Bypassing Tunnels: Leaking VPN Client Traffic by Abusing Routing Tables

Nian Xue, New York University
Yashaswi Malla, Zihang Xia, Christina Pöpper, New York University Abu Dhabi
Mathy Vanhoef, imec-DistriNet, KU Leuven

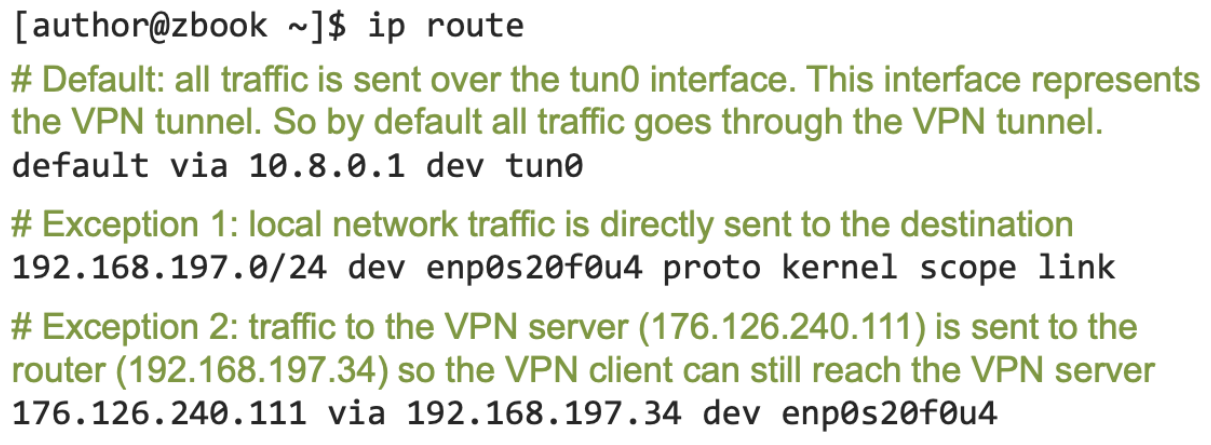
Writer: Akib Jawad Nafis

1 Introduction

VPN (Virtual Private Network) has been widely used for a for security and privacy both in personal and professional setting. VPNs encrypt network packets to protect user's privacy and ensure secure access to internal resources for employees when they are working remotely. A study [8] published by researchers actually built two attacks to bypass VPNs encryption. On both of those attacks, researchers manipulated the routing table to bypass VPNs encryption and leak user traffic outside of the VPN tunnel. Goal of this project was to reproduce the attack scenarios proposed in the study [8].

2 How VPNs Work?

VPNs operate by creating a virtual network interface called tun device. After that VPN client application modifies the routing table of user device in way so that all network is sent to the tun interface. Inside the tun interface it will encrypt the network packet and repackage the encrypted network packet so that encrypted packet in sent to the VPN server. To keep functionality of the network, traffic to VPN server must not be encrypted. Therefore in the routing table, VPN client creates an exception for the VPN server's network. Additionally, to keep access to any service running local network, VPN client also add another exception for traffic to local network. Figure:1 presents the state of the routing table when VPN is enabled.



```
[author@zbook ~]$ ip route
# Default: all traffic is sent over the tun0 interface. This interface represents
the VPN tunnel. So by default all traffic goes through the VPN tunnel.
default via 10.8.0.1 dev tun0

# Exception 1: local network traffic is directly sent to the destination
192.168.197.0/24 dev enp0s20f0u4 proto kernel scope link

# Exception 2: traffic to the VPN server (176.126.240.111) is sent to the
router (192.168.197.34) so the VPN client can still reach the VPN server
176.126.240.111 via 192.168.197.34 dev enp0s20f0u4
```

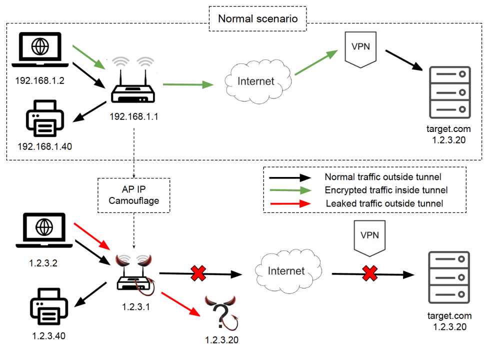
Figure 1: Routing Table with VPN enabled

3 Manipulation of Routing Exceptions: Attacks

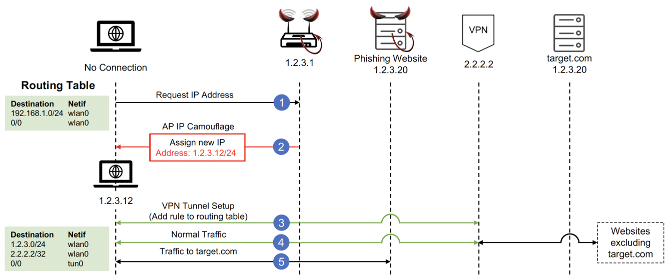
Due to the network route exceptions, packets from user device can flow outside of the VPN tunnel. If traffic to any server which is neither the VPN server nor local network can be sent through these exception routes, then purpose of using VPN will be nullified. Authors of the paper [8] built two attacks to send traffic to any specific website through the unencrypted routes.

3.1 LocalNet Attack

In this attack, routing exception to local network is abused to send traffic to a target website via unencrypted channel. Here authors setup a rouge access point which will distrubute an ip address from the subnet of the target website. As a result user's device will think the target website is in the local network. Hence traffic to target network will be sent through unencrypted channel, bypassing the VPN tunnel. Figure:2a presents the attack and Figure:2b presents steps to carry out local net attack for target.com.



(a) Local net attack using rouge access points



(b) Steps to carry out local net attack

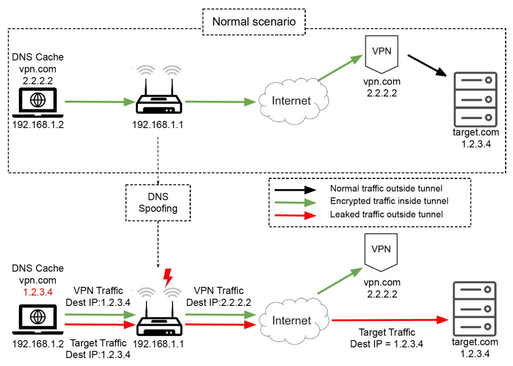
Figure 2: Formation of LocalNet Attack

3.2 ServerIP Attack

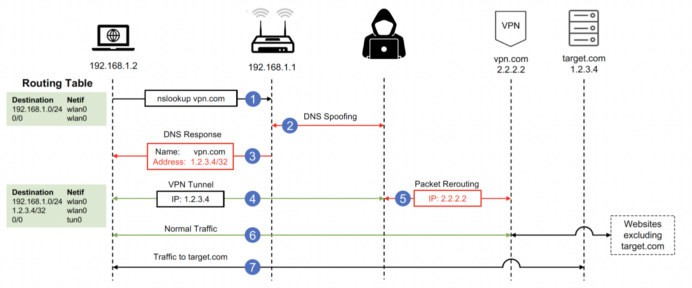
In this attack, routing exception for traffic to VPN server will be misused to leak traffic to target website outside of the VPN tunnel. In this case, rogue access point will work as a malicious DNS server. During the VPN tunnel setup, user's device, which is connected to the rogue access point, will ask for the IP address of the VPN server. Then malicious DNS server in the rouge access point will send the IP address of the target website as the IP address of the VPN server. As a result, user's device will think traffic to target website is traffic to VPN server. Hence, it will send the traffic through the exception route to the VPN server. As a result, traffic to target website will be sent outside of the encrypted VPN tunnel. Figure:3a presents the ServerIP attack and Figure:3b shows the steps to carry out ServerIP-Attack.

4 Experimental Setup

To reproduce the attack developed in the paper [8], I set up my environment following the description given in the paper. Figure:4 shows my setup for LocalNet attack. To create a rogue access point I used a Panda PAU04 network card [4] and create_ap script [1]. For the LocalNet attack, I



(a) Server IP attack using rouge access points



(b) Steps to carry out ServerIP attack

Figure 3: Formation of ServerIP Attack

set up an aws server to have a target website which just runs a nginx server [7]. This is a public server with the IP address 35.164.87.16. I also created a local clone of the server with the same IP address to catch when LocalNetAttack is successful. The local server also runs nginx server, but on the home page it says "CLONE". When user's device is using VPN and user is visiting the target website 35.164.87.16, if user device shows the local webpage with "CLONE" in it, we can confirm that VPN client is leaking.

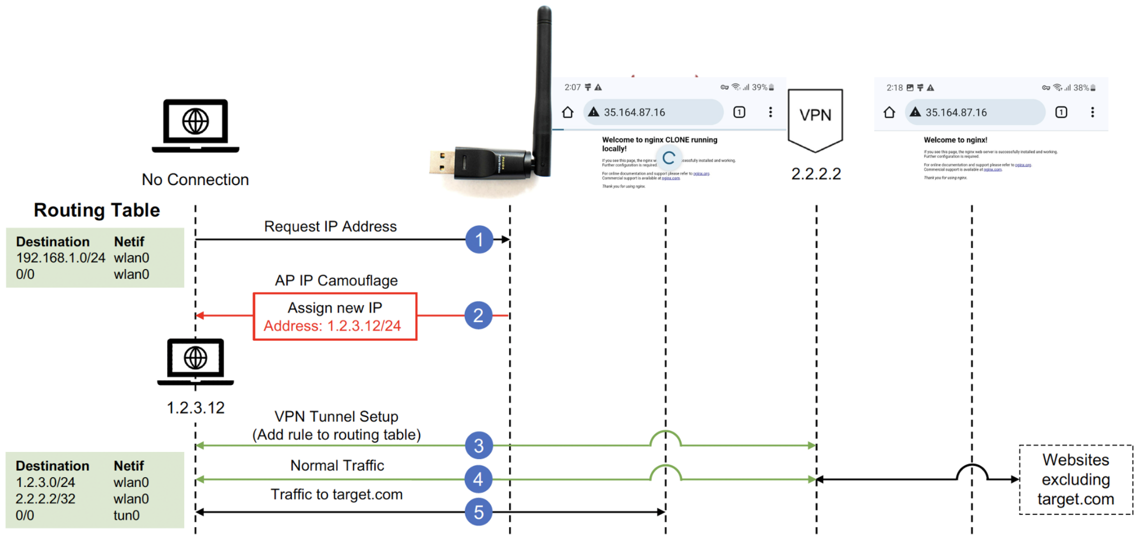


Figure 4: My experimental setup for LocalNet attack on VPN clients.

For the ServerIP attack same setup has been used. But this time, in the rogue access point I also setup a DNS server to spoof DNS response from the VPN server.

4.1 Test Subjects

For the LocalNet Attack, I choose two VPN clients on 4 operating systems resulting in 8 test-subjects.

1. Hide.Me VPN [2] on: iOS, Android, Windows, Mac
2. Hotspot Shield VPN [3] on: iOS, Android, Windows, Mac

For the ServerIP attack I tested built-in VPN clients on 3 operating systems (iOS, Android, Mac) with VPNGate [5] as the VPN server. As mentioned in the paper [8], none of the custom VPN clients is actually vulnerable to ServerIP attack. The reason is custom VPN clients do not use DNS to find the IP address of the VPN server. As a result, authors tried to test the built-in VPN clients with profiles from VPN servcice provider. There was only one free VPN service provider which is VPNGate[5]. When testing for ServerIP attacks, we look for plain text DNS queries in Wireshark [6]. Because when connecting to VPN server, if the VPN client sends a DNS query in plain text, then the client is vulnerable to ServerIP attack..

5 Experimental Results

Behavior of VPN clients under LocalNet attack or ServerIP attack can be categorized into these 4 categories:

- Leaking: These VPN will leak traffic outside of VPN tunnel.
- Not Leaking: These VPN clients never leak traffic outside of VPN tunnel.
- Conditional Leaking: These VPN client leaks traffic if access to local network in enabled.
- Blocking: These VPN clients do not leak traffic outside of VPN client but blocks access to the target website.

I will explain result of LocalNet with Hotspot Shield VPN on Android and iOS. Rest of the results are presented in the table 1. I will also explain ServerIP attack on iOS, remaining results

VPN(version)	Operating System (version)	Result
Hotspot Shield (10.9.0)	Android (13)	Not Leaking
Hotspot Shield (8.10.0)	iOS (17.1.2)	Leaking
Hotspot Shield (12.5.1)	Windows 10	Leaking
Hotspot Shield (6.5.0)	Mac (14.1)	Leaking
Hide.Me(4.1.x)	Android (13)	Leaking when local network connection is enabled.
Hide.Me(5.0.1)	iOS (17.1.2)	Leaking
Hide.Me(3.14.0)	Windows 10	Sends ARP request for targets IP then blocks local connection
Hide.Me(5.2.2)	Mac (14.1)	Leaking
VPN Unlimited paid (9.1.19)	iOS 17.2.1	Leaking

Table 1: Summary of LocalNet attack on 2 VPN clients: Hide.Me VPN and Hotspot Shield VPN

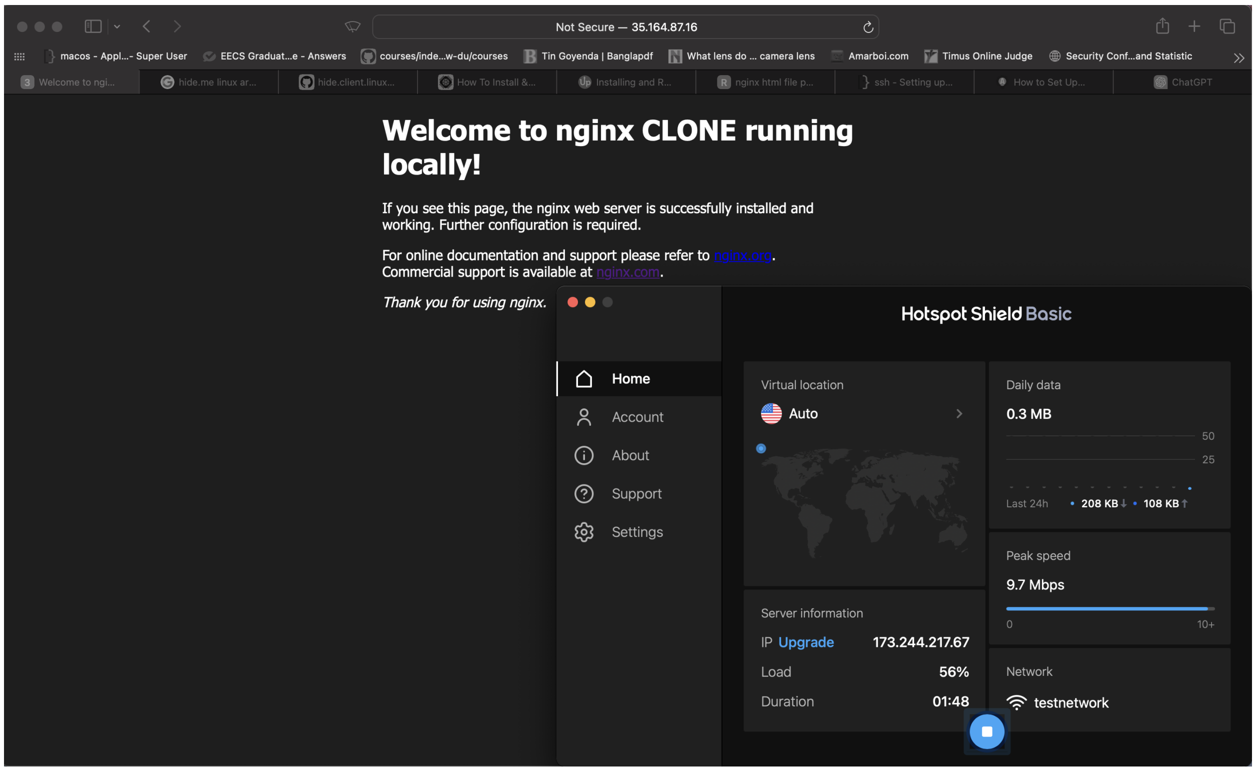
are presented on table 2.

VPN Client (Operating System)	Service Provider	Result
Built-in (iOS 17.1.2)	VPN Gate	Leaking
Built-in (Mac OS 14.1)	VPN Gate	Leaking
Android (13)	VPN Gate	Leaking

Table 2: Result of ServerIP attacks on built-in VPN clients on Mac, iOS and Android

5.1 LocalNet Attack on Hotspot Shield VPN 6.5.0 on Mac 14.1

Hotspot Shield VPN 6.5.0 on Mac OS 14.1 is vulnerable to LocalNet attack. As we can see on Figure:5 VPN is connected using Hotspot Shield app. When visiting our target website http://35.164.87.16, loaded web page does have "CLONE" mark in it. Because browser is loading the website from the local nginx server.



Welcome to nginx CLONE running locally!

If you see this page, the nginx web server is successfully installed and working. Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

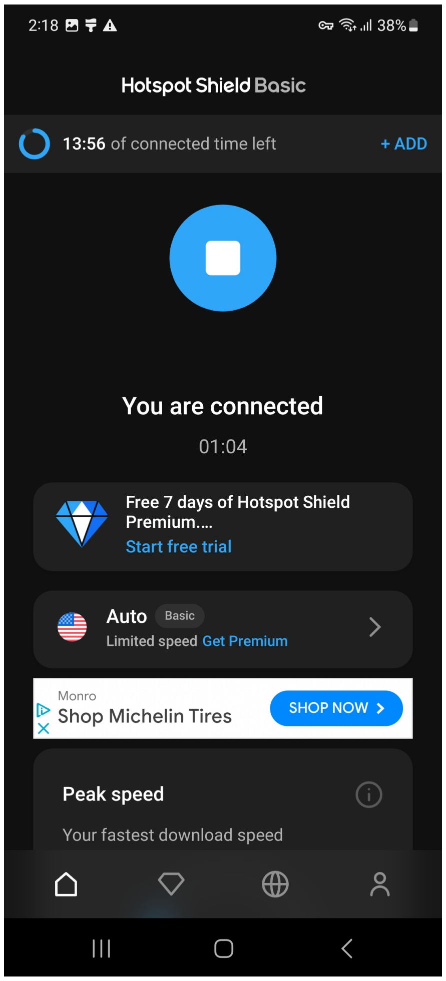
Thank you for using nginx.

Virtual location: Auto
Daily data: 0.3 MB — 208 KB ↓ 108 KB ↑
Peak speed: 9.7 Mbps
Server information
IP Upgrade 173.244.217.67
Load 56%
Duration 01:48
Network: testnetwork

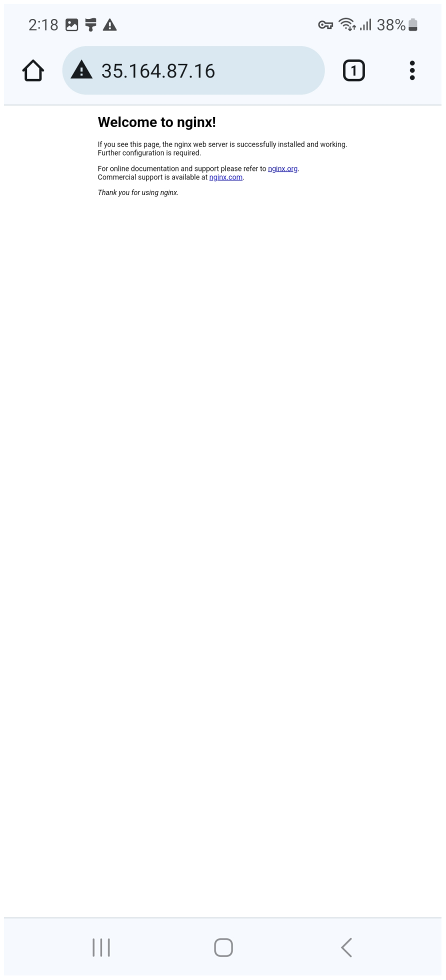
Figure 5: Hotspot Shield VPN on Mac in vulnerable to LocalNet attack

5.2 LocalNet Attack on Hotspot Shield VPN 10.9.0 on Android 13

Hotspot Shield VPN 10.9.0 on Android 13 is not vulnerable to LocalNet attack. As we can see on Figure:6a VPN is connected using Hotspot Shield app, on Figure:6b when visiting our target website http://35.164.87.16, loaded web page does not have "CLONE" mark in it. Because browser is loading the website from the public aws server.


2:18
38%
Hotspot Shield Basic
13:56 of connected time left
+ ADD
You are connected
01:04
Free 7 days of Hotspot Shield Premium....
Start free trial
Auto
Basic
Limited speed Get Premium
Monro
Shop Michelin Tires
SHOP NOW
Peak speed
Your fastest download speed

(a) Hotspot Shield VPN app on Android



Welcome to nginx!

If you see this page, the nginx web server is successfully installed and working.
Further configuration is required.

For online documentation and support please refer to nginx.org.
Commercial support is available at nginx.com.

Thank you for using nginx.

(b) Loading target website http://35.164.87.16

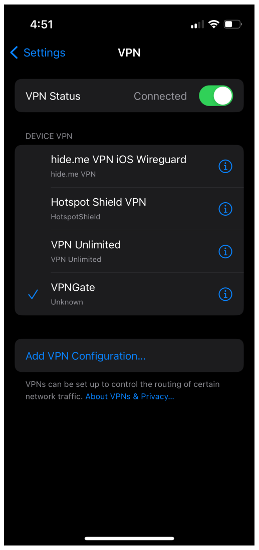
Figure 6: Testing Hotspot Shield VPN on Android for LocalNet attack

5.3 ServerIP Attack on Built-In VPN clinet on iOS 17.1.2

As shown in Figure:7a built-in VPN client of iOS 17.1.2 is connecting to VPN server of VPN-Gate. At the same time, if we look at wireshark on Figure:7b, we can see a plain text DNS qwery for the IP address of VPN server. Meaning, built-in VPN client of iOS 17.1.2 is vulnerable to ServerIP attack. Similar result has been noticed on Android and Mac.

6 Discussion

My goal on this project was to reproduce all categories of result presented in the paper [8]. In the original paper, authors tested large number of VPN clients. Big number of those VPN clients were paid clients. I tested only free VPN clients. I added one paid VPN client which I use for my personal usage. That VPN client was also vulnerable to the attack. In the end, I was actually able to reproduce results of all 4 categories.



4:51

Settings　VPN

VPN Status　Connected

DEVICE VPN

hide.me VPN iOS Wireguard
hide.me VPN

Hotspot Shield VPN
HotspotShield

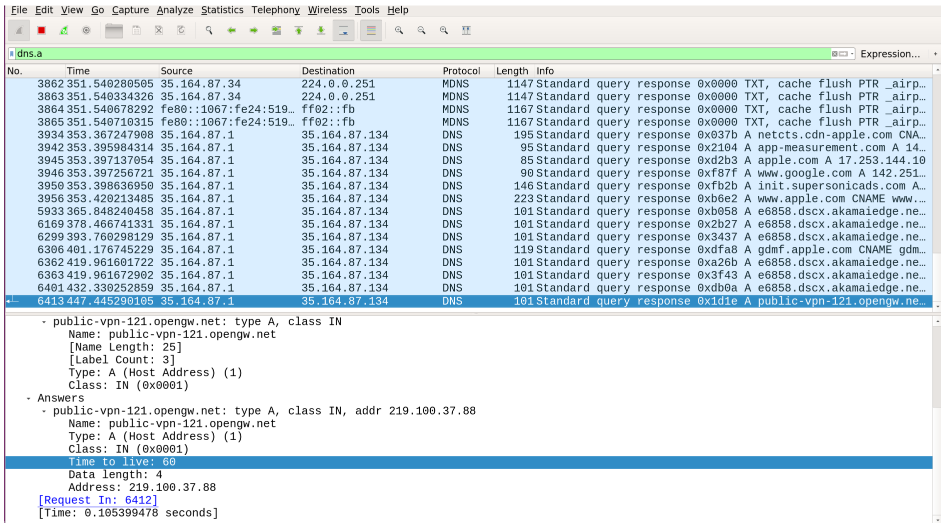
VPN Unlimited
VPN Unlimited

✓ VPNGate
Unknown

Add VPN Configuration...

VPNs can be set up to control the routing of certain network traffic. About VPNs & Privacy...

(a) Connecting to VPNGate with built in VPN client of iOS 17.2.1



File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
Expression...

```
dns.a
```

```
No.     Time          Source                Destination     Protocol  Length Info
   3862 351.540280505 35.164.87.34          224.0.0.251     MDNS      1147 Standard query response 0x0000 TXT, cache flush PTR _airp…
   3863 351.540334326 35.164.87.34          224.0.0.251     MDNS      1147 Standard query response 0x0000 TXT, cache flush PTR _airp…
   3864 351.540678292 fe80::1067:fe24:519…  ff02::fb        MDNS      1167 Standard query response 0x0000 TXT, cache flush PTR _airp…
   3865 351.540710315 fe80::1067:fe24:519…  ff02::fb        MDNS      1167 Standard query response 0x0000 TXT, cache flush PTR _airp…
   3934 353.367247908 35.164.87.1           35.164.87.134   DNS        195 Standard query response 0x037b A netcts.cdn-apple.com CNA…
   3942 353.395984314 35.164.87.1           35.164.87.134   DNS         95 Standard query response 0x2104 A app-measurement.com A 14…
   3945 353.397137054 35.164.87.1           35.164.87.134   DNS         85 Standard query response 0xd2b3 A apple.com A 17.253.144.10
   3946 353.397256721 35.164.87.1           35.164.87.134   DNS         90 Standard query response 0xf87f A www.google.com A 142.251…
   3950 353.398636950 35.164.87.1           35.164.87.134   DNS        146 Standard query response 0xfb2b A init.supersonicads.com A…
   3956 353.420213485 35.164.87.1           35.164.87.134   DNS        223 Standard query response 0xb6e2 A www.apple.com CNAME www.…
   5933 365.848240458 35.164.87.1           35.164.87.134   DNS        101 Standard query response 0xb058 A e6858.dscx.akamaiedge.ne…
   6169 378.466741331 35.164.87.1           35.164.87.134   DNS        101 Standard query response 0x2b27 A e6858.dscx.akamaiedge.ne…
   6299 393.760298129 35.164.87.1           35.164.87.134   DNS        101 Standard query response 0x3437 A e6858.dscx.akamaiedge.ne…
   6306 401.176745229 35.164.87.1           35.164.87.134   DNS        119 Standard query response 0xdfa8 A gdmf.apple.com CNAME gdm…
   6362 419.961601722 35.164.87.1           35.164.87.134   DNS        101 Standard query response 0xa26b A e6858.dscx.akamaiedge.ne…
   6363 419.961672902 35.164.87.1           35.164.87.134   DNS        101 Standard query response 0x3f43 A e6858.dscx.akamaiedge.ne…
   6401 432.330252859 35.164.87.1           35.164.87.134   DNS        101 Standard query response 0xdb0a A e6858.dscx.akamaiedge.ne…
   6413 447.445290105 35.164.87.1           35.164.87.134   DNS        101 Standard query response 0x1d1e A public-vpn-121.opengw.ne…
```

```
    - public-vpn-121.opengw.net: type A, class IN
        Name: public-vpn-121.opengw.net
        [Name Length: 25]
        [Label Count: 3]
        Type: A (Host Address) (1)
        Class: IN (0x0001)
  - Answers
    - public-vpn-121.opengw.net: type A, class IN, addr 219.100.37.88
        Name: public-vpn-121.opengw.net
        Type: A (Host Address) (1)
        Class: IN (0x0001)
        Time to live: 60
        Data length: 4
        Address: 219.100.37.88
    [Request In: 6412]
    [Time: 0.105399478 seconds]
```

(b) Plain text DNS query when connecting to VPN server of VPNGate

Figure 7: Testing built-in VPN client on iOS for ServerIP attack

References

[1] Create_ap script to create access points, dns servers etc. URL: https://github.com/oblique/create_ap.

[2] Hide.me vpn. URL: https://hide.me/en/.

[3] Hotspot shield vpn. URL: https://www.hotspotshield.com.

[4] Panda 150mbps midrange wireless n usb adapater. URL: https://www.amazon.com/Panda-150Mbps-Wireless-Adapter-Antenna/dp/B004AC0L4Y.

[5] Vpngate: Free vpn service provider. URL: https://www.vpngate.net.

[6] Wireshark: A packet inspection tool. URL: https://www.wireshark.org.

[7] Nginx: An open source server. URL: https://nginx.org/en/.

[8] Nian Xue, Yashaswi Malla, Zihang Xia, Christina Pöpper, and Mathy Vanhoef. Bypassing tunnels: Leaking VPN client traffic by abusing routing tables. In *32nd USENIX Security Symposium (USENIX Security 23)*, pages 5719–5736, Anaheim, CA, August 2023. USENIX Association. URL: https://www.usenix.org/conference/usenixsecurity23/presentation/xue.