SSSH

```
###SSH into a remote machine###
ssh user@mydomain.com
#or by ip address
ssh user@192.168.1.1
exit: exit ###Install Something###
#If it's a new server, update apt-
get first thing
sudo apt-get update
#then you can install something -
say Git
sudo apt-get install git
###Copy/Deploy files###
#copy all of the files in this
directory to the /home/will/newapp
directory
rsync -av .
will@domain.com:~/newapp
#delete a file and run rsync again,
and it only copies the one mising
file
###Generate an SSH keypair for
passwordless SSH###
#on your computer
cd ~/.ssh
#you might need to make the .ssh
directory
ssh-keygen -C "my@email.com"
#hit enter a few times to generate
key
#copy the file contents to the
clipboard
cat id_rsa.pub | pbcopy
#log into your machine
ssh user@mydomain.com
#make the .ssh directory and get in
it
mkdir .ssh
cd .ssh
#open authorized_keys in nano and
paste the contents in
nano authorized_keys
#paste contents in and save by
hitting ctrl+x
```

SSSH (cont)

```
#exit and you can now ssh without a
password!
```

Vagrant

```
###Let's install Vagrant###
- _Install VirtualBox:
https://www.virtualbox.org/_
- _Install Vagrant:
http://vagrantup.com/_
###Select a Vagrant Box from
https://vagrantcloud.com###
`
#add it to your list of boxes
vagrant box add
hashicorp/precise32
#create a new folder for your
project & init vagrant
vagrant init hashicorp/precise32
#run your new machine
vagrant up
#ssh into your new machine
vagrant ssh
`
_Now you are connected to a new
machine!_
- hit exit to disconnect
###Let's make this feel like the
real world by adding a fake domain
name###
- open the Vagrantfile and
uncomment the private_network line
& change to your desired IP
    - config.vm.network
"private_network", ip:
"22.22.22.22"
    - run vagrant reload
- make a fake domain for that ip
    - run sudo open /etc/hosts -a
"Sublime Text" to open your
/etc/hosts file in Sublime Text
```

Vagrant (cont)

```
    - add this line to the end of
the file and save 22.22.22.22
mytestsite.com
_Nice! Now going to mytestsite.com
will connect to our machine!_
###Let's install nginx so we can
see if our "domain" is working###
`
#always update apt-get on a new
machine
sudo apt-get update
#install nginx
sudo apt-get install nginx
#start-er up
sudo service nginx start
`
Now, go to mytestsite.com...you
should see
_Welcome to nginx!_
```