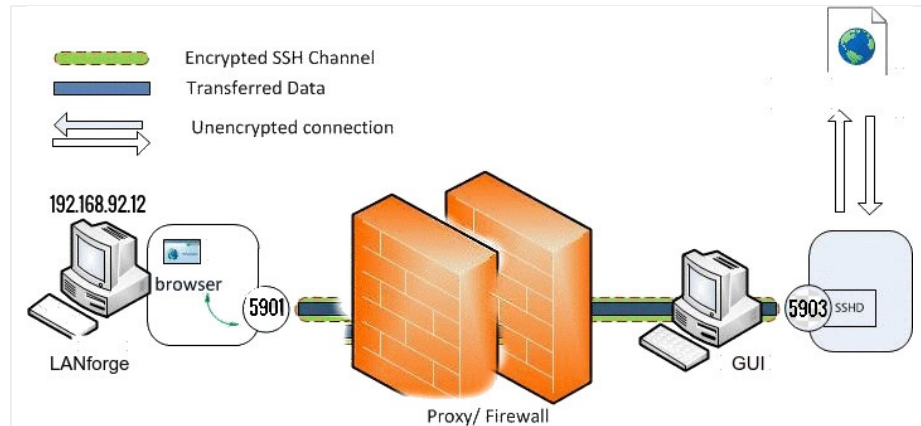


Set up an SSH-tunnel on Windows, Linux, or Mac

Goal: Connect to a LANforge Linux system via a compressed tunnel connection

When connecting to your remote LANforge hardware (presumably accessible over a VPN) you will notice poor response time and lag in your LANforge GUI or your VNC connection. Many VPN connections are based on UDP protocols and packet loss might be affecting your connection quality. Below we explain how to set up SSH tunnels that increase the quality of your connection.



1. Linux SSH tunnel Setup

- A. From the computer that you are trying to connect your SSH tunnel from, open the `.bashrc` file from `'/home/user/'`. The `.bashrc` file can be opened via `gedit`, `vim`, or `nano`. This `.bashrc` file is where the alias will be setup to properly invoke your `ssh`.

```
dipti@muffin:/home$ gedit ~/.bashrc
dipti@muffin:/home$
```

- B. Once the `.bashrc` file is open, type in your alias in any blank spot (that is not within another for-loop or definition). Flags used in the bash alias creation example below are as followed:
- `-C` = Requests compression of data. This is desirable for slower connections. Recommended.
 - `n` = redirects `stdin` `/dev/null`. Required when SSH is running in the background.
 - `N` = do not execute a remote command, useful for forwarding ports.
 - `v` = Verbose mode. Causes SSH to print debugging messages about its progress.
 - `-L` = `bind_address:port:host:hostport` format. Use this flag multiple times to forward multiple ports with one command.
 - Please visit the SSH man page for further flags and switches: https://linuxcommand.org/lc3_man_pages/ssh1.html

- C. The generic setup of bash aliases for SSH are `ssh -flags localport:ipaddress:remoteport`. In the example included above we've included the following ports for the according reasons:

```
124 alias FreyaTunnel="ssh -CnNv -L 5903:192.168.92.13:5901 -L 4131:192.168.92.13:4001 -L 4132:192.168.92.13:4002 lanforge@192.168.92.13"
```

- A. 4001 -- perl scripts use this for ascii connection to LANforge server
- B. 4002 -- GUI uses this for binary connection to LANforge server
- C. 5901 -- VNC port for display :1
- D. 4131 -- local port for perl scripts to connect to .92.12
- E. 4132 -- local port for GUI to connect to .92.12
- F. 5903 -- local port for VNC to connect to .92.12

- D. Further example ssh aliases include:

```
124 alias FreyaTunnel="ssh -CnNv -L 5903:192.168.92.13:5901 -L 4131:192.168.92.13:4001 -L 4132:192.168.92.13:4002 lanforge@192.168.92.13"
```

- A. `alias FreyaTunnel="ssh -CnNv -L 5903:192.168.0.6:5901 -L 4131:192.168.0.6:4001 -L 4132:192.168.0.6:4002 lanforge@192.168.0.6"`
- B. `alias SaltTunnel="ssh -CnN -L 4001:192.168.200.18:4001 salt@10.253.1.6"`

- E. After editing the `.bashrc` file, source the file in order for the changes to be applied. This will enable the code without the need to turn off and restart the device.

```
dipti@muffin:~$ . ~/.bashrc
dipti@muffin:~$
```

- F. In order for our machine to remember certain passwords and access configurations, some additional edits in the `~/.ssh/config` file or `$HOME/.ssh/config`.
- G. Next, this ssh connection will require an ssh key. This means that one needs to be generated. The private key and public key of the key pair must be saved to the local computer while only the public key of the pair must be saved to the remote server.
- H. Add your SSH key to the device being forwarded. Finally, add your public key that you generated earlier via SSH. This can be done by typing in `ssh-copy-id user@ipaddress` (see below example).

```
dipti@muffin:~$ ssh-copy-id lanforge@192.168.95.101
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompt
ed now it is to install the new keys
lanforge@192.168.95.101's password:

Number of key(s) added: 1

Now try logging into the machine, with:  "ssh 'lanforge@192.168.95.101'"
and check to make sure that only the key(s) you wanted were added.
```

- I. Once the alias is added to `.bashrc` file and the ssh key is added to the remote device, open any terminal and simply type in the alias name. This will initiate the tunnel. For example, "FreyaTunnel" in this example would be the alias typed into any terminal. This should incur an instance of your tunnel.

2. Windows SSH tunnel Setup

- A. There are many ways to set up an SSH tunnel, however, this cookbook will utilize PuTTY.



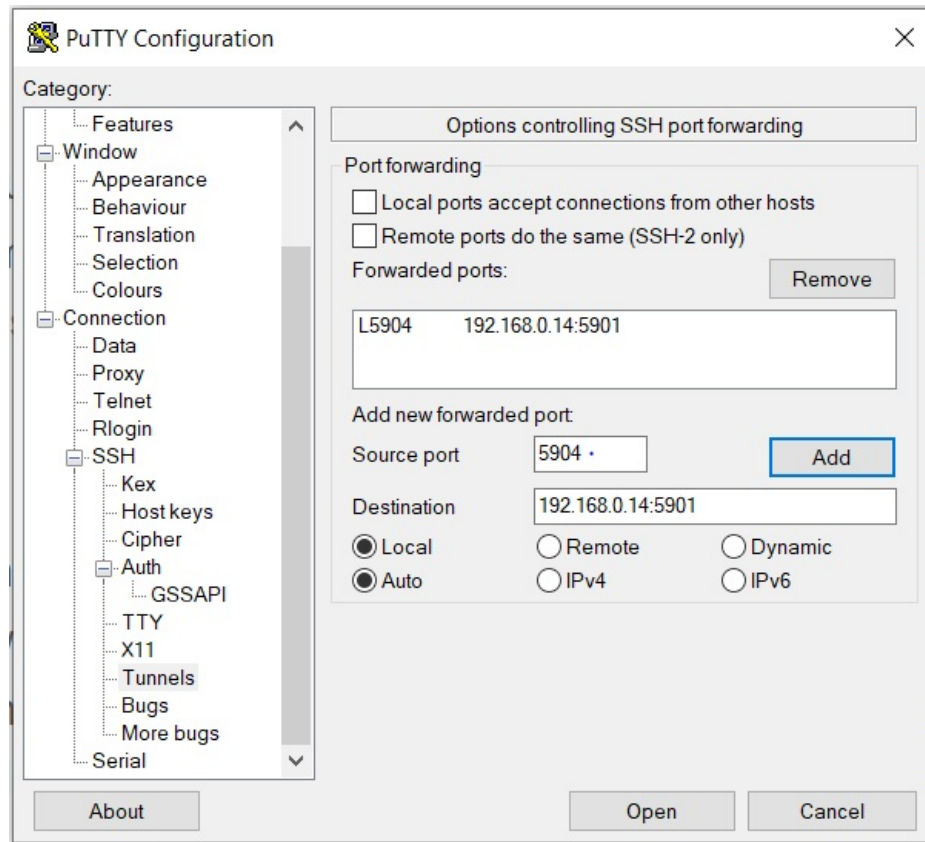
Download PuTTY

PuTTY is an SSH and telnet client, developed originally by Simon Tatham for the Windows platform. PuTTY is open source software that is available with source code and is developed and supported by a group of volunteers.

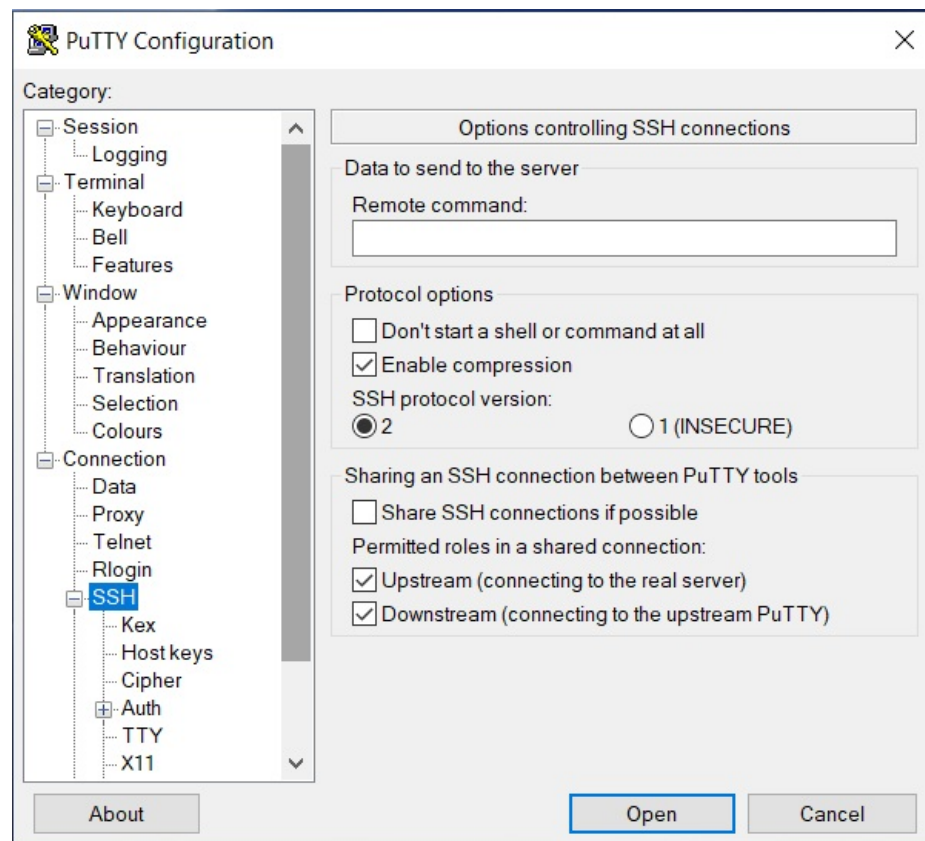
You can download PuTTY [here](#).

- B. Once PuTTY is downloaded, configure the SSH connection before adding the tunnel. For more information see [Connecting with PuTTY](#).

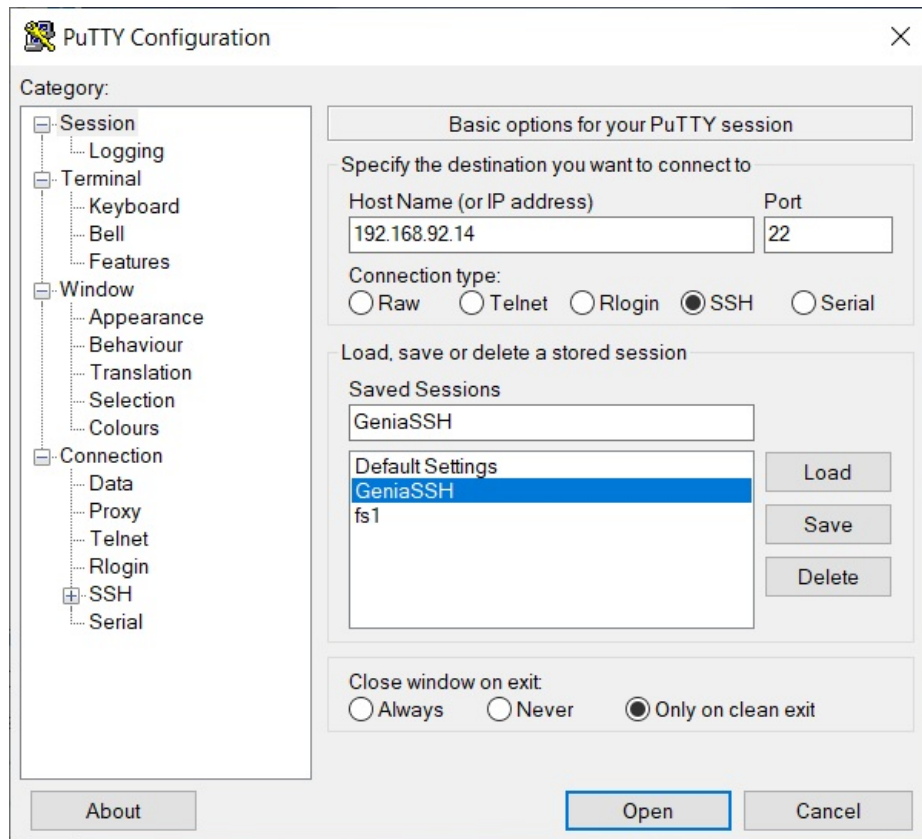
- C. Once your session is setup, select your session that was just saved from the last cookbook, then on the left-hand panel, select Connection -> SSH -> Tunnels.



- D. After setting up the tunnel, select SSH and enable compression. This will ensure that the tunnel uses data compression.



- E. Once all the settings desired are configured, select Session, highlight the session again in Saved Sessions and hit Save for the new session settings. This will make sure that the next time logged in will include all the settings here.



- F. Now, the session is saved and can be opened by clicking Open