Lab 1: Linux and Python

Overview

This lab will test your understanding of both linux command line and python.

You should learn or gain experience with:

- Learn how to use the command line for when we work on the Roomba
- You will be exposed (a little) to end-to-end encryption and data transfer
- Basic testing of your understanding of Python and some of the data types and flow controls we will use on the robots
- We will also gain some familiarity with jupyter notebooks

For this coding, let the EE's do most of the work. CompE's can help out, but they already have more experience with python than EE's do. CompE's should only take over if time is running out in the lab

Authorized Resources

You may only get help within your group or from the instructor. Do not talk to other groups or other cadets about this lab.

Setup a Jupyter Notebook

You will use a Jupyter notebook for this lab. From the command line, start the `jupyter` program by typing:

```
jupyter notebook
```

A web browser should open. From the top right, create a new `Python 2` notebook called *Lab1*. Set up your notebook with the following:

```
%matplotlib inline
from __future__ import division
from __future__ import print_function
```

```
import numpy as np
from matplotlib import pyplot as plt
```

Using Markdown (see the references or Google for proper syntax) when you complete all of the Tasks, copy and paste your answers into the notebook. I expect your results to look like lesson 3, 4 & 5. If you don't know how, take a look at the markdown in those notebooks.

Note: this notebook is running on and saved to your laptop … *not* the roomba.

[5 pts] Prelab: Install

If you have not already, please install the required software on your computer following the software install instructions from the course website. **This could take some time!!!**

[15 pts] Task 1: SSH and Crypto

Login username/passwords:

Class Day	Username	Password
T5 class	t5	raspberry
T6 class	t6	raspberry

Note: I changed the login process late in the game. If you see something in the documentation saying login in as `pi`, ignore it. Use the `t5` or `t6` logins.

1. Using `gitbash`, open a terminal window and change to your home directory with: `cd ~`
2. Create a cryptographic public/private key with: `ssh-keygen`. We are not going to do anything *smart* here, just hit enter always accepting the defaults no mater what. If we really cared about this key, we would put in a pass phrase (just like you have with your CAC … your PIN). This will create a private key (`~/.ssh/id_rsa`) and a public key (`~/.ssh/id_rsa.pub`).
3. Make sure to connect your laptop's wifi to the roomba's SSID first. Note, you may have to disconnect from .EDU to do this.
4. Next, open a browser and goto: 10.10.10.1:8080. A webpage should pop-up with your robot's name on it and some other stuff. If you see this, then you are properly connected and all is good.
5. Next share your key with your roomba: `ssh-copy-id p=<username>@10.10.10.1`. It will ask for a password: *raspberry*. It will also say something about adding roomba's IP address to a list of known hosts … that is fine.
6. Copy/paste the ssh key ascii art finger print into your notebook. If you ever want to reproduce this *amazing* art form again, just do:

```
[kevin@Tardis ~]$ ssh-keygen -lvf ~/.ssh/id_rsa.pub
  2048 b1:58:41:c5:93:b3:bc:c7:34:5b:e8:be:bc:15:ff:55  kevin@tardis.local (RSA)
  +--[ RSA 2048]----+
  |       .oo..     |
  |         .=      |
  |        o. + .   |
  |       o oo + .  |
  |      . S  = +. E|
  |          . =  o.|
  |           o  . o|
|           ...  o|
|            +o  .|
+-----------------+
```

Note: This computer you are using can now login to the roomba. If anyone else on the team wants/needs to, then just repeat the above steps. It is probably a good idea everyone can login with their own notebook just in case.

[50 pts] Task 2: Linux Command Line

1. Login to your assigned roomba with the following command: `ssh <username>@10.10.10.1`. Since you created a key in the previous task and shared it, you *should not* be asked for a username or password. If you are, notify your instructor.
2. Once you have logged in, copy/paste the welcome you see into your notebook
3. Checkout what is running on the machine by typing `htop` and pressing `q` for quit when done. Take a screen grab and save it as a jpeg or png. Then add the picture to your notebook.
4. Next, let's see what is attached to the i2c bus. Issue the command and copy/paste the results into your notebook: `sudo i2cdetect -y 1`. There should be a couple devices attached, those hex numbers are the i2c addresses for the IMU sensors: accelerometer, gyros, and magnetometer. We will talk more about them later. You should see *something* like this:

```
mday@create ~ $ sudo i2cdetect -y 1
     0  1  2  3  4  5  6  7  8  9  a  b  c  d  e  f
00:          -- -- -- -- -- -- -- -- -- -- -- -- --
10: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- 1f
20: -- 21 -- -- -- -- -- -- -- -- -- -- -- -- -- --
30: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
40: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
50: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
60: -- -- -- -- -- -- -- -- -- -- -- -- -- -- -- --
70: -- -- -- -- -- -- -- --
```

5. Lets see how much disk space we have. Use the command `df -h`, copy/paste the results into your notebook. The `/dev/root` entry with the mount point `/` is your main drive. It should have 10-11 GB of space available.
6. Now do the following folder/file manipulations. Show your instructor when you are done. Feel free to look at the command line reference material for this block to help you.
    1. If not already there, change to your home directory with `cd ~` or just `cd`.
        1. *Note:* On most Linux/Unix systems, `cd` is aliased to the command `cd ~`
    2. Create a folder called `ece387`
    3. Inside that folder, create a text file called `team.txt` using the `pico` program: `pico team.txt`. Enter the name of your team there, then close and save it using `Ctrl-X`.

[20 pts] Task 3: Python Math

1. In a new cell, calculate the results of the following 2 matrices. Remember to use `numpy` and print out the results of `C` and `D`

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \tag{1}$$

$$B = \begin{bmatrix} 4 & 3 \\ 2 & 1 \end{bmatrix} \tag{2}$$

$$C = A * B \tag{3}$$

$$D = C + B \tag{4}$$

2. Next let's plot some stuff. In a new cell, write a program that plots the following 2 lines in plots side-by-side when x is from 0 to 10.

$$y = 2x + 6 \tag{5}$$

$$z = \frac{1}{2}x - 4 \tag{6}$$

3. In a new notebook cell, write a function that takes in a number, sums everything up to that number and returns the result. Finally, within the same cell, call that function with the number 10 and print its result.

$$return = \sum_{i=0}^{n} i$$

[10 pts] Task 4: Samba

Follow the setup guide on the website, "Interfacing Windoze and Linux file systems", and login to the roomba. Put a screen shot of your home directory (`/home/t5 or t6`) in this notebook. From Windoze finder, you should be able to read/write files. This will be the easiest way to edit programs on your robot.

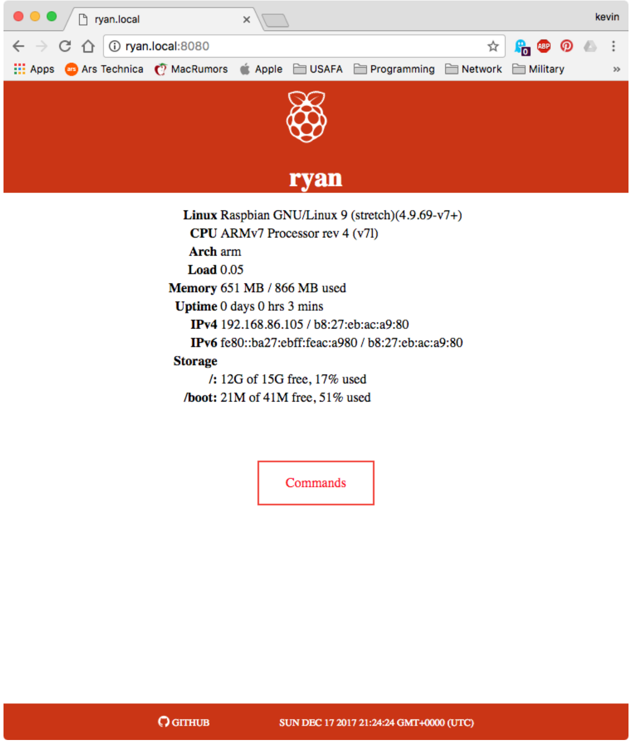
Turn-in

At the end of lab, print your `jupyter notebook` out and turn it in.

If this is the last class of the day

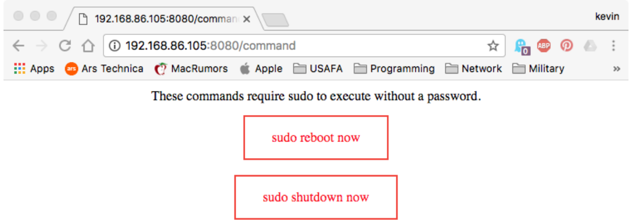
DO NOT JUST SHUT OFF POWER TO THE RPI

We need to shut everything down nicely. Our Roomba's have archeyjs installed on them. Open a web browser to `10.10.10.1:8080` and you should see this:



Linux Raspbian GNU/Linux 9 (stretch)(4.9.69-v7+)
CPU ARMv7 Processor rev 4 (v7l)
Arch arm
Load 0.05
Memory 651 MB / 866 MB used
Uptime 0 days 0 hrs 3 mins
IPv4 192.168.86.105 / b8:27:eb:ac:a9:80
IPv6 fe80::ba27:ebff:feac:a980 / b8:27:eb:ac:a9:80
Storage
/: 12G of 15G free, 17% used
/boot: 21M of 41M free, 51% used

1) Click on the `Commands` button on the webpage.



These commands require sudo to execute without a password.

2) Then select `sudo shutdown now`. Wait 1 minute and then unplug the RPi.