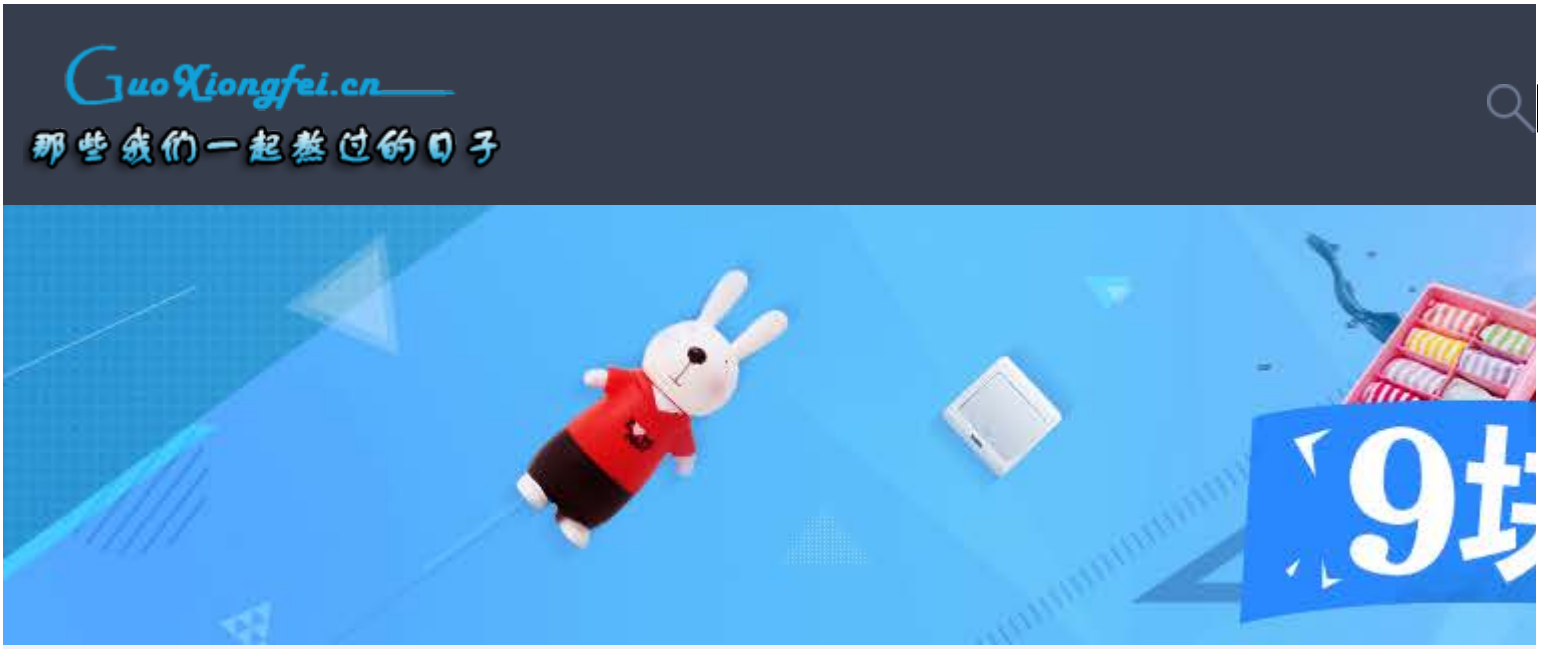
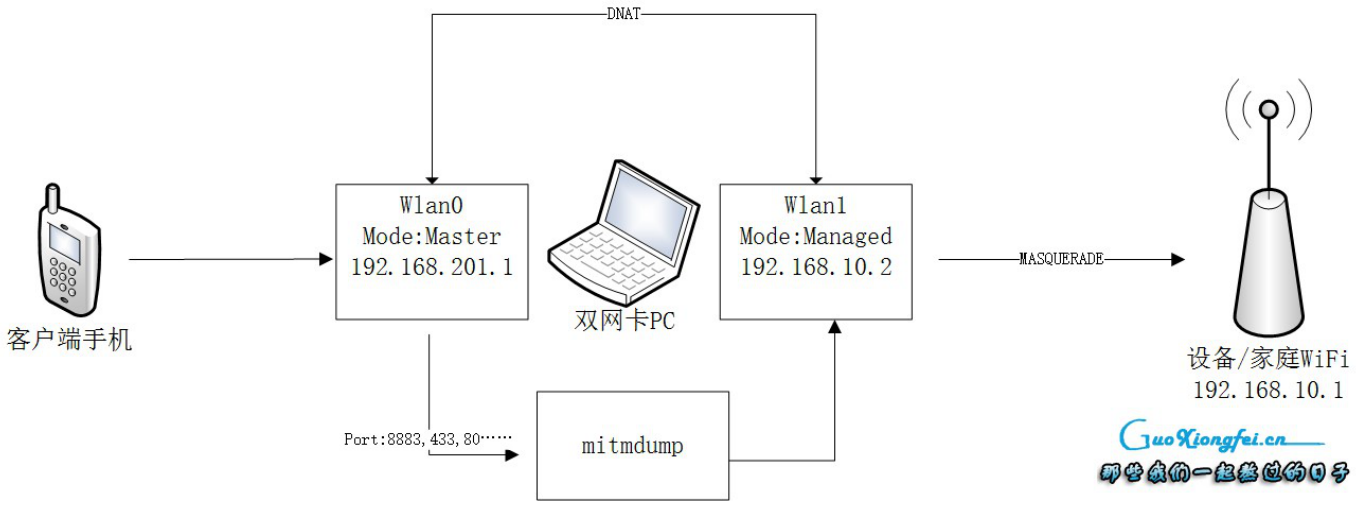


双网卡+mitmproxy+iptables搭建SSL中间人（支持非HTTPS协议）

作者：www.GuoXiongfei.cn / 时间：2018-08-31 21:39:04 / 浏览：93,942次 /

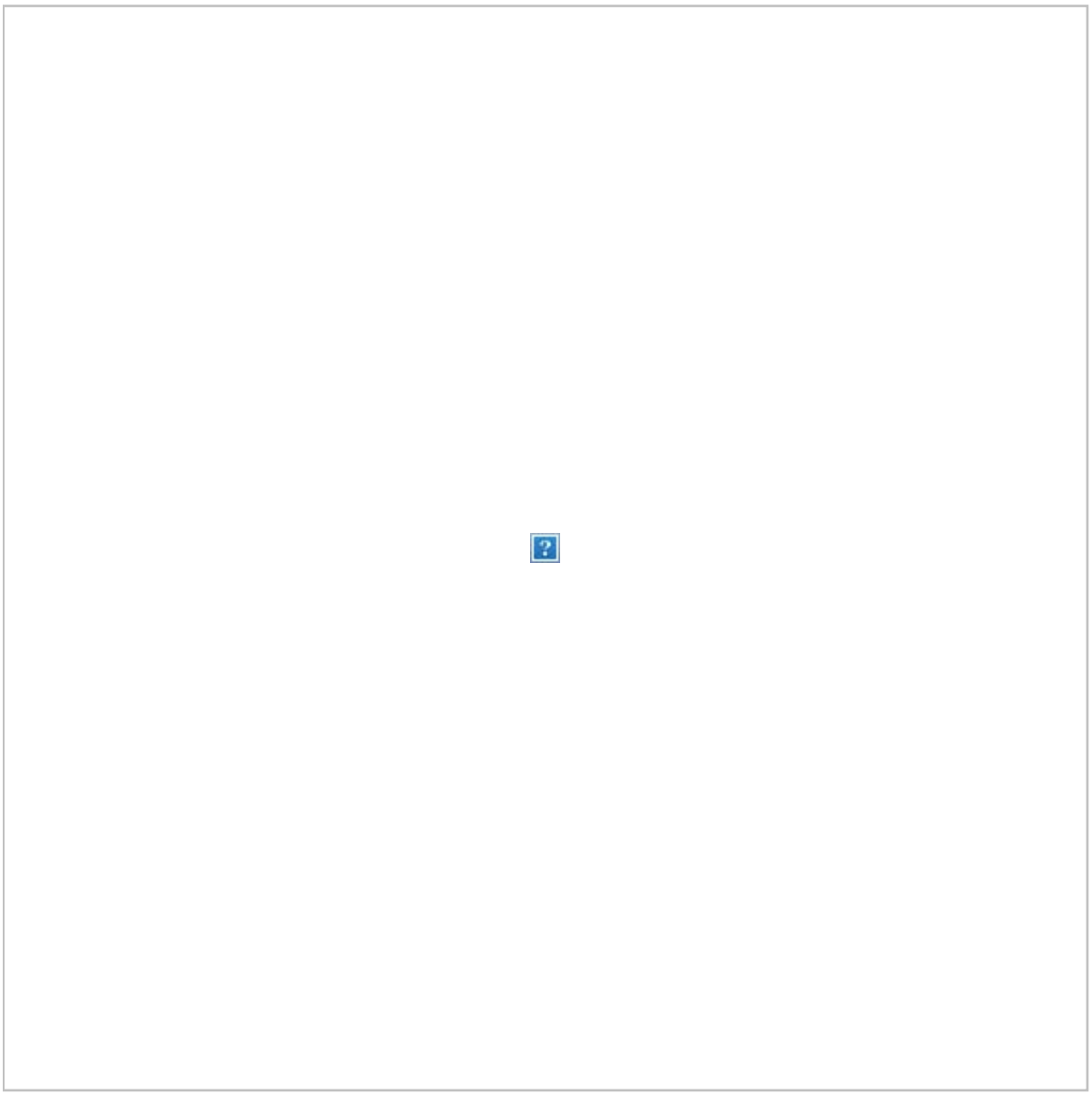
> “想要解决一个问题，最根本方法的就是了解这一切是如何工作的，而不是玄学。”——ASCII0X03最近学习发现现在很多现成的安卓SSL中间人工具和教程都只针对HTTPS流量，比如注册为安卓VPN的Packetcapture，以及设置http代理来抓包的Fiddler。他们对于分析http流量都很好，但是遇到局域网通信或者非HTTPS流量如MQTT，就无能为力了。因此，特利用双网卡和mitmproxy...



“想要解决一个问题，最根本方法的就是了解这一切是如何工作的，而不是玄学。”

——ASCII0X03

最近学习发现现在很多现成的安卓SSL中间人工具和教程都只针对HTTPS流量，比如注册为安卓VPN的Packetcapture，以及设置http代理来抓包的Fiddler。他们对于分析http流量都很好，但是遇到局域网通信或者非HTTPS流量如MQTT，就无能为力了。因此，特利用双网卡和mitmproxy神器（叫做神器是有原因的，这应该是做中间人最好的工具了，可满足各种需求）搭建了一个支持SSL上的MQTT协议的中间人环境，基本框架如下图。



注意，此方案选用俩网卡自己当网关是基于如下考虑：手机设置代理的方法不是所有app的所有流量都会使用，ARP欺骗不稳定，并且有的设备自己做AP热点根本就不转发任何其他流量。当然，还是需要Xpose或者重打包来绕过App上的SSL pinning。鄙人认为最好的方法还是在安卓端写一个类似于Packetcapture的VPN软件配合Xpose Hook并支持各种端口的协议（不太好的一

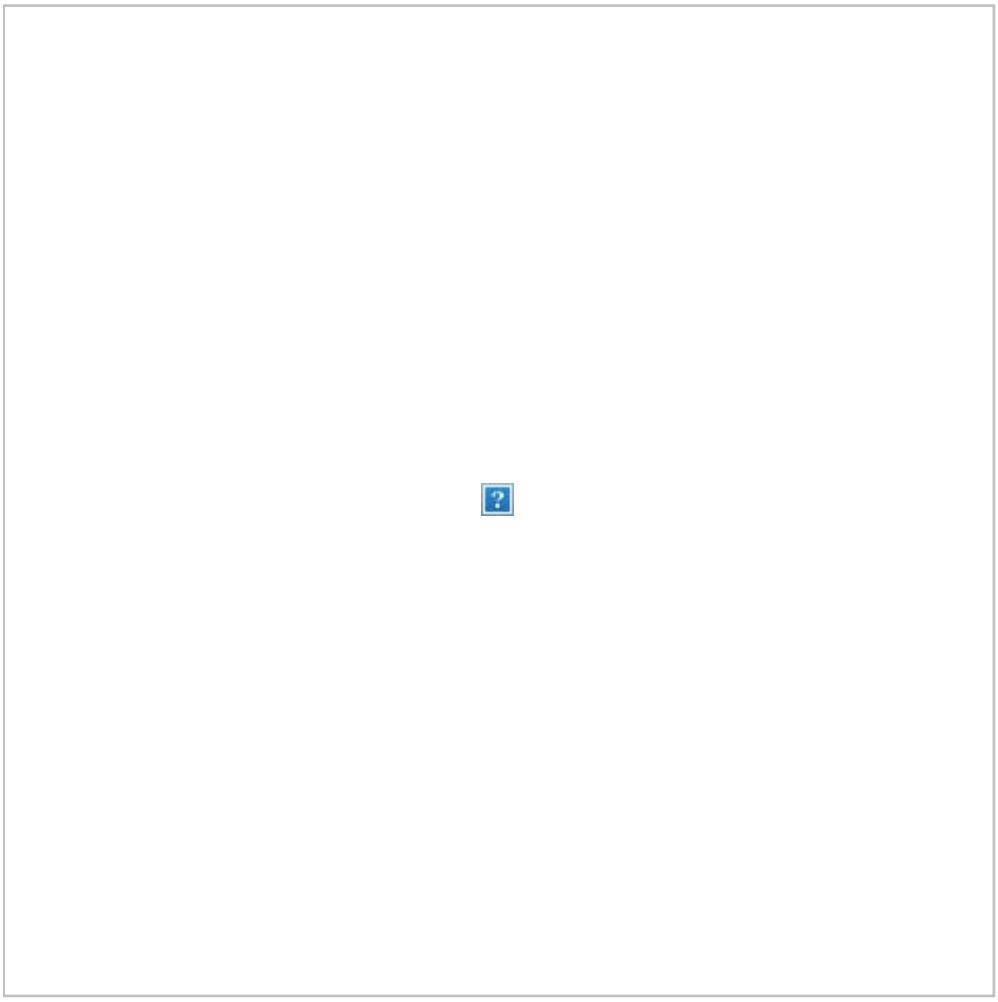
个参考：https://blog.csdn.net/XXOOYC/article/details/78223242），奈何太菜不懂安卓，哪位达人有兴趣可以git 来一发。

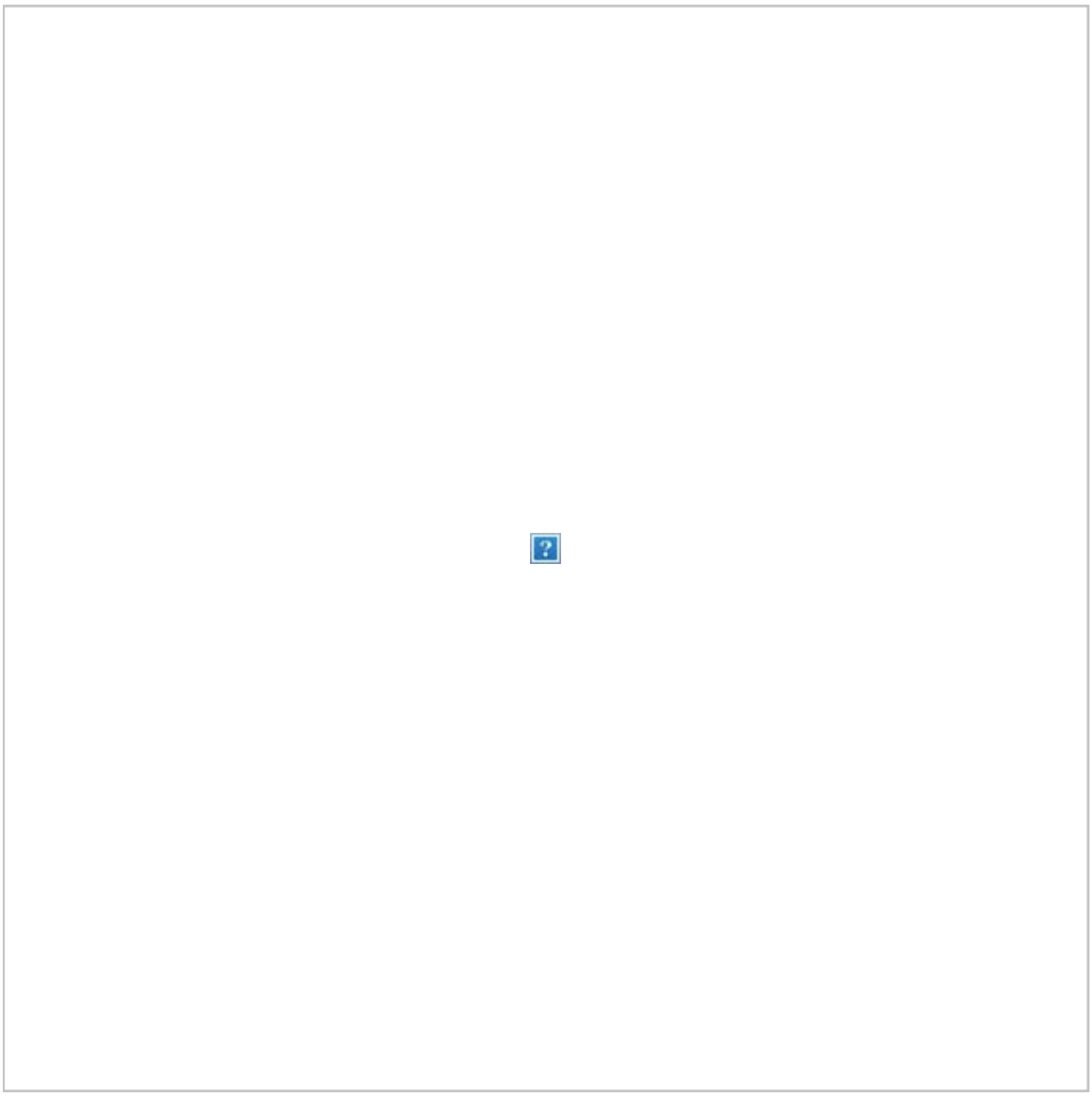
一、基础知识

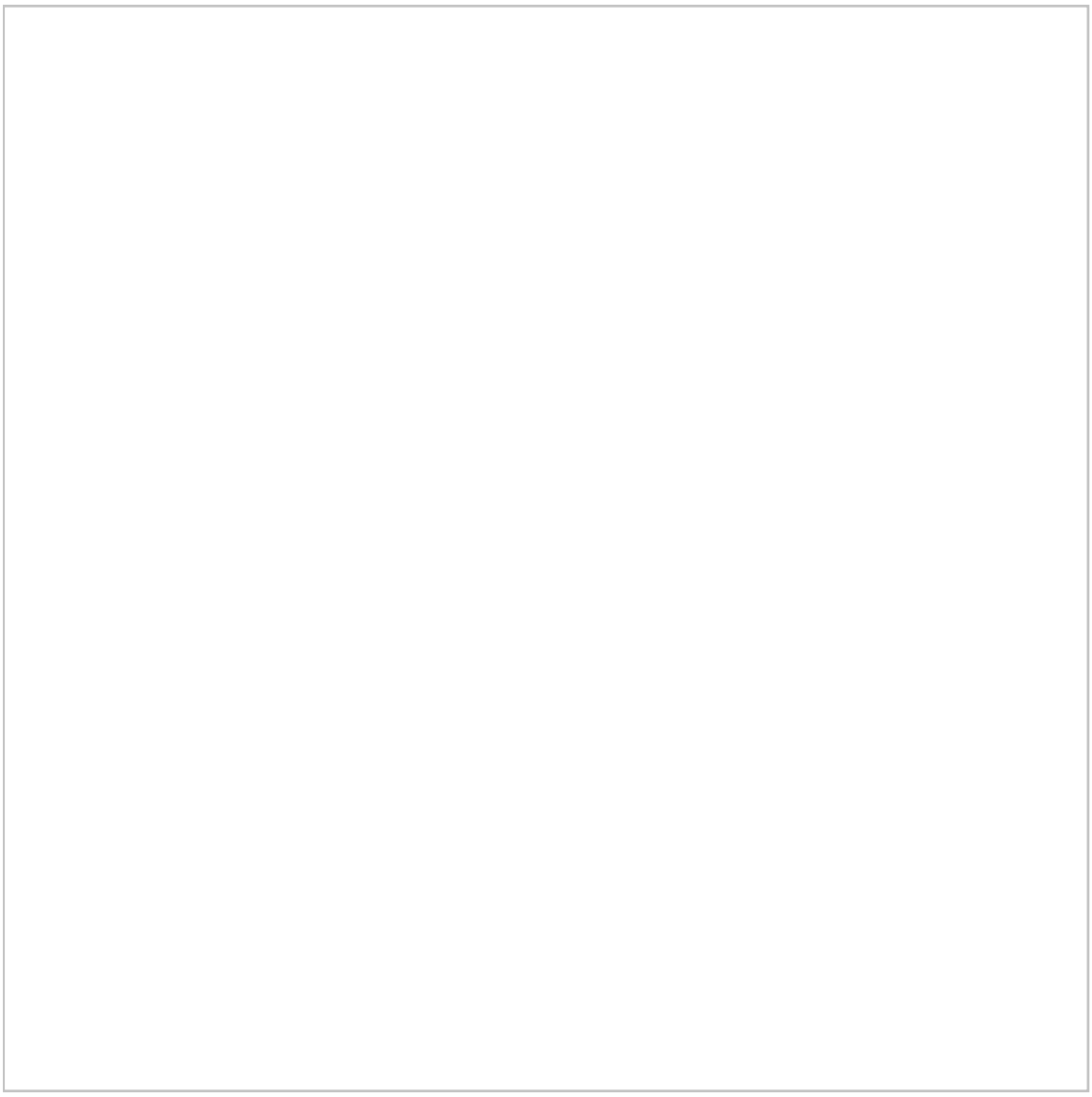
1.mitmdump基础

参考：https://docs.mitmproxy.org/stable/concepts-modes/#transparent-proxy

由于我们不想只监听HTTP流量，因此需要使用mitmproxy的“透明”（Transparent Proxy）模式，即从网络层转发至代理，监听IP层以上的所有流量。基本原理如下图所示，在router层面(使用iptables)将数据包重定向到mitmproxy，然后中间人代理会嗅探并自动转发。注意：代理会将源IP地址修改为本机；NAT不应该在代理之前做，因为会让代理无法判断真正的目的地址，第三张图是错误的示例。



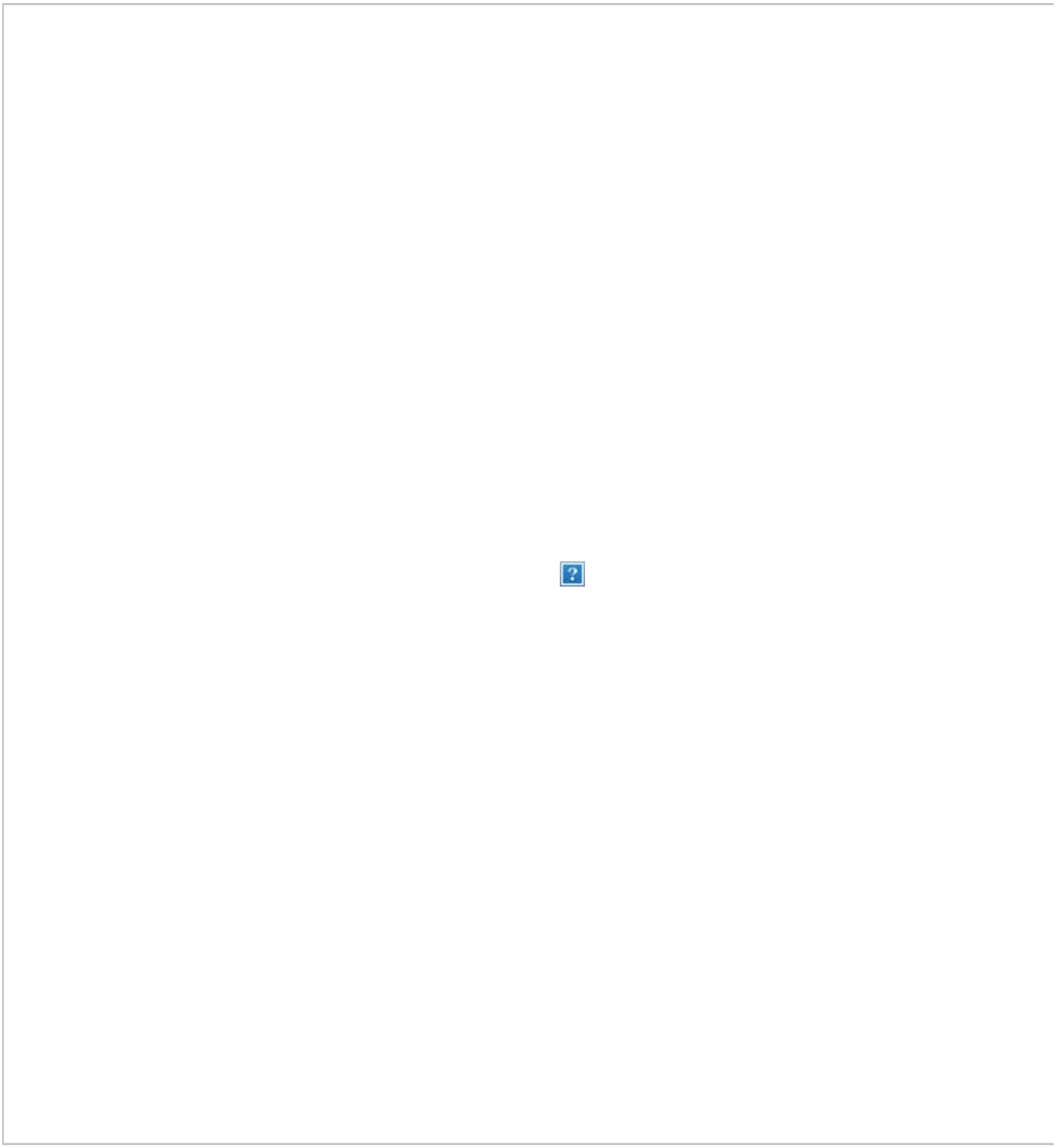




2.iptables基础

参考系列教程：http://www.zsythink.net/archives/1199

尤其是其中的这张图，画的非常好，这里转载过来备用，侵删。



3.安卓校验SSL实现

安卓如何实现https可以参考：https://www.v2ex.com/t/309553

绕过安卓App的SSl可参考：https://blog.netspi.com/four-ways-bypass-android-ssl-verification-certificate-pinning/

对于安卓上的SSL pinning，不一定要使用Frida，也可以使用Xpose的Just trust me模块来Hook

SSLContext.init函数，给一个空的TrustManager，但是，校验的顺序是先校验证书，再校验域名，即setHostnameVerifier。通常这个函数是不用我们处理的，因为mitmproxy已经自动的生成对方服务器的hostname来构建中间人证书，但是个别情况下我们还是需要找到并且Hook掉这个App自己的HostnameVerifier类，比如App连接设备的WiFi并且校验SSL证书是不是和WiFi的名字相同。

二、方案说明

1.wlan1接入具有外网的热点或者按需接入设备建立的配网热点。记录下需要和App通信的ip地址，以方便转发udp流量。

2. wlan0建立热点，并让手机接入。可参考：

```
################################################################################
# File Name: setup_wifi_ap.sh
# Author: ascii0x03
# mail:
# Created Time: 2018年01月22日 星期一 15时28分50秒
################################################################################
#!/bin/bash
#
/etc/init.d/hostapd stop
service isc-dhcp-server stop
/etc/init.d/isc-dhcp-server stop
ifconfig wlan0 down

vim /etc/hostapd/hostapd.conf
vim /etc/dhcp/dhcpd.conf
#设置网卡wlan的IP和子网，要跟dhcpd.conf网关一致
ifconfig wlan0 192.168.201.1 netmask 255.255.255.0 up
sleep 1
rfkill unblock wifi
sleep 1
```

```
hostapd -B /etc/hostapd/hostapd.conf
#initctl reload-configuration
service isc-dhcp-server start
#/etc/init.d/isc-dhcp-server start

#iptables转发上网
sudo sysctl -w net.ipv4.ip_forward=1
sudo iptables -F
sudo iptables -t nat -A POSTROUTING -o eth0 -j MASQUERADE
sudo iptables -L
```

View Code

3.设置转发规则，例如将App的UDP等数据通过DNAT转发至设备，设备也DNAT至App，因为IoT经常使用这些UDP广播来进行设备发现。

4.设置转发规则，将想要监听的tcp端口转发至mitmproxy。

5.在wlan1设置NAT规则进行转发（在出口位置肯定在mitmproxy之后）

```
1 ##############################################################################
2 # File Name: mitm_config.sh
3 # Author: ascii0x03
4 # mail:
5 # Created Time: 2017年12月06日 星期三 16时24分37秒
6 ##############################################################################
7 #!/bin/bash
8 #首先，建立好AP为wlan0，并且开启dhcp服务器分配ip，之后运行如下脚本即可。
9 sysctl -w net.ipv4.ip_forward=1
10 sysctl -w net.ipv6.conf.all.forwarding=1
11 sysctl -w net.ipv4.conf.all.send_redirects=0
12
13 iptables -F
14 iptables -t nat -F
```

```
15 iptables -A FORWARD -i wlan1 -o wlan0 -j ACCEPT
16 iptables -A FORWARD -i wlan0 -o wlan1 -j ACCEPT
17
18 #UDP for discovering device in wlan
19 iptables -t nat -A PREROUTING -i wlan0 -p udp -m udp --dport 5678 -j DNAT --to-destination
192.168.1.101
20 iptables -t nat -A PREROUTING -i wlan1 -p udp -m udp --sport 5678 -j DNAT --to-destination
192.168.201.156
21
22 iptables -t nat -A PREROUTING -i wlan0 -p tcp -m tcp --dport 80 -j REDIRECT --to-ports 8080
23 iptables -t nat -A PREROUTING -i wlan0 -p tcp -m tcp --dport 443 -j REDIRECT --to-ports 8080
24 #MQTTS
25 iptables -t nat -A PREROUTING -i wlan0 -p tcp -m tcp --dport 8883 -j REDIRECT --to-ports 8080
26
27 #all
28 #iptables -t nat -A PREROUTING -i wlan0 -p tcp -j REDIRECT --to-ports 8080
29
30 iptables -t nat -A POSTROUTING -o wlan1 -j MASQUERADE
31
32 iptables -S
33 iptables -t nat -S
34 #存放ssl密钥的位置
35 export MITMPROXY_SSLKEYLOGFILE=~/IoT/keylog
36
37 #mitmdump -T -v --cert *=~/iRobot/irobot_cert.pem   --insecure -tcp -w 11111
38 mitmdump -T -v --insecure --raw-tcp -w 11111
```

View Code

三、附录常用指令

1.连接未加密WiFi

```
iw dev wlan1 connect Roomba-3147C60040239620
```

2.连接加密WiFi

wpa_supplicant -i wlan1 -c ./wifi.conf

wifi.config的内容参考如下

```
network={
    ssid="NIPC"
    psk="11111111"
    #psk=e68b3b7ddb0aba50c582753dfbe33cc372b080eb0d11fcaa6b096ec466f82343
}
```

3.连接后请求获得IP地址

dhclient wlan1

4.

源文地址：https://www.cnblogs.com/ascii0x03/p/9001777.html

文章标签： 网卡 mitmproxy iptables 搭建 SSL 中间人 支持 HTTPS 0

猜你喜欢

武汉成中国第二个无现金城市！全行业普及支付宝

华为终端回应英国禁售令：不影响业务 已提起上诉

微软宣布作为金牌会员加入Cloud Foundry基金会

报告称苹果正扩展CareKit 组建团队推动医疗数据数字化

3D感应模组下半年出货 iPhone 8面部识别功能有戏了

硅谷房价飙升 谷歌斥资3000万美元为员工购置模块化公寓

叫板马云、向客户道歉 顺丰王卫是个什么样的人