



**Institute for the Wireless
Internet of Things**

at Northeastern University

Wi-Fi on Colosseum

<https://github.com/colosseum-wiot/colosseum-school-2024/tree/main/wifi-assignment>

Stefano Maxenti

Institute for the Wireless Internet of Things

Northeastern University, Boston, MA

maxenti.s@northeastern.edu

Our goal

- Ad hoc Wi-Fi on Software Defined Radios in Colosseum
- Do experiments by generating traffic over the emulated channel



Step 1

- Creating a reservation on Colosseum



Number of SRNs: 67 max available

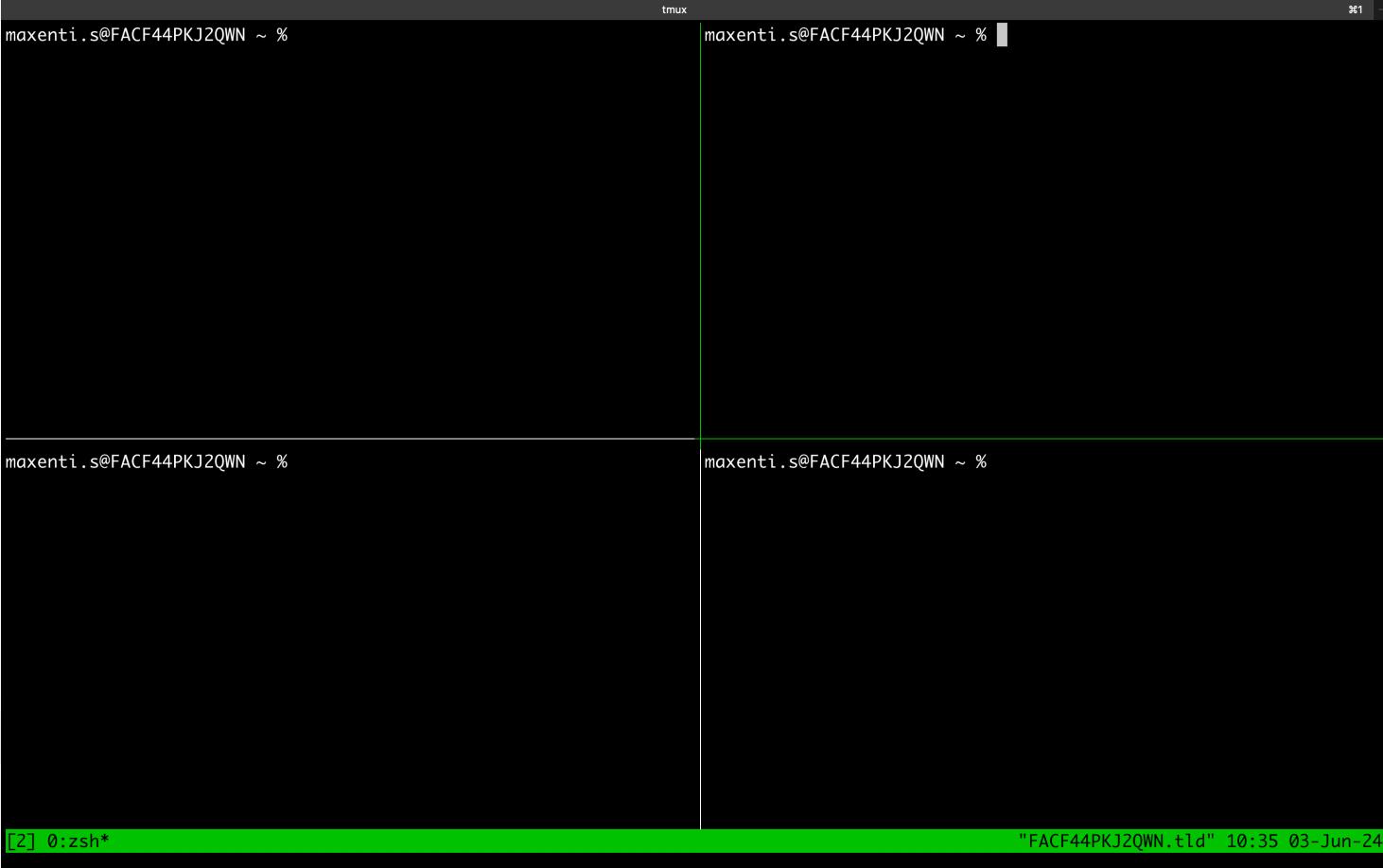
Default image:

Node 1:

Node 2:

Step 2

- Accessing the containers using SSH in 4 panes



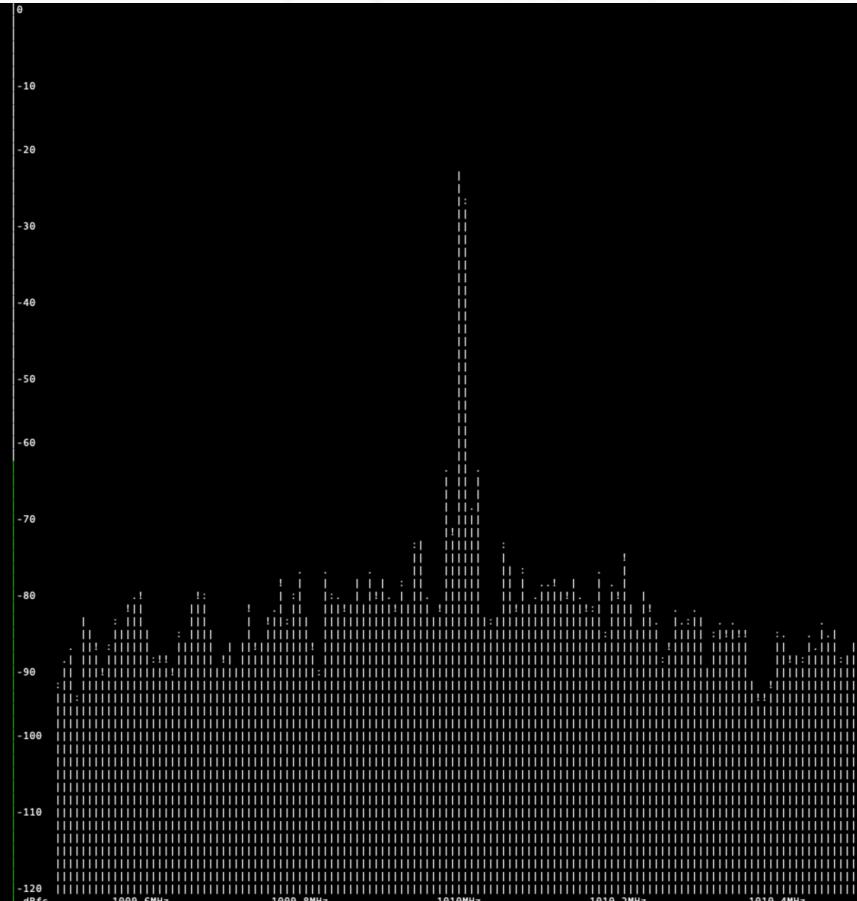
```
maxenti.s@FACF44PKJ2QWN ~ %
maxenti.s@FACF44PKJ2QWN ~ %
maxenti.s@FACF44PKJ2QWN ~ %
maxenti.s@FACF44PKJ2QWN ~ %

[2] 0:zsh* "FACF44PKJ2QWN.tld" 10:35 03-Jun-24
```

Step 3

- Starting an RF scenario and testing it

```
+-----+  
| Field | Value |  
+-----+  
| scenario_id | 1009 |  
| scenario_status | RUNNING |  
| scenario_start_time | 16:29:06 |  
| scenario_cycle | True |  
| Number of Nodes | 10 |  
+-----+  
  
root@neu-webinar-interactive-v1-srn1:~# cd /root/utils  
root@neu-webinar-interactive-v1-srn1:~/utils# ls  
uhd_rx_fft.sh uhd_tx_tone.sh  
root@neu-webinar-interactive-v1-srn1:~/utils# ./uhd_tx_tone.sh  
linux: GNU C++ version 5.4.0 20160609; Boost_105808; UHD_003.009.005-0-g32951af2  
  
Creating the usrp device with: ...  
-- X300 initialization sequence...  
-- Determining maximum frame size... 8000 bytes.  
-- Setup basic communication...  
-- Loading values from EEPROM...  
-- Setup RF frontend clocking...  
-- Radio 1x clock:200  
-- Detecting internal GPSDO... No GPSDO found  
-- Initialize Radiotap control...  
-- Performing register loopback test... pass  
-- Initialize Radiotap control...  
-- Performing register loopback test... pass  
Using Device: Single USRP:  
Device: X-Series Device  
Mboard 0: X310  
RX Channel: 0  
RX DSP: 0  
RX Oboard: A  
RX Subdev: UBX RX  
RX Config: 1  
RX DSP: 1  
RX Oboard: B  
RX Subdev: UBX RX  
TX Channel: 0  
TX DSP: 0  
TX Oboard: A  
TX Subdev: UBX TX  
TX Channel: 1  
TX DSP: 1  
TX Oboard: B  
TX Subdev: UBX TX  
  
Setting TX Rate: 0.100000 Msps...  
  
UHD Warning:  
The hardware does not support the requested TX sample rate:  
Target sample rate: 0.100000 Msps  
Actual sample rate: 0.390625 Msps  
  
UHD Warning:  
The hardware does not support the requested TX sample rate:  
Target sample rate: 0.100000 Msps  
Actual sample rate: 0.390625 Msps  
Actual TX Rate: 0.390625 Msps...  
  
Setting TX Freq: 1010.000000 MHz...  
Actual TX Freq: 1010.000000 MHz...  
  
Setting TX Gain: 20.000000 dB...  
Actual TX Gain: 20.000000 dB...  
  
Setting device timestamp to 0...  
Checking TX: TX0: locked ...  
Press Ctrl + C to stop streaming...
```



RECEIVER

SENDER

Step 4

- Setting up network connectivity
- Starting the Wi-Fi modem
- Testing the generated network with different tools
 - ping
 - iPerf3
 - TGEN

Let's start!

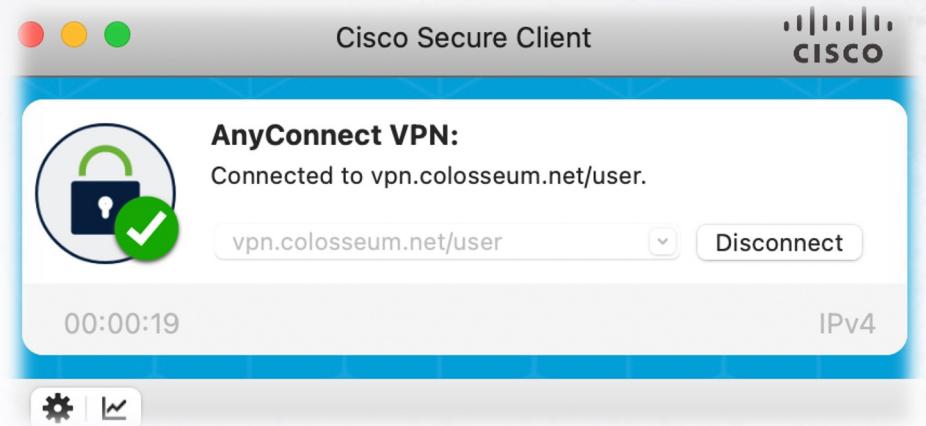
- <https://github.com/colosseum-wiot/colosseum-school-2024/tree/main/wifi-assignment>



Let's start!

Getting ready

- Connecting to the VPN



- Starting a reservation with **2** nodes
using *webinar-interactive-v1*

Number of SRNs:	<input type="text" value="2"/> 67 max available
Default image:	<input type="text" value="webinar-interactive"/> <input type="button" value="Reset all"/>
Node 1:	<input type="text" value="webinar-interactive-v1"/>
Node 2:	<input type="text" value="webinar-interactive-v1"/>

Getting ready

- Connect to the two SRNs when the status becomes “allocated” (root/sunflower)

ID: 145327

Status: Future

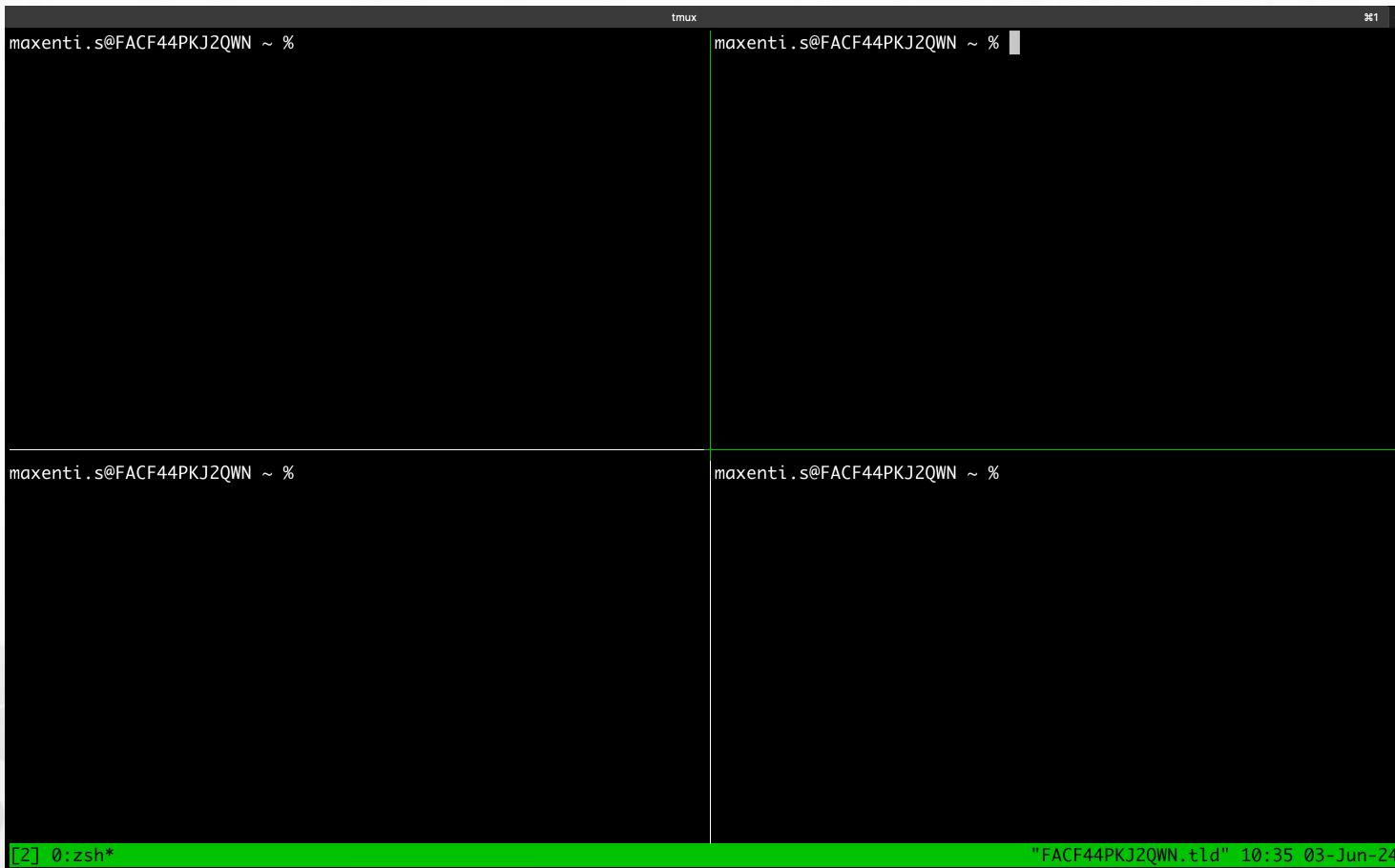
Nodes: 2

Time: 2024/06/03 10:39:00 AM - 2024/06/03 11:39:00 AM (60 minutes)

Node	Image	Hostname	Port	Status
SRN -98	webinar-interactive-v1	wineslab-098	-	-
SRN -99	webinar-interactive-v1	wineslab-099	-	-

Getting ready

- We will need 4 terminals, use the layout that you prefer, I will be using tmux with 4 panes



Testing if Colosseum is working

- In any of the SRNs, run:
 - `colosseumcli rf start 1009 -c`
 - Check if the scenario is active with `colosseumcli rf info`
- In one terminal, run: `/root/utils/uhd_tx_tone.sh`
- In the other, run: `/root/utils/uhd_rx_fft.sh`
- In `uhd_tx_tone.sh`, try to change the center frequency and see what happens

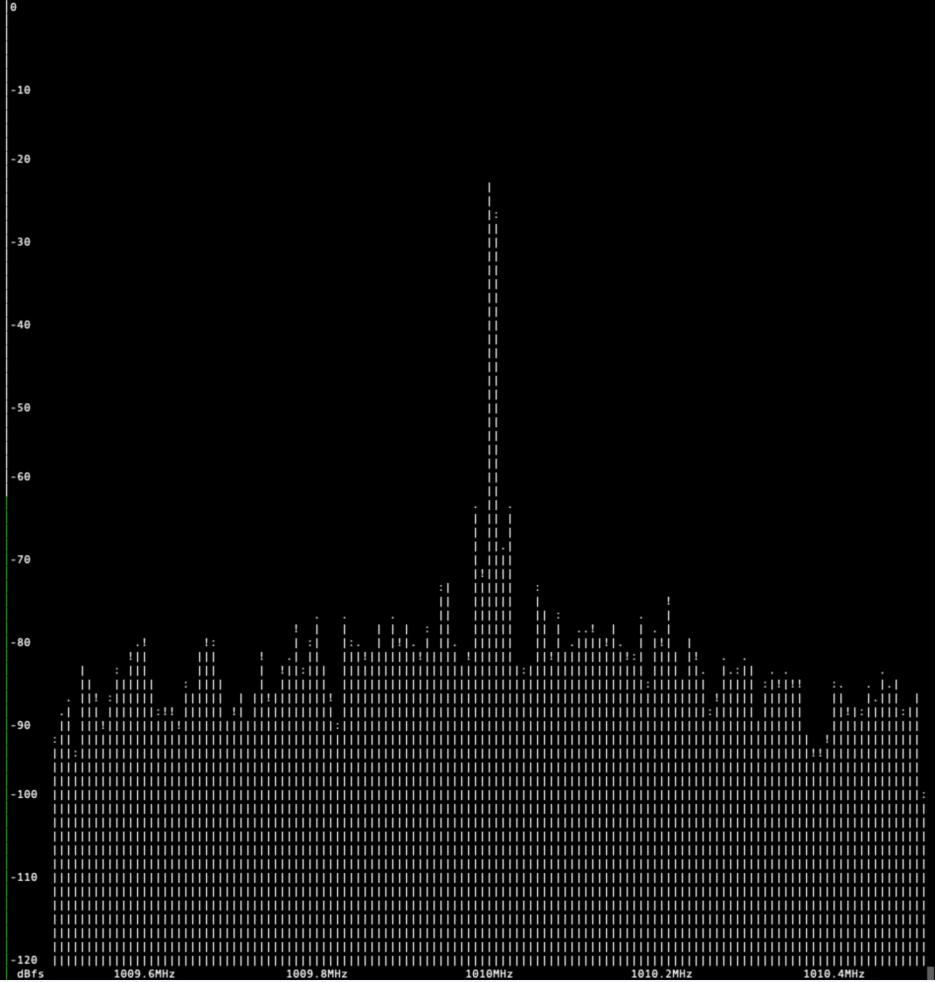
What you should see

```
| Field | Value |
| scenario_id | 1009 |
| scenario_status | RUNNING |
| scenario_start_time | 16:29:06 |
| scenario_cycle | True |
| Number of Nodes | 10 |

root@neu-webinar-interactive-v1-srn1:~# cd /root/utils
root@neu-webinar-interactive-v1-srn1:~/utils# ls
uhd_rx_fft.sh uhd_tx_tone.sh
root@neu-webinar-interactive-v1-srn1:~/utils# ./uhd_tx_tone.sh
linux: GNU C++ version 5.4.0 20160609: Boost_105800; UHD_003.009.005-0-g32951af2

Creating the usrp device: ...
-- X300 initialization sequence...
-- Determining maximum frame size... 8000 bytes.
-- Setup basic communication...
-- Loading values from EEPROM...
-- Setup RF frontend clocking...
-- Radio Ix clock:200
-- Detecting internal GPSDO... No GPSDO found
-- Initialize Radio0 control...
-- Performing register loopback test... pass
-- Initialize Radio1 control...
-- Performing register loopback test... pass
Using Device: Single USRP:
Device: X-Series Device
Mboard: 0 - X310
RX Channel: 0
    RX DSP: 0
    RX Board: A
    RX Subdev: UBX RX
RX Channel: 1
    RX DSP: 1
    RX Board: B
    RX Subdev: UBX RX
TX Channel: 0
    TX DSP: 0
    TX Board: A
    TX Subdev: UBX TX
TX Channel: 1
    TX DSP: 1
    TX Board: B
    TX Subdev: UBX TX
Setting TX Rate: 0.100000 Msps...
UHD Warning:
The hardware does not support the requested TX sample rate:
Target sample rate: 0.100000 Msps
Actual sample rate: 0.390625 Msps
UHD Warning:
The hardware does not support the requested TX sample rate:
Target sample rate: 0.100000 Msps
Actual sample rate: 0.390625 Msps
Actual TX Rate: 0.390625 Msps...
Setting TX Freq: 1010.000000 MHz...
Actual TX Freq: 1010.000000 MHz...
Setting TX Gain: 20.000000 dB...
Actual TX Gain: 20.000000 dB...
Setting device timestamp to 0...
Checking TX: TXLO: locked ...
Press Ctrl + C to stop streaming...
```

SENDER



RECEIVER

Getting ready: almost there

- Setting up routes to connect the two SRNs
 - `/root/interactive_scripts/tap_setup.sh`
 - In each SRNs, run:
 - `/root/interactive_scripts/route_setup.sh <ID of the other SRN>`
 - This sets up routes at L3 between the two SRNs and allows traffic routing
 - On the TAP interface, each SRN will have IP like this: 192.168.**\$(ID+100)**.1.

Finally, let's transmit!

- On each SRN, run
 - */root/interactive_scripts/modem_start.sh*
- This starts the transmission between the two nodes
- You should see some small traffic flowing between the nodes
- Connectivity can be checked with the **ping** command, or with **iperf3**

Let's create a lot of traffic

- We will be using the other two terminals you created
- We use TGEN
 - `colosseumcli tg start 10090`
 - `colosseumcli tg info` to check if it works
 - It takes around ~5min to start
- In the meanwhile, start on each node a traffic sniffer:
 - `tcpdump -i tr0`

What you should see



Institute for the Wireless Internet of Things

at Northeastern

Closing the experiment and ending the reservation

- `ctrl-c` on all terminals
- `colosseumcli tg stop #` to close the traffic generator
- `colosseumcli rf stop #` to close the channel emulator
- exit on all terminals
- **DONE!** You have transmitted over Wi-Fi on Colosseum

Thank you, any final question?