

《计算机网络第三次实验》

班级：信安 2302 班

学号：202308060227

姓名：石云博

目录

一. 问题描述.....	2
二. 问题分析.....	3
三. 实验过程及代码.....	3
四. 结论.....	14
五. 参考文献.....	14

一.问题描述

Assignment 3: Measure the WiFi hotspot

1. Write a program that can probe and record the WiFi hotspots around Yuelu Mountain, visualize them in a map, as many as possible

2. Measure/eavesdrop/estimate their network performance: Loss, delay, bandwidth, find their home AS

3. Probe their security modes, find the proportion of different security settings Talk about your methods and results, any interesting point?

二.问题分析

问题中要求我们探测岳麓山附近的热点情况,我们首先要知道怎么将手机上的热点情况记录下来,即怎么获取手机网络状态,Android 可以使用软件 TelephonyManager, iOS 可以使用软件 CoreTelephony 获取移动网络信息。

在 Linux 系统上,则可使用 nmcli 或 qmicli。

然后,我们要记录地址信息以在地图上显示化表述出来,这里首先想到的是 gps,但是网上的资料表明 wifi 和蓝牙定位都是可行的。

最后,我们要绘制信息地图,可以使用 Matplotlib 或 Foliu 在地图上绘制基站和网络模式。

对于这些网络的测试,测试丢包可以使用使用 ping 连续发送 ICMP 数据包,计算丢包,延时可以使用 ping 计算 RTT (Round Trip Time),带宽可以下载 Speedtest 来进行测试,要找到 Wi-Fi 热点 IP 的归属 AS,需要获取其外网 IP 并查询 ASN 信息,我们可以使用网站 <https://ipinfo.io/>来进行在线的查询。

最后,找出其安全模式,网上找资料,得到一般网络的安全模式有以下几种:

WPA3-Personal: 家庭 WiFi 网络的最佳安全设置, **WPA3-企业**: 企业的最佳安全设置, **WPA2 (AES)**: 第二好的安全设置,在更多路由器上可用, **WPA/WPA2-PSK (TKIP/AES)**: 对于旧设备网络的最佳安全设置,但在大多数路由器上不可用, **WPA2-PSK (TKIP)**: 仍然可用,但仅提供最低限度的安全性。

三.实验过程及代码

首先,我们要获取周围的 wifi 热点信息,可以使用 linux 中的 nmcli 也可以使用

windows 中自带的指令“netsh wlan show interfaces”。

我们首先使用 linux 中的 nmcli，使用 kali 打开：

```
(kali㉿kali)-[~/桌面]
$ nmcli -help
用法：nmcli [选项] 对象 { 命令 | help }

选项
-a, --ask                询问缺少的参数
-c, --colors auto|yes|no 是否在输出中使用颜色
-e, --escape yes|no      转义值中的列分隔符
-f, --fields <字段, ... >|all|common 指定要输出的字段
-g, --get-values <字段, ... >|all|common -m tabular -t -f 的快捷方式
-h, --help                打印此帮助
-m, --mode tabular|multiline 输出模式
-o, --overview            概览模式
-p, --pretty              美化输出
-s, --show-secrets        允许显示密码
-t, --terse               简介输出
-v, --version              显示程序版本
-w, --wait <秒数>         设定操作完成的等待超时

对象
g[eneral]                NetworkManager 的常规状态和操作
n[etworking]              整体网络控制
r[adio]                   NetworkManager 无线电开关
c[onnection]              NetworkManager 的连接
d[evice]                  NetworkManager 管理的设备
a[gent]                   NetworkManager 机密 (secret) 或 polkit 代理
m[onitor]                 监视 NetworkManager 更改
```

使用指令“nmcli device”查看当前设备的网络设施：

```
(kali㉿kali)-[~/桌面]
$ nmcli device
DEVICE      TYPE      STATE      CONNECTION
eth0         ethernet  已连接      Wired connection 1
lo           loopback  连接 (外部)  lo
wlan0        wifi      已断开      --
wlan1        wifi      已断开      --
p2p-dev-wlan0 wifi-p2p   已断开      --
p2p-dev-wlan1 wifi-p2p   已断开      --
hwsim0       unknown  未托管      --
```

我们需要用到 wlan0 中的 wifi，这里显示已断开，我们试着检查 wifi 状态，使用

指令“nmcli radio wifi”来查看：

```
(kali㉿kali)-[~/桌面]
$ nmcli radio wifi
enabled
```

我们发现 wifi 是打开了的，我们尝试查找附近 wifi，使用指令“nmcli device wifi list”：

```
(kali㉿kali)-[~/桌面]
$ nmcli device wifi list
IN-USE  BSSID  SSID  MODE  CHAN  RATE  SIGNAL  BARS  SECURITY
IN-USE  BSSID  SSID  MODE  CHAN  RATE  SIGNAL  BARS  SECURITY
```

可以看到还是没有结果。再检查 NetworkManager 是否运行：

```
(kali㉿kali)-[~/桌面]
$ systemctl status NetworkManager
● NetworkManager.service - Network Manager
   Loaded: loaded (/usr/lib/systemd/system/NetworkManager.service; enabled; preset: enabled)
   Active: active (running) since Fri 2025-03-14 15:35:01 CST; 2h 45min ago
  Invocation: e9af1766cbe94aeca9872d71b823a954
     Docs: man:NetworkManager(8)
    Main PID: 856 (NetworkManager)
      Tasks: 4 (limit: 4511)
    Memory: 14.9M (peak: 15.8M)
       CPU: 1.201s
    CGroup: /system.slice/NetworkManager.service
            └─856 /usr/sbin/NetworkManager --no-daemon

3月 14 18:15:48 kali NetworkManager[856]: <info> [1741947348.6289] device (p2p-dev-wlan0): supplicant management interface state>
3月 14 18:15:48 kali NetworkManager[856]: <info> [1741947348.6289] device (wlan1): supplicant interface state: inactive → inter>
3月 14 18:15:48 kali NetworkManager[856]: <info> [1741947348.6289] device (p2p-dev-wlan1): supplicant management interface state>
3月 14 18:15:48 kali NetworkManager[856]: <info> [1741947348.6297] device (wlan0): supplicant interface state: disconnected → i>
3月 14 18:15:48 kali NetworkManager[856]: <info> [1741947348.6298] device (p2p-dev-wlan0): supplicant management interface state>
3月 14 18:15:48 kali NetworkManager[856]: <info> [1741947348.6566] device (wlan1): supplicant interface state: interface_disable>
3月 14 18:15:48 kali NetworkManager[856]: <info> [1741947348.6566] device (p2p-dev-wlan1): supplicant management interface state>
3月 14 18:16:25 kali NetworkManager[856]: <info> [1741947385.1878] device (wlan1): supplicant interface state: disconnected → i>
3月 14 18:16:25 kali NetworkManager[856]: <info> [1741947385.1879] device (p2p-dev-wlan1): supplicant management interface state>
3月 14 18:19:03 kali NetworkManager[856]: <info> [1741947543.5169] dhcp4 (eth0): state changed new lease, address=192.168.219.130
lines 1-22/22 (END)
```

可以看到处于运行状态，试着重新加载 wifi 驱动，再重新打开 WiFi：

```
(kali㉿kali)-[~/桌面]
$ sudo modprobe -r iwlwifi && sudo modprobe iwlwifi

[sudo] kali 的密码：

(kali㉿kali)-[~/桌面]
$ nmcli radio wifi on

(kali㉿kali)-[~/桌面]
$ nmcli device
DEVICE          TYPE          STATE          CONNECTION
eth0            ethernet     已连接         Wired connection 1
lo              loopback     连接（外部）   lo
wlan0           wifi         已断开         --
wlan1           wifi         已断开         --
p2p-dev-wlan0   wifi-p2p     已断开         --
p2p-dev-wlan1   wifi-p2p     已断开         --
hwsim0          unknown     未托管         --
```

可以看到 wifi 还是处于断开状态。网上查原因，可能是当系统内无第三方网络管理工具（如 nm）时，系统默认使用 interfaces 文件内的参数进行网络配置。

当系统内安装了 nm 之后，nm 默认接管了系统的网络配置，使用 nm 自己的网络配置参数来进行配置。

没有办法，我们转用 windows 系统下的工具来探测网络。

在 windows 终端中使用指令“netsh wlan show networks mode=bssid”：

```
C:\Users\lian ton>netsh wlan show networks mode=bssid

接口名称 : WLAN
当前有 5 个网络可见。
```

可以看到我们这里得到了五个结果，每个结果里面包含了该 wifi 的具体信息：

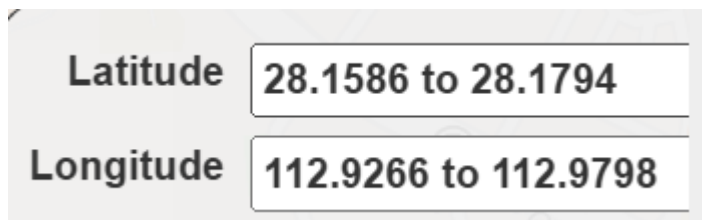
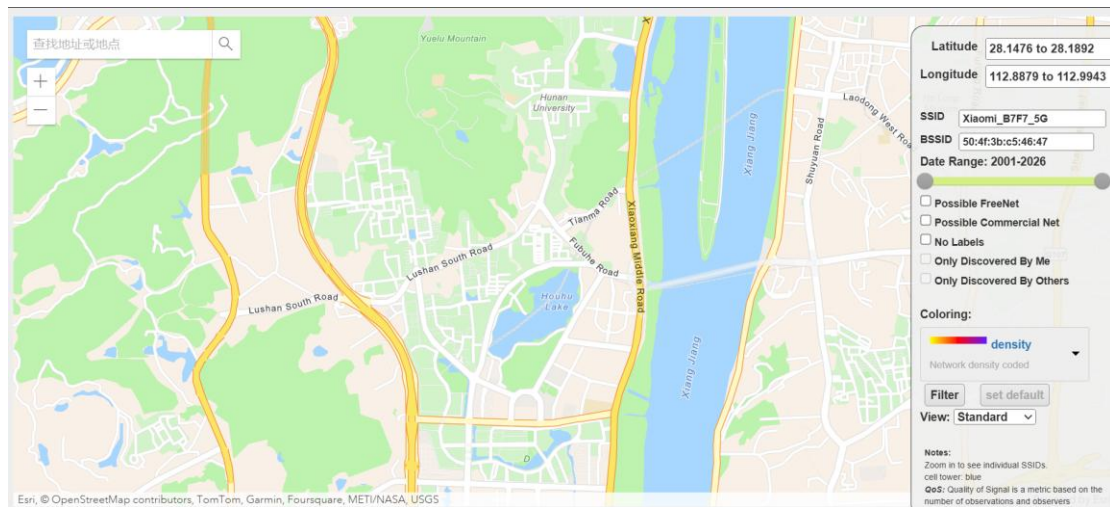
```
SSID 3 : Xiaomi_B7F7_5G
    Network type           : 结构
    身份验证               : WPA2 - 个人
    加密                   : CCMP
    BSSID 1                : 50:4f:3b:c5:46:47
        信号               : 14%
        无线电类型         : 802.11ax
        波段               : 5 GHz
        频道               : 44
        Bss 负载 :
            连接的电台 : 1
            频道利用率 : 9 (3% )
            中可用容量 : 31250 (1000000 us/s)
支持的 QoS MSCS : 0
    支持的 QoS 映射 : 0
    基本速率(Mbps) : 6 12 24
    其他速率(Mbps) : 9 18 36 48 54

SSID 4 : 3-11-714
    Network type           : 结构
    身份验证               : WPA2 - 个人
    加密                   : CCMP
    BSSID 1                : 48:8a:d2:ae:0d:33
        信号               : 81%
        无线电类型         : 802.11n
        波段               : 2.4 GHz
        频道               : 10
        Bss 负载 :
            连接的电台 : 1
            频道利用率 : 0 (0% )
            中可用容量 : 31250 (1000000 us/s)
支持的 QoS MSCS : 0
    支持的 QoS 映射 : 0
    基本速率(Mbps) : 1 2 5.5 11
    其他速率(Mbps) : 6 9 12 18 24 36 48 54
```

里面包含了 ssid，bssid，类型，加密，速率等等。

得到了这些信息之后，我们就要开始绘制信息地图了。这里我们可以用到一个地

图网站 <https://wicle.net/index>:



右边可以设置的信息中，我们可以输入 ssid 和 bssid 来获得其经纬度以及定位到大概的位置，可以看到图上大概就是岳麓山附近位置。

但是这里不好添加信息到地图中，而且最多只能使用五次，所以我转头使用 folium 画图。我们使用 python 来实现这个画图：

```
# 创建地图
map_object = folium.Map(location=[28.1716, 112.9444], zoom_start=15) # 默认中心点

for wifi_name, wifi_info in wifi_locations.items():
    popup_content = [f"<b>{wifi_name}</b><br>"
                    f"SSID: {wifi_info['SSID']}<br>"
                    f"BSSID: {wifi_info['BSSID']}<br>"
                    f"加密模式: {wifi_info['SECURITY']}<br>"
                    f"信号强度: {wifi_info['Signal']}<br>"]
    popup = folium.Popup(popup_content, max_width=300)

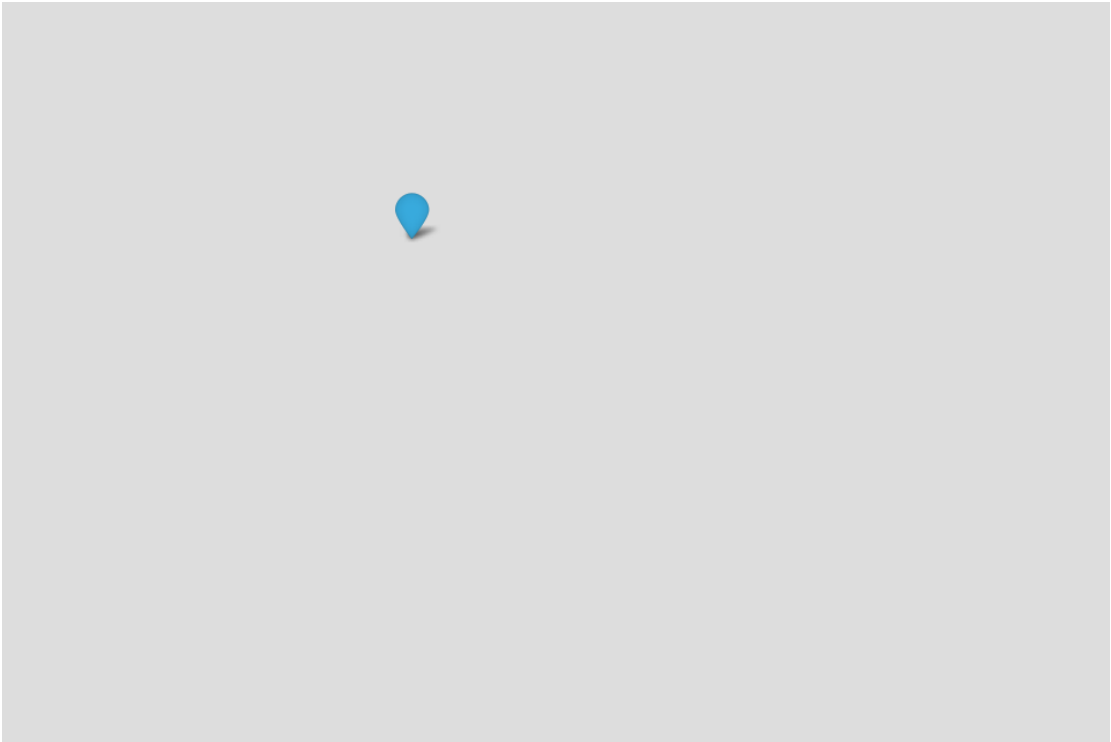
    folium.Marker(location=wifi_info["Location"], popup=popup).add_to(map_object)

# 保存地图
map_object.save("wifi_hotspots_map.html")
print("地图已保存为 wifi_hotspots_map.html")

# 运行
generate_map()
```

使用 python3 来运行该 py 文件，生成地图文件。

打开地图文件：



发现地图加载不出来，只有节点，查看浏览器的网络流量情况，发现有很多内容没有被正确地 get 到。

Status	Method	Domain	File	Initiator	Type	Transferred	Size	Headers	Cookies	Request	Response	Timings	Stack Trace
GET	tile.openstr...	13708.png	leaflet.js-5 (img)	NS_ERROR_CO...	0 B			Filter Headers					Block Resend
GET	tile.openstr...	13708.png	leaflet.js-5 (img)	NS_ERROR_CO...	0 B								
GET	tile.openstr...	13711.png	leaflet.js-5 (img)	NS_ERROR_CO...	0 B								
GET	tile.openstr...	13711.png	leaflet.js-5 (img)	NS_ERROR_CO...	0 B			GET https://tile.openstreetmap.org/15/2666/13711.png					
200	GET	cdn.jsdelivr...	marker-icon.png	leaflet.js-5 (img)	png	cached	1.47 kB	Transferred	0 B (0 B size)				
200	GET	cdn.jsdelivr...	marker-shadow.png	leaflet.js-5 (img)	png	cached	618 B	Referer Policy	strict-origin-when-cross-origin				
								Request Priority	Low				
								DNS Resolution	System				
79 requests, 25.70 kB / 0 B transferred, Finish: 41.76 s, load: 1.24 min													
Request Headers (472 B)													

把这个问题反馈给 chatgpt，得到答案“如果 tile.openstreetmap.org 速度慢或被屏蔽，可在 folium（或其他地图库）中更换为其他 Tile Provider。例如使用 openstreetmap.fr 镜像”，于是我们在原有代码中加上

```
tiles="https://{s}.tile.openstreetmap.fr/hot/{z}/{x}/{y}.png",
```


```
attr="OpenStreetMap FR"
```

再次生成地图文件，打开：



第二部分：测量/窃听/评估他们的网络性能：丢包、延迟、带宽、找到他们的归属 AS

测试丢包可以使用使用 ping 连续发送 ICMP 数据包，计算丢包，延时可以使用 ping 计算 RTT (Round Trip Time)，带宽可以下载 Speedtest 来进行测试，要找到 Wi-Fi 热点 IP 的归属 AS，需要获取其外网 IP 并查询 ASN 信息，我们可以使用网站 <https://ipinfo.io/>来进行在线的查询：


Products Solutions Why IPinfo? Pricing Resources Docs Login [Sign up](#)

My IP or

Search

The Trusted Source For IP Address Data

Accurate IP address data that keeps pace with secure, specific, and forward-looking use cases.

[Sign up for free](#)
[Contact sales](#)

```

{
  "country": "US",
  "loc": "39.9523,-75.1638",
  "postal": "19187",
  "timezone": "America/New_York",
  "asn": {
    "asn": "AS12167",
    "name": "LightWave Networks",
    "domain": "lightwavenetworks.com",
    "route": "107.161.144.0/23",
    "type": "hosting"
  },
  "company": {
    "name": "LightWave Networks",
    "domain": "lightwavenetworks.com",
    "type": " "
  }
}

```

对于延迟和丢包，我选择了使用软件 winMTR 来进行测试。首先我们电脑连接要测试的对象，然后对于 host: 8.8.8.8 来进行测试，得到如下结果：

Host:

Stop

Options

Exit

Copy Text to clipboard

Copy HTML to clipboard

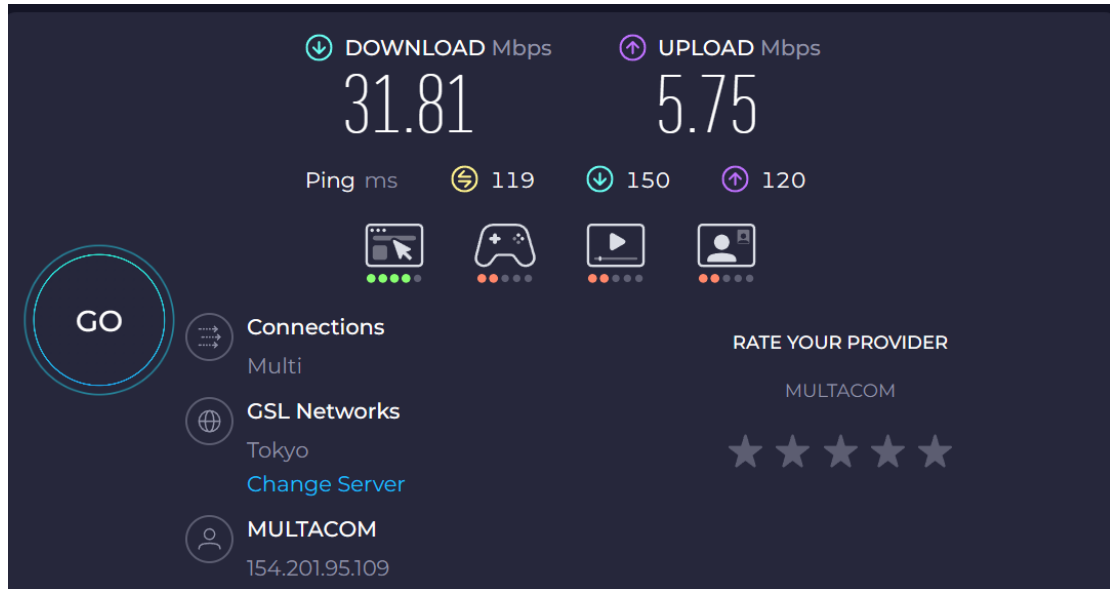
Export TEXT

Export HTML

Hostname	Nr	Loss %	Sent	Recv	Best	Avg	Worst	Last
192.168.26.163	1	1	278	277	3	15	142	9
10.68.0.1	2	6	233	221	6	26	187	13
10.62.255.250	3	70	75	23	0	35	157	31
210.43.96.121	4	1	278	277	7	30	212	20
10.1.1.9	5	1	274	272	7	34	480	11
222.244.139.1	6	82	66	12	0	27	78	32
61.187.30.201	7	82	66	12	0	39	117	31
No response from host	8	100	56	0	0	0	0	0
61.137.7.249	9	89	62	7	0	33	79	79
No response from host	10	100	56	0	0	0	0	0
202.97.57.157	11	49	95	49	31	44	106	37
202.97.33.154	12	9	211	193	28	46	294	55
202.97.6.6	13	7	223	208	41	62	357	46

我们可以看到对于只有一条的情况，也就是 Nr=1，其丢包率为百分之一，最低延迟 3，最高延迟 143，平均延迟为 15.

接下来测试带宽，我们选择在 Speedtest 网站上直接进行在线测速，得到结果：



经该网站测试，下载带宽为 31.81Mbps，上载带宽为 5.75Mbps。

测试所属 AS：

我们首先找到自己的公网 ip，windows 下可以使用指令“nslookup myip.opendns.com resolver1.opendns.com”来得到：

```
C:\Users\lian ton>nslookup myip.opendns.com resolver1.opendns.com
服务器: dns.sse.cisco.com
Address: 208.67.222.222

非权威应答:
DNS request timed out.
    timeout was 2 seconds.
名称: myip.opendns.com
Address: 222.244.139.196
```

下面的 address 后面的数字就是我们的公网 ip，将其复制之后到网站

<https://ipinfo.io/>中进行查找：

[Products](#)
[Solutions](#)
[Why IPinfo?](#)
[Pricing](#)
[Resources](#)
[Docs](#)
[Login](#)
[Sign up](#)

222.244.139.196

Shanghai, Shanghai, China

Need more data or want to access it via API or data downloads? Sign up to get free access
 [Sign up for free](#)

Summary

Geolocation

Privacy

ASN

Company

Abuse

Summary

ASN	AS4134 - CHINANET-BACKBONE
Hostname	No Hostname
Range	222.240.0.0/13
Company	CHINANET Hunan province network
Hosted domains	0
Privacy	False
Anycast	False
ASN type	ISP
Abuse contact	anti-spam@chinatelecom.cn

我们可以看到，ASN 为 AS4134-CHINANET-BACKBONE

Company 是 CHINANET Hunan province network。

最后一个部分：探究其网络模式，找出不同安全模式的比例。因为我们第一个任务中测试网络时，我们将测得的网络信息存在了“wifi.txt”文件中，所以我们可以直接从里面统计出不同安全模式的比例。

我们使用一个简单的 python 程序来进行统计：

```

with open("wifi.txt", "r", encoding="utf-8") as f:
    for line in f:
        # 每行形如 "SSID: wifi_1, Security: WPA2-个人"
        if "Security:" in line:
            parts = line.split("Security:")
            if len(parts) >= 2:
                sec = parts[1].strip()
                if sec in security_counts:
                    security_counts[sec] += 1

print("安全模式统计结果 :", security_counts)

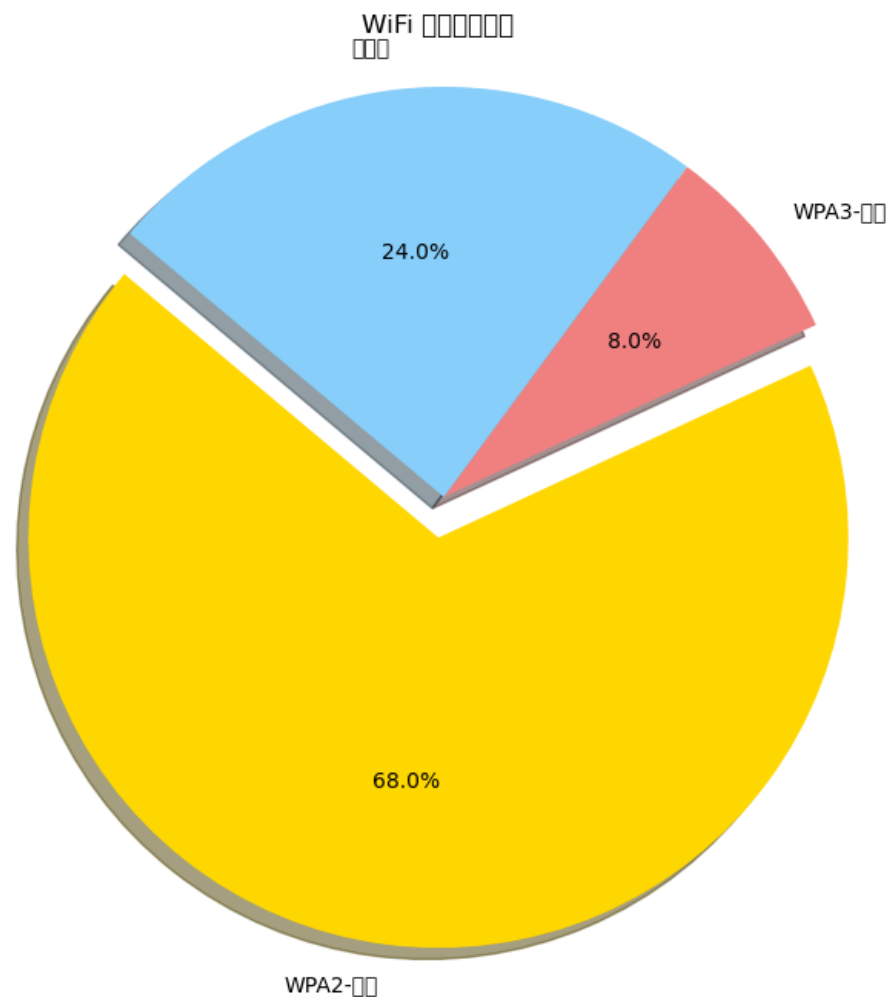
```



刚开始生成的比例图中，没有办法显示中文，出现了错误

“/usr/lib/python3.13/tkinter/__init__.py:865: UserWarning: Glyph 23433 (\N{CJK UNIFIED IDEOGRAPH-5B89}) missing from current font.

func(*args)”表示 Tkinter 正在尝试显示一个当前字体中没有的字符，即 Unicode 字符“安”（CJK UNIFIED IDEOGRAPH-5B89）。这通常发生在使用默认字体时，该字体不支持某些中文字符。我找了很多攻略试了很多次都没有办法完全消除这个问题，故最终结果只能这样表示了：



蓝色的是开放式的网络，黄色是 WPA2-个人，红色的是 WPA3-个人。可以看到 WPA2-个人最多，而 WPA3-个人最少，开放式居中。

四.结论

我首先对所要解决的问题进行了分析,通过查阅资料和对题目中安全模式等方面的研究,提出了解决思路。具体而言,我在 Windows 环境下利用 Python 脚本调用“netsh wlan show networks mode=bssid”命令,收集了周边个 WiFi 热点信息。随后,我查询了 WIGLE 数据库,并使用统计图呈现了本小区周边的 WiFi 热点分布情况。接着,通过 Python 脚本将这些 WiFi 热点的经纬度信息生成了一个 HTML 文件,能够直观展示各热点的地理位置。

在对这些 WiFi 热点进行分析后发现,除了主流的 WPA2 安全模式外,WPA3 和开放式网络也同时存在,并且它们的延迟和带宽特性各有不同。总体来看,所收集的中国电信 WiFi 热点主要采用“WPA2-个人”、“WPA3-个人”以及“开放式”三种安全模式。通过这些结果,我对热点分布和安全模式的规律进行了探讨。

五.参考文献

1 <https://blog.csdn.net/coc61987/article/details/106068658>, Matlab2019 中文显示问题(乱码与方框). FJUR. 2020.5.26.

[2] <https://www.cnblogs.com/surt/p/15601654.html>. WLAN 安全策略-WEP、WPA/WPA2、WPA3. 贪知猪. 2023.03.22.

[3] <https://zhuanlan.zhihu.com/p/681806463>. WiFi 测试的核心思路 and 主要工具. Ankie Wan.2021.01.24.

[4] <https://blog.csdn.net/HGJ515/article/details/115914249>. 测试局域网或 wifi 实际最大带宽. 请叫我阿进.2023.8.15.

- [5] https://blog.csdn.net/2409_89014517/article/details/144028757. 使用 WinMTR 软件简单分析跟踪检测网络路由情况. 网硕互联的小客服.2022.3.24
- [6] https://blog.csdn.net/python2021_/article/details/123652555. Python 绘制地图神器 folium 介绍及安装使用教程. python2021_.2022.03.22.
- [7] <https://zhuanlan.zhihu.com/p/604459423>. 彩云之南. 测试 WiFi 信号、选用 WiFi 信道的 App: WiFi Analyzer. 2022.8.15.