BL(u)E CRAB:

RSSI Detection Pattern Analysis for Flagging System Development

Zhi Qu

Bluetooth Low Energy (BLE)

- Smaller devices that use battery and are often portable
- Virtually all laptops, tablet computers, computer printers and cellphones now have 802.11 wireless modems using the 2.4 and 5.7 GHz ISM bands.
- Transfers small pieces of information often are
- Not compatible with Bluetooth classic rather they connect with a device with both types to relay information. Cellphones have both and can act as the connection
- Smart home, Fitness device, **GPS replacement**
- Line of sight 50 m
- New feature of Bluetooth 5 now the range is up to 800 m

What is the Threat?

Stalking

- Making unwanted and persistent phone calls
- Approaching or showing up in places uninvited
- Following and watching the person
- Sending unwanted texts, emails, and social media messages
- Delivering unwanted gifts
- Utilizing technology for monitoring and tracking

What is the Threat?

Stalking

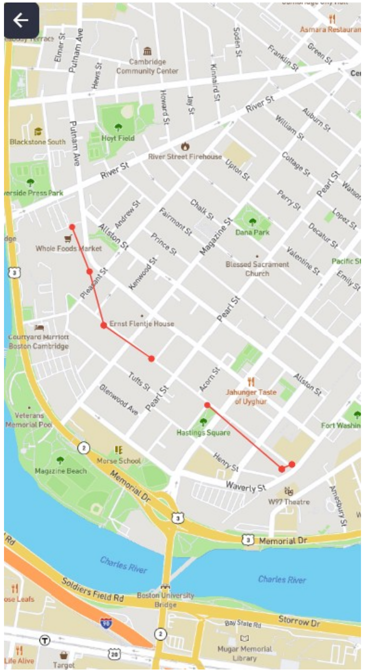
- Making unwanted and persistent phone calls
- Approaching or showing up in places uninvited
- Following and watching the person
- Sending unwanted texts, emails, and social media messages
- Delivering unwanted gifts
- Utilizing technology for monitoring and tracking

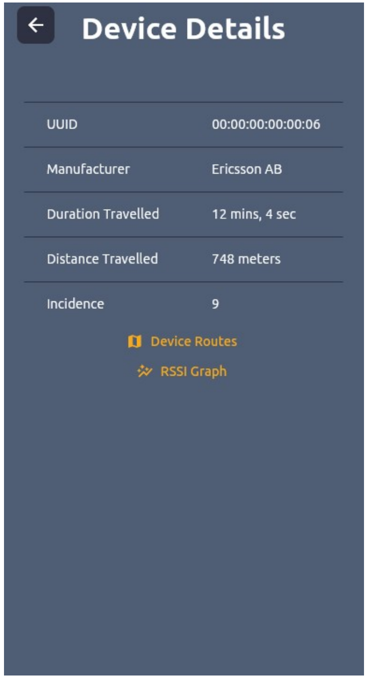
BL(u)E CRAB

- It can be used across many platforms
- Scans for BLE nearby
- Assesses device risk by z-score to grade each
- And displays information like device mac address
- Logs device info

Information log

- location
- time
- RSSI





Device Details

UUID	00:00:00:00:00:06
Manufacturer	Ericsson AB
Duration Travelled	12 mins, 4 sec
Distance Travelled	748 meters
Incidence	9

What is this?

Received Signal Strength Indicator

As the name suggests, RSSI is an indication of the strength level received in radio signals. the greater the RSSI value, the stronger the signal. In this case the focus is on BLE device transmissions

Question

How does the RSSI values differ for suspicious and non-suspicious devices?

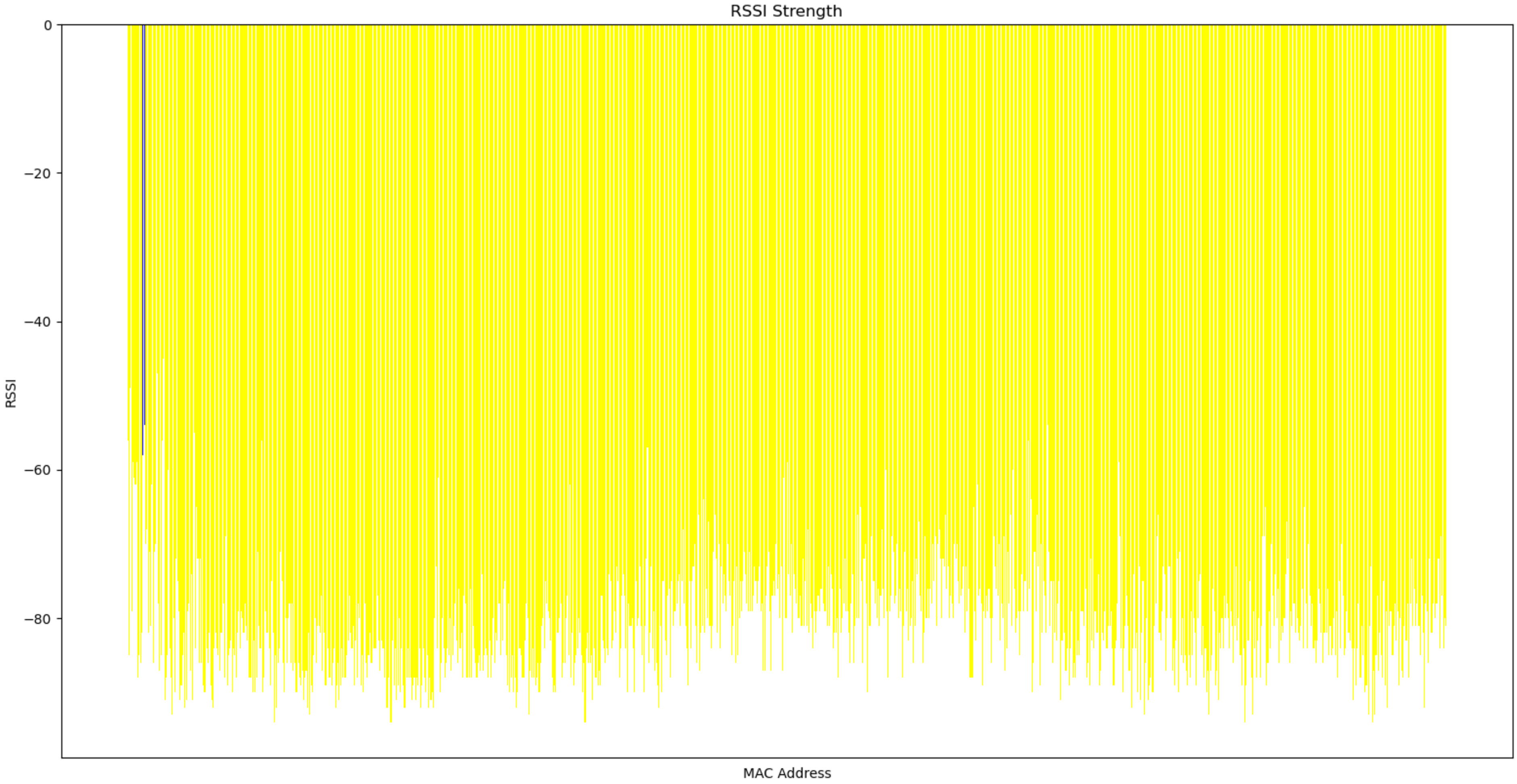
- Are they correlated?
- Is there a pattern throughout?

Methods

Reading the data

- The data is from a previous study

Graph

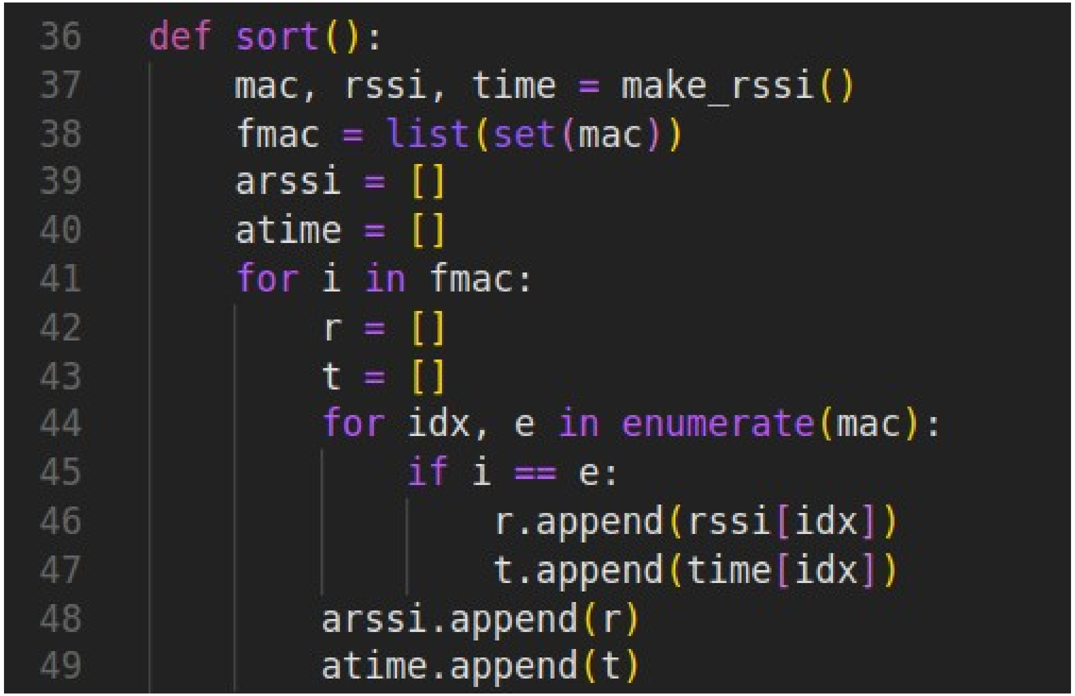


Graph

RSSI Strength
0
−20
−80
MAC Address

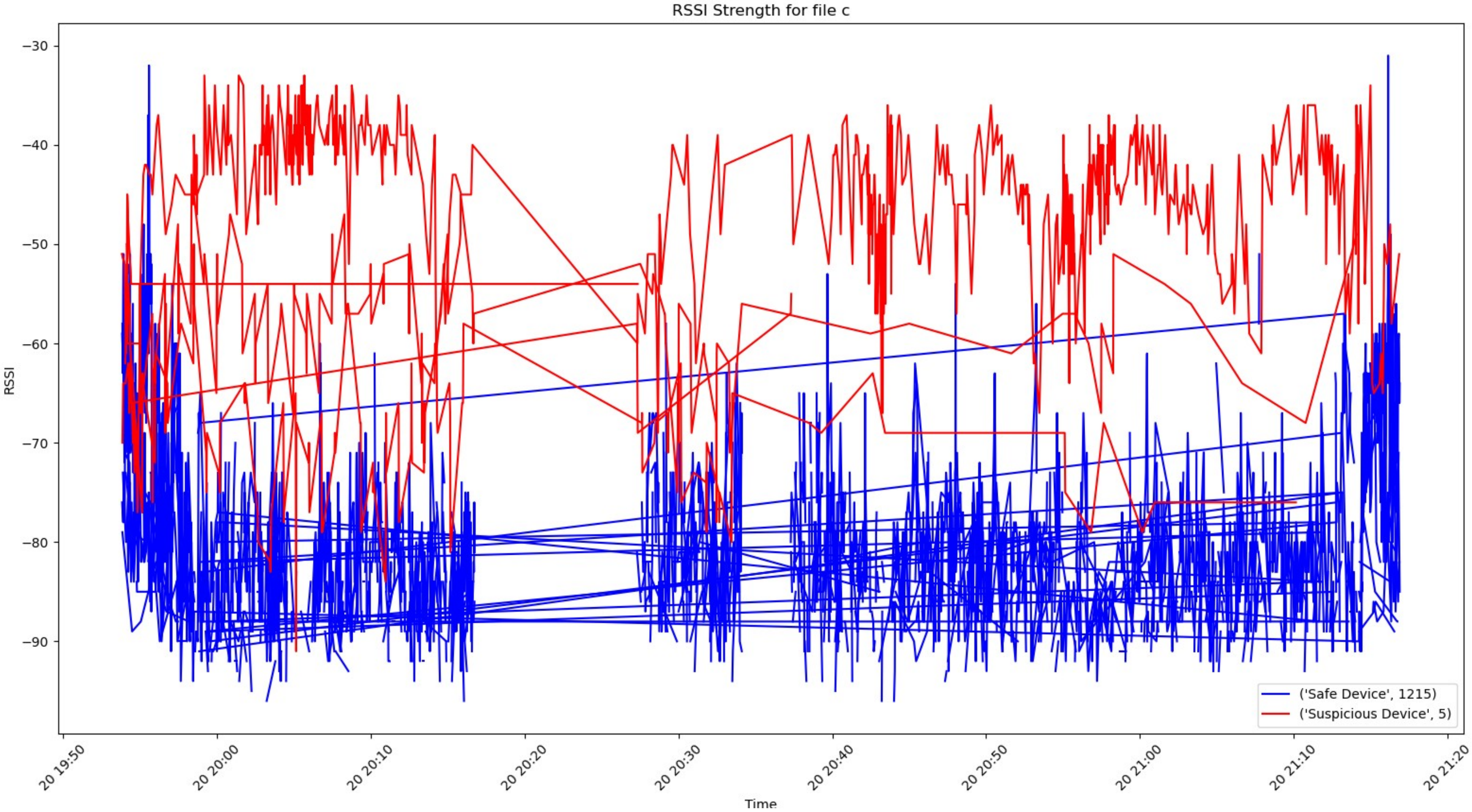
Wait. What is this?

Identifying each mac id

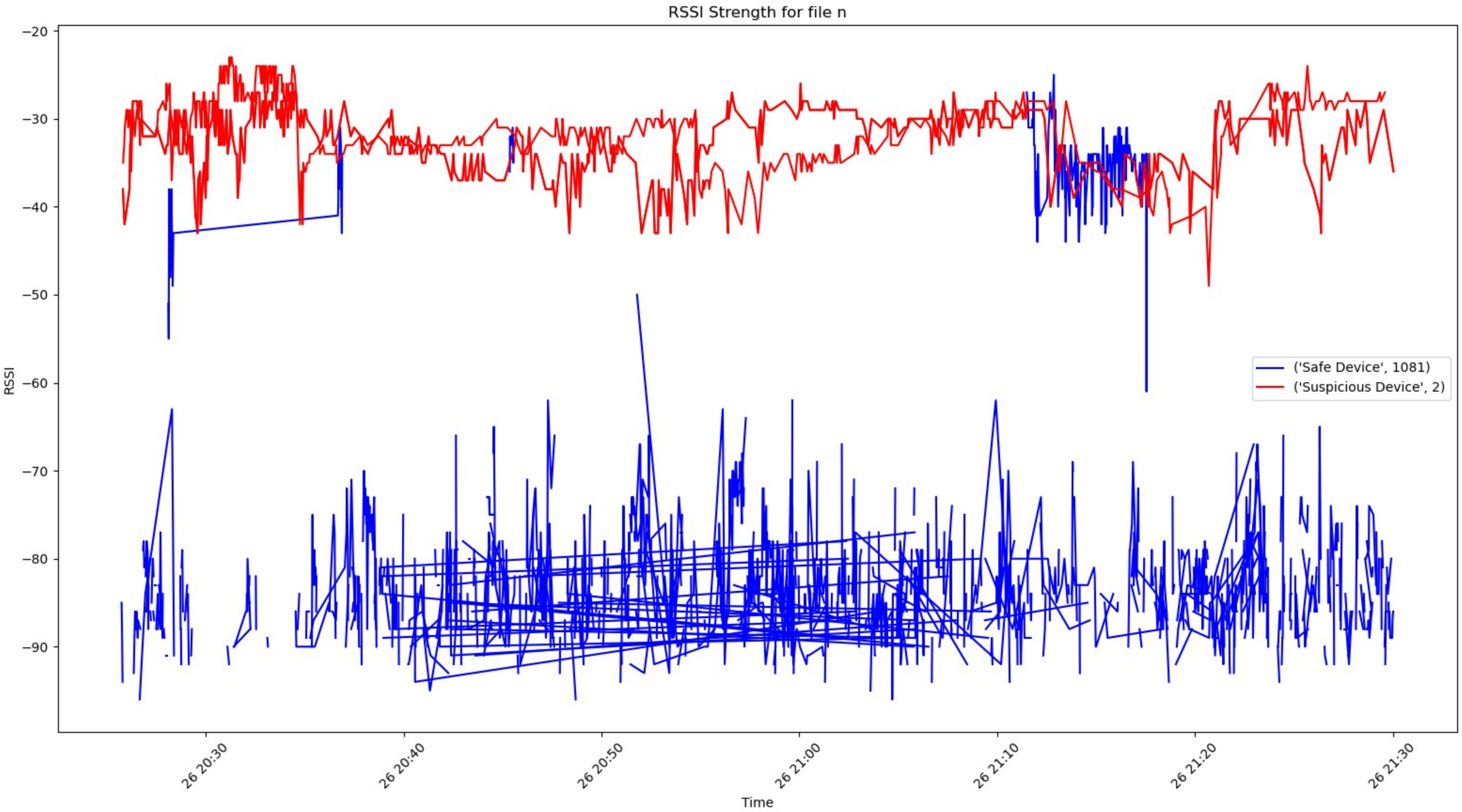


```python
def sort():
    mac, rssi, time = make_rssi()
    fmac = list(set(mac))
    arssi = []
    atime = []
    for i in fmac:
        r = []
        t = []
        for idx, e in enumerate(mac):
            if i == e:
                r.append(rssi[idx])
                t.append(time[idx])
        arssi.append(r)
        atime.append(t)
```

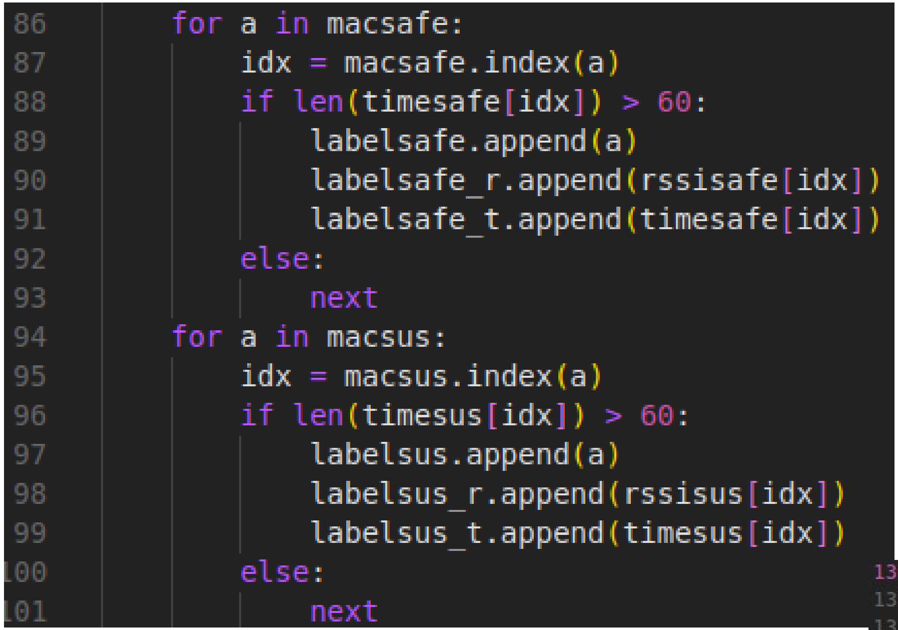
Data file c


RSSI Strength for file c
−30
−40
−50
−60
−70
−80
−90
RSSI
20 19:50
20 20:00
20 20:10
20 20:20
20 20:30
20 20:40
20 20:50
20 21:00
20 21:10
20 21:20
Time
('Safe Device', 1215)
('Suspicious Device', 5)

Data file n


RSSI Strength for file n
RSSI
−20
−30
−40
−50
−60
−70
−80
−90
Time
26 20:30
26 20:40
26 20:50
26 21:00
26 21:10
26 21:20
26 21:30
('Safe Device', 1081)
('Suspicious Device', 2)

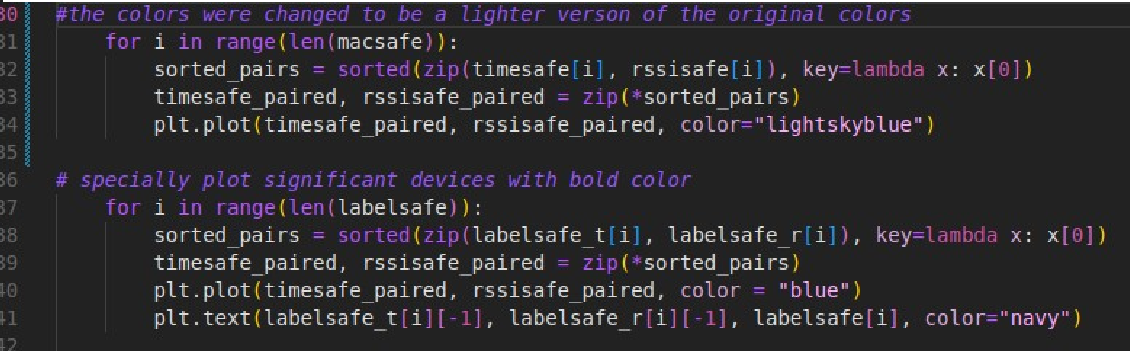
Significant time Detected



```
    for a in macsafe:
        idx = macsafe.index(a)
        if len(timesafe[idx]) > 60:
            labelsafe.append(a)
            labelsafe_r.append(rssisafe[idx])
            labelsafe_t.append(timesafe[idx])
        else:
            next
    for a in macsus:
        idx = macsus.index(a)
        if len(timesus[idx]) > 60:
            labelsus.append(a)
            labelsus_r.append(rssisus[idx])
            labelsus_t.append(timesus[idx])
        else:
            next
```

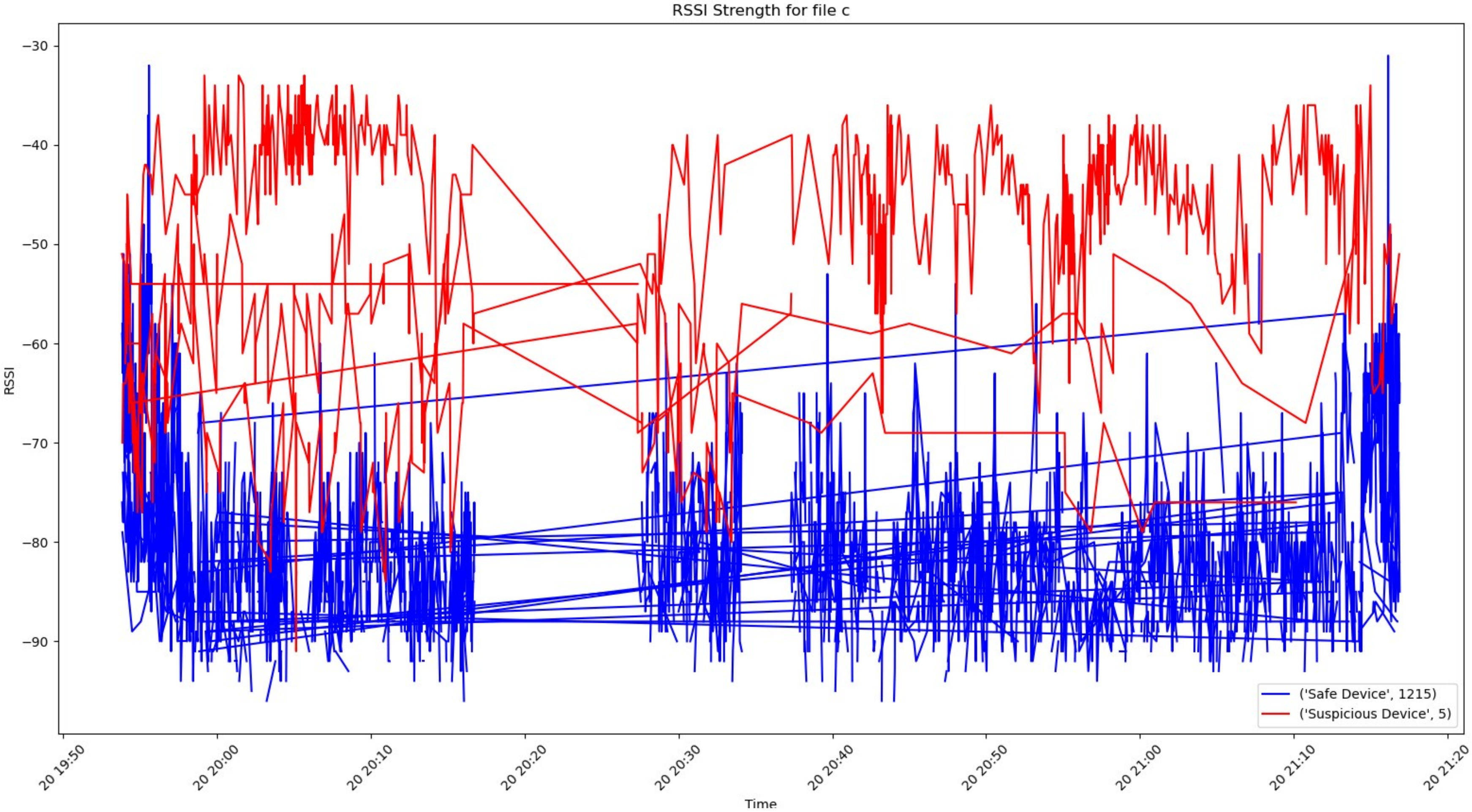
Setting threshold to 60 seconds

Plotting with different colors

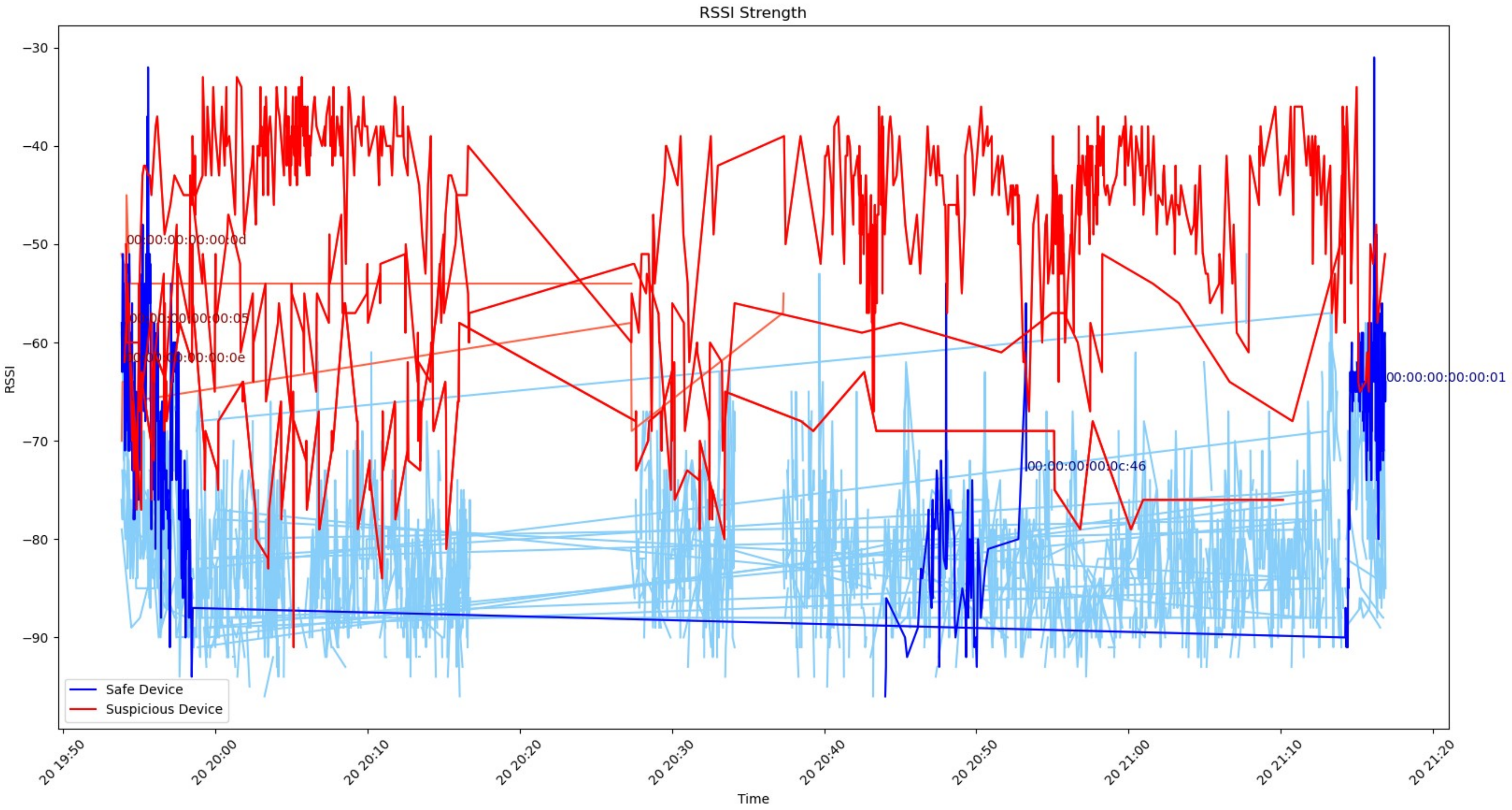


```
#the colors were changed to be a lighter verson of the original colors
    for i in range(len(macsafe)):
        sorted_pairs = sorted(zip(timesafe[i], rssisafe[i]), key=lambda x: x[0])
        timesafe_paired, rssisafe_paired = zip(*sorted_pairs)
        plt.plot(timesafe_paired, rssisafe_paired, color="lightskyblue")

# specially plot significant devices with bold color
    for i in range(len(labelsafe)):
        sorted_pairs = sorted(zip(labelsafe_t[i], labelsafe_r[i]), key=lambda x: x[0])
        timesafe_paired, rssisafe_paired = zip(*sorted_pairs)
        plt.plot(timesafe_paired, rssisafe_paired, color = "blue")
        plt.text(labelsafe_t[i][-1], labelsafe_r[i][-1], labelsafe[i], color="navy")
```

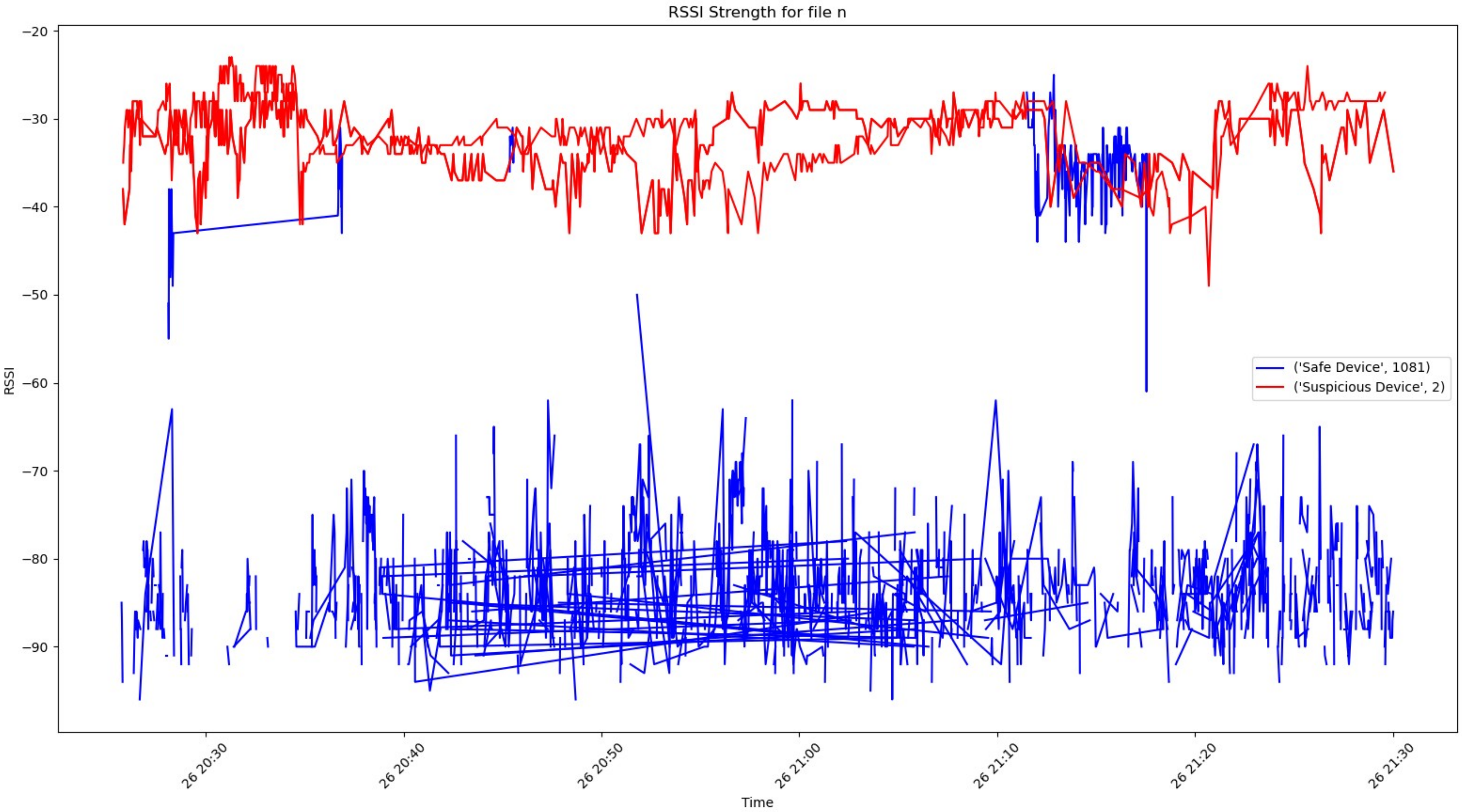
Data file c


RSSI Strength for file c
RSSI
−30
−40
−50
−60
−70
−80
−90
Time
20 19:50
20 20:00
20 20:10
20 20:20
20 20:30
20 20:40
20 20:50
20 21:00
20 21:10
20 21:20
('Safe Device', 1215)
('Suspicious Device', 5)

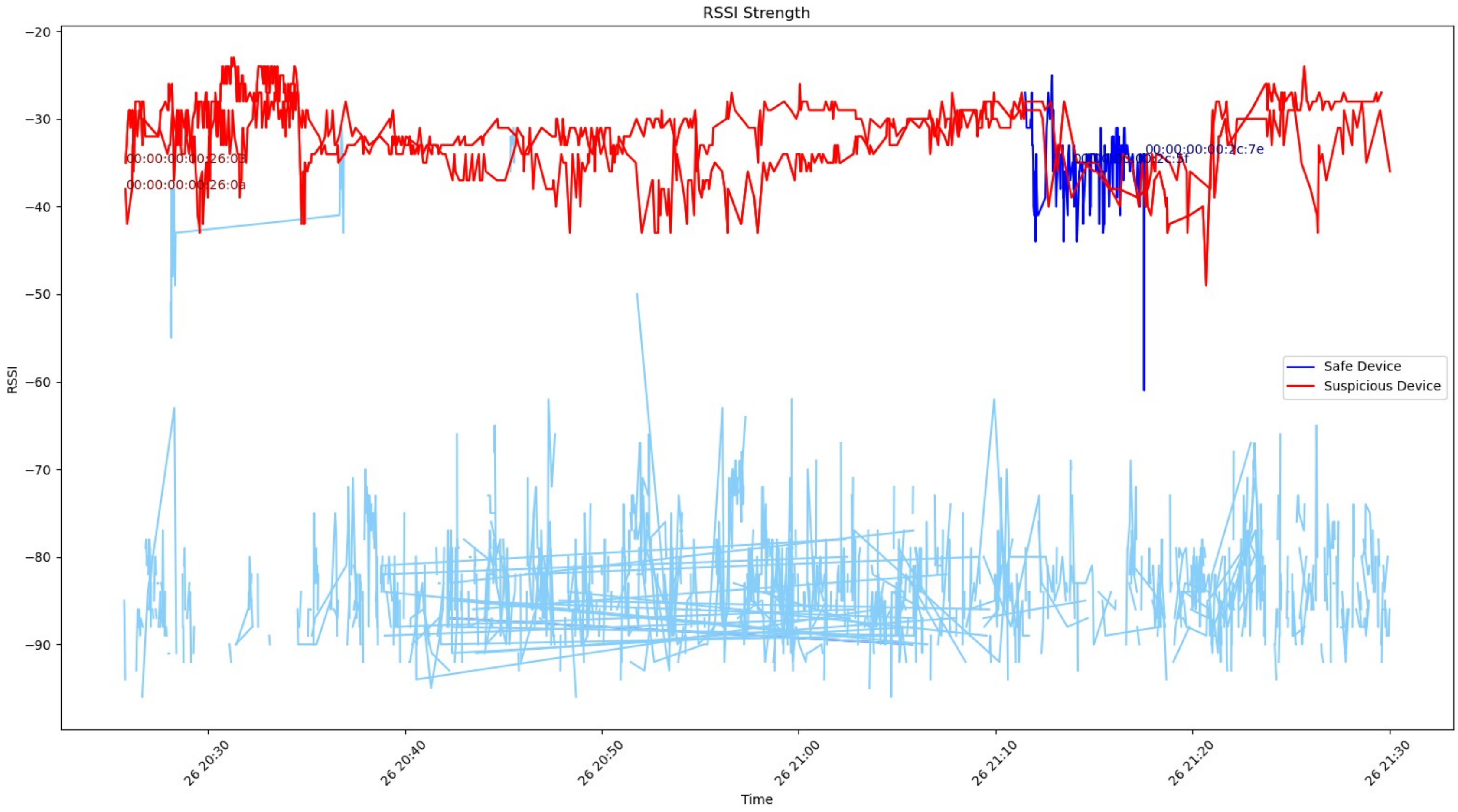
Data file c – long detection



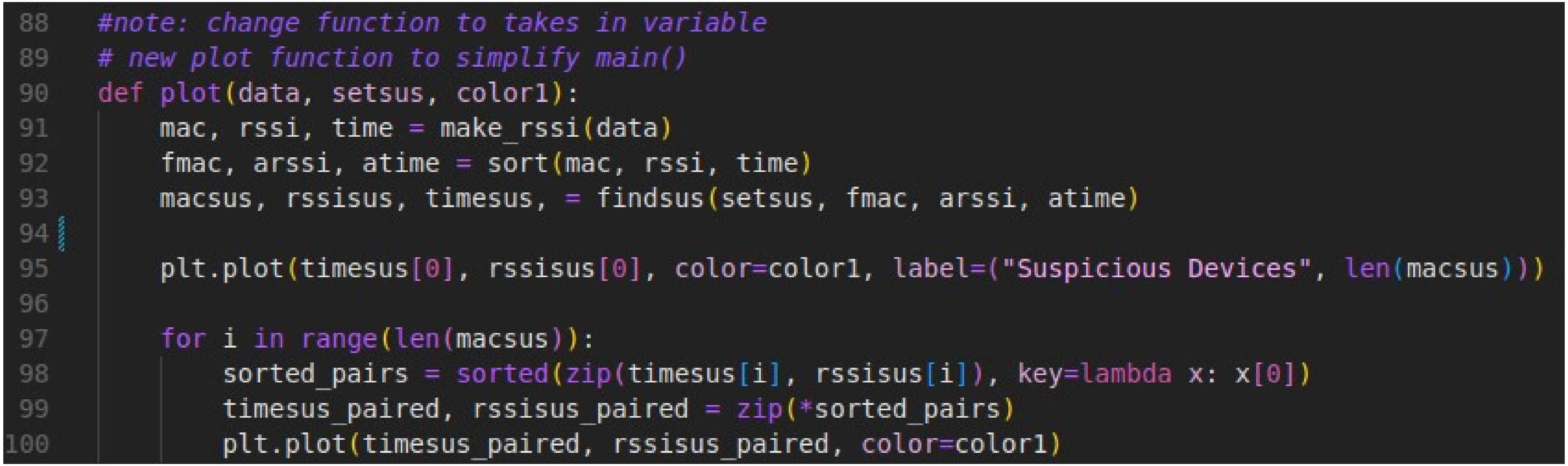
Data file n


RSSI Strength for file n
RSSI
−20
−30
−40
−50
−60
−70
−80
−90
Time
26 20:30
26 20:40
26 20:50
26 21:00
26 21:10
26 21:20
26 21:30
('Safe Device', 1081)
('Suspicious Device', 2)

Data file n - long detection

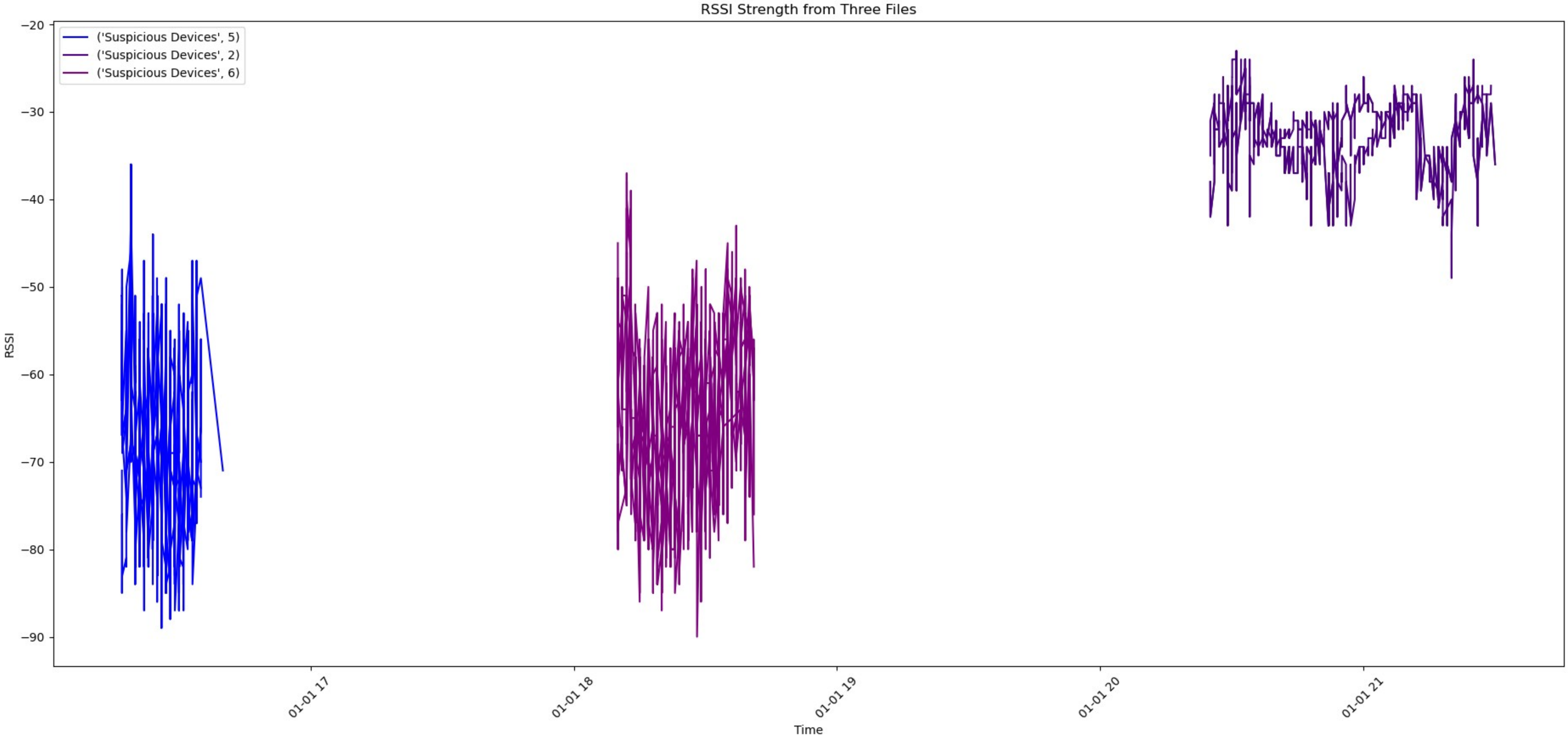


Time of Day

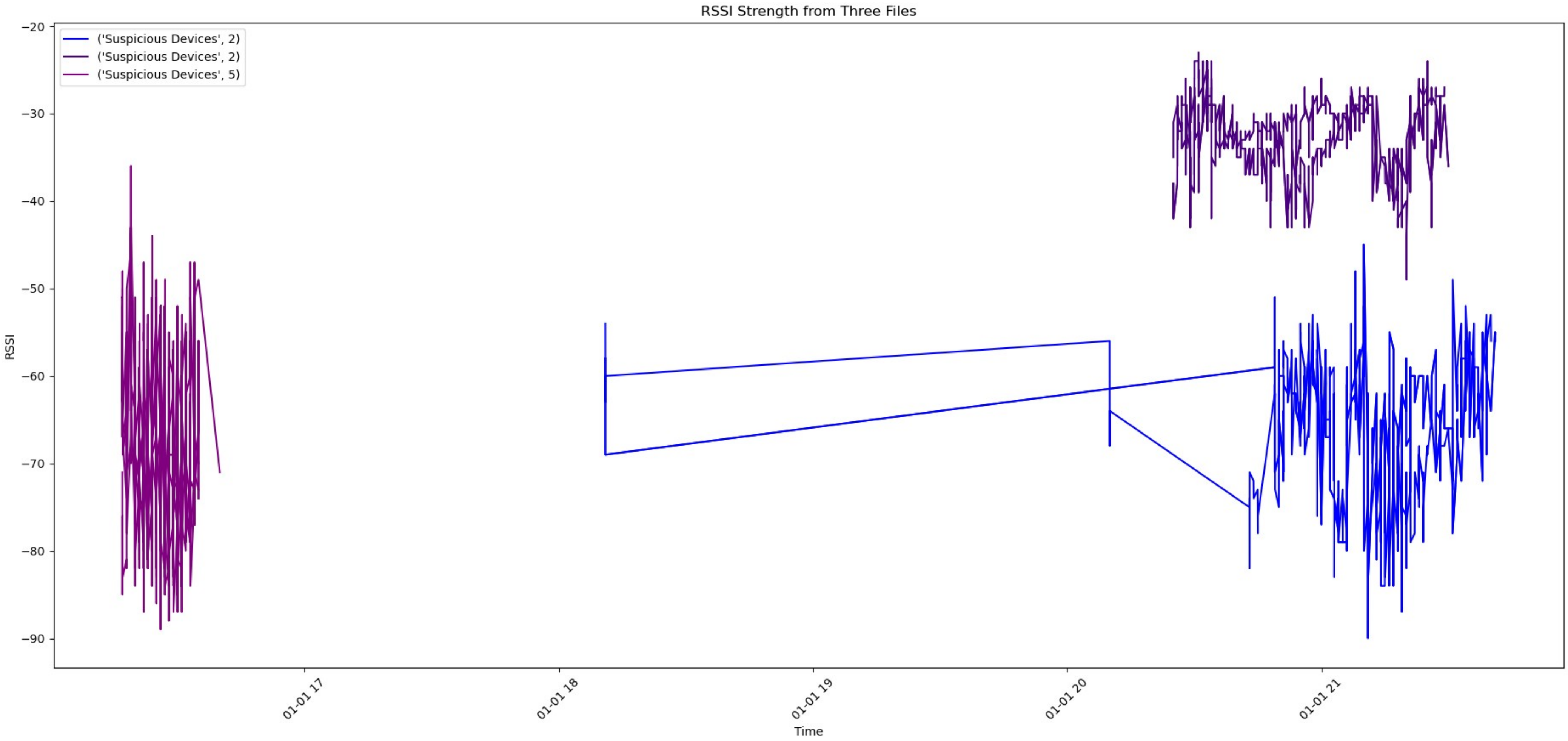


```
#note: change function to takes in variable
# new plot function to simplify main()
def plot(data, setsus, color1):
    mac, rssi, time = make_rssi(data)
    fmac, arssi, atime = sort(mac, rssi, time)
    macsus, rssisus, timesus, = findsus(setsus, fmac, arssi, atime)

    plt.plot(timesus[0], rssisus[0], color=color1, label=("Suspicious Devices", len(macsus)))

    for i in range(len(macsus)):
        sorted_pairs = sorted(zip(timesus[i], rssisus[i]), key=lambda x: x[0])
        timesus_paired, rssisus_paired = zip(*sorted_pairs)
        plt.plot(timesus_paired, rssisus_paired, color=color1)
```

Time of day – G,N,J



Time of day – G,N,A



Conclusion

Discussion