

This documentation talks about how to setup a Raspberry PI Zero W, do the SSH, setup the Wi-Fi, write code in your local system and then scp the files to RPI Zero using SCP. This setup will work on both MacOS and Linux.

1. For Windows, you can use tool like GUI tool like Putty.
2. For MacOS and Linux, you can use tool ssh and scp (relatively straight forward).
3. Last way is to set-up a git repo (tracking on the RPI) but make sure never to commit your certificate, ARN and any other important credentials.

SSH and SCP tools

1. Connect one end of the cable (micro-USB) to RPI micro USB port and other side to laptop.



2. Use following command to enter the RPI using USB Cable.

```
>> ssh pi@raspberrypi.local
```

```
(awsiot) Shailendras-MacBook-Air:AWSIoT shailendra0408$ sudo ssh pi@raspberrypi.local
Password:
Warning: Permanently added the ECDSA host key for IP address 'fe80::5140:732b:56c1:8147%en3' to the list of known hosts.
pi@raspberrypi.local's password:
Linux raspberrypi 5.10.36+ #1418 Thu May 13 18:12:25 BST 2021 armv6l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Sat Nov  4 14:43:24 2023

SSH is enabled and the default password for the 'pi' user has not been changed.
This is a security risk - please login as the 'pi' user and type 'passwd' to set a new password.

pi@raspberrypi:~$
```

You can use raspberry as default password. Make sure you change the password post first login.

- Once you enter inside the RPI, create one folder and give it any name of your choice. Use `mkdir <folder name>` command to create the folder. Suppose I made one director awsiotbangalore.
- Enter inside the folder using command `cd <folder name>`
- Once inside the directory, use `pwd` to know the path of the directory.

```
pi@raspberrypi:~ $ cd awsiotbangalore/
pi@raspberrypi:~/awsiotbangalore $ pwd
/home/pi/awsiotbangalore
pi@raspberrypi:~/awsiotbangalore $
```

- Now return to your original OS. Probably move inside the same directory where you have saved the password you downloaded from the AWS IOT core while creating the thing.
- Now use scp command to copy the certificates to RPI directory which you created in step 3.
- Use following command to copy the certificates.

>> `sudo scp private.pem.key pi@raspberrypi.local:/home/pi/awsiotbangalore`

- Here private.pem.key is the file in your local directory which you want to copy to the RPI.
 - /home/pi/awsiotbangalore is the directory in RPI where you want to transfer the private.pem.key file (from step 5).
 - Scp is the command that you use to transfer the file from one source to destination.
- Once you have transferred the certificates, transfer the code files also to the RPI. Make sure that in the code, you have added the right certificate file name and end point for AWS IOT Core.

```
pi@raspberrypi:~/awsiotbangalore $ pwd
/home/pi/awsiotbangalore
pi@raspberrypi:~/awsiotbangalore $ ls
Amazon-root-CA-1.pem device.pem.crt private.pem.key __pycache__ rangesensor.py RPIDistanceSensor.py
pi@raspberrypi:~/awsiotbangalore $
```

- Once you have trasnfered all the file, check if the RPI is connected to internet using `ping www.yuktix.com`. If you get a successful ping that means you are connected. In case if you are not, then run `ifconfig` command to see the available network options.

```
pi@raspberrypi:~/awsiotbangalore $ ifconfig
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

usb0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.100 netmask 255.255.0.0 broadcast 192.168.1.255
    inet6 fe80::208:1fff:fe00:0000 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:00:00:00 txqueuelen 1000 (Ethernet)
    RX packets 376 bytes 69545 (67.9 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 223 bytes 38282 (37.3 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

wlan0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.1.101 netmask 255.255.255.0 broadcast 192.168.1.255
    inet6 fe80::208:1fff:fe00:0000 prefixlen 64 scopeid 0x20<link>
    ether 08:00:27:00:00:00 txqueuelen 1000 (Ethernet)
    RX packets 236 bytes 21176 (20.6 KiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 191 bytes 19847 (19.3 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

11. If you get wlan0 in the output and no IP assigned, that's mean, you have wifi but its not connected.
12. Use following set of command to connect to wifi

- a. Open the wpa_supplicant.conf file using nano.
>> **sudo nano /etc/wpa_supplicant/wpa_supplicant.conf.**
- b. Now change the following content. Use Cntrl + x to save the file.
>> **network={ ssid="The SSID of your network (eg. Network name)"
psk="Your Wifi Password" }**
- c. Run the following command.
>> **sudo ifdown wlan0.**
- d. Now below command
>> **sudo ifup wlan0.**
- e. Run below command if you are connected to internet now.
>> **ifconfig.**

13. Now run the RPIDistancesensor.py file using python3 and as a result you should be able to see following log.

```
pi@raspberrypi:~/awsiotbangalore $ python3 Temp1.py
Using RPI ZERO and HC SR04 for distace measurement...in porogres
Waiting for senosr to settle down
0.004825115203857422
Distance: 82.75 cm
35.2
{"message":"OK","traceId":"25b0ec74-1523-00b7-90e2-d4d93e69e969"}
Using RPI ZERO and HC SR04 for distace measurement...in porogres
Waiting for senosr to settle down
0.004817008972167969
Distance: 82.61 cm
34.7
{"message":"OK","traceId":"fed4991f-1ebf-d314-19f0-2d8d3b357dbb"}
Using RPI ZERO and HC SR04 for distace measurement...in porogres
Waiting for senosr to settle down
0.004818916320800781
Distance: 82.64 cm
35.2
{"message":"OK","traceId":"385a2447-c290-5638-79b9-5737a0ce67c3"}
Using RPI ZERO and HC SR04 for distace measurement...in porogres
Waiting for senosr to settle down
```