Robotics Teaming Installation Tutorial

DFEC

C1C Matthew U. DeMaso, C1C Anthony J. Tolbert

Setting Up The NUC

1. The NUC comes with Ubuntu 20.04 and ROS already, so you will not need to worry about installing any of those packages.

2. Login:

   ```
   Username: DFEC
   Password: dfec3141
   ```

3. Open a new Terminal on the NUC `Ctrl` + `Alt` + `t`

4. Install the following libraries using *sudo apt install xxx*

   A. rospy
   B. time
   C. signal
   D. csv
   E. copy
   F. os
   G. datetime
   H. statistics
   I. numpy

Adding the NUC to the Network

```
There are two setups that have been used: An isolated local network and
an internet enabled network (Makerspace). The main difference is the
availability of a DHCP server, which is only available on the internet
connected networks. Instalation is similar for each with key
differences.
```

1. Connect to the network with the GUI (using \"Moku_Station_3_5Ghz\" as an example).

Isolated LAN ONLY

1. Open the Wi-Fi settings to configure the IP settings:

   A. Select the gear icon next to the network name
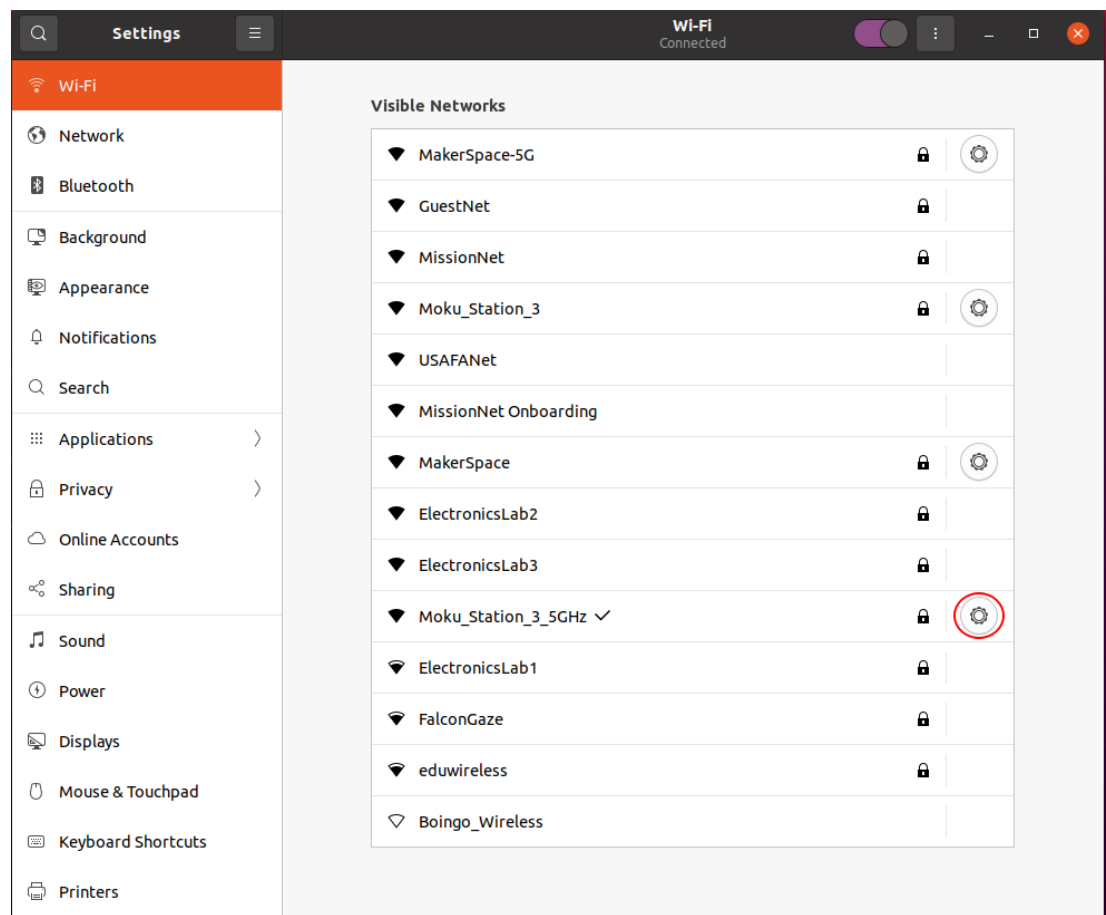
B. Select the IPv4 tab and change IPv4 Method to "Manual"

C. Use the IP address format for your router (likely 192.168.1.xxx) to give the NUC a unique IP.

D. As of now the IP addresses are 192.168.1.2XX and 192.168.1.1XX for the NUCs and robots respectivly where XX is replaced with either the usafabot number or NUC number

(i.e. robotics2 = 192.168.1.202 & usafabot22 = 192.168.1.122)

Cancel	Moku_Station_3_5GHz	Apply

Details   Identity   **IPv4**   IPv6   Security

IPv4 Method   ◯ Automatic (DHCP)   ◯ Link-Local Only

⦿ Manual   ◯ Disable

◯ Shared to other computers

Addresses

Address	Netmask	Gateway	
192.168.1.2XX	255.255.255.0	192.168.1.1	🗑
			🗑

DNS   Automatic ⬤

Separate IP addresses with commas

Routes   Automatic ⬤

Address	Netmask	Gateway	Metric	
				🗑

2. Add other nodes' IP addresses to hosts file. The file can be copied from any of the other NUCs or robots, but the NUC should be added to the file first and copied to all NUCs and robots.

   A. The file is located is *etc/hosts* and can be modified by nano with sudo permissions.
   B. The file can also be pulled from the capstone21-22 or copied via SCP.

```
                                    dfec@robotics2: ~

        dfec@robotics2: ~           ×        dfec@robotics2: ~/robotics_ws/src/ca...   ×

dfec@robotics2:~$ sudo nano /etc/hosts
[sudo] password for dfec: █
```

```
  GNU nano 4.8
127.0.0.1           localhost
127.0.1.1           robotics2

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters

192.168.1.201     robotics1
192.168.1.202     robotics2
192.168.1.203     robotics3
192.168.1.100     usafabot0
192.168.1.101     usafabot1
192.168.1.102     usafabot2
192.168.1.103     usafabot3
192.168.1.104     usafabot4
192.168.1.105     usafabot5
192.168.1.106     usafabot6
192.168.1.107     usafabot7
192.168.1.108     usafabot8
192.168.1.110     usafabot10
192.168.1.111     usafabot11
192.168.1.112     usafabot12
192.168.1.113     usafabot13
192.168.1.114     usafabot14
192.168.1.115     usafabot15
192.168.1.116     usafabot16
```

```
192.168.1.117    usafabot17
192.168.1.118    usafabot18
192.168.1.119    usafabot19
192.168.1.120    usafabot20
192.168.1.121    usafabot21
192.168.1.122    usafabot22
192.168.1.123    usafabot23
192.168.1.124    usafabot24
```

Setting Up the Robot

The robot should come loaded with an image of Ubuntu that has the neccessary source files loaded. If not, contact Mr. Steven Beyer for the .iso file to be flashed to the raspberry pi. Ensure that the hostname matches the lablel and is what is expected. Hostnames can be updated by `hostnamectl set-hostname new-hostname`. Once the robot is added to the same network as the NUCs, updates to files can be accomplished via SCP.

Adding the Robot to the Network

> Adding a robot to the network is similar to the NUCs and should be done on the robots locally (not via SSH). Changes take effect after a reboot.

1. Connect the robot's Pi to a monitor and keyboard then turn it on.

2. After Ubuntu launches, Login:

       Username: pi
       Password: dfec3141

3. `sudo nano /etc/netplan/50-cloud-init.yaml` Change the network settings and IP address to match the robot

```
  GNU nano 4.8
# This file is generated from information provided by the datasource.  Changes
# to it will not persist across an instance reboot.  To disable cloud-init's
# network configuration capabilities, write a file
# /etc/cloud/cloud.cfg.d/99-disable-network-config.cfg with the following:
# network: {config: disabled}
network:
    ethernets:
        eth0:                          Yes for internet/DHCP enabled networks .
            dhcp4: true                        and delete everything below.
            optional: true
    version: 2
    wifis:
        wlan0:
            optional: true
            access-points:
  SSID              "Moku_Station_3_5GHz":
                    password: "dfec3141"
            dhcp4: no
            addresses:                     Change for each robot
                - 192.168.1.124/24
            gateway4: 192.168.1.1
            nameservers:
                addresses: [8.8.8.8, 1.1.1.1]
```

4. Add other nodes' IP addresses to hosts file. The file can be copied from any of the other NUCs or robots, but the robot should be added to the file first and copied to all NUCs and robots.

   A. The file is located is /etc/hosts and can be modified by nano with sudo permissions
   B. The file can also be copied via SCP.

Downloading Files and Synching Device with GitHub

1. Ensure you have a working GitHub account.

2. Open a new Terminal on the NUC `Ctrl` + `Alt` + `t`

3. Generate a New SSH Key substituting your e-mail adress instead of **mudemaso@gmail.com**

```
robotics1@robotics1:~$ ssh-keygen -t ed25519 -C "mudemaso@gmail.com"
```

4. You will be prompted in which file to save the key. Press `Enter` when your terminal output is identical to the below picture.

```
Generating public/private ed25519 key pair.
Enter file in which to save the key (/home/robotics1/.ssh/id_ed25519):
```

5. If a key already exists at the location, press `y`.

```
/home/robotics1/.ssh/id_ed25519 already exists.
Overwrite (y/n)?
```

6. Enter a new, easy to remember passphrase when prompted (**dfec3141**).

```
Enter passphrase (empty for no passphrase):
```

7. Enter passphrase again when prompted. The final output should look like this.

```
Your identification has been saved in /home/robotics1/.ssh/id_ed25519
Your public key has been saved in /home/robotics1/.ssh/id_ed25519.pub
The key fingerprint is:
SHA256:bkJCNRQ3JgdR3CanRZVe+qTCU0J4FjauNpsqqH9b0u8 mudemaso@gmail.com
The key's randomart image is:
+--[ED25519 256]--+
|      .O**+=o..   |
|      . *++O.. .  |
|     .    X.. o   |
|    .    ... + .  |
|     . . S. o +   |
|      + o ++ . .  |
|     .. + =  o    |
|    . oo =        |
|.o...o..E         |
+----[SHA256]-----+
```

8. Run the following commands To ensure the SSH key is working properly and add in your new SSH ID. Enter the passcode when prompted. You should get a successfully added message when finished.

```
robotics1@robotics1:~$ eval "$(ssh-agent -s)"
Agent pid 96060
robotics1@robotics1:~$ ssh-add ~/.ssh/id_ed25519
Enter passphrase for /home/robotics1/.ssh/id_ed25519:
Identity added: /home/robotics1/.ssh/id_ed25519 (mudemaso@gmail.com)
```

9. Open the Public Key.

```
robotics1@robotics1:~$ nano ~/.ssh/id_ed25519.pub
```

10. Expand the Window to see the whole contents of the file (ends in your email). Copy The contents of the file. You must use right mouse, copy. `Ctrl` + `v` **DOES NOT WORK**

11. Open a web browser and sign into your GitHub Account.

12. In the top right-hand of the window, select the dropdown menu from your account picture and click on settings.
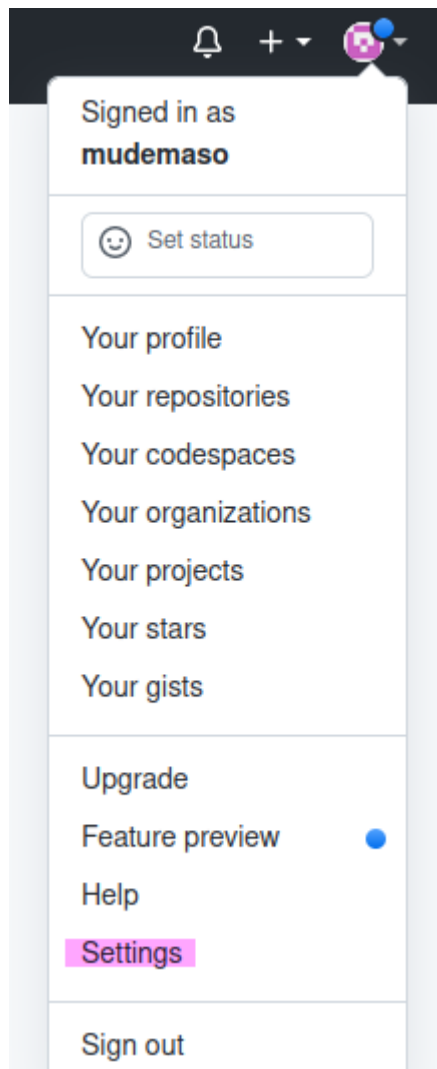
13. Open the **SSH and GPG Keys** section.

Account settings

Profile

Account

Appearance

Accessibility

Account security

Billing & plans

Security log

Security & analysis

Sponsorship log

Emails

Notifications

Scheduled reminders

SSH and GPG keys

Repositories

Packages

Pages

Organizations

Saved replies

Applications

14. Click new SSH Key.

SSH keys   New SSH key

15. In the "Title" Field, put a descriptive title.

16. In the "Key" Field, paste the copied key from Step 10 into this field. When Finished, press "Add SSH Key".
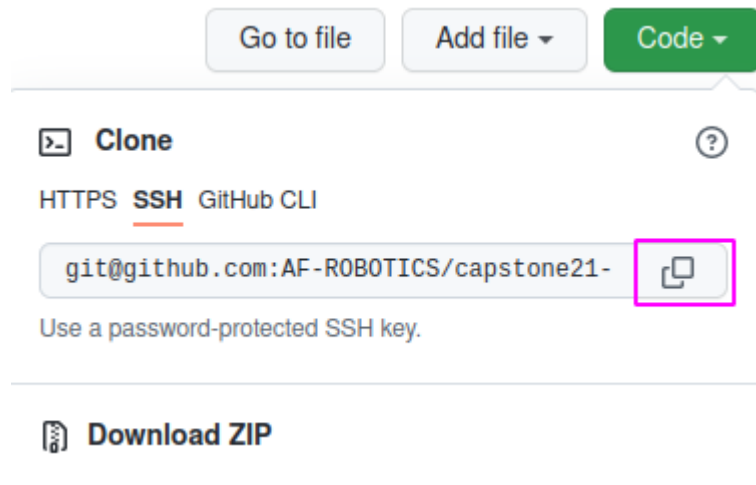
SSH keys / Add new

Title

```
roboticsTeamingCapstone
```

Key

```
ssh-ed25519 AAAAC3NzaC1lZDI1NTE5AAAAIIs8fR5Um3mSzoBq0NlAwc+vGkYNbLOLl3Ec2rG8eOL8
mudemaso@gmail.com
```

[Add SSH key]

17. Navigate to the Capstone Reposity and click "Code". Copy the repo adress using SSH mode.

18. In the File Browser Gui, navigate to *Home->robotics_ws->src*.

19. Ensure the "capstone21-22" folder is deleted.

20. Open a new Terminal on the NUC `Ctrl` + `Alt` + `t`

21. Navigate to the *robotics_ws->src* directory.

```
robotics1@robotics1:~$ cd robotics_ws/src/
```

22. Clone the repo using the username and passphrase you used in Step 6 when prompted. Paste the repo adress from Step 11.

```
robotics1@robotics1:~/robotics_ws/src$ git clone git@github.com:AF-ROBOTICS/capstone21-22.git
```

23. Upon Completion the Repo is copied and you are now able to use all typical git commands from the terminal with the Repo!

24. Attain usafabot directory and gazebo directory from Dr. Baek / Mr. Beyer for simulations and working on real robot. Put them in your *robotics_ws->src directory.*

Create your ROS workspace:

Source the ROS setup file:

```
source /opt/ros/noetic/setup.bash
mkdir -p ~/robotics_ws/src
cd ~/robotics_ws/
catkin_make
```

Setup ROS environment variables and setup scripts within the `~/.bashrc` file. Open the `~/.bashrc` file with your favorite command line editor and add the following to the bottom:

```
source /opt/ros/noetic/setup.bash
source ~/robotics_ws/devel/setup.bash
export ROS_PACKAGE_PATH=~/robotics_ws/src:/opt/ros/noetic/share
export ROS_HOSTNAME=`hostname` # note these are backticks, not apostrophes
export ROS_MASTER_URI=http://roboticsX:11311 # replace 'X' with label of
your NUC
```

Any time you make changes to your `~/.bashrc` file you must source it:

```
source ~/.bashrc
```

Dependencies

There are a number of ROS packages required to operate the USAFABot. To ensure ROS message compatibility you will want all of these libraries to be downloaded on your desktop as well. Some of these can be installed using `apt install` and others have to be installed from source. Change directories to your source folder and follow the below steps to install each dependency.

USAFABot

```
cd ~/robotics_ws/src
git clone git@github.com:AF-ROBOTICS/usafabot.git
cd ..
catkin_make
```

If you have compile errors, you may need to install the following packages

```
sudo apt install ros-noetic-amcl
sudo apt install ros-noetic-apriltag-ros
sudo apt install ros-noetic-slam-gmapping
sudo apt install ros-noetic-teleop-twist-keyboard
```

```
cd ~/robotics_ws/src
git clone git@github.com:Slamtec/rplidar_ros.git
cd ~/robotics_ws
catkin_make
```

```
cd ~/robotics_ws/src
git clone https://github.com/ros-drivers/um7.git
cd ~/robotics_ws
```

```
rosdep install --from-paths src --ignore-src -r -y
catkin_make
```

Serial

```
cd ~/robotics_ws/src
git clone https://github.com/wjwwood/serial
```

This will take a while.

Now we can make and source our workspace:

```
cd ~/robotics_ws
catkin_make
source ~/.bashrc
```

Installing Jupyter-Notebook For Documentation (Personal Computer / Cadet Laptop)

1. Download Py-Charm edu from their Educational Website

2. Download Windows Terminal from the Microsoft Store

3. Open Windows Terminal on your computer.

4. Check what version of python you have installed. It should be **3.8**. exit() python when finished.

   ```
   PS C:\Users\C22Matthew.DeMaso> python
   Python 3.8.2 (tags/v3.8.2:7b3ab59, Feb 25 2020, 23:03:10) [MSC v.1916 64 bit (AMD64)] on win32
   Type "help", "copyright", "credits" or "license" for more information.
   ```

5. Install jupyter-notebook.

   ```
   PS C:\Users\C22Matthew.DeMaso> pip install notebook
   ```

6. When installing, you will see yellow text with an error saying a path is not current. i.e. C:\Users\CadetName...

7. Copy the path that was not active and in your computer settings go *System->About->Advanced System Settings*.

8. Go to Environment Variables.

System Properties

Computer Name | Hardware | **Advanced** | System Protection | Remote

You must be logged on as an Administrator to make most of these changes.

Performance

Visual effects, processor scheduling, memory usage, and virtual memory

[Settings...]

User Profiles

Desktop settings related to your sign-in

[Settings...]

Startup and Recovery

System startup, system failure, and debugging information
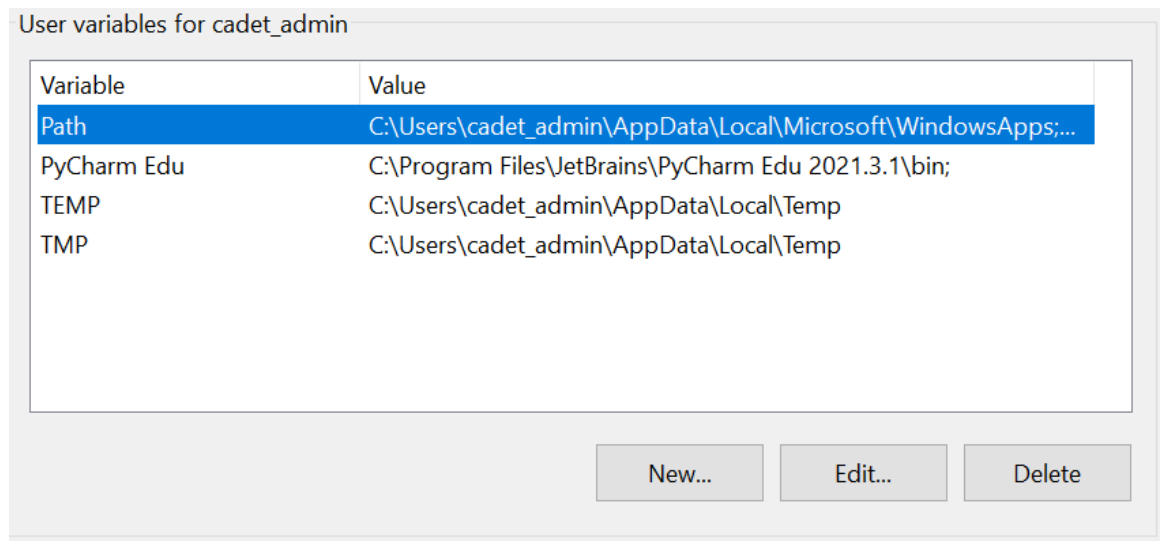
[Settings...]

[Environment Variables...]

[OK] [Cancel] [Apply]

9. Select the *Path* variable under variables for cadet_admin and click *Edit*.

User variables for cadet_admin

Variable	Value
Path	C:\Users\cadet_admin\AppData\Local\Microsoft\WindowsApps;...
PyCharm Edu	C:\Program Files\JetBrains\PyCharm Edu 2021.3.1\bin;
TEMP	C:\Users\cadet_admin\AppData\Local\Temp
TMP	C:\Users\cadet_admin\AppData\Local\Temp

New...   Edit...   Delete

10. Select *New* and *paste* path copied from *step 6*.

    <img src"" alt="path2.png is not in the github">

11. Repeat steps 9-10 for *path* within the *system variables*.

12. Click *Okay* and exit out of all settings.

13. Close Windows Terminal.

14. Clone the Capstone Repository to the machine.

15. After files have synched, navigate to the *Documentation* folder in your local file browser.

16. Right Click in the folder and click *Open in Windows Terminal*

17. In windows termial, type *jupyter-notebook*

PS C:\Users\C22Matthew.DeMaso\afacademy.af.edu\Capstone – Robotics Teaming – General\Capstone_Documentation> jupyter-notebook

18. Jupyter-Notebook will open in your prefered browser. You can now access all documentation notebooks and edit with jupyter-notebook.

19. For a tutorial on how to use it, look at this helpful website.

In [ ]: