Informative Wireless Raspberry Pi Router Design Document

By Jonathan Saenz

CSCE 462 - 502

April 30, 2020

Table of Contents

Abstract

With the increase of mobile devices, people have become more reliant on wifi hotspots to connect to the internet. Reliable, safe, and fast are some of the main characteristics consumers expect their internet service to have. Unfortunately, safe and protected are characteristics that are overlooked when deciding to connect to public wifi. It is common for many businesses and organizations to offer free wifi. For the end-user, there is an inherent risk associated with using a free hotspot that is managed and owned by someone else, especially if the network is available for any to connect to. Open network connections do not require a passphrase to connect to the Wi-Fi or use encryption to encode the traffic being sent through the network. This means that there is no restriction to who can use the access point and anyone can read unencrypted information. To eliminate the security risk of relying on a service and connecting to an insecure network, in my project, a Raspberry Pi is used to create an internal network with security features while also providing an application for users to see relevant information about the newly created network.

System Requirements

The finished project should meet similar functionality of a wireless router which includes connecting multiple devices simultaneously, encrypted and secure traffic, and having an accessible dashboard using the Pi's IP address.

Any type of device with a wifi chip such as mobile phones, tablets, and laptops should be able to connect using the SSID and passphrase of the network regardless of the operating system. It is important to note that during the discovery phases from the client to Pi, both their wifi cards have to support the same data rates in order for the acknowledgment phase. For every successful connection to the Pi, a new IP address has to be assigned to the workstation in order for the proper packet to be sent and received correctly. Eth0 will mimic a WAN by providing internet connection and the built-in Wi-Fi (wlan0) will be used to interface with the clients connecting to the Raspberry Pi for internet access.

The wireless connectivity should be as effortless as connecting to an insecure network however it should be password protected. After authentication with the network, the router should already configure the network with the device to give it access to the internet almost instantly by forwarding the packets. Once connected, the internal network created should have an uptime until the user decides to turn off the Pi.

Network Topology

One of the challenges this project tries to overcome is restricted access to network equipment and configuration like in schools, apartmentments, and hotels. This project was implemented in an apartment complex where each apartment had 3 access points. These three access points have a seperate SSID but are able to communicate with each other. However, different sets of three access points could not communicate with each other even if they are in the same subnet. This means that the network is configured as a partially connected mesh shown in the figure below. To see a real example of the switch the connects the access points, see in the image in **appendix D.**

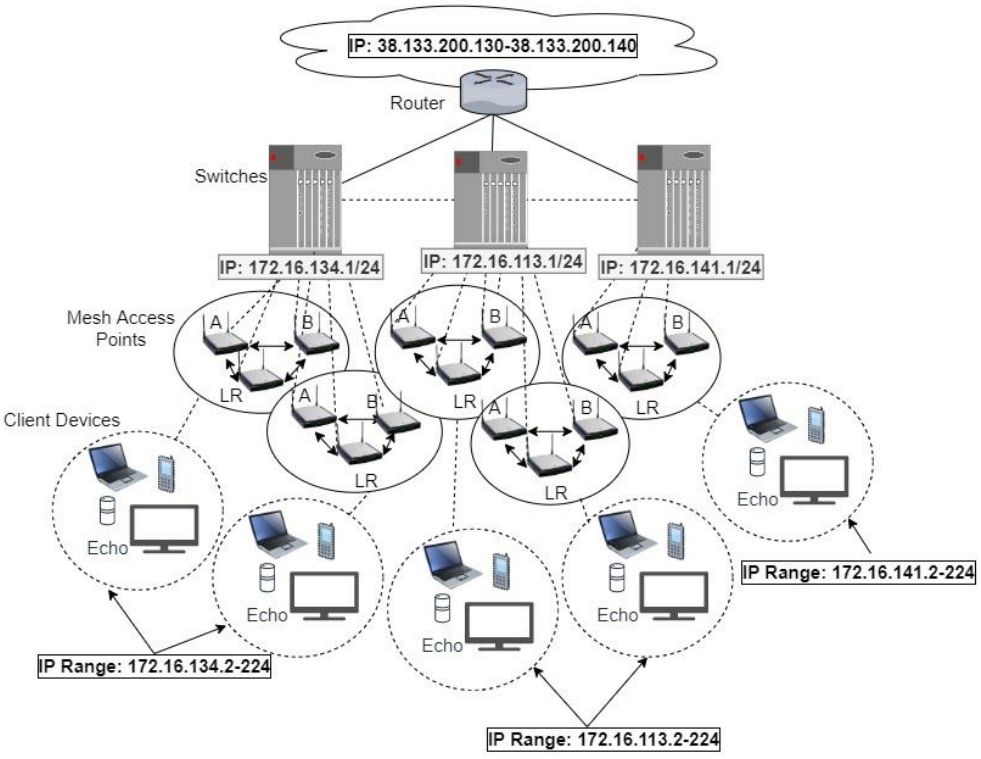

Figure 1 - Apartment Network Topology

In the figure above, each mesh access point is open for the public to connect to without a passphrase. This allows a person to connect to any network even if it is not the one intended for their apartment. To solve the issue, a new internal network is created using an ethernet connection and a Raspberry Pi that creates a network on a different subnet. This can be shown in the figure below with the addition of a new network with subnet 192.168.220.1/24.

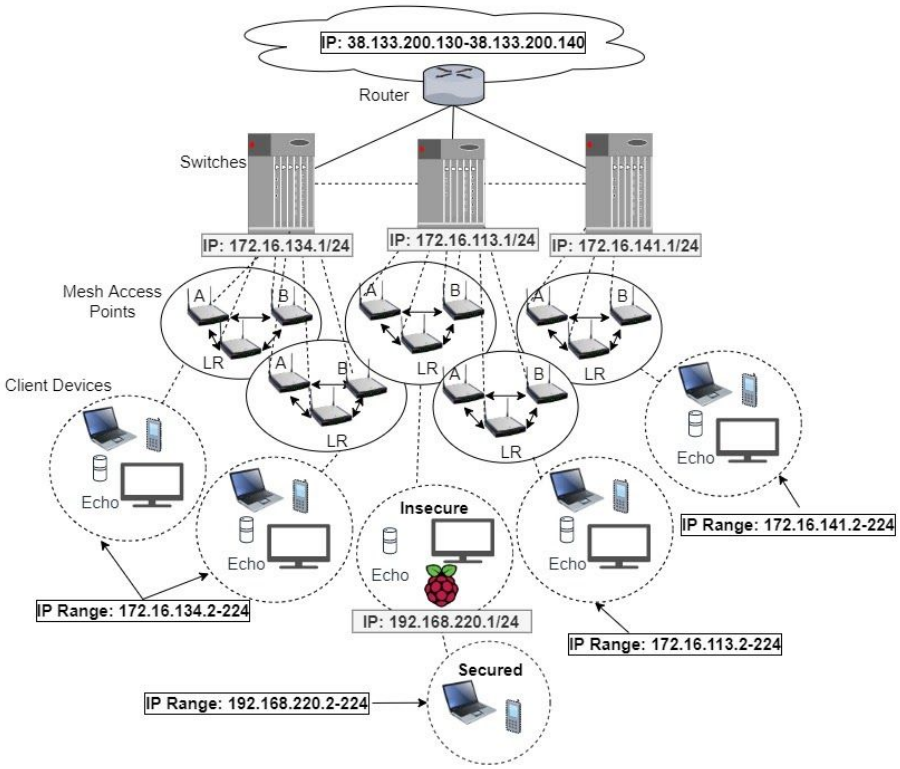

Figure 2 - Added LAN to existing network

WiFi

The Wi-Fi chip on the Raspberry Pi is the Cypress CYW43455 which is compatible with 802.11ac dual-band (2.4 and 5GHz) wireless LAN. In this project, the raspberry pi wlan0 will be configured in infrastructure mode which will allow it to act as a central access point for nearby devices to connect to it.

802.11 Wireless Standards

Year	IEEE Standard	Speed	Transmission Range (interference with material ca decrease range)	Freq
1999	802.11a	up tp 54mbit/sec	25 to 75 feet indoors	5GHz
1999	802.11b	up to 11mbit/sec	Up to 150 feet indoors	2.4GHz
2003	802.11g	up to 54mbit/sec	Up to 150 feet indoors	2.4GHz
2009	802.11n	up tp 600Mbit/sec	Up to 175+ feet indoors	2.4/5GHz

2013	802.11ac	100-1300mbit/sec	Up to 150 feet indoors	5GHz

Table 1 - Common Wifi Standards

The standard used in this project for wlan0 is 802.11g - 2.4GHz. This 802.11g protocol combines both 802.11a and 802.11b with backwards compatibility. Also, 802.11ac protocol is backwards compatible with g and n. This is why when referring to 802.11ac, it also means 802.11b/g/n/ac Wi-Fi .

Channels

802.11 protocols with freq 2.4GHz have 14 channels that are 22MHz wide. Each separated 5Hhz apart from each other except channel 14. The only channels that don't overlap are channels 1,6, 11, and 14. No overlapping channels decreases interference. For this reason, when we configure the hotspot, it will be on a channel that doesn't overlap. However, 14 can not be used as it is illegal to operate on channel 14.

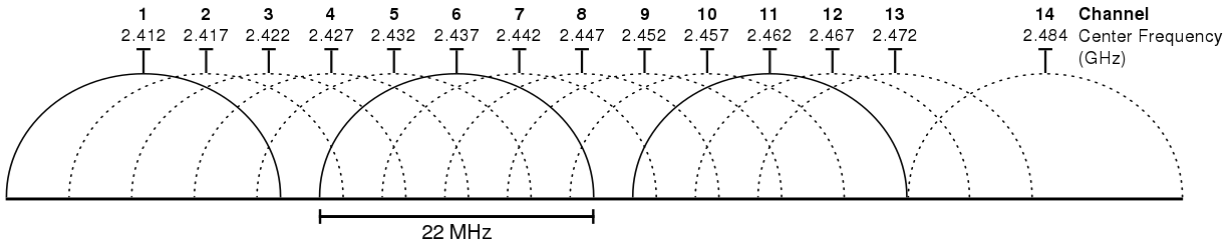

Figure 3 - 802.11b/g channels in 2.4 GHz band [1]

Security

The network will be secured using WPA2-Personal which uses a Pre-Shared Key (PSK) authentication. It will create access control by requiring a password to connect to the network and provide data protection using the password to generate a key for network traffic encryption and decryption.

802.11 Association Process

The association process is used for a device to authenticate and associate before data is transmitted. The figure below depicts the 4 way handshake.

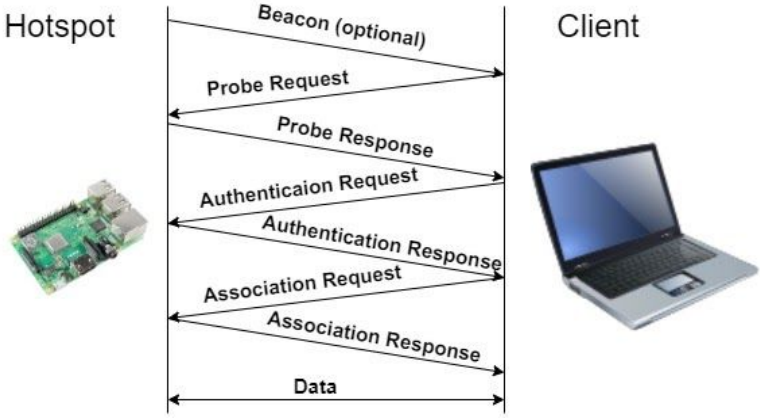

Figure 4 - Association process between Raspberry Pi and laptop

Ethernet

The Raspberry Pi uses a LAN7515 single-chip, hi-speed USB 2.0 to 10/100/1000 Gigabit Ethernet controller for an ethernet connection. After testing, the Raspberry Pi uses the max data transfer rate of 100Mbps.

Figure 5 - Current speed of ethernet provided by apartment complex with laptop

Figure 6 - Current speed of ethernet provided by apartment complex with Raspberry Pi

Design

Software

- Django Web Framework and Djano's development server
- Bootstrap to make it responsive on any screen resolution
- Linux packages hostapd, dnsmasq, vnstati

Layout

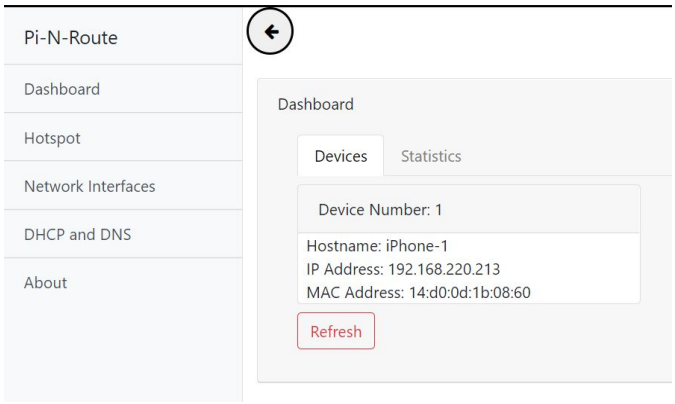

Figure 7 - Desktop version

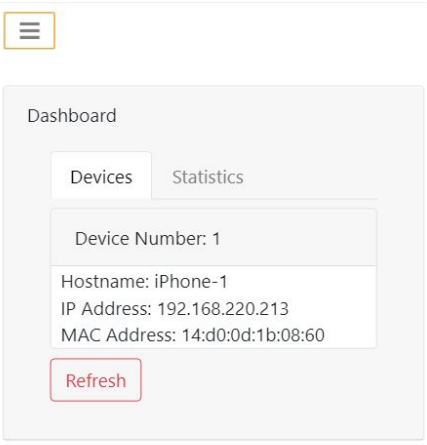

Figure 8 - Mobile version

Backend

```
def dashboard(request):
    devices = os.popen('arp -a').readlines()
    devicesLL = []
    i = 0

    for device in devices:
        deviceInfo = device.split(" ")
        if deviceInfo[0] != '?' and deviceInfo[3] != '<incomplete>' and deviceInfo[1] != '(1.1.1.1)':
            devicesLL.append({})
            devicesLL[i]["hostname"] = deviceInfo[0]
            devicesLL[i]["ip"] = deviceInfo[1][1:-1]
            devicesLL[i]["mac"] = deviceInfo[3].strip()
            i += 1
    context = {
        "devices": devicesLL
    }
    # os.system("sudo vnstat -u -i wlan0")
    os.system("rm status/static/status/img/summary1.png")
    os.system("rm status/static/status/img/summary3.png")
    os.system("vnstati -vs -c 1 -i wlan0 -o status/static/status/img/summary1.png")
    os.system("vnstati -s -i wlan0+eth0 -o status/static/status/img/summary3.png")

    return render(request, 'status/dashboard.html', context)
```

Figure 9 - Example of a python script in Django

The figure above is a python script used in django that reads the devices connected to the Raspberry Pi network and analytical bandwidth data in real-time. Each navigation section is broken into a python function.

Linux Packages

- Hostapd - Knows as host access point dameon and is used to create a network for devices to discover and connect to
- Dnsmasq - DHCP and DNS server to assign devices connected to Raspberry Pi a new IP address in subnet mask, forward packets correctly, allow access to the internet through eth0.
- Vnstati - network monitoring tool that can provide past and current information about data being transferred or received on a interface

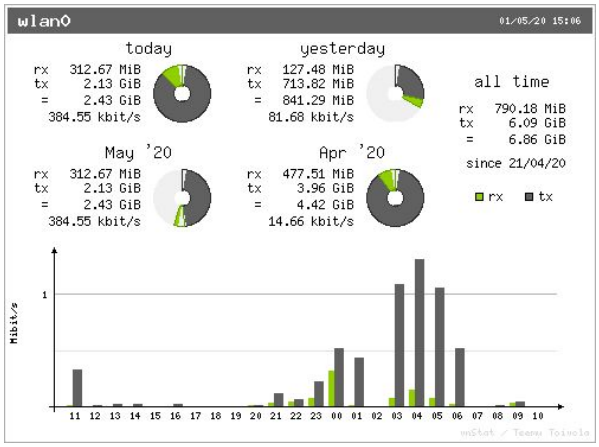

Figure 10 - Vnstati wlan0 summary

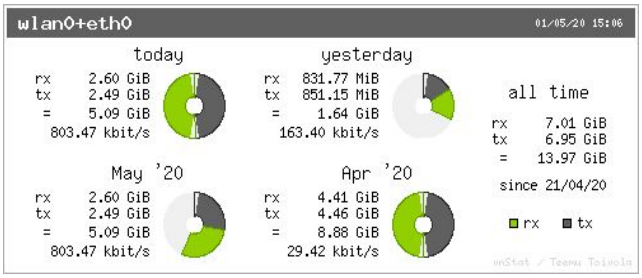

Figure 11 - Vnstati wlan0+eth0 summary

Hardware

- Raspberry Pi 3 B+ or 4
- Ethernet Connection

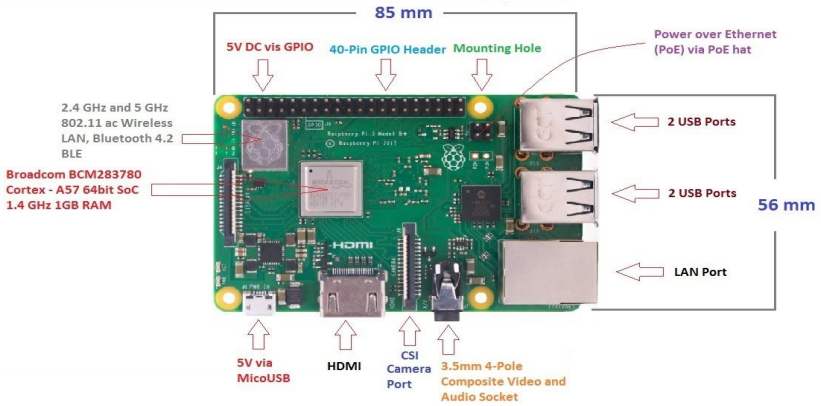

Figure 12 - Raspberry 3 B+ specs

Implementation

Steps done by running the" automateAP" on Raspberry Pi and can be view in **appendix B**:

1. Assign Raspberry Pi's interface wlan0 with a static IP. Ex: 192.168.220.1
2. Configure hostapd with information like SSID, passphrase, and security scheme
3. Configure DHCP ip range and lease time and to point to cloudflare DNS
4. Enable kernel IPv4 forwarding which allows devices connected to RPi to access the internet
5. Enable private IP address to communicate with LAN to get have internet connection and save rules on boot
6. Restart and enable services

Results

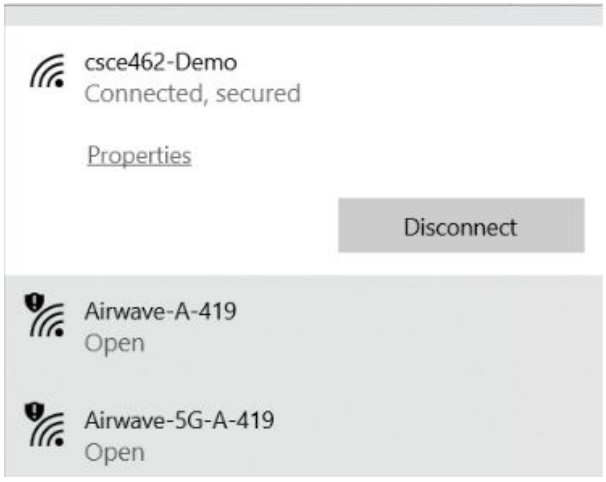

Figure 13 - Successful connection

Figure 14 - Network properties

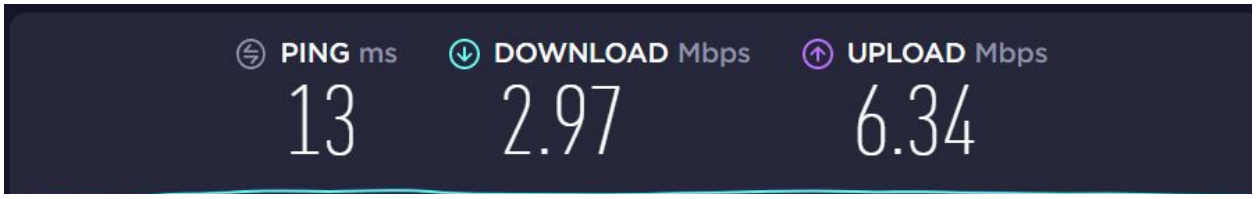

Figure 15 - Internet speed test

After connecting the network, there was a decrease in speeds but the rate speeds are enough for a device to watch youtube videos or surf the web. After testing, using three devices on the network at the same time did not have problems however using the internet for more than 3 devices simultaneously made the network unresponsive or really slow.

Appendix A material build

Quantity	Part name	
1	Sandisk Ultra 32GB Micro SD Card	$9.00
1	Raspberry Pi 3 Model B+	$30.00
1	CanaKit 5V 2.5A Power Supply USB Charger	$10.00
1	Ethernet Cable	$11.99

Appendix B automateAP script

```
sudo apt-get install hostapd dnsmasq -y
sudo systemctl stop hostapd
sudo systemctl stop dnsmasq

echo "interface wlan0" >> /etc/dhcpcd.conf
echo "    static ip_address=192.168.220.1/24" >> /etc/dhcpcd.conf
echo "    nohook wpa_supplicant" >> /etc/dhcpcd.conf

sudo systemctl restart dhcpcd

echo "# config file for AP" >> /etc/hostapd/hostapd.conf
echo "ssid=csce462-Demo" >> /etc/hostapd/hostapd.conf
echo "wpa_passphrase=abc12308" >> /etc/hostapd/hostapd.conf # has to at least have 8 length
echo "interface=wlan0" >> /etc/hostapd/hostapd.conf
echo "driver=nl80211" >> /etc/hostapd/hostapd.conf
echo "hw_mode=g" >> /etc/hostapd/hostapd.conf
echo "channel=6" >> /etc/hostapd/hostapd.conf
echo "ieee80211n=1" >> /etc/hostapd/hostapd.conf
echo "wmm_enabled=0" >> /etc/hostapd/hostapd.conf
echo "macaddr_acl=0" >> /etc/hostapd/hostapd.conf
echo "ignore_broadcast_ssid=0" >> /etc/hostapd/hostapd.conf
```

```
echo "auth_algs=1" >> /etc/hostapd/hostapd.conf
echo "wpa=2" >> /etc/hostapd/hostapd.conf
echo "wpa_key_mgmt=WPA-PSK" >> /etc/hostapd/hostapd.conf
echo "wpa_pairwise=TKIP" >> /etc/hostapd/hostapd.conf
echo "rsn_pairwise=CCMP" >> /etc/hostapd/hostapd.conf

sed -i 's/#DAEMON_CONF=\"\"/DAEMON_CONF=\"\/etc\/hostapd\/hostapd.conf\"/'
/etc/default/hostapd
sudo mv "/etc/dnsmasq.conf" "/etc/dnsmasq.conf.orig"

echo "# Using interface wlan0 and Cloudflare DNS" >> /etc/dnsmasq.conf
echo "interface=wlan0" >> /etc/dnsmasq.conf
echo "server=1.1.1.1" >> /etc/dnsmasq.conf
echo "# IP range and 24h lease time" >> /etc/dnsmasq.conf
echo "dhcp-range=192.168.220.2,192.168.220.224,24h" >> /etc/dnsmasq.conf

sed -i 's/#net.ipv4.ip_forward=1/net.ipv4.ip_forward=1/' /etc/sysctl.conf

sudo sh -c "echo 1 > /proc/sys/net/ipv4/ip_forward"

sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
sudo sh -c "iptables-save > /etc/iptables.ipv4.nat"
sed -i 's/exit 0/iptables-restore < \/etc\/iptables.ipv4.nat/' /etc/rc.local
echo "exit 0" >> /etc/rc.local

sudo systemctl unmask hostapd
sudo systemctl enable hostapd
sudo systemctl start hostapd
sudo service dnsmasq start
```

Appendix C Application Code

https://github.com/saenzjonathan11/CSCE-462-Raspberry-Pi-Documentation.git

Appendix D Images

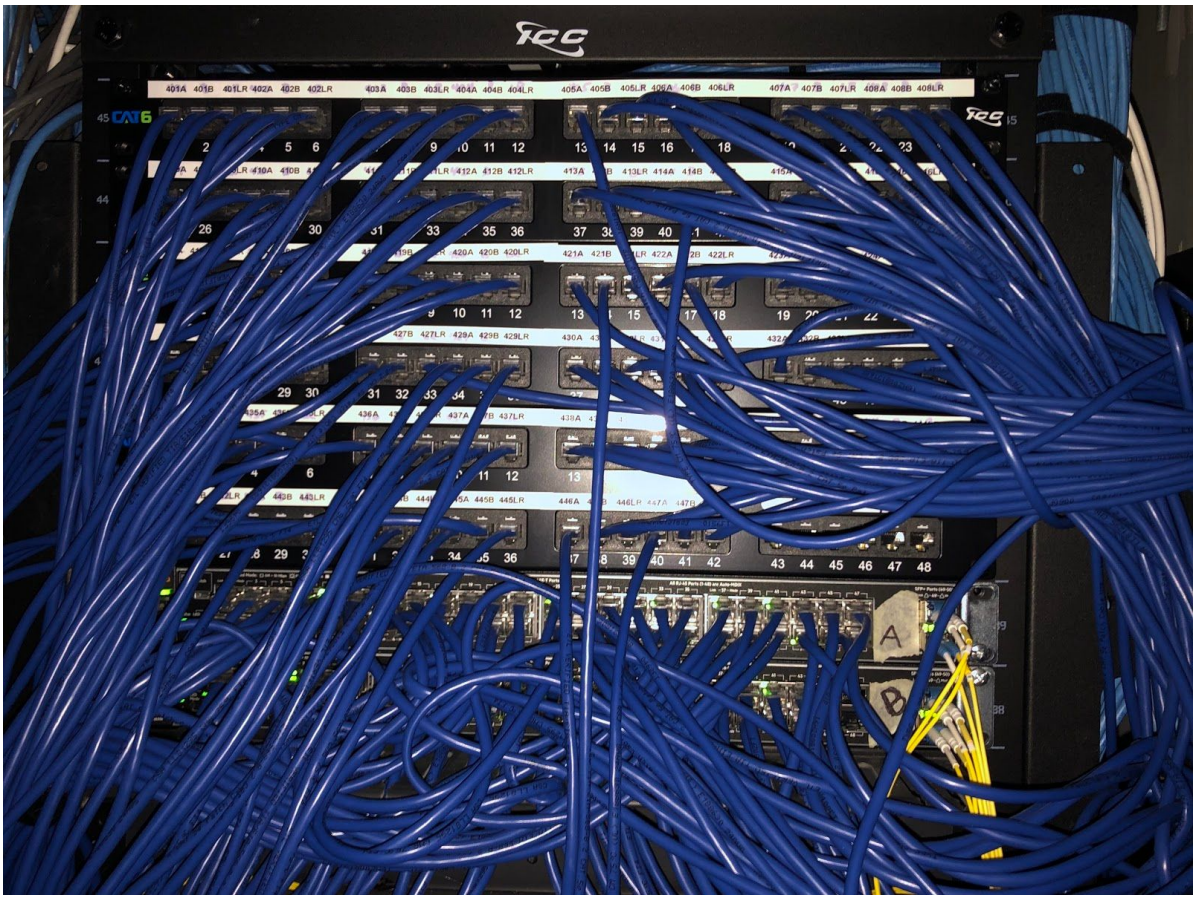

Figure 16 - Apartment complex switch rack

References

[1]https://upload.wikimedia.org/wikipedia/commons/thumb/8/8c/2.4_GHz_Wi-Fi_channels_%28802.11b%2Cg_WLAN%29.svg/1245px-2.4_GHz_Wi-Fi_channels_%28802.11b%2Cg_WLAN%29.svg.png

[2]https://www.theengineeringprojects.com/wp-content/uploads/2018/07/introduction-to-raspberry-pi-3-b-plus-1-300x203.jpg

[3]https://www.speedtest.net/

[4]https://raspberry-valley.azurewebsites.net/RaspAP-Wifi-Hotspot/