Lauren Fromm
404751250

Lab 7 Log

I first needed to connect my
beaglebone green wireless board.
When I first connected it, I tried
to download the necessary software,
but it wasn't originally working on
my mac. After trying again, I couldn't
find the beaglebone connected by my usb
cord.

Since I wasn't ever able to fix my
beaglebone, I am just going to write
what I would have done if it had worked.

First, I use
$sudo ssh root@192.168.7.2
to SSH into my beaglebone.

I then want to connect to the wifi,
so I use:
$sudo connmanctl
$connmanctl> enable wifi
$connmanctl> scan wifi
$connmanctl> services
Which lists all of the discovered access
points. Each access point has the following
format:
CSD-Guest wifi_506583d4fc5e_544e434150413937414239_managed_psk

Then, I register as the prospective client:
$connmanctl> agent on

After finding the access point I want to connect to,
I use:
$connmanctl> connect
wifi_506583d4fc5e_544e434150413937414239_managed_psk
I then enter the wifi password after it asks for Passphrase?

Now, I'm connected to the wifi, so I exit the connmanctl
$connmanctl> quit

By running:
$ifconfig
I find the IP address of my beaglebone, which will
allow my teammate to SSH into my beaglebone.
My IP address is:

128.97.244.18

I then update the database:
$sudo apt-get update

I install xauth tool for X11 forwarding:
$sudo apt-get install xauth

I then install FireFox so I can do the X11
forwarding:
$apt-get install firefox-esr-l10n-en-gb

Finally, I want to start the actual lab now that
the set up is done.

For the server:
I generate public and private keys by using:

$ssh-keygen

And I use no passwords for these keys.
I then create an account for a client
on the server using:

$ sudo useradd –d /home/jeanne –m jeanne
$ sudo passwd jeanne

Both the password and username are set to 'jeanne'

Next I create the ssh directory for the new user:
$ cd /home/jeanne
$ sudo mkdir .ssh

I want to change the ownership and permission
on the .ssh directory:
$ sudo chown –R jeanne .ssh
$ sudo chmod 700 .ssh

I then disable password based authentication:

$ emacs /etc/ssh/sshd_config

And then change password authentication to 'no'

For the client:

I again use:
$ ssh-keygen
to generate public and private keys.

I copy my public key to the server
for key-based authentication.

```
$ ssh-copy-id -i jeanne@128.97.244.18
```

I add my private key to the authentication
agent (ssh-agent)

```
$ ssh-add
```

Then I SSH to server:

```
$ ssh jeanne@128.97.244.18
$ ssh -X jeanne@128.97.244.18
```

I then run

```
$ firefox
```

on the remote host to see if I successfully
SSH'd onto the server, which I did.