

Setup Remote Machine

Add user

```
1. # while as root user
   useradd -m -s /bin/bash -G sudo manas
   # -m is to create a home directory
   # -s to provide default shell
   # -G is to provide groups

   # To add password
   passwd manas
```

2. You can check user added and shell in file: `/etc/passwd`. To check group, type `groups manas` in terminal.
3. Login into new user created. You may want to change hostname (`username@hostname:cwd_path>`)
 1. Using `sudo hostnamectl set-hostname newhostname` OR
 2. Changing text in file `/etc/hostname`

Setup SSH Keys

```
1. ssh-keygen -f ~/.ssh/[key-name]
```

Leave passphrase blank to keep things simple.

This creates a public key (with extension `.pub` and a private key in the `~/.ssh` directory).

```
2. ssh-copy-id -i ~/.ssh/[key-name] [remote-user]@[remote-ip]
```

This transfers the `~/.ssh/[key-name].pub` to remote `.ssh` directory.

```
3. ssh [remote-user]@[remote-ip] -i ~/.ssh/[key-name]
```

4. To make the process of logging in convenient,
 1. Edit `~/.ssh/config` file.
 2. Add Host entry

```
Host [shortcut]
  Hostname [remote-ip]
```

```
User [remote-user]
IdentityFile /home/[local-user]/.ssh/[key-name]
```

3. Now you can login using `ssh [shortcut]`

5. To make remote system more secure, modify `/etc/ssh/sshd_config`

1. PermitRootLogin No
2. PublicKeyAuthentication yes
3. PasswordAuthentication no Restart ssh daemon: `sudo systemctl restart sshd`

Optional: If you have a domain registered, Add an A record to create an alias for the ip. After that you may login by `ssh [remote-user]@[domain]`

Nginx

Virtual Servers: A web server can host multiple websites (called virtual servers). In layman terms, it is sites inside sites-available directory.

Setup Nginx

1. `sudo apt install nginx`
2. `sudo systemctl status nginx` to check if nginx is running.

Configure bare-minimum website

1. `cd /etc/nginx/sites-available`

2.

```
#To remove the default website of nginx.
sudo rm default
```

3.

```
#It is a good practice to name the
#configuration file same as domain.
sudo touch [website-domain]
```

4. Add the bare minimum config details to the `[website-domain]` file:

```
server{
    listen 80;
    server_name [website-domain];

    location / {
        root [path-to-website-directory];
    }
}
```

```
}  
}
```

add another location to server static files: This in essence replaces /static url at start with the alias, ie [website-domain]/static/test.js will give file [website-domain]/[path-to-static-directory]/test.js

```
location /static/ {  
    alias [path-to-static-directory];  
}
```

To use nginx as proxy,

```
location / {  
    proxy_pass http://localhost:[port-for-actual-server];  
}
```

5. Create a symbolic link to the [website-domain] file, in sites-enabled directory.

```
sudo ln -s /etc/nginx/sites-available/[website-domain]  
/etc/nginx/sites-enabled/[website-domain]
```