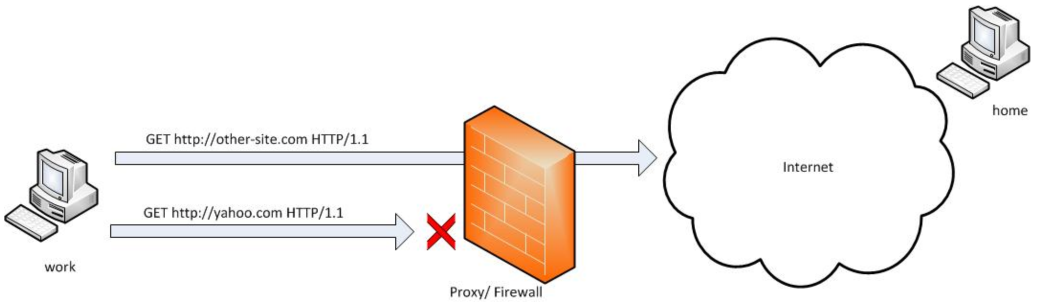
How to create ssh tunnels / ssh tunneling on Linux and FreeBSD with openssh

Author : admin


GET http://other-site.com HTTP/1.1
GET http://yahoo.com HTTP/1.1
work
Proxy/ Firewall
Internet
home

SSH tunneling allows to send and receive traffic using a dedicated port. Using an ssh traffic can have many reasons one most common usage reason is to protect the traffic from a host to a remote server or to access port numbers which are by other means blocked by firewall, e.g.: **(get around firewall filtering)** SSH tunneling works only with TCP traffic. The way to make ssh tunnel is with cmds:

```
  host:/root# ssh -L localhost:deshost:destport username@remote-server.net
host:/root# ssh -R restport:desthost:localport username@remote-server.net
host:/root# ssh -X username@remote-server.net
```

This command will make ssh to bind a port on localhost of the host **host:/root#** machine to the host **desthost:destport** (destination host : destinationport). Important to say *deshost* is the host destination visible from the **remote-server.net** therefore if the connection is originating from *remote-server.net* this means *desthost* will be localhost.
Mutiple ssh tunnels to multiple ports using the above example commands is possible. Here is one **example of ssh tunneling**
Let's say its necessery to access an FTP port (21) and an http port (80), listening on *remote-server.net* In that case **desthost** will be *localhost* , we can use locally the port (8080) insetad of 80, so it will be no necessery to make the ssh tunnel with root (admin privileges). After the ssh session gets opened both services will be accessible on the local ports.

```
  host:/home/user$ ssh -L 21:localhost:21 -L 8080:localhost:80 user@remote-server.net
```

That's all enjoy ;)