SSH Client side configuration
SSH using hostnames instead of IP address
{Example from Linux Administration Cookbook}

Edit /etc/hosts on local system to resolve your remote system's name to its IP address
$ echo "192.168.33.11 centos2" | sudo tee -a /etc/hosts

Now your /etc/hosts file includes the resolved name

```
[vagrant@centos1 ~]$ cat /etc/hosts
127.0.0.1       centos1 centos1
127.0.0.1   localhost localhost.localdomain localhost4 localhost4.localdomain4
::1         localhost localhost.localdomain localhost6 localhost6.localdomain6
192.168.33.11 centos2
```

And you're able to ssh using its hostname instead of your remote's IP address
$ ssh centos2

```
[vagrant@centos1 ~]$ ssh centos2
The authenticity of host 'centos2 (192.168.33.11)' can't be established.
ECDSA key fingerprint is SHA256:Dsk0YbK2AADHW7zhGkS6AkhuEuF5kovOxsgcZVQFQfk.
ECDSA key fingerprint is MD5:2e:78:33:c1:09:67:5c:73:2c:cc:0a:cb:66:d3:a6:2d.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'centos2' (ECDSA) to the list of known hosts.
Last login: Wed Mar 25 22:42:47 2020 from 192.168.33.10
```

Tips & Tricks
To end an ssh session if and when it hangs
$ ~.

Read ssh config file while ssh'ing to include your typical arguments

- Config file is /etc/ssh/ssh_config

Create a file for your client
$ touch ~/.ssh/config
$ chmod 600 ~/.ssh/config

Edit ~/.ssh/config:

```
# every entry except CentOS2-V6 will use port 22 and ed25519 key
Host * !CentOS2-V6
  IdentityFile ~/.ssh/id_ed25519
  Port 22

Host CentOS2-V4
```

```
  Hostname 192.168.33.11
  User vagrant

Host CentOS2-V6
  Hostname fe80::a00:27ff:fe5c:7f1b%%eth1
  IdentityFile ~/.ssh/id_rsa
  Port 22
  User vagrant

Host CentOS2-Hostname
  Hostname centos2
  User vagrant
```