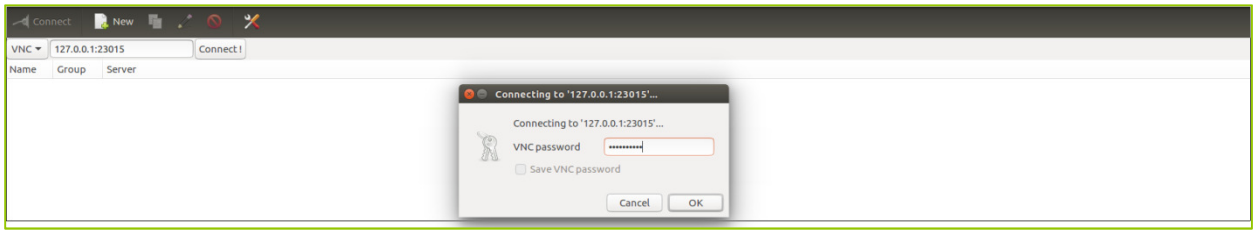
Remote Desktop

Your VNC client of choice may be used by configuring it with the following values. Adjust the port number to match your Telesploit deployment.

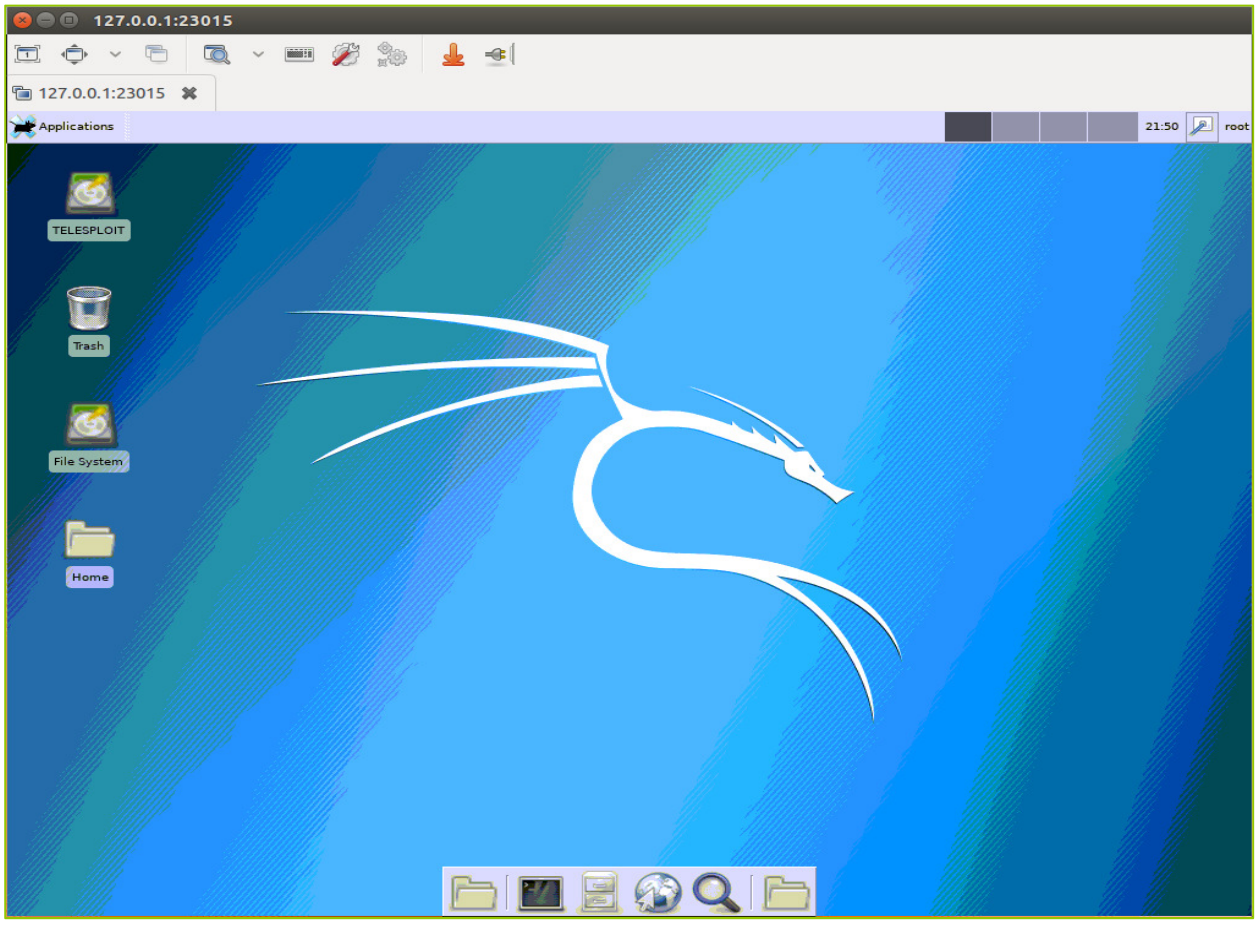
Example VNC Configuration

Host: localhost (127.0.0.1)
Username: <NONE>
Password: telesploit
Port: 23015

The following example uses the Remmina Remote Desktop Client. You will be prompted to enter the VNC password. As the VNC server is only listening on localhost, and connectivity requires SSH key authentication, this password is superfluous and has thus been set to 'telesploit' for all deployments.



Selecting the OK button will establish a remote desktop session on the Telesploit server.



Web Proxy

Your browser and web application assessment tools of choice (e.g. Chrome, Edge, Burp Suite, Zap) may be used by configuring them with the following values. Adjust the port number to match your Telesploit deployment.

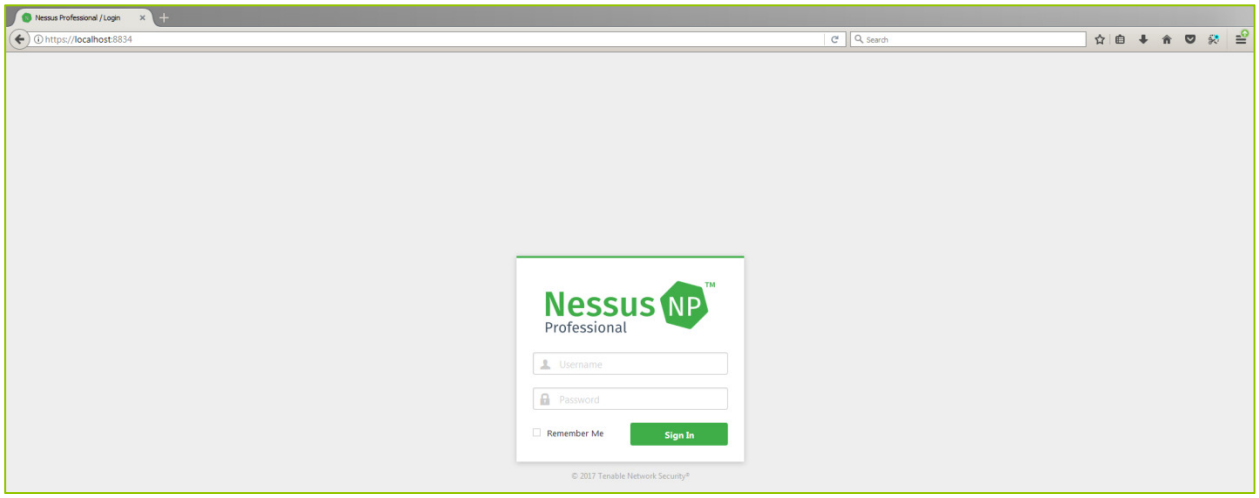
Example Web Proxy Configuration

Host: localhost (127.0.0.1)
Username: <NONE>
Password: <NONE>
Port: 33015

In Firefox these settings can be found under Options -> Network Proxy -> Settings.

Web applications within the target environment can then be accessed by entering their IP address or Fully Qualified Domain Name along with port they are running on just as if you were testing from the local network.

Web-enabled applications running on the Telesploit server itself, such as Nessus (license not included), can be accessed by entering localhost or 127.0.0.1 and the port number.



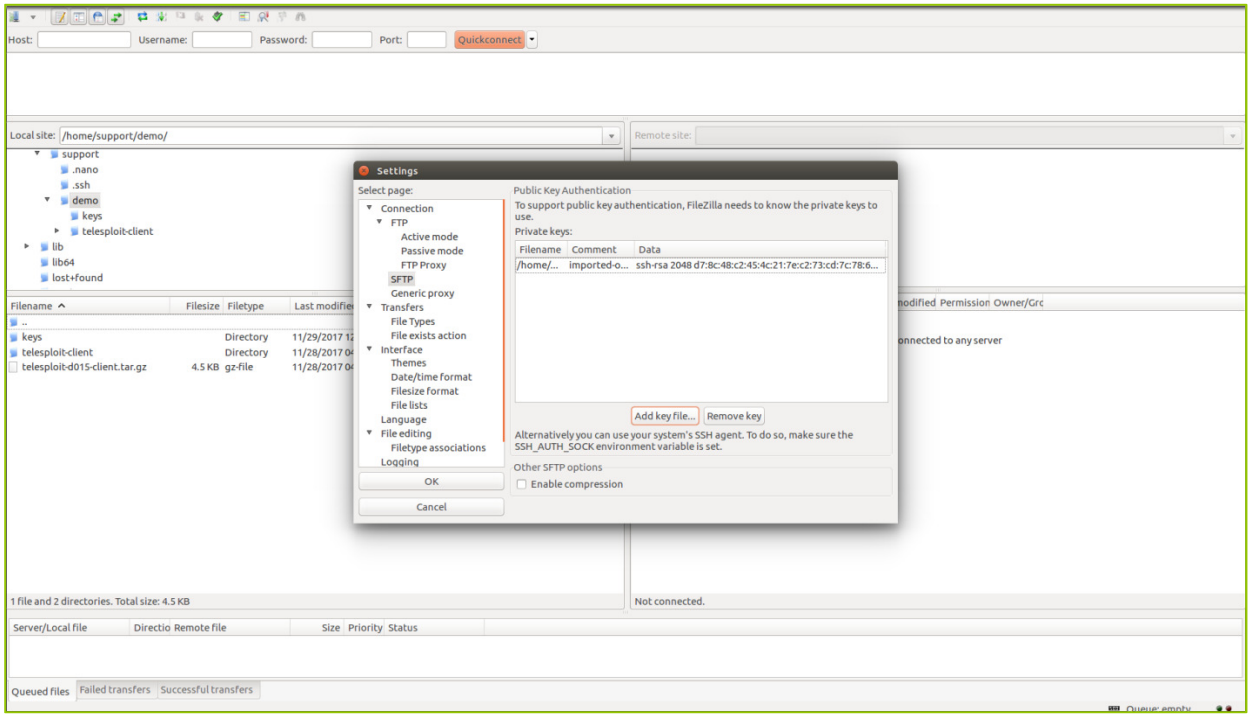
File Transfer

In addition to command line utilities, such as scp, file transfer tools like FileZilla may be used by configuring them with the following values. Adjust the port number to match your Telesploit deployment.

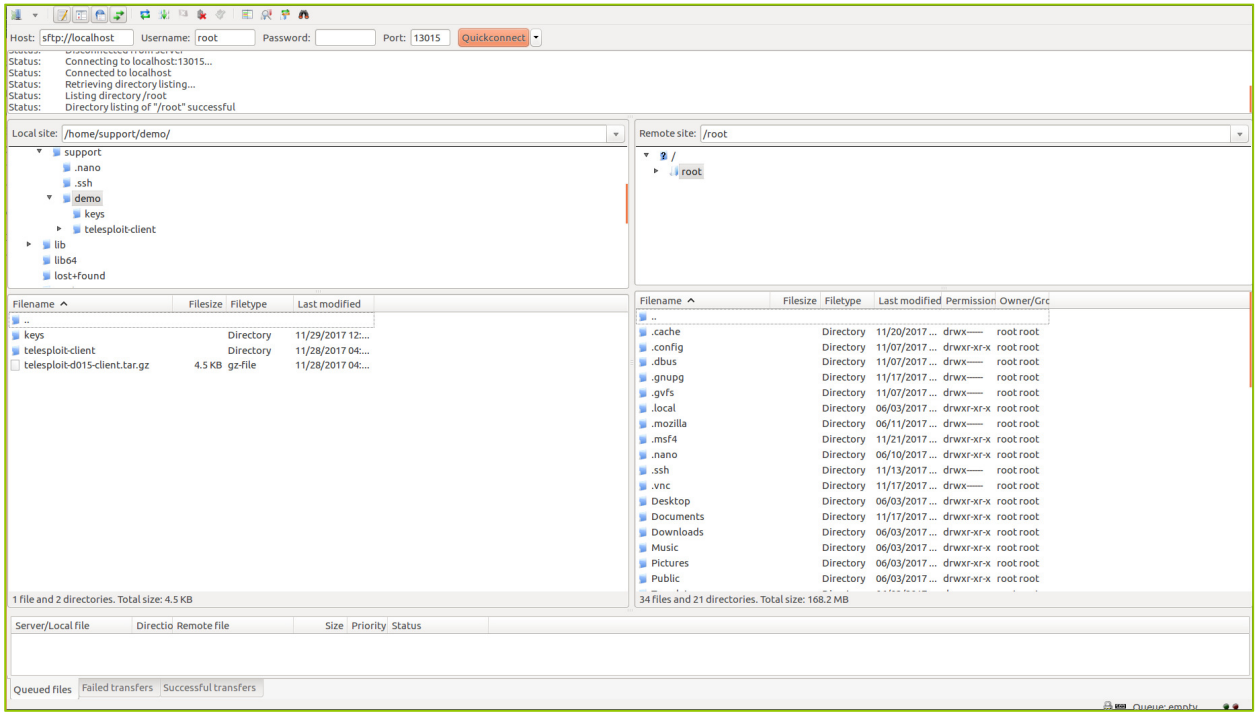
Example File Transfer Configuration

Host: localhost (127.0.0.1)
Username: root
Password: N/A
Private Key: Your SSH private key
Port: 13015

The following example uses FileZilla. To configure the application to use your SSH private key select Edit -> Preferences -> SFTP -> Add key file.



Enter sftp://localhost in the Host field, leave the password blank to force key-based authentication, and enter the SSH port number assigned to your deployment in the Port field. Select Quickconnect, accept the SSH fingerprint, and enter the password to your private key.



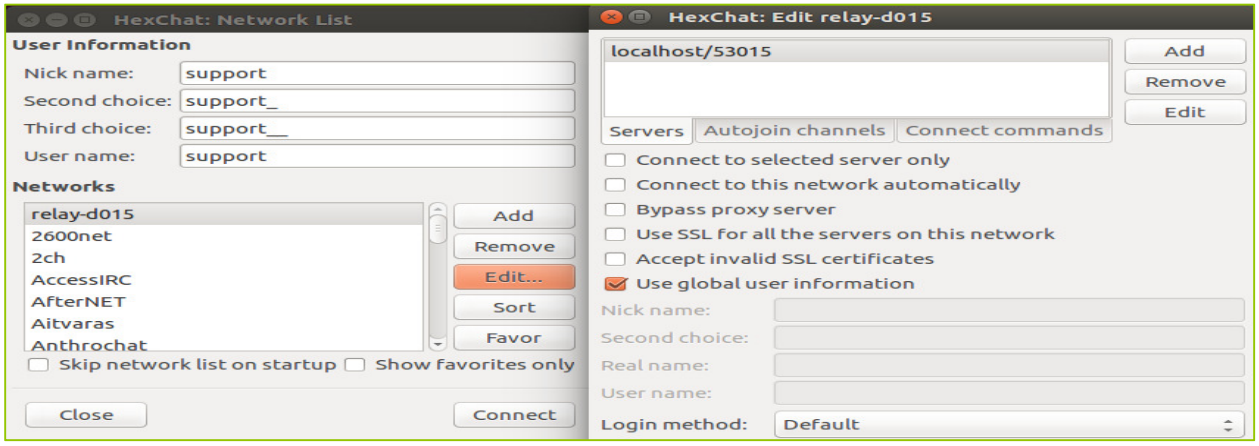
Internet Relay Chat

The Telesploit relay has an IRC server built in and both the client and server create SSH tunnels to communicate with it. Your IRC client of choice may be used by configuring it with the following values. Adjust the port number to match your Telesploit deployment.

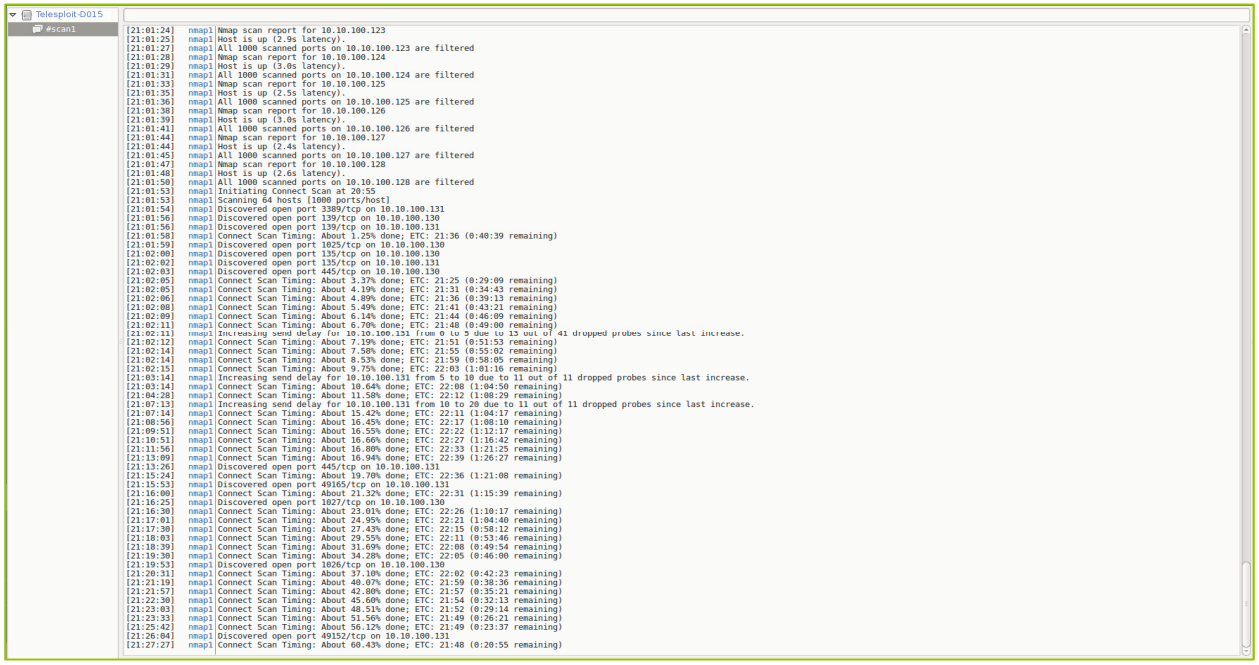
Example IRC Configuration

Host: localhost (127.0.0.1)
Username: <ANY>
Password: <NONE>
Port: 53015

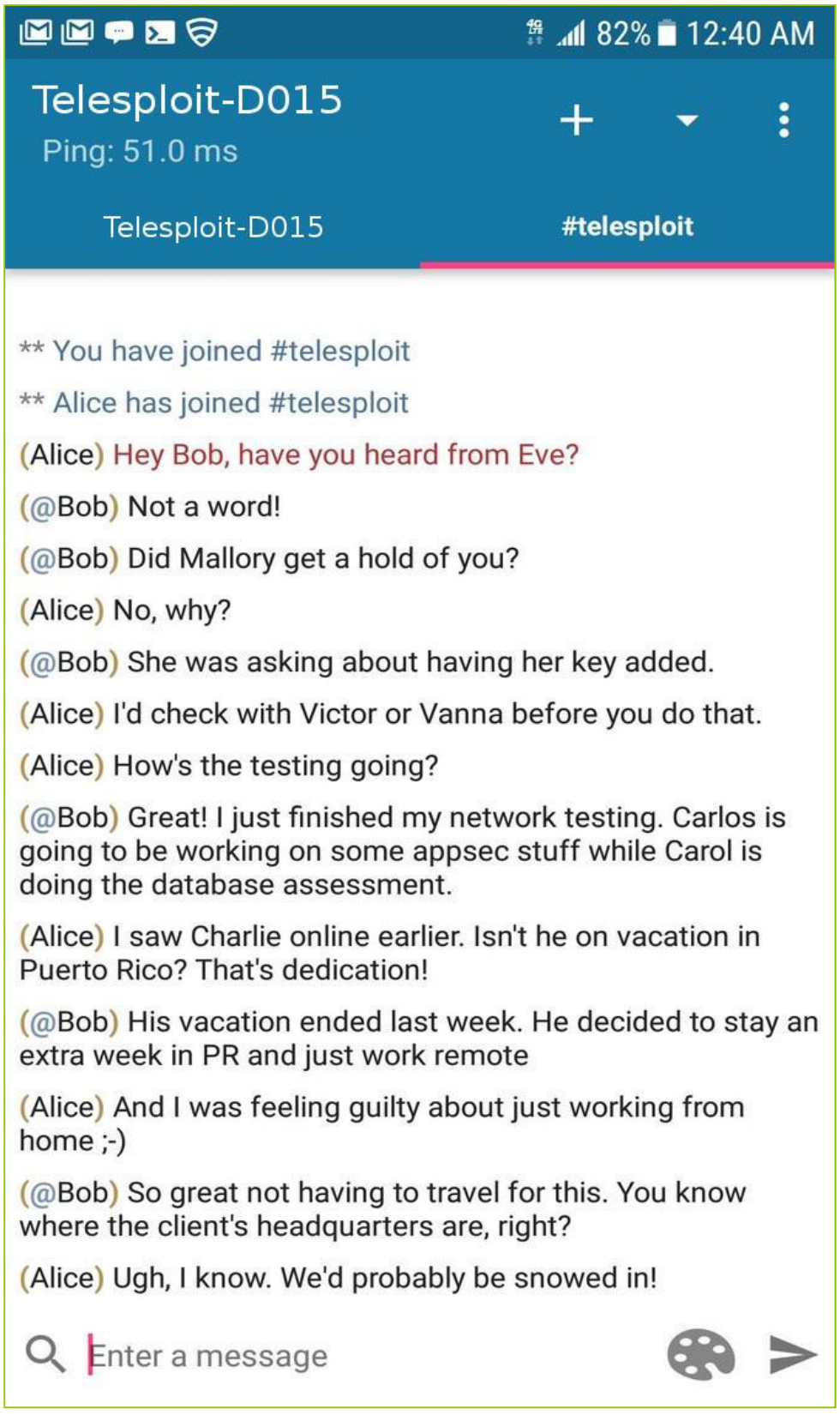
The following example uses HexChat.



In addition to communication between team members, the IRC server can be used to provide updates from scanners and other tools running on the Telesploit server. Dedicate a channel for the output of each discrete activity and immediately know its status without cycling through multiple consoles or screen sessions.



Stay connected on the go with mobile SSH and chat applications.



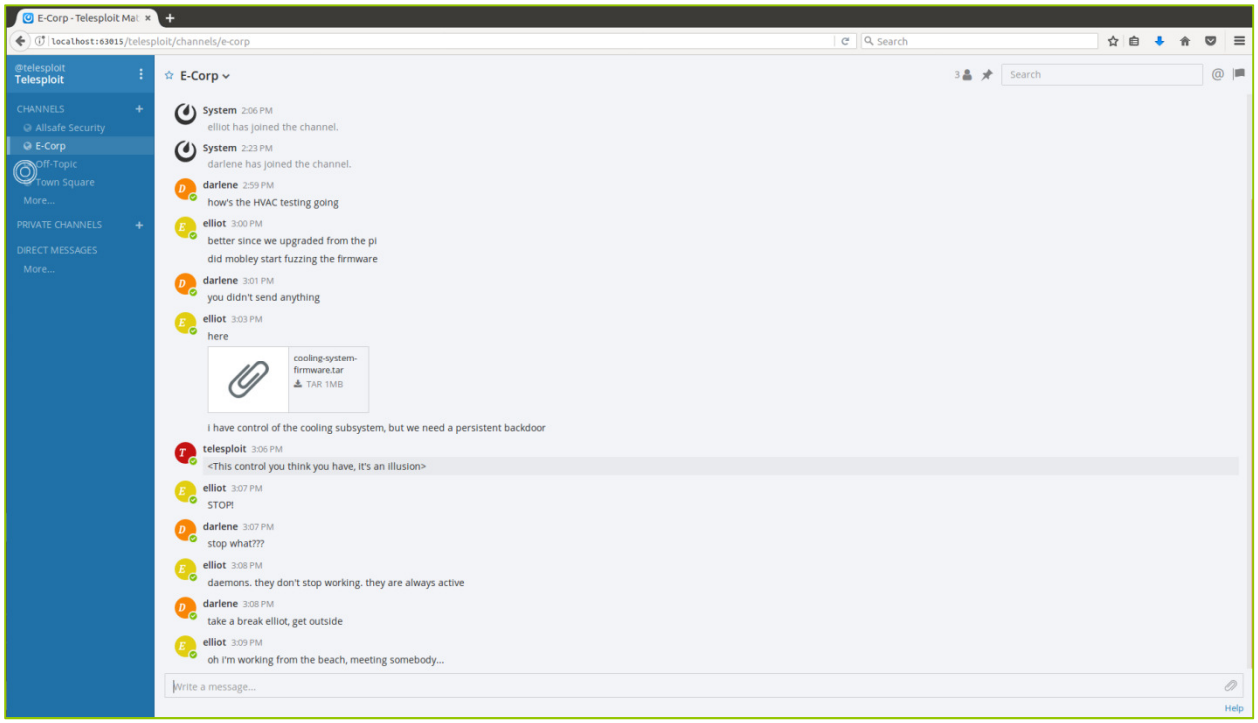
Collaboration

The Telesploit relay has a Mattermost instance installed and both the client and server create SSH tunnels to communicate with it. A web browser or Mattermost client may be used by configuring them with the following values. Adjust the port number to match your Telesploit deployment.

Example Mattermost Configuration

Host: localhost (127.0.0.1)
Username: <Assigned by Mattermost Admin>
Password: <Assigned by Mattermost Admin>
Port: 63015

The first configured user will become the Mattermost admin. The following example uses a standard browser to access the collaboration platform.



Troubleshooting

If you are unable to connect to the Telesploit server, then verify that you are able to directly connect to the relay.

If your client is configured to use an SSH connection, then try directly accessing the SSH server on the relay. You will not be able to successfully login with the following command, but it will validate that nothing is blocking your access and that the relay is up.

ssh test@relay-d014.telesploit.com

```
support@telesploit:~/demo/telesploit-client/telesploit-d015$ ssh test@relay-d015.telesploit.com
The authenticity of host 'relay-d015.telesploit.com (52.14.156.116)' can't be established.
ECDSA key fingerprint is SHA256:zxnvFlxtNVPkdLRvukdr8mFkaLzlzIHCn2/VCFkLIzE.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'relay-d015.telesploit.com,52.14.156.116' (ECDSA) to the list of known hosts.
Permission denied (publickey).
```

If you are using TLS or proxy connections, then try accessing the SSH server through the HA Proxy running on the relay using ncat. You should see output similar to the following if nothing is blocking your access and the HA Proxy and SSH server are up.

ncat -v --ssl relay-d015.telesploit.com 443

```
support@telesploit:~/demo/telesploit-client/telesploit-d015$ ncat -v --ssl relay-d015.telesploit.com 443
Ncat: Version 7.01 ( https://nmap.org/ncat )
Ncat: SSL connection to 52.14.156.116:443.
Ncat: SHA-1 fingerprint: 117D 8A44 F399 4A7D F9CF 84B8 07DF 2358 E4B0 6E84

SSH-2.0-OpenSSH_7.2p2 Ubuntu-4ubuntu2.2
Protocol mismatch.
```

If either of the above tests result in timeouts or no connections then verify that your outbound connections are not being blocked by a firewall or Intrusion Prevention System. Contact Telesploit support for additional assistance.

If the tests are successful then verify that the tunnels have been created by running netstat and reviewing the output.

If the tunnels have not been created then the output should look similar to the following depending on what services you have running.

netstat -plnt

```
support@telesploit:~/demo/telesploit-client/telesploit-d015$ netstat -plnt
(No info could be read for "-p": geteuid()=1001 but you should be root.)
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
tcp        0      0 127.0.0.1:5141          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.1.1:53            0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      -
tcp6       0      0 ::1:631                 :::*                    LISTEN      -
```

If the tunnels have been established then the output should look similar to the following, again depending on what services you have running.

netstat -plnt | sort -k7

```
support@telesploit:~/demo/telesploit-client/telesploit-d015$ netstat -plnt | sort -k7
(Not all processes could be identified, non-owned process info
 will not be shown, you would have to be root to see it all.)
Active Internet connections (only servers)
tcp        0      0 127.0.0.1:5141          0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN      -
tcp        0      0 127.0.1.1:53            0.0.0.0:*               LISTEN      -
tcp6       0      0 ::1:631                 :::*                    LISTEN      -
tcp        0      0 127.0.0.1:13015         0.0.0.0:* SSH           LISTEN      5339/ssh
tcp        0      0 127.0.0.1:23015         0.0.0.0:* VNC           LISTEN      5339/ssh
tcp        0      0 127.0.0.1:33015         0.0.0.0:* Squid         LISTEN      5339/ssh
tcp        0      0 127.0.0.1:43015         0.0.0.0:* SOCKS         LISTEN      5339/ssh
tcp        0      0 127.0.0.1:53015         0.0.0.0:* IRC           LISTEN      5339/ssh
tcp        0      0 127.0.0.1:63015         0.0.0.0:* Mattermost    LISTEN      5339/ssh
Proto Recv-Q Send-Q Local Address           Foreign Address         State       PID/Program name
```

If the tunnels have not been established then re-run the script create_tunnels.sh. If the tunnels have been established run the script kill_tunnels.sh followed by create_tunnels.sh to clear any hung connections.

If you receive an error similar to the following after running create_tunnels.sh then the tunnels have already been established.

```
support@telesploit:~/demo/telesploit-client/telesploit-d015$ ./create_tunnels.sh
Enter passphrase for key '/home/support/demo/keys/telesploit-d015':
bind: Address already in use
channel_setup_fwd_listener_tcpip: cannot listen to port: 13015
bind: Address already in use
channel_setup_fwd_listener_tcpip: cannot listen to port: 23015
bind: Address already in use
channel_setup_fwd_listener_tcpip: cannot listen to port: 33015
bind: Address already in use
channel_setup_fwd_listener_tcpip: cannot listen to port: 43015
bind: Address already in use
channel_setup_fwd_listener_tcpip: cannot listen to port: 53015
bind: Address already in use
channel_setup_fwd_listener_tcpip: cannot listen to port: 63015
Could not request local forwarding.
```

If you receive an error similar to the following after running kill_tunnels.sh then there are no active tunnels to teardown.

```
support@telesploit:~/demo/telesploit-client/telesploit-d015$ ./kill_tunnels.sh
error: list of process IDs must follow -p

Usage:
 ps [options]

 Try 'ps --help <simple|list|output|threads|misc|all>'
  or 'ps --help <s|l|o|t|m|a>'
 for additional help text.

For more details see ps(1).
```

If you are still unable to create SSH tunnels to the relay then run the script kill_tunnels.sh followed by setup_client.sh before once again executing create_tunnels.sh. If this still fails then please contact Telesploit support.