Wi-Jacking：无需破解就可以连接邻居的WiFi

angel010 / 2018-09-12 00:59:54 / 浏览数 4313 技术文章 技术文章 顶(0) 踩(0)

本文翻译自：https://www.surecloud.com/sc-blog/wifi-hijacking

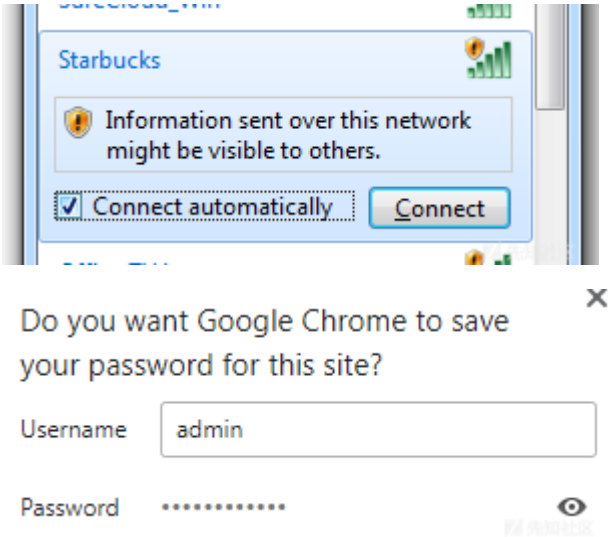
Wi-Jacking是一种滥用by design的行为，可以在无需破解握手信息的情况下，获取凭证信息，攻击上百万家用WiFi网络。

简介

研究人员最近发现一个浏览器行为交互和几乎所有家用路由器的漏洞。浏览器的行为于保存的凭证是相关的。如果凭证保存在浏览器中，其中的凭证会与URL相关联，当再次

结合这两个组件，就可能在不破解握手协议的情况下获取不同网络的访问权限（当前破解WPA/WPA2网络最常用的方法就是破解握手）。该攻击在大多数网络上都使用，但

- 目标攻击网络上必须要用活动的客户端设备；
- 客户端设备必须连接过其他开放网络并允许自动重连；
- 客户端设备应该使用基于Chromium的浏览器，比如Chrome和Opera；
- 客户端设备应该在浏览器中记住（保存）过路由器管理接口凭证；
- 目标网络路由器管理接口必须配置为使用非加密的HTTP。



没有这5个先决条件，攻击是不可能实现的。但大多数浏览器都会建议用户自动保存凭证。降低这种可能性的主要先决条件是使用Chromium和保存路由器凭证，但仍会影响

Firefox、IE/Edge和Safari浏览器需要明显的用户交互，所以攻击也可以实现，但主要是基于社会工程的攻击。如果路由器的管理接口凭证未保存，仍然可以尝试猜测默认凭

虽然攻击主要针对的是可以从web接口直接提取WiFi密钥的家用路由器，但其他设备也有可能成为攻击目标。因为使用的是非加密的HTTP，所以可以对URL进行预测。理论

在正式讲述攻击前，需要对Karma/Jassager攻击有所了解。

KARMA是在多层次评估无线客户端安全性的工具集。无线嗅探工具被动地通过802.11探测请求帧发现客户和他们的首选/可信网络。然后，恶意攻击者可以通过创建一个非

Karma就是通过捕抓无线客户端的带有ESSID的Probe Qequest，然后模拟相关的ESSID，从而使得无线客户端连接到虚假的AP当中，然后进行后续的攻击。

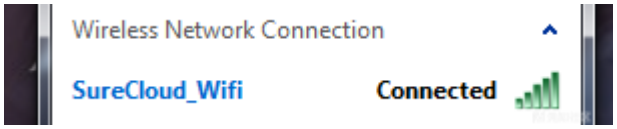
Jasager基于KARMA的一个Linux固件。它可以在多数接入点与Atheros的无线卡上运行，提供一组Linux工具来发现无线客户端的安全漏洞，与WiFish Finder类似，但最大的区别是可以被用于进行无线蜜罐攻击。Jasager可以运行在FON或者WiFi Pineapple路由器上。它能配置软AP，生成附近无线客户端搜索的SSID，同时还能向无线客户端提供DHCP、DNS、HTTP服务。其中HTTP服务器可以讲网络访问请求导向特定网站。

关于Karma攻击的信息参见：
https://wiki.wifipineapple.com/legacy/#!karma.md

Wi-Jacking攻击

Step 1. 将设备接入研究人员控制的网络:

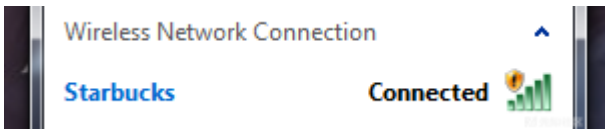
首先用aireplay-ng发送deauthentication请求，并用hostapd-wpe发起Karma攻击，使用的无线网卡为Alfa AWUS036NHA。



连接到家用WiFi

```
root@sc-elliott:~# aireplay-ng -0 0 -a 2C:FD:A1:0E:2C:A0 -c 00:23:6C:8B:1C:F4 wlan1
01:54:22  Waiting for beacon frame (BSSID: 2C:FD:A1:0E:2C:A0) on channel 13
01:54:23  Sending 64 directed DeAuth (code 7). STMAC: [00:23:6C:8B:1C:F4] [46|70 ACKs]
01:54:24  Sending 64 directed DeAuth (code 7). STMAC: [00:23:6C:8B:1C:F4] [45|66 ACKs]
01:54:24  Sending 64 directed DeAuth (code 7). STMAC: [00:23:6C:8B:1C:F4] [42|69 ACKs]
01:54:26  Sending 64 directed DeAuth (code 7). STMAC: [00:23:6C:8B:1C:F4] [65|64 ACKs]
01:54:26  Sending 64 directed DeAuth (code 7). STMAC: [00:23:6C:8B:1C:F4] [54|63 ACKs]
01:54:27  Sending 64 directed DeAuth (code 7). STMAC: [00:23:6C:8B:1C:F4] [44|68 ACKs]
01:54:28  Sending 64 directed DeAuth (code 7). STMAC: [00:23:6C:8B:1C:F4] [43|70 ACKs]
01:54:29  Sending 64 direct^C DeAuth (code 7). STMAC: [00:23:6C:8B:1C:F4] [12|22 ACKs]
root@sc-elliott:~#
```


deauth攻击



连接到开放网络

Step 2. 触发浏览器加载URL:

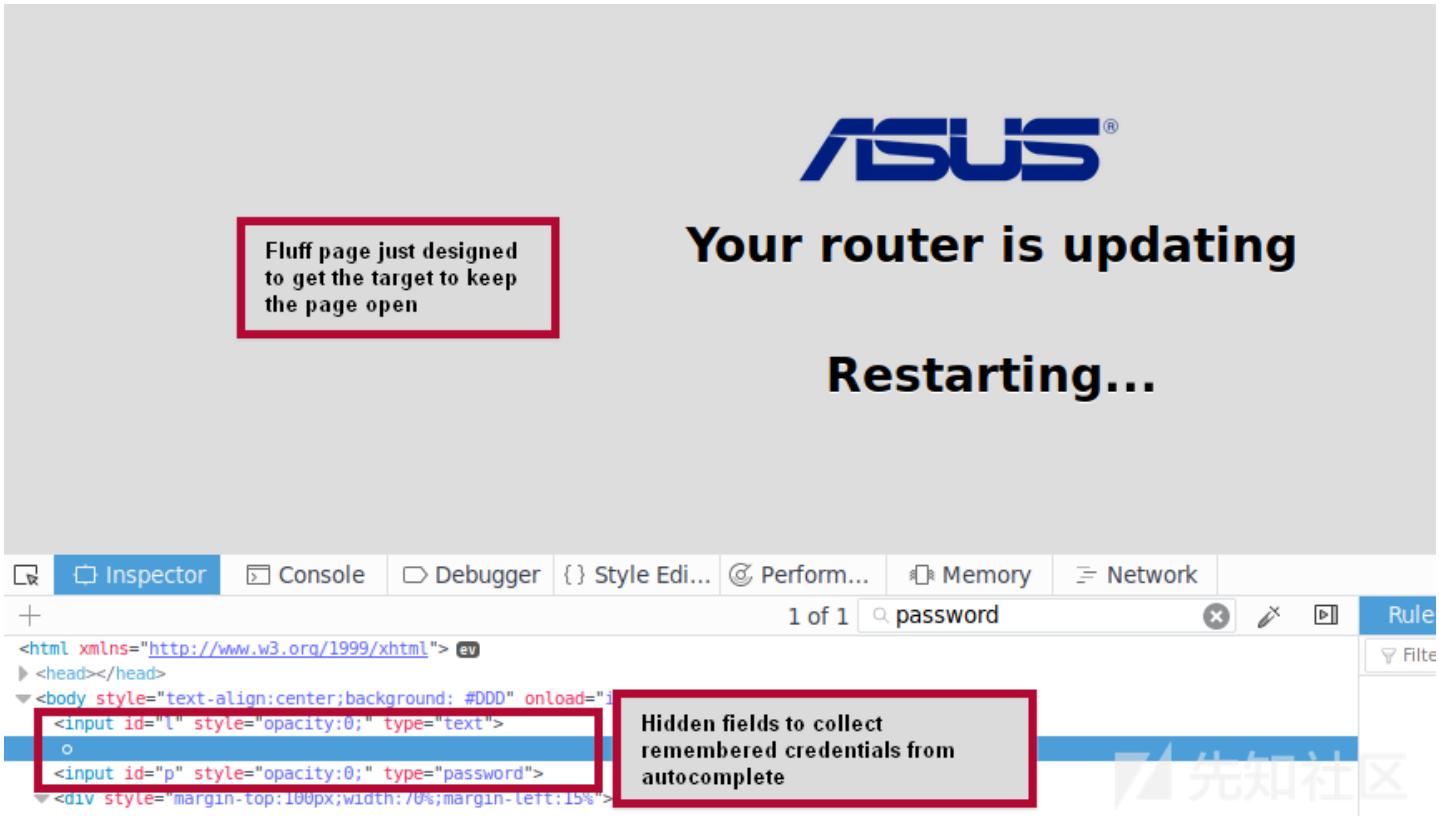
用dnsmasq和python脚本完成触发动作。当看到HTTP请求时，就会创建一个重定向到研究人员URL的响应和服务的主页。

根据攻击的路由器的不同，对应的URL和服务的主页也是不同的。研究人员可以可以基于BSSID和ESSID或者纯猜测来检测发送哪个URL和主页面，但可选范围也不是无限的

重定向有很多额外的选项。默认情况下，允许所有HTTPS通过，并等待HTTP请求。但如果等待时间太长，触发了Windows上的WiFi captive portal detection，就会自动启动默认浏览器并打开研究人员指定的URL。但触发captive portal也有一定的限制，尤其是在MacOS中，启动的是一个处理captive portal的隔离的浏览器，防止研究人员获取保存的凭证。

```
root@sc-elliott:Captive# ./5-portal.py
ASUS payload chosen based on BSSID
 * Serving Flask app "5-portal" (lazy loading)
 * Environment: production
   WARNING: Do not use the development server in a production environment.
   Use a production WSGI server instead.
 * Debug mode: on
 * Running on http://0.0.0.0:80/ (Press CTRL+C to quit)
 * Restarting with inotify reloader
ASUS payload chosen based on BSSID
 * Debugger is active!
```


portal flask app



wifi credential capturing page

Step 3. 窃取自动填充凭证：

当页面加载时，浏览器会进行两项检查：

- URL源是否与路由器的admin接口源（协议、IP地址、hostname）匹配；
- 页面上的input域是否与浏览器记住的路由器接口一致。

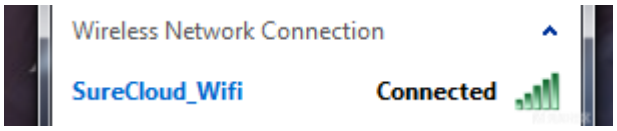
如果这两项检查都通过，浏览器就会自动用保存的凭证填充到页面中。在这里是有路由器admin的详细情况的，但一般情况下input域是完全隐藏的。

如果目标使用的Chrome浏览器，还需要进行额外的一个步骤：Chromium的`PasswordValueGatekeeper`特征需要用户与页面进行一定形式的交互。比如在页面上的任何

如果目标使用的Firefox、Internet
Explorer、Safari或Edge，这些input域是无法隐藏的。如果目标点击了form域并从下拉菜单中选择了保存的凭证，那么攻击也是有效的。这样的话，攻击是需要一些社会工

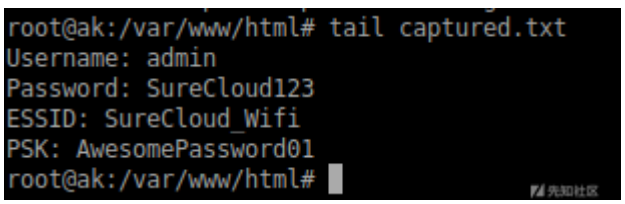
Step 4. 发送目标到家用WiFi

获取凭证后，还是希望页面打开的时间能够长一点。此时，停止Karma攻击，将目标释放会原来的网络中。



connected to home wifi

一旦目标设备成功连接回原来的网络，位于路由器管理接口源的主页就在JS中加载管理凭证。然后用XMLHttpRequest登陆并获取PSK，还可以改变其他配置。研究人员在测
PSK，而不需要获取到网络的握手信息。但如果路由器隐藏了密钥，研究人员还可以用已知的密钥来开启WPS，在路由器的接口中创建新的AP。



credentials captured

攻击使用的工具

除了路由器特定的payload和选择脚本外，模拟攻击中所使用的所有工具都是标准Kali组件。脚本地址如下：
https://gitlab.com/eth01/Wi-Jacking-PoC

总结

目前的攻击仍处于PoC阶段，但毫无疑问，现实的攻击会很快出现。长期目标是构建一个WiFi pineapple模块来自动化执行攻击。

上一篇：noxCTF部分web题writeup 下一篇：代码审计之某汽车网源码
1. 0 条回复
   - 动动手指，沙发就是你的了！