

Identifying the Nexus of Scaled Ad Fraud

 spur.us/identifying-the-nexus-of-scaled-ad-fraud/

Riley Kilmer

May 17, 2023

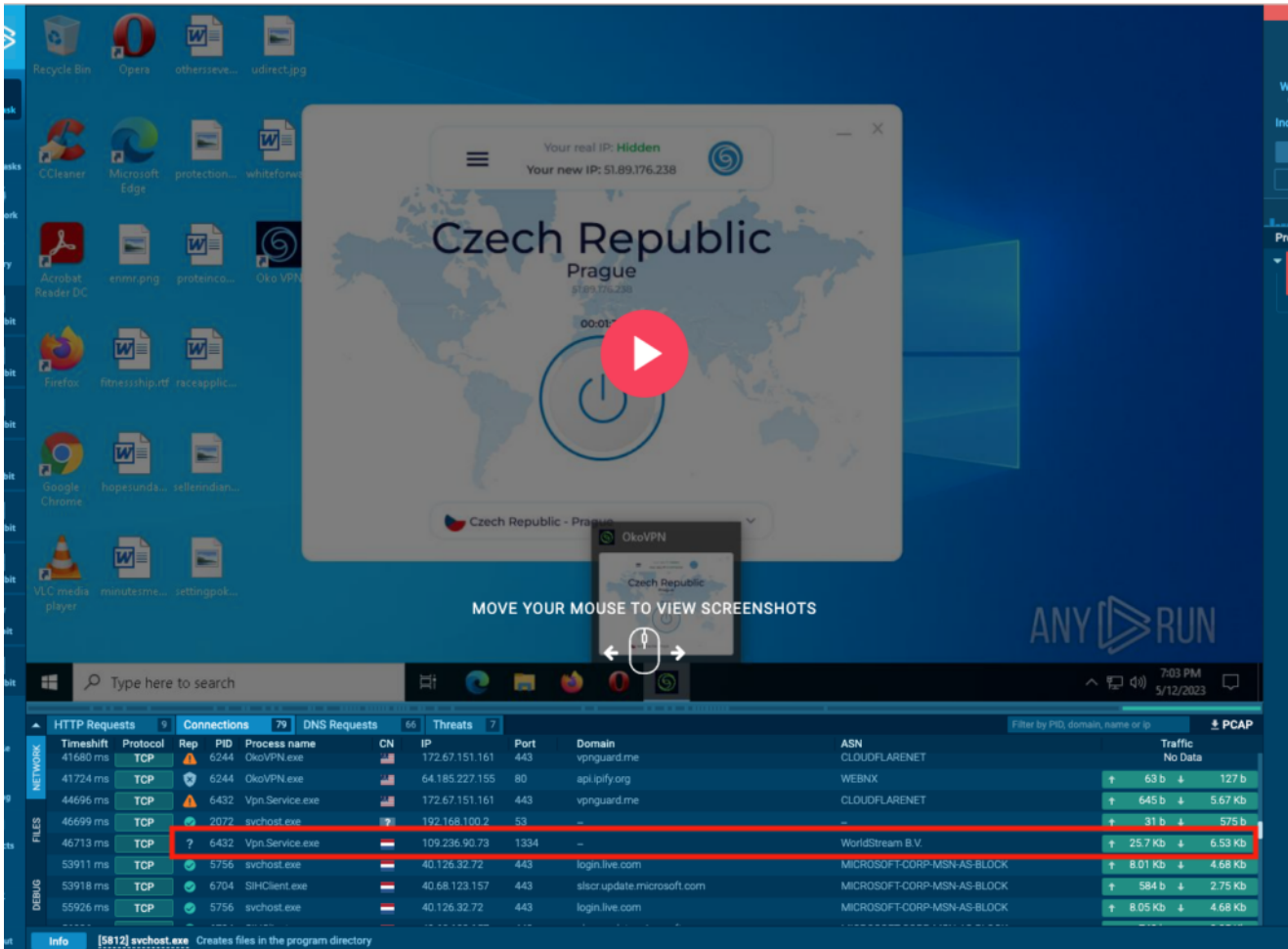
The Problem

Late last week, I was procrastinating perusing LinkedIn and encountered an [article](#) that referenced a scaled ad-fraud campaign powered by a free VPN application called Oko VPN (okovpn[.]com). The second I saw the article title, I had a gut feeling it had to involve a residential proxy service. First, I wanted to know if that was true. Second, was it a service we already track? Unfortunately, the article stopped short of identifying which service fueled the reported fraud.

I had to know...

The Journey Begins

Since there were not any real indicators of compromise (IOCs) provided by the article, I set out to find them. The best way to determine which service is utilizing Oko is identifying the backend callback infrastructure. I noticed they had a Windows application which makes this trivial to check. After a few minutes in Any.Run, we were off to the races. You can look at the Any.Run report [here](#).



The IP address **109.236.90.73** really stood out to me during this analysis. It is an odd port and no associated domain within the sandbox. Using my DNSDB CLI from DomainTools, I found some interesting domains.

```

→ ~ dnsdb 109.236.90.73
109.236.90.73    2022-11-08T07:08:05Z    2023-05-16T13:24:58Z    nsignal.net.
109.236.90.73    2022-11-28T09:07:37Z    2023-05-16T11:02:11Z    ts13.p2proxy.net.
109.236.90.73    2022-11-08T15:55:56Z    2023-05-16T03:03:05Z    109-236-90-
73.hosted-by-worldstream.net.

```

p2proxy[.]net and **nsignal[.]net** both look like very likely candidates. But before diving deeper into these domains, I wanted to look at the contents of this TCP stream.

Network stream

109.236.90.73: 1334 ↗ VM: 64217

RAW data flow between two hosts

8 of 40 Show all View **HEX** Text Highlight chars

Recv: 107 B Timeshift: 136.11 s Download Hide

```

00000000  57 00 00 00 02 00 64 47 45 54 20 2F 6A 73 6F 6E  W.....dGET /json
00000010  20 48 54 54 50 2F 31 2E 31 0D 0A 48 6F 73 74  HTTP/1.1..Host
00000020  20 70 72 6F 78 79 63 68 65 63 6B 2E 6C 69 6E 6B  proxycheck.link
00000030  0D 0A 55 73 65 72 2D 41 67 65 6E 74 3A 20 47 6F  ..User-Agent: Go
00000040  2D 68 74 74 70 2D 63 6C 69 65 6E 74 2F 31 2E 31  -http-client/1.1
00000050  0D 0A 41 63 63 65 70 74 2D 45 6E 63 6F 64 69 6E  ..Accept-Encodin
00000060  67 3A 20 67 7A 69 70 0D 0A 0D 0A  g: gzip....

```

Send: 369 B Timeshift: 136.34 s Download Hide

```

00000000  57 00 00 00 02 01 6A 48 54 54 50 2F 31 2E 31 20  W.....jHTTP/1.1
00000010  32 30 30 20 4F 4B 0D 0A 53 65 72 76 65 72 3A  200 OK..Server:
00000020  6E 67 69 6E 78 2F 31 2E 31 34 2E 31 0D 0A 44 61  nginx/1.14.1..Da
00000030  74 65 3A 20 46 72 69 2C 20 31 32 20 4D 61 79 20  te: Fri, 12 May
00000040  32 30 32 33 20 31 39 3A 30 31 3A 35 31 20 47 4D  2023 19:01:51 GM
00000050  54 0D 0A 43 6F 6E 74 65 6E 74 2D 54 79 70 65 3A  T..Content-Type:
00000060  20 74 65 78 74 2F 70 6C 61 69 6E 3B 20 63 68 61  text/plain; cha
00000070  72 73 65 74 3D 75 74 66 2D 38 0D 0A 43 6F 6E 74  rset=utf-8..Cont
00000080  65 6E 74 2D 4C 65 6E 67 74 68 3A 20 31 39 38 0D  ent-Length: 198.
00000090  0A 43 6F 6E 6E 65 63 74 69 6F 6E 3A 20 6B 65 65  .Connection: kee
000000a0  70 2D 61 6C 69 76 65 0D 0A 0D 0A 7B 22 71 75 65  p-alive....{"que
000000b0  72 79 22 3A 22 34 35 2E 39 31 2E 32 30 2E 31 33  ry":"45.91.20.13
000000c0  22 2C 22 63 6F 75 6E 74 72 79 43 6F 64 65 22 3A  ","countryCode":
000000d0  22 49 54 22 2C 22 63 6F 75 6E 74 72 79 22 3A 22  "IT" "country":

```

This definitely looks like plain-text proxying through my sandbox to check IP address information. Before long, there was a TLS connection made using this tunnel to Instagram and other services using this same TCP tunnel. This information really affirmed my thought that this host was the responsible command-and-control for the residential proxy service. Unfortunately, [nsignal\[.\]net](#) nor [p2proxy\[.\]net](#) were associated with any of the services we already track.

Sometimes, these IOCs are easy to track back to the associated services. Unfortunately, simple Google searches only yielded results from years ago connecting to some blockchain proxy pool. VirusTotal searches for [nsignal\[.\]net](#) only yielded more OkoVPN samples. Investigations into this infrastructure were fairly frustrating and did not yield any hints as to which network was reselling this bandwidth.

Subdomains (3)				
s2.nsignal.net	0 / 87	185.183.35.137		
nsignal.net	0 / 87	148.72.170.53	103.66.180.3	192.95.29.203 ...
dev.nsignal.net	0 / 87	185.2.83.145		

Communicating Files (37)			
Scanned	Detections	Type	Name
2023-04-27	0 / 65	Android	oko_vpn.apk
2022-11-03	1 / 62	Android	Oko VPN_1.5.1_apkcombo.com.apk
2023-02-22	0 / 64	Android	OkoVPN.apk
2023-04-22	0 / 64	Android	VPN Ultra-V1.1.apk
2023-05-01	0 / 64	Android	Oko VPN_1.6_Apkpure.apk
2023-05-10	0 / 64	Android	litevpn.apk
2023-04-14	0 / 64	Android	VPN Ultra_1.0.1_Apkpure.apk
2023-05-04	0 / 64	Android	Oko VPN_1.5.4_apkcombo.com.apk
2023-03-27	0 / 64	Android	Oko VPN_1.7.1@PJAPK.apk
2023-01-26	0 / 65	Android	Okovpn 5.11.apk

Historical Whois Lookups (3)

Back to the beginning

Our only real connection to this mystery network was Oko VPN. There is always the possibility that the VPN itself is owned by the service directly. If this is the case, the terms of service and/or privacy policy can be very revealing. I didn't even reach the end of the terms of service before I was met with a revelation:

In return for some of the premium features of "Oko VPN", you may choose to be a peer on the D.M.D.D. network. By doing so you agree to have read and accepted the Terms of Service of the D.M.D.D.'s SDK EULA: https://lumiapps.io/End_User_License_Agreement.pdf and D.M.D.D.'s Privacy Policy: https://lumiapps.io/DMDD_SDK_Privacy_Policy.pdf. You may opt out of the D.M.D.D.'s network by clicking: Personal settings => switch the tubler of => agree that you want to cancel your participation in D.M.D.D. network.

We have talked a lot about how different residential proxies source their IP addresses. Many of these free VPN apps play the game of embedding consent deep within their TOS or privacy policies. The lumiapps[.]io website did not have any obvious answer. I was really hoping for some area titled "Want to buy bandwidth from us?" or similarly phrased section. But now we have a handful of clues and hopefully one of these domains or IP addresses they are using links them to a residential proxy service.

SurfaceBrowser by Recorded Future is one of my favorite tools for just throwing spaghetti at the wall. I saw a subdomain that looked really promising. Or at least pretty suspect. The mail server being hosted in Russia definitely felt like a clue worth pursuing.

Pivoting on the IP address `45.80.205[.]121` led me to a whole lot of other mail servers that could definitely be the culprits.

Hostname	Rank
mail.nexusnet.pro	-
mail.dmdd.io	-
mail.any-page.io	-
mail.oksy.org	-
mail.avgustorg.ru	-
mail.sneakerproxy.io	-
mail.asocks-mail.com	-
mail.nexusmail.pro	-
mail.lumiapps.io	-
mail.broxy.one	-
mail.asocks-subscribe.com	-

When I showed this to my research team, one of my teammates was quick to saying “I have heard of NexusNet”. Their primary service is operated at `nexusnet[.]io`. This service was on a list we were actively investigating for addition to our tracking system.

After a couple of hours of enumeration, we had a list of NexusNet proxies. Using our android sandbox environment running a longer job of Oko VPN, we were able to see our activity within our own sandbox using the [nsignal\[.\]net](#) tunnel. This was the last piece of evidence that helped us tie up this investigation.

Conclusion

If a VPN service is free and claiming no-logs, there has to be a catch. Whether that is advertisements or other monetization like bandwidth re-selling, you are the product. Installing Oko VPN on your device would have included your IP address in this reported Ad fraud campaigns or any other fraud being performed by customers of NexusNet.

Take a look at our active intelligence on NexusNet using our community [dashboard](#). And if you are looking for ways to prevent these networks from abusing your platforms, check out our feeds, API tools, and [Monocle](#).