BL(u)E CRAB:

RSSI Detection Pattern Analysis for Flagging System Development

Zhi Qu

Bluetooth Low Energy (BLE)

- Low power devices
- 2.4 GHz ISM band
- IoT connections
- Not compatible with Bluetooth classic
- Applications
- Range

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

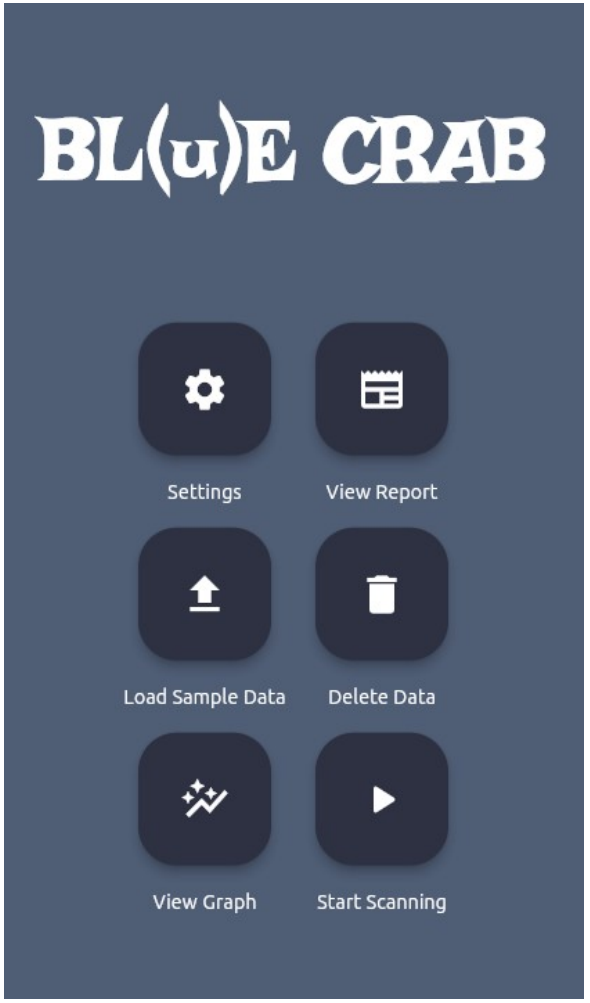
What is the Threat?

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

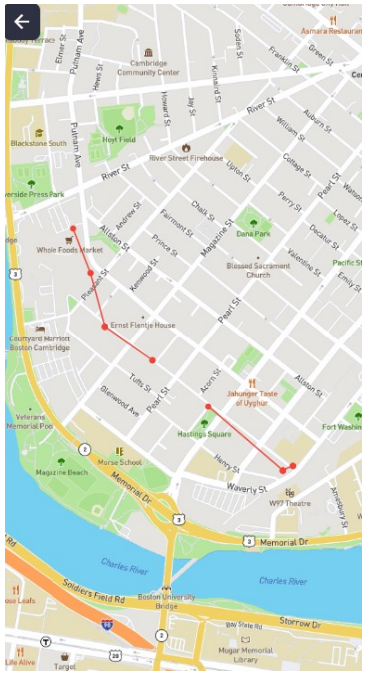
BL(u)E CRAB

- Flutter app
- Scans for BLE devices nearby
- Assess risk
- Flags device
- Logs device info



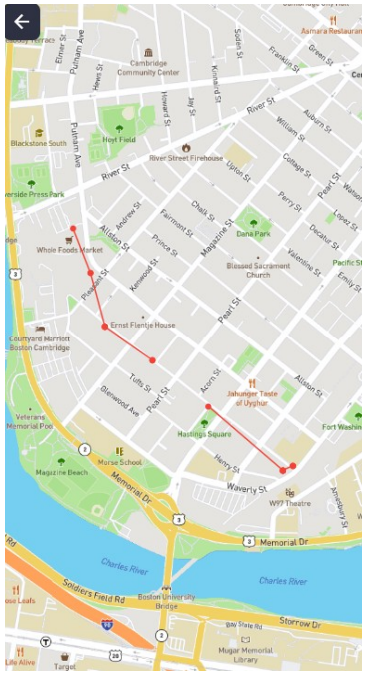
Information Log

- location
- time
- RSSI

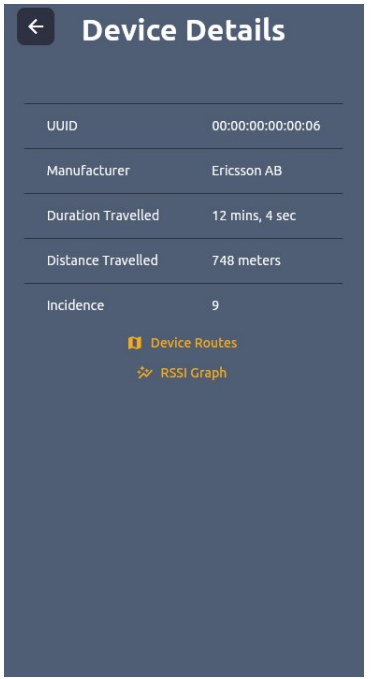


Information Log

- location



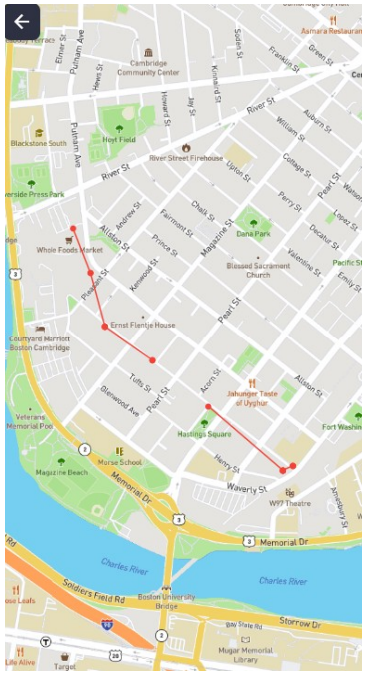
- time

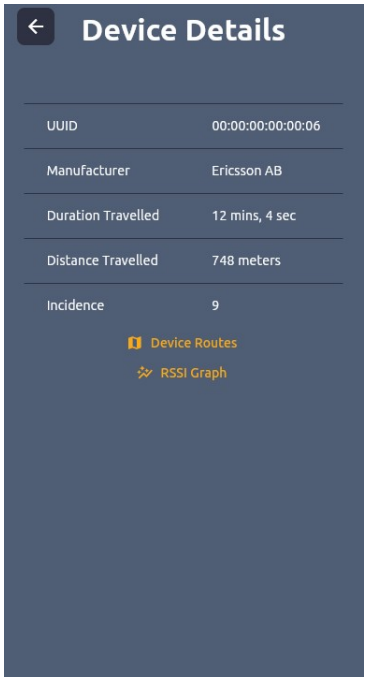


- RSSI

Information Log

- location
- time
- RSSI

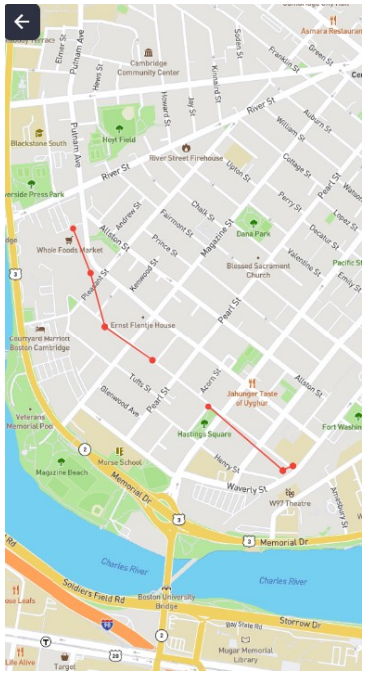




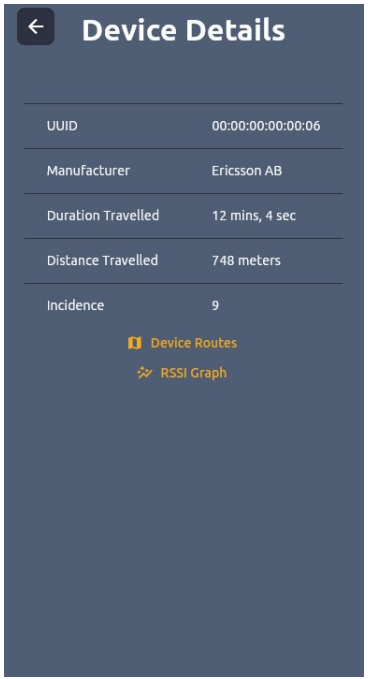
What is this?

Information Log

- location



- time



- RSSI

Received Signal Strength Indicator

Question

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

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

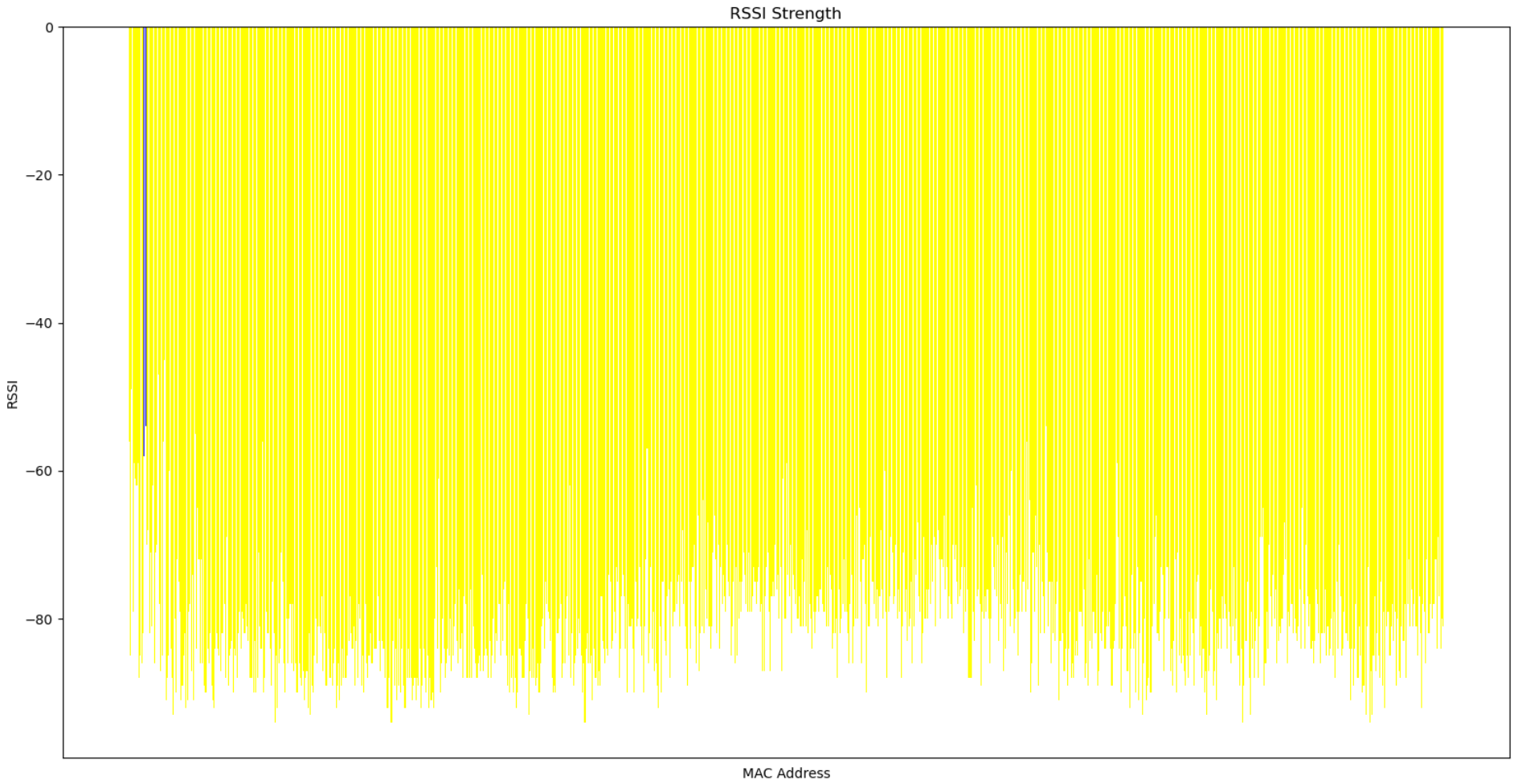
Methods

Reading the data

- The data is from a previous study

A	Walking	Backpack
B	Walking	Backpack
C	Walking	
D	Walking	
E	Car	
F	Jogging	Backpack
G	Walking	Backpack
H	Walking	Backpack
I	Train	Backpack
J	Train	Backpack
K	Walking	
L	Walking	
M	Train	
N	Car	

Graph



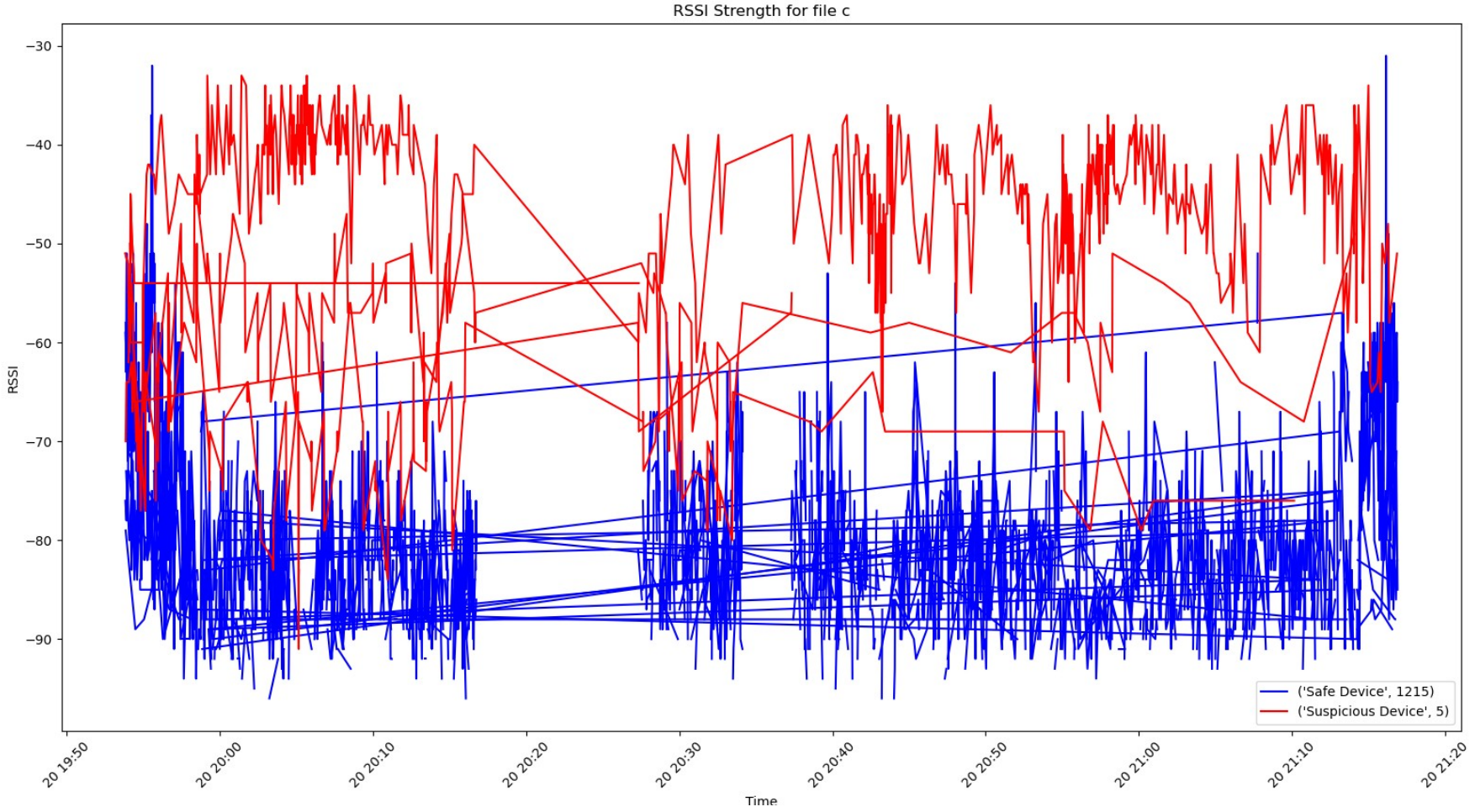
Graph

Wait. What is this?

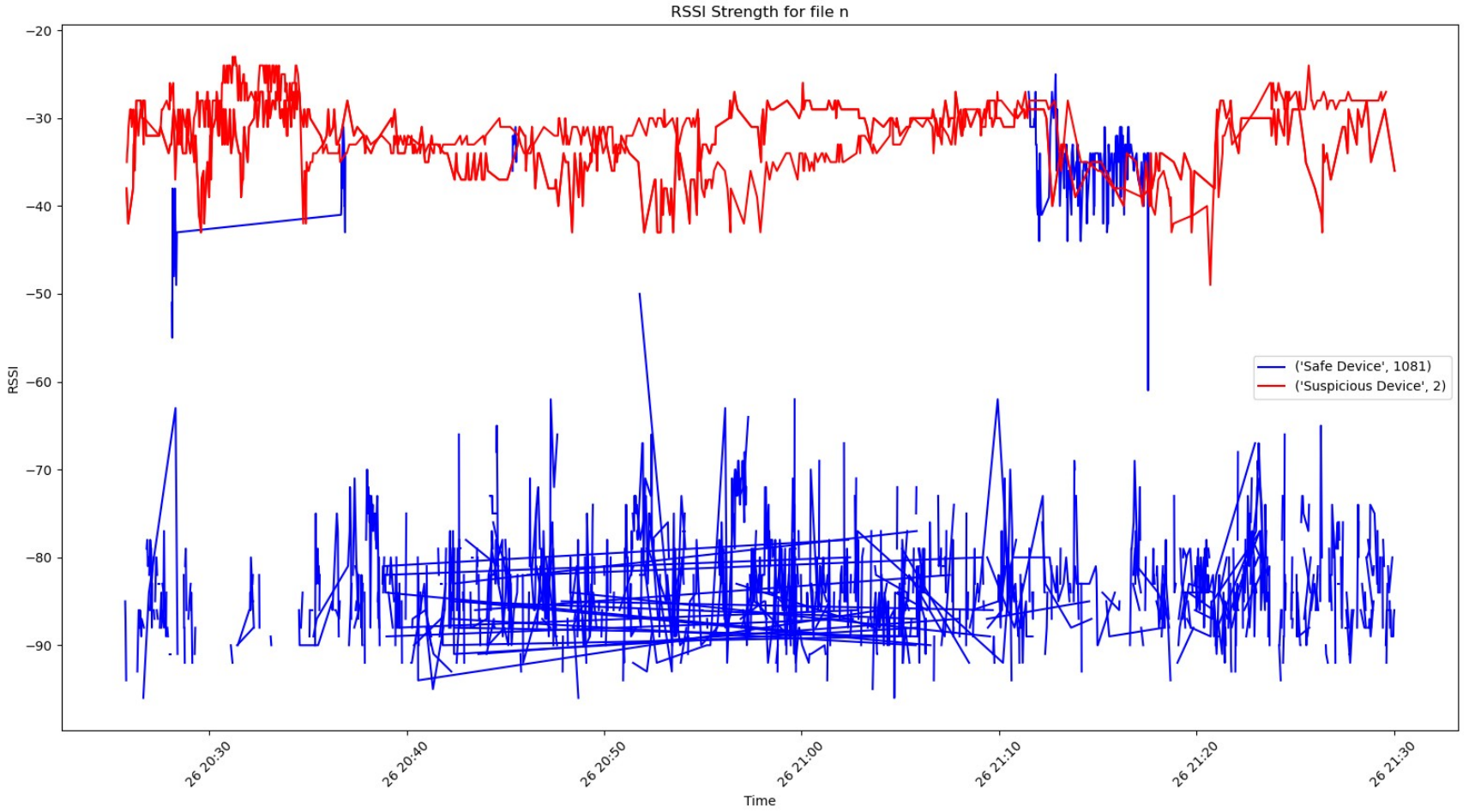
Identifying each mac id

```
36  def sort():
37      mac, rssi, time = make_rssi()
38      fmac = list(set(mac))
39      arssi = []
40      atime = []
41      for i in fmac:
42          r = []
43          t = []
44          for idx, e in enumerate(mac):
45              if i == e:
46                  r.append(rssi[idx])
47                  t.append(time[idx])
48          arssi.append(r)
49          atime.append(t)
```

Data file c


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

Significant time Detected

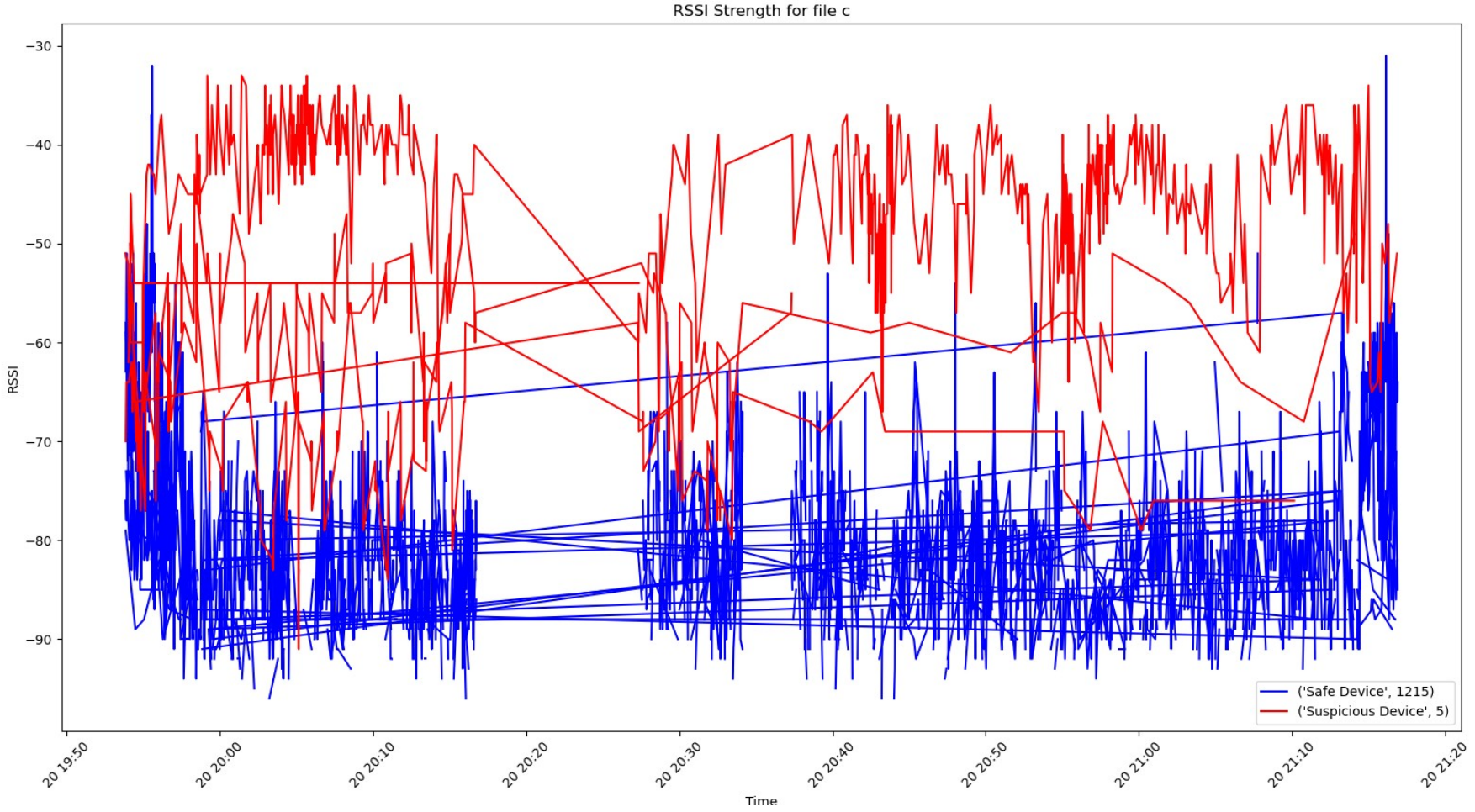
```
86      for a in macsafe:
87          idx = macsafe.index(a)
88          if len(timesafe[idx]) > 60:
89              labelsafe.append(a)
90              labelsafe_r.append(rssisafe[idx])
91              labelsafe_t.append(timesafe[idx])
92          else:
93              next
94      for a in macsus:
95          idx = macsus.index(a)
96          if len(timesus[idx]) > 60:
97              labelsus.append(a)
98              labelsus_r.append(rssisus[idx])
99              labelsus_t.append(timesus[idx])
100         else:
101             next
```

Setting threshold to 60 seconds

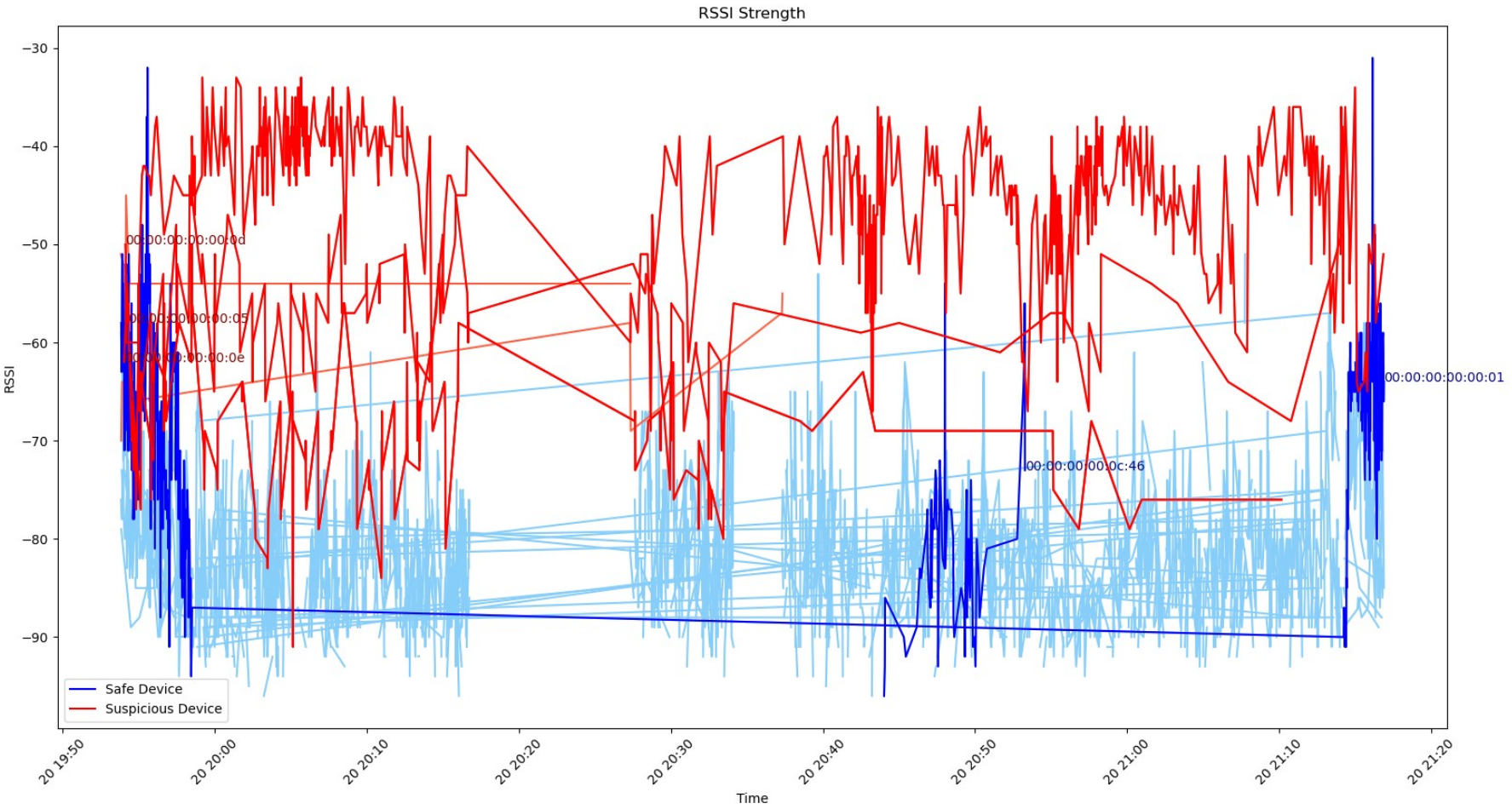
Plotting with different colors

```
130 #the colors were changed to be a lighter verson of the original colors
131     for i in range(len(macsafe)):
132         sorted_pairs = sorted(zip(timesafe[i], rssisafe[i]), key=lambda x: x[0])
133         timesafe_paired, rssisafe_paired = zip(*sorted_pairs)
134         plt.plot(timesafe_paired, rssisafe_paired, color="lightskyblue")
135
136 # specially plot significant devices with bold color
137     for i in range(len(labelsafe)):
138         sorted_pairs = sorted(zip(labelsafe_t[i], labelsafe_r[i]), key=lambda x: x[0])
139         timesafe_paired, rssisafe_paired = zip(*sorted_pairs)
140         plt.plot(timesafe_paired, rssisafe_paired, color = "blue")
141         plt.text(labelsafe_t[i][-1], labelsafe_r[i][-1], labelsafe[i], color="navy")
142
```

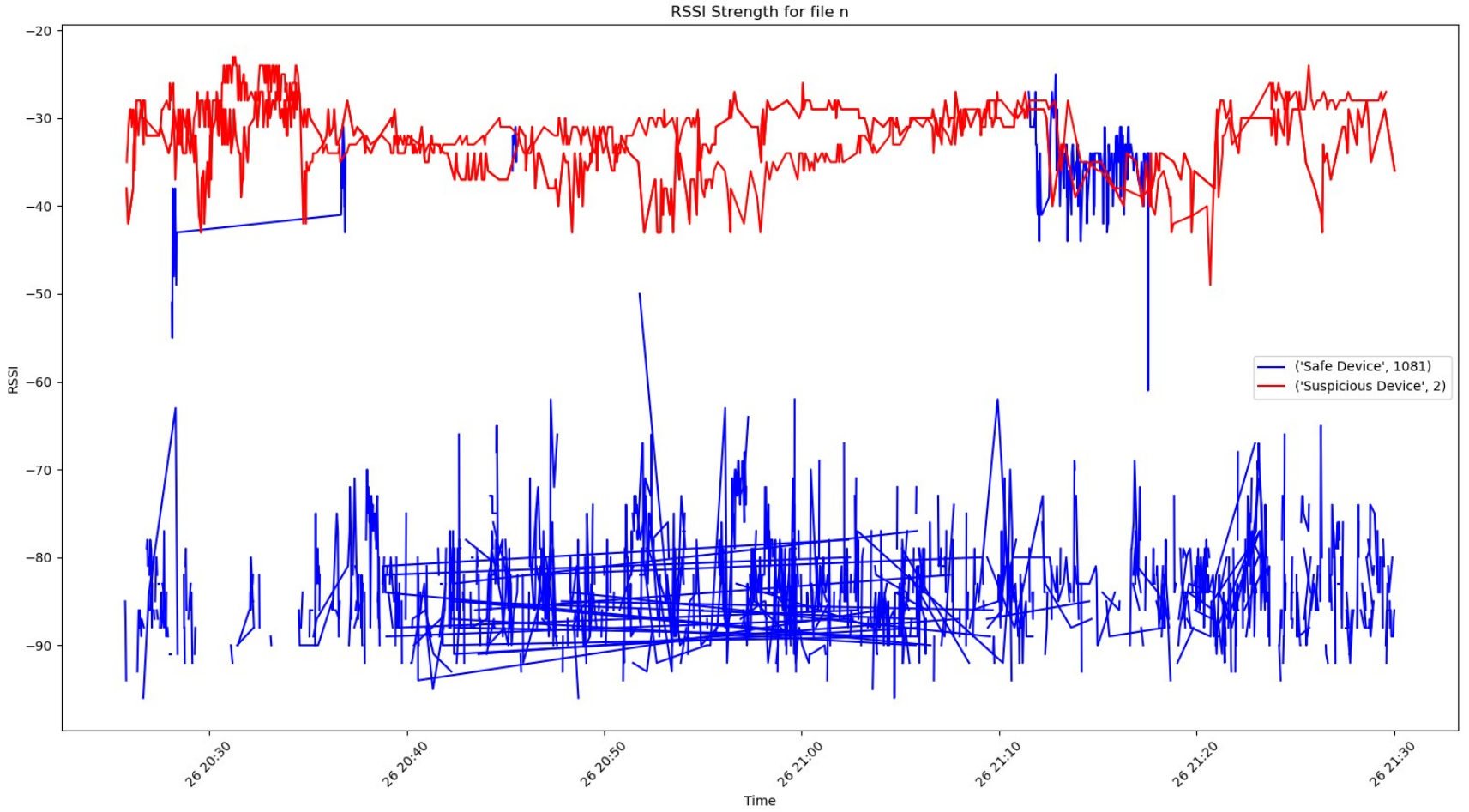
Data file c


RSSI Strength for file c
RSSI
−30
−40
−50
−60
−70
−80
−90
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)

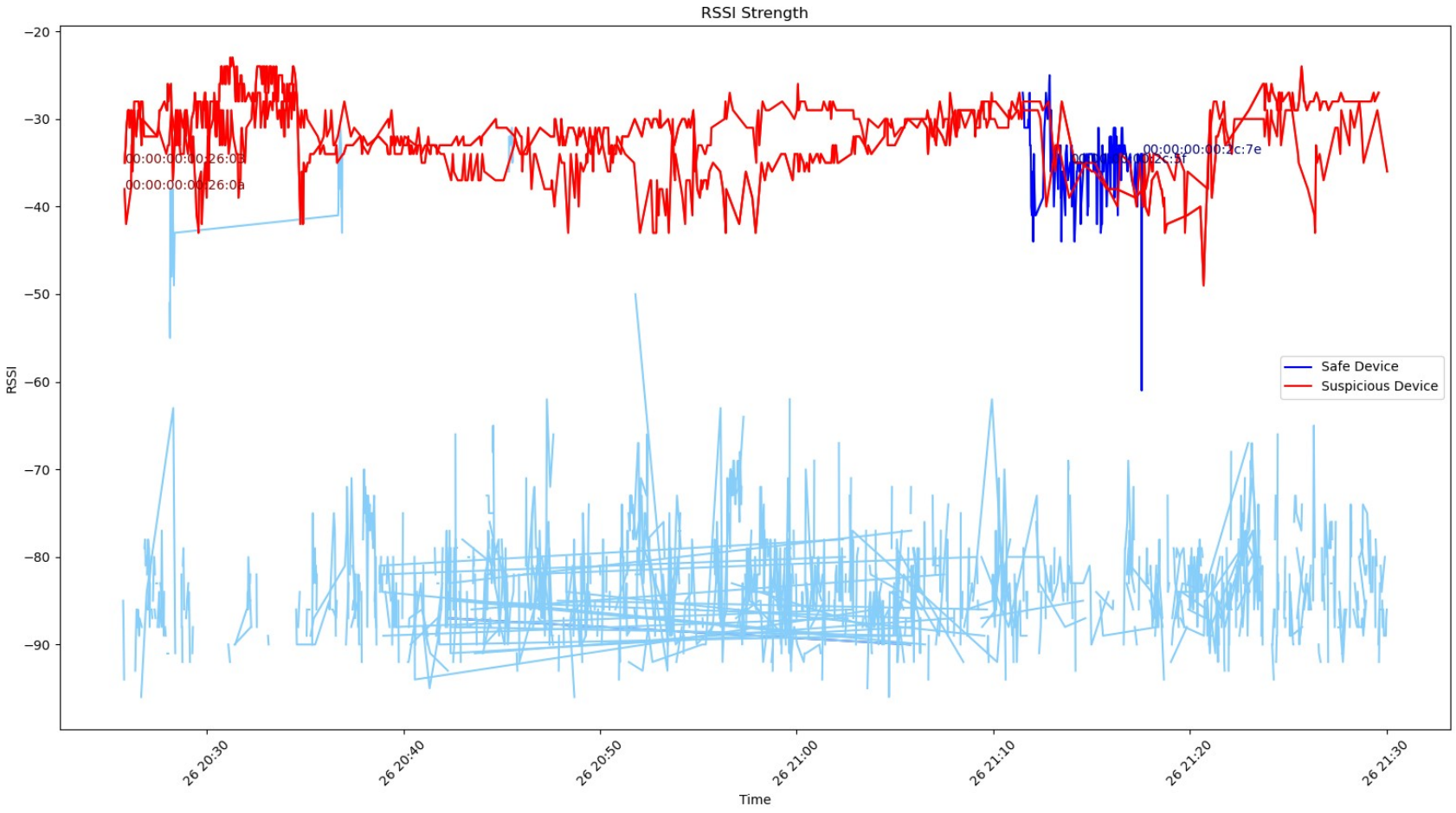
Data file c – long detection



Data file n


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

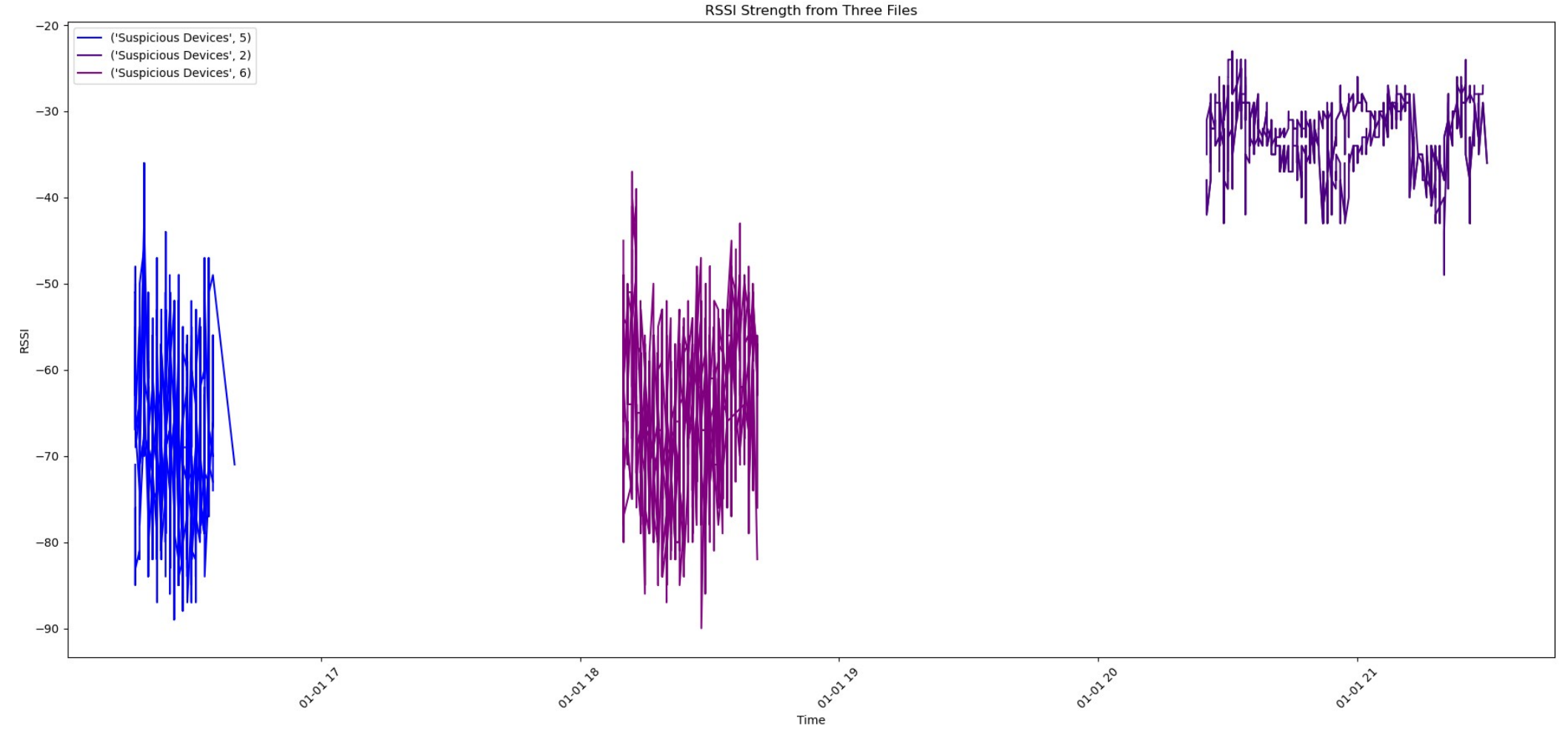
Data file n - long detection



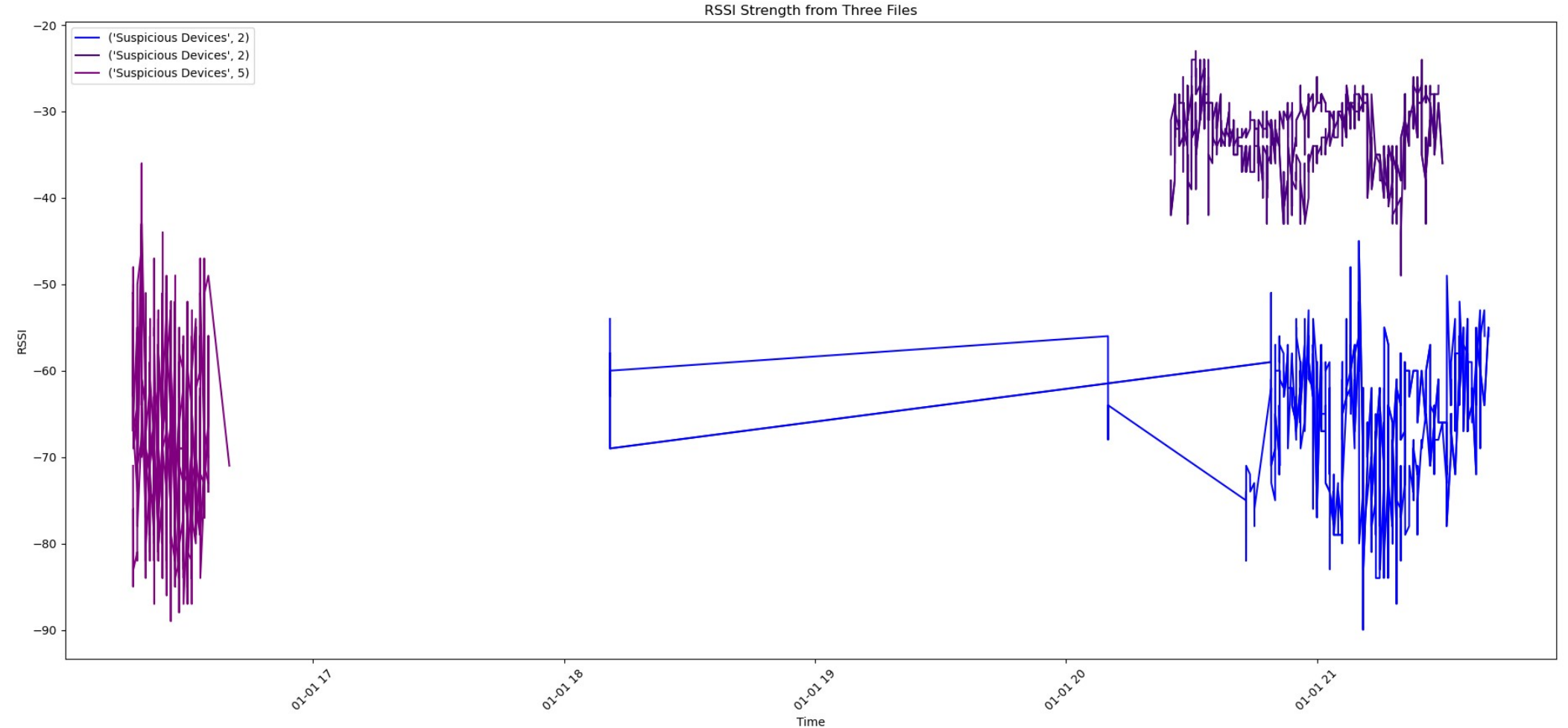
Time of Day

```
88  #note: change function to takes in variable
89  # new plot function to simplify main()
90  def plot(data, setsus, color1):
91      mac, rssi, time = make_rssi(data)
92      fmac, arssi, atime = sort(mac, rssi, time)
93      macsus, rssisus, timesus, = findsus(setsus, fmac, arssi, atime)
94
95      plt.plot(timesus[0], rssisus[0], color=color1, label=("Suspicious Devices", len(macsus)))
96
97      for i in range(len(macsus)):
98          sorted_pairs = sorted(zip(timesus[i], rssisus[i]), key=lambda x: x[0])
99          timesus_paired, rssisus_paired = zip(*sorted_pairs)
100         plt.plot(timesus_paired, rssisus_paired, color=color1)
```

Time of day – G,N,J



Time of day – G,N,A



Conclusion

Discussion