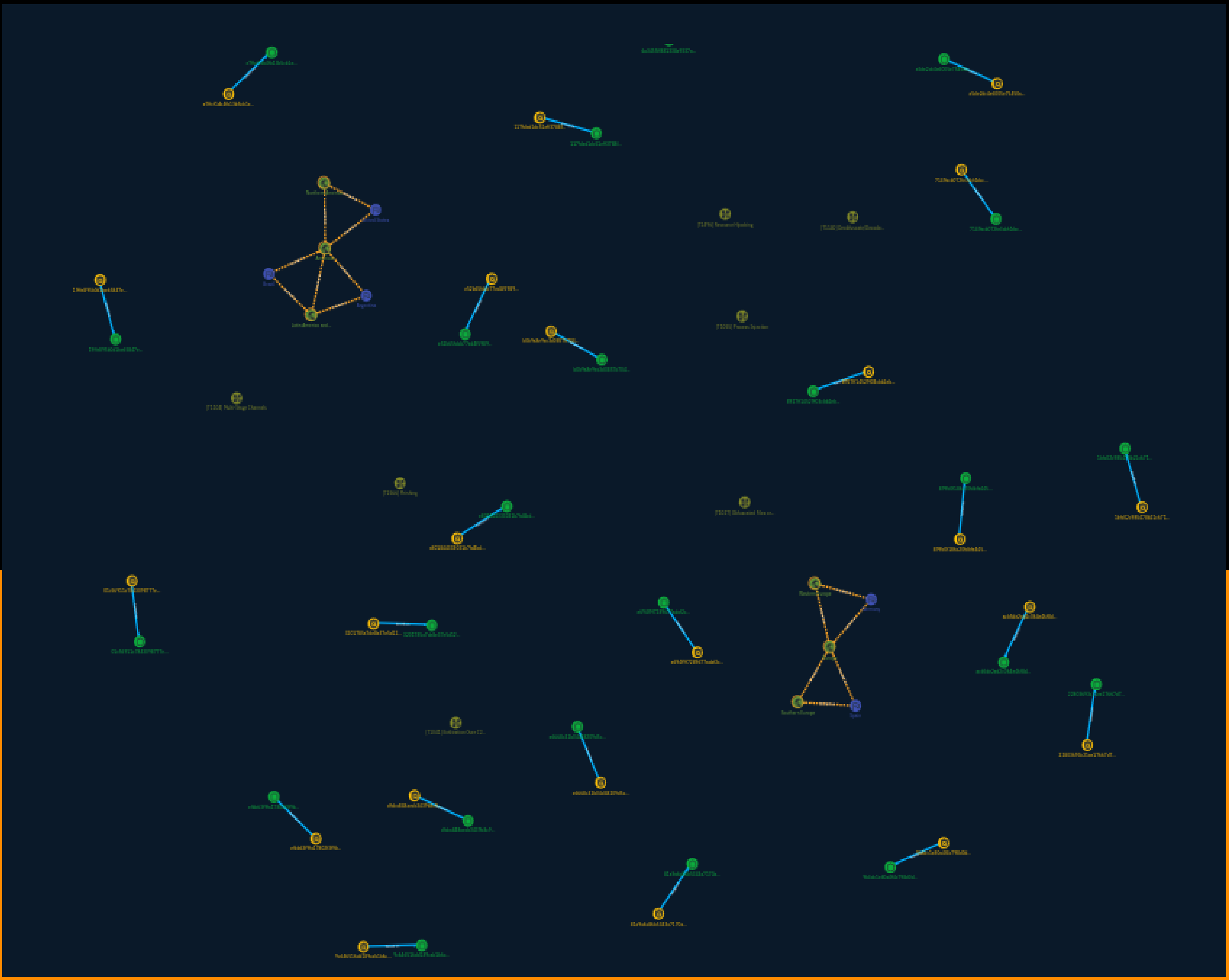
Dec 27 2023

NETMANAGEIT

# Intelligence Report

# Stealth Backdoor “Android/Xamalicious” Actively Infecting Devices



# Table of contents

## Overview

## Entities

## Observables

## External References

# Overview

## Description

McAfee Mobile Research Team identified an Android backdoor implemented with Xamarin, an open-source framework that allows building Android and iOS apps with .NET and C#.

## Confidence

*This value represents the confidence in the correctness of the data contained within this report.*

15 / 100

# Content

N/A

# Attack-Pattern

| Name |
|---|
| Process Injection |
| **ID** |
| T1055 |
| **Description** |
| Adversaries may inject code into processes in order to evade process-based defenses as well as possibly elevate privileges. Process injection is a method of executing arbitrary code in the address space of a separate live process. Running code in the context of another process may allow access to the process's memory, system/network resources, and possibly elevated privileges. Execution via process injection may also evade detection from security products since the execution is masked under a legitimate process. There are many different ways to inject code into a process, many of which abuse legitimate functionalities. These implementations exist for every major OS but are typically platform specific. More sophisticated samples may perform multiple process injections to segment modules and further evade detection, utilizing named pipes or other inter-process communication (IPC) mechanisms as a communication channel. |
| **Name** |
| Phishing |
| **ID** |
| T1566 |

## Description

Adversaries may send phishing messages to gain access to victim systems. All forms of phishing are electronically delivered social engineering. Phishing can be targeted, known as spearphishing. In spearphishing, a specific individual, company, or industry will be targeted by the adversary. More generally, adversaries can conduct non-targeted phishing, such as in mass malware spam campaigns. Adversaries may send victims emails containing malicious attachments or links, typically to execute malicious code on victim systems. Phishing may also be conducted via third-party services, like social media platforms. Phishing may also involve social engineering techniques, such as posing as a trusted source, as well as evasive techniques such as removing or manipulating emails or metadata/headers from compromised accounts being abused to send messages (e.g., [Email Hiding Rules](https://attack.mitre.org/techniques/T1564/008)).(Citation: Microsoft OAuth Spam 2022)(Citation: Palo Alto Unit 42 VBA Infostealer 2014) Another way to accomplish this is by forging or spoofing(Citation: Proofpoint-spoof) the identity of the sender which can be used to fool both the human recipient as well as automated security tools.(Citation: cyberproof-double-bounce) Victims may also receive phishing messages that instruct them to call a phone number where they are directed to visit a malicious URL, download malware,(Citation: sygnia Luna Month)(Citation: CISA Remote Monitoring and Management Software) or install adversary-accessible remote management tools onto their computer (i.e., [User Execution](https://attack.mitre.org/techniques/T1204)).(Citation: Unit42 Luna Moth)

## Name

Resource Hijacking

## ID

T1496

## Description

Adversaries may leverage the resources of co-opted systems to complete resource-intensive tasks, which may impact system and/or hosted service availability. One common purpose for Resource Hijacking is to validate transactions of cryptocurrency networks and earn virtual currency. Adversaries may consume enough system resources to negatively impact and/or cause affected machines to become unresponsive.(Citation: Kaspersky Lazarus Under The Hood Blog 2017) Servers and cloud-based systems are common targets because of the high potential for available resources, but user endpoint systems may also be compromised and used for Resource Hijacking and cryptocurrency mining.(Citation: CloudSploit - Unused AWS Regions) Containerized environments may also be targeted due to the ease of deployment via exposed APIs and the potential for scaling mining activities by deploying or compromising multiple containers within an environment or cluster.(Citation: Unit 42 Hildegard Malware)(Citation: Trend Micro Exposed Docker APIs) Additionally, some cryptocurrency mining malware identify then kill off processes for competing malware to ensure it's not competing for resources.(Citation: Trend Micro War of Crypto Miners) Adversaries may also use malware that leverages a system's network bandwidth as part of a botnet in order to facilitate [Network Denial of Service](https://attack.mitre.org/techniques/T1498) campaigns and/or to seed malicious torrents.(Citation: GoBotKR) Alternatively, they may engage in proxyjacking by selling use of the victims' network bandwidth and IP address to proxyware services.(Citation: Sysdig Proxyjacking)

## Name

Obfuscated Files or Information

## ID

T1027

## Description

Adversaries may attempt to make an executable or file difficult to discover or analyze by encrypting, encoding, or otherwise obfuscating its contents on the system or in transit. This is common behavior that can be used across different platforms and the network to evade defenses. Payloads may be compressed, archived, or encrypted in order to avoid detection. These payloads may be used during Initial Access or later to mitigate detection. Sometimes a user's action may be required to open and [Deobfuscate/Decode Files or Information](https://attack.mitre.org/techniques/T1140) for [User Execution](https://attack.mitre.org/techniques/T1204). The user may also be required to input a password to open a password protected compressed/encrypted file that was provided by the adversary.(Citation: Volexity PowerDuke November 2016) Adversaries may also use compressed or archived scripts, such as JavaScript. Portions of files can also be encoded to hide the plain-text strings that would otherwise help defenders with discovery. (Citation: Linux/Cdorked.A We Live Security Analysis) Payloads may also be split into separate, seemingly benign files that only reveal malicious functionality when reassembled. (Citation: Carbon Black Obfuscation Sept 2016) Adversaries may also abuse [Command Obfuscation](https://attack.mitre.org/techniques/T1027/010) to obscure commands executed from payloads or directly via [Command and Scripting Interpreter](https://attack.mitre.org/techniques/T1059). Environment variables, aliases, characters, and other platform/language specific semantics can be used to evade signature based detections and application control mechanisms. (Citation: FireEye Obfuscation June 2017) (Citation: FireEye Revoke-Obfuscation July 2017)(Citation: PaloAlto EncodedCommand March 2017)

**Name**

Multi-Stage Channels

**ID**

T1104

**Description**

Adversaries may create multiple stages for command and control that are employed under different conditions or for certain functions. Use of multiple stages may obfuscate the command and control channel to make detection more difficult. Remote access tools will call back to the first-stage command and control server for instructions. The first stage may have automated capabilities to collect basic host information, update tools, and upload additional files. A second remote access tool (RAT) could be uploaded at that point to redirect the host to the second-stage command and control server. The second stage will likely be more fully featured and allow the adversary to interact with the system through a reverse shell and additional RAT features. The different stages will likely be hosted separately with no overlapping infrastructure. The loader may also have backup first-stage callbacks or [Fallback Channels](https://attack.mitre.org/techniques/T1008) in case the original first-stage communication path is discovered and blocked.

**Name**

Deobfuscate/Decode Files or Information

**ID**

T1140

**Description**

Adversaries may use [Obfuscated Files or Information](https://attack.mitre.org/techniques/T1027) to hide artifacts of an intrusion from analysis. They may require separate mechanisms to decode or deobfuscate that information depending on how they intend to use it. Methods for doing that include built-in functionality of malware or by using utilities present on the system. One such example is the use of [certutil](https://attack.mitre.org/software/S0160) to decode a remote access tool portable executable file that has been hidden inside a certificate file.(Citation: Malwarebytes Targeted Attack against Saudi Arabia) Another example is using the Windows `copy /b` command to reassemble binary fragments into a malicious payload.(Citation: Carbon Black Obfuscation Sept 2016) Sometimes a user's action may be required to open it for deobfuscation or decryption as part of [User Execution](https://attack.mitre.org/techniques/T1204). The user may also be required to input a password to open a password protected compressed/encrypted file that was provided by the adversary. (Citation: Volexity PowerDuke November 2016)

**Name**

Exfiltration Over C2 Channel

**ID**

T1041

**Description**

Adversaries may steal data by exfiltrating it over an existing command and control channel. Stolen data is encoded into the normal communications channel using the same protocol as command and control communications.

# Indicator

**Name**

e7ffcf1db4fb13b5cb1e9939b3a966c4a5a894f7b1c1978ce6235886776c961e

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' =
'e7ffcf1db4fb13b5cb1e9939b3a966c4a5a894f7b1c1978ce6235886776c961e']

**Name**

acb5de2ed2c064e46f8d42ee82feabe380364a6ef0fbfeb73cf01ffc5e0ded6b

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' =
'acb5de2ed2c064e46f8d42ee82feabe380364a6ef0fbfeb73cf01ffc5e0ded6b']

**Name**

28a4ae5c699a7d96e963ca5ceec304aa9c4e55bc661e16c194bdba9a8ad847b7

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' =
'28a4ae5c699a7d96e963ca5ceec304aa9c4e55bc661e16c194bdba9a8ad847b7']

**Name**

6a3455ff881338e9337a75c9f2857c33814b7eb4060c06c72839b641b347ed36

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' =
'6a3455ff881338e9337a75c9f2857c33814b7eb4060c06c72839b641b347ed36']

**Name**

1bfc02c985478b21c6713311ca9108f6c432052ea568458c8bd7582f0a825a48

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' =
'1bfc02c985478b21c6713311ca9108f6c432052ea568458c8bd7582f0a825a48']

**Name**

01c56911c7843098777ec375bb5b0029379b0457a9675f149f339b7db823e996

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' =
'01c56911c7843098777ec375bb5b0029379b0457a9675f149f339b7db823e996']

**Name**

a5de2dc4e6005e75450a0df0ea83816996092261f7dac30b5cf909bf6daaced0

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' =
'a5de2dc4e6005e75450a0df0ea83816996092261f7dac30b5cf909bf6daaced0']

**Name**

e801844333031b7fd4bd7bb56d9fb095f0d89eb89d5a3cc594a4bed24f837351

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' =
'e801844333031b7fd4bd7bb56d9fb095f0d89eb89d5a3cc594a4bed24f837351']

**Name**

b0b9a8e9ec3d0857b70464617c09ffffce55671b227a9fdbb178be3dbfebe8ed

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' =
'b0b9a8e9ec3d0857b70464617c09ffffce55671b227a9fdbb178be3dbfebe8ed']

**Name**

e694f9f7289677adaf2c2e93ba0ac24ae38ab9879a34b86c613dd3c60a56992d

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' =
'e694f9f7289677adaf2c2e93ba0ac24ae38ab9879a34b86c613dd3c60a56992d']

**Name**

dfdca848aecb3439b8c93fd83f1fd4036fc671e3a2dcae9875b4648fd26f1d63

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' =
'dfdca848aecb3439b8c93fd83f1fd4036fc671e3a2dcae9875b4648fd26f1d63']

**Name**

e52b65fdcb77ed4f5989a69d57f1f53ead58af43fa4623021a12bc11cebe29ce

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' =
'e52b65fdcb77ed4f5989a69d57f1f53ead58af43fa4623021a12bc11cebe29ce']

**Name**

e6668c32b04d48209d5c71ea96cb45a9641e87fb075c8a7697a0ae28929913a6

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' = 'e6668c32b04d48209d5c71ea96cb45a9641e87fb075c8a7697a0ae28929913a6']

**Name**

19ffe895b0d1be65847e01d0e3064805732c2867ce485dfccc604432faadc443

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' = '19ffe895b0d1be65847e01d0e3064805732c2867ce485dfccc604432faadc443']

**Name**

8927ff14529f03cbb2ebf617c298f291c2d69be44a8efa4e0406dea16e53e6f9

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' = '8927ff14529f03cbb2ebf617c298f291c2d69be44a8efa4e0406dea16e53e6f9']

**Name**

81a9a6c86b5343a7170ae5abd15f9d2370c8282a4ed54d8d28a3e1ab7c8ae88e

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' = '81a9a6c86b5343a7170ae5abd15f9d2370c8282a4ed54d8d28a3e1ab7c8ae88e']

**Name**

5fffb10487e718634924552b46e717bbcbb6a4f9b1fed02483a6517f9acd2f61

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' = '5fffb10487e718634924552b46e717bbcbb6a4f9b1fed02483a6517f9acd2f61']

**Name**

22803693c21ee17667d764dd226177160bfc2a5d315e66dc355b7366b01df89b

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' = '22803693c21ee17667d764dd226177160bfc2a5d315e66dc355b7366b01df89b']

**Name**

3201785a7de8e37e5d12e8499377cfa3a5b0fead6667e6d9079d8e99304ce815

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' = '3201785a7de8e37e5d12e8499377cfa3a5b0fead6667e6d9079d8e99304ce815']

**Name**

7149acb072fe3dcf4dcc6524be68bd76a9a2896e125ff2dddefb32a4357f47f6

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' = '7149acb072fe3dcf4dcc6524be68bd76a9a2896e125ff2dddefb32a4357f47f6']

**Name**

efbb63f9fa17802f3f9b3a0f4236df268787e3d8b7d2409d1584d316dabc0cf9

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' = 'efbb63f9fa17802f3f9b3a0f4236df268787e3d8b7d2409d1584d316dabc0cf9']

**Name**

117fded1dc51eff3788f1a3ec2b941058ce32760acf61a35152be6307f6e2052

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' = '117fded1dc51eff3788f1a3ec2b941058ce32760acf61a35152be6307f6e2052']

**Name**

9b4dc1e80a4f4c798d0d87a52f52e28700b5b38b38a532994f70830f24f867ba

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' = '9b4dc1e80a4f4c798d0d87a52f52e28700b5b38b38a532994f70830f24f867ba']

**Name**

9c646516dd189cab1b6ced59bf98ade42e19c56fc075e42b85d597449bc9708b

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' = '9c646516dd189cab1b6ced59bf98ade42e19c56fc075e42b85d597449bc9708b']

**Name**

6953ba04233f5cf15ab538ae191a66cb36e9e0753fcaeeb388e3c03260a64483

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' = '6953ba04233f5cf15ab538ae191a66cb36e9e0753fcaeeb388e3c03260a64483']

**Name**

899b0f186c20fdbfe445b4722f4741a5481cd3cbcb44e107b8e01367cccfdda3

**Pattern Type**

stix

**Pattern**

[file:hashes.'SHA-256' = '899b0f186c20fdbfe445b4722f4741a5481cd3cbcb44e107b8e01367cccfdda3']

# Country

| Name |
|---|
| Brazil |
| **Name** |
| Germany |
| **Name** |
| Argentina |
| **Name** |
| Spain |
| **Name** |
| United States |

# Region

| Name |
|---|
| Europe |
| **Name** |
| Southern Europe |
| **Name** |
| Northern America |
| **Name** |
| Western Europe |
| **Name** |
| Americas |
| **Name** |
| Latin America and the Caribbean |

# StixFile

| Value |
|---|
| 5fffb10487e718634924552b46e717bbcbb6a4f9b1fed02483a6517f9acd2f61 |
| 22803693c21ee17667d764dd226177160bfc2a5d315e66dc355b7366b01df89b |
| e694f9f7289677adaf2c2e93ba0ac24ae38ab9879a34b86c613dd3c60a56992d |
| 117fded1dc51eff3788f1a3ec2b941058ce32760acf61a35152be6307f6e2052 |
| b0b9a8e9ec3d0857b70464617c09ffffce55671b227a9fdbb178be3dbfebe8ed |
| e6668c32b04d48209d5c71ea96cb45a9641e87fb075c8a7697a0ae28929913a6 |
| 7149acb072fe3dcf4dcc6524be68bd76a9a2896e125ff2dddefb32a4357f47f6 |
| 8927ff14529f03cbb2ebf617c298f291c2d69be44a8efa4e0406dea16e53e6f9 |
| 6a3455ff881338e9337a75c9f2857c33814b7eb4060c06c72839b641b347ed36 |
| dfdca848aecb3439b8c93fd83f1fd4036fc671e3a2dcae9875b4648fd26f1d63 |
| efbb63f9fa17802f3f9b3a0f4236df268787e3d8b7d2409d1584d316dabc0cf9 |
| 899b0f186c20fdbfe445b4722f4741a5481cd3cbcb44e107b8e01367cccfdda3 |
| acb5de2ed2c064e46f8d42ee82feabe380364a6ef0fbfeb73cf01ffc5e0ded6b |

| |
|---|
| 6953ba04233f5cf15ab538ae191a66cb36e9e0753fcaeeb388e3c03260a64483 |
| 81a9a6c86b5343a7170ae5abd15f9d2370c8282a4ed54d8d28a3e1ab7c8ae88e |
| e52b65fdcb77ed4f5989a69d57f1f53ead58af43fa4623021a12bc11cebe29ce |
| e801844333031b7fd4bd7bb56d9fb095f0d89eb89d5a3cc594a4bed24f837351 |
| 9c646516dd189cab1b6ced59bf98ade42e19c56fc075e42b85d597449bc9708b |
| 19ffe895b0d1be65847e01d0e3064805732c2867ce485dfccc604432faadc443 |
| 1bfc02c985478b21c6713311ca9108f6c432052ea568458c8bd7582f0a825a48 |
| 9b4dc1e80a4f4c798d0d87a52f52e28700b5b38b38a532994f70830f24f867ba |
| e7ffcf1db4fb13b5cb1e9939b3a966c4a5a894f7b1c1978ce6235886776c961e |
| 3201785a7de8e37e5d12e8499377cfa3a5b0fead6667e6d9079d8e99304ce815 |
| 01c56911c7843098777ec375bb5b0029379b0457a9675f149f339b7db823e996 |
| a5de2dc4e6005e75450a0df0ea83816996092261f7dac30b5cf909bf6daaced0 |
| 28a4ae5c699a7d96e963ca5ceec304aa9c4e55bc661e16c194bdba9a8ad847b7 |

# External References

- https://otx.alienvault.com/pulse/658c40da58889532fbfe245c
- https://www.mcafee.com/blogs/other-blogs/mcafee-labs/stealth-backdoor-android-xamalicious-actively-infecting-devices/