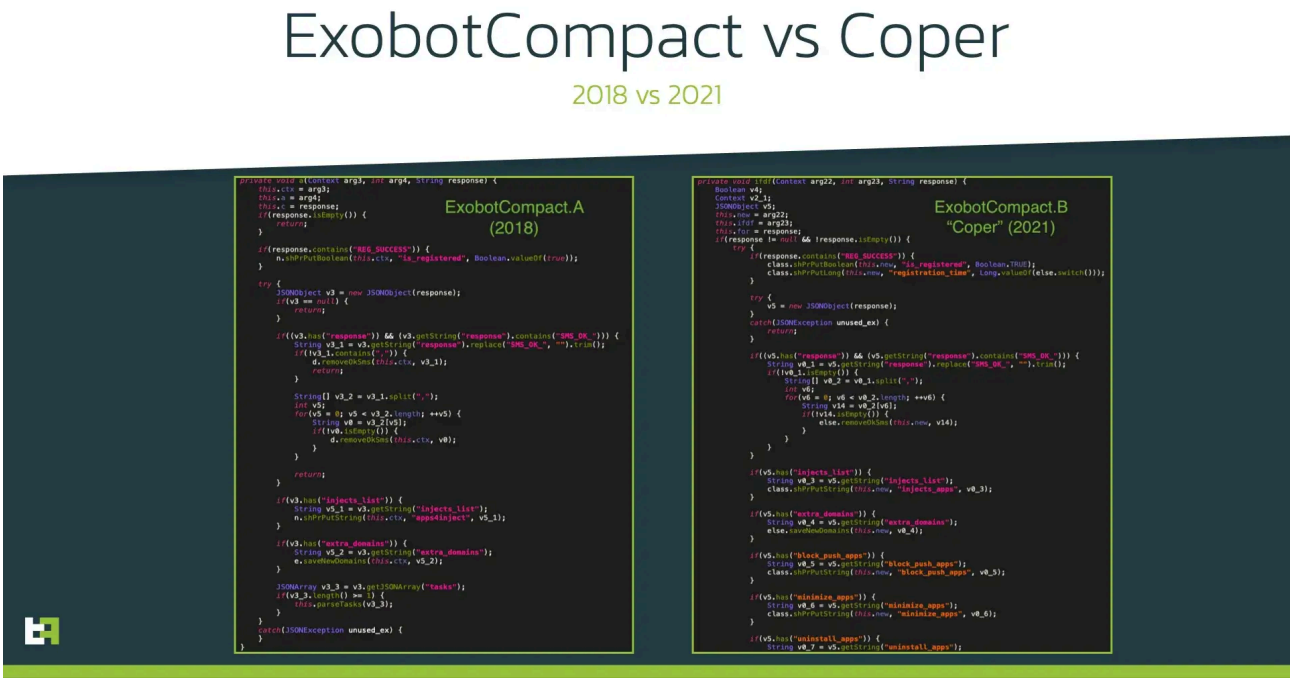
Look out for Octo's tentacles! A new on-device fraud Android Banking Trojan with a rich legacy

Published: 2024-10-01 · Archived: 2026-04-05 13:41:44 UTC

Intro

In mid-2021, a new Android banking malware strain was spotted in the wild. While some AV companies dubbed it as a new family with the name “Coper”, ThreatFabric threat intelligence pointed towards it being a direct descendant of the quite well-known malware family **Exobot**. First observsed in 2016, and based on the source code of the banking Trojan Marcher, Exobot was maintained until 2018 targeting financial institutions with a variety of campaigns focused on Turkey, France and Germany as well as Australia, Thailand and Japan. Subsequently, a “lite” version of it was introduced, named ExobotCompact by its author, the threat actor known as **“android”** on dark-web forums.

ThreatFabric analysts were able to establish a direct connection between ExobotCompact and this newly spotted malware strain, that was dubbed as ExobotCompact.B on our MTI Portal. After some iterations of updates in ExobotCompact, the latest variant was introduced in November 2021, referred to as ExobotCompact.D.



The latest activity of this malware family, and actors behind it, involves distribution through several malicious applications on Google Play Store. These applications were installed more then **50k+ times** and were targeting financial organisations all over the world, both with broad and generic campaigns with large amount of targets, as well as very narrow and focused campaigns throughout Europe.

On January 23, 2022, ThreatFabric analysts spotted a post on one of the darknet forums, in which a member was looking for Octo Android botnet. Further analysis, as it will be shown in this blog, uncovered a direct connection between Octo and ExobotCompact: in fact, ExobotCompact was updated with several features and rebranded to Octo. This blog covers details of attribution made by ThreatFabric analysts and provides more details of Modus Operandi of this Android banking Trojan.

On-device fraud is here

The major update made to ExobotCompact brought **remote access capability**, thus allowing the threat actors behind the Trojan to perform **on-device fraud** (ODF). ODF is the most dangerous, risky, and inconspicuous type of fraud, where transactions are initiated from the same device that the victim uses every day. In this case, anti-fraud engines are challenged to identify the fraudulent activity with significantly smaller number of suspicious indicators compared to other types of fraud performed through different channels.

In general, to get remote control over the device, cybercriminals need screen-streaming to see the contents of the screen and some mechanism to execute actions on the device. To establish remote access to the infected device, ExobotCompact.D relies on built-in services that are part of Android OS: **MediaProjection** for screen streaming and **AccessibilityService** to perform actions remotely. Even though this solution cannot be deemed completely reliable, it is a realistic way to have remote control over the device. Screen streaming with MediaProjection is based on sending screenshots at high rate (1 per second), which gives operator close to live representation of what is happening on remote device.

When ExobotCompact.D receives “start_vnc” command, it parses the configuration sent together with this command:

Option	Description
STREAM_SCREEN	Enables screen streaming with MediaProjection
BLACK	Enables black screen overlay to hide remote actions from victim
SILENT	Disables all notifications (no interruption mode), sets screen brightness to 0

“BLACK” and “SILENT” options help to not raise suspicion in victims as all remote actions and events caused by them will be hidden and performed invisibly. Besides screen streaming, ExobotCompact.D is able to read all the contents of the screen, including elements’ ID, type, and location on the screen. Having this information, the actor is able to re-create the layout of the screen on the C2 backend and have visibility on the internal structure of any app installed on the device. This information is later used when interacting with the remote device to point the element that should be interacted with (i.e., clicked).

Having this real-time visibility, including the internal layout of applications, the operator can send actions to be executed on the device with the help of the “vnc_tasks” command. The supported actions are listed in the table below:

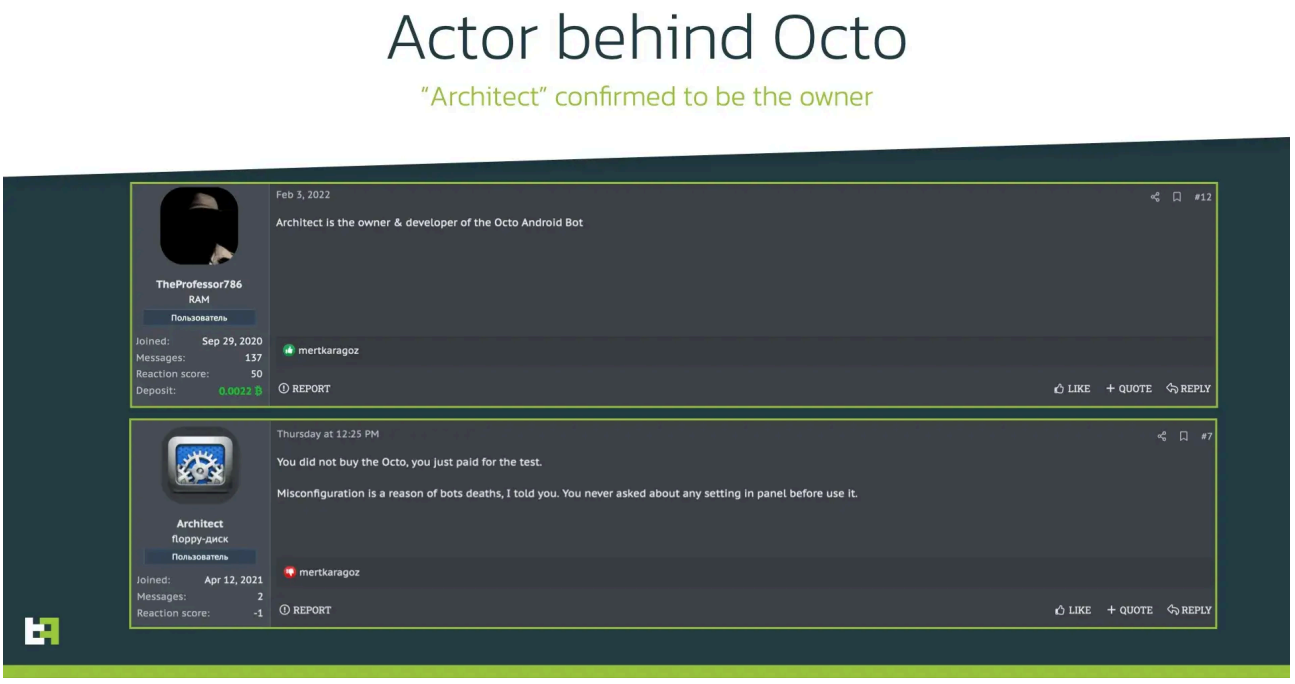
VNC task	Description
click_at	Performs click at specified coordinates X, Y
gesture	Performs gesture
set_text	Sets specified text in specified element
long_click	Performs long click
action	Performs specified action
set_clip	Sets clipboard text to specified one
paste	Pastes data from clipboard
send_pattern	Performs gesture based on the specified pattern
scroll	Performs scroll up/down

We would like to point out that these set of actions that the Trojan is able to perform on victim’s behalf is sufficient to implement (with certain updates made to source code of the Trojan) an Automated Transfer System (ATS). In that case the operator does not have to manually interact with the remote device, but can simply send a sequence of actions to execute. Its execution can lead to automatic initiation of fraudulent transactions and its authorization without manual efforts from the operator, thus allowing fraud on significantly larger scale.

Octo is the new Exo

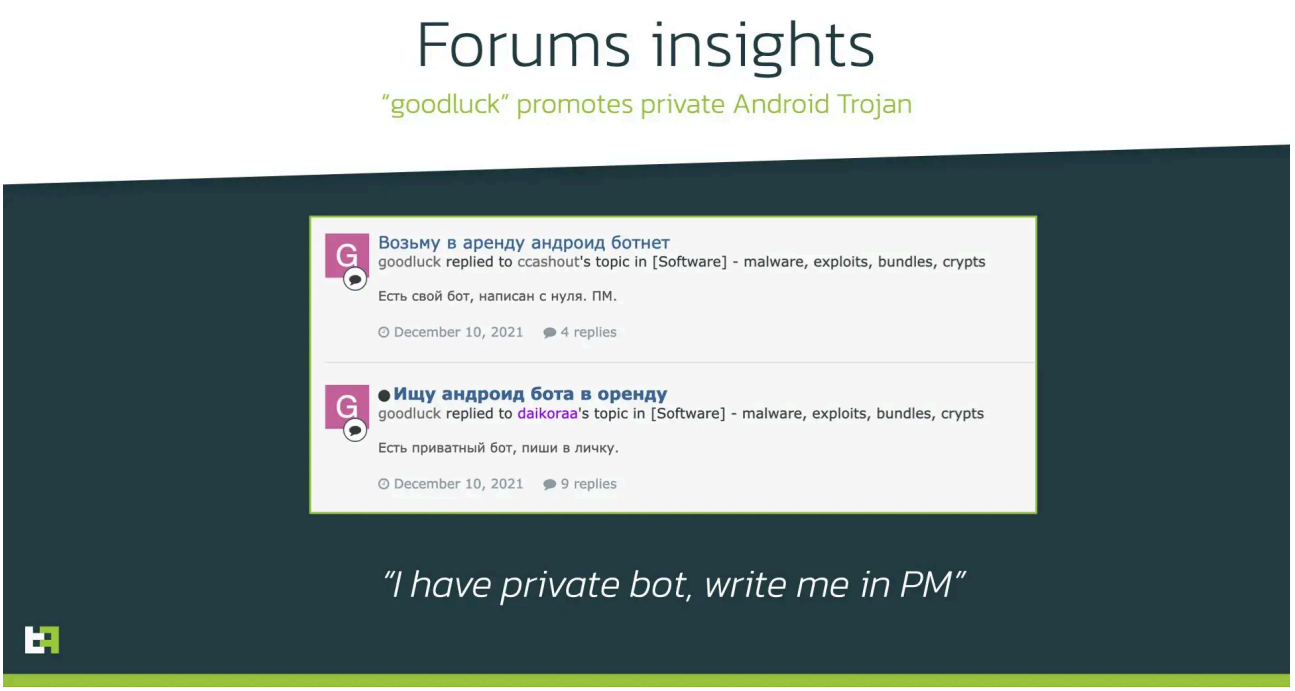
At the time when Octo Android botnet was first mentioned on forums, it was unclear what botnet this was, whether it was some new malware family or just some well-known family rebranded.

On February 3, 2022, another member revealed the owner of Octo botnet, a member of the forum known as “Architect”. Later in March Architect confirmed he/she is the owner and seller of Octo botnet:



Earlier post by Architect reveals his/her skills. A search by telegram contact reveals another nickname used by “Architect” on another forum: “goodluck”.

On this forum, “goodluck” mentioned that he/she has private Trojan written from scratch on December 10:

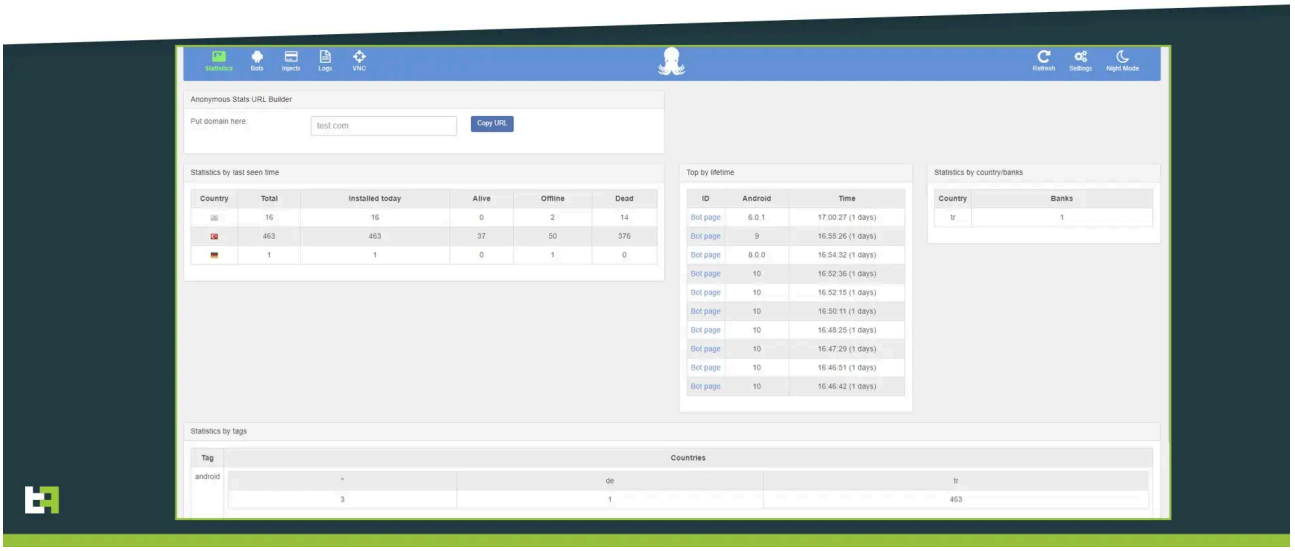


While investigating Octo botnet, ThreatFabric analysts spotted certain similarities between ExobotCompact features and skills of Octo botnet owner, “Architect”:

- “Source code protection from reverse (with native wrapper in C++)” - as we will show in this blog, ExobotCompact uses proprietary payload obfuscation implemented in native library that protects it from reverse engineering.
- “Publication in Google Play with 100% approve” – ExobotCompact was seen distributed by several droppers uploaded to official Google Play store.
- “Disable Google Protect” – one of the first actions that ExobotCompact makes upon the installation.

At this point ThreatFabric analysts made a hypothesis that Octo botnet is a rebranding of ExobotCompact, and “Architect” is either a new owner of the source code or the same actor who was behind Exobot and ExobotCompact.A.

To prove this hypothesis, ThreatFabric analysts examined the supported commands of ExobotCompact, its capabilities and commands available on the administrator panel of Octo banking Trojan.



Here is a summary of our findings:

- Both ExobotCompact and Octo have remote access capability, and it is called “VNC” in both cases.
- Octo panel has six time-based configurations that configure delays before executing some action. This list exact matches the same delays that ExobotCompact can receive from C2. Some of the configurations, like “minimize_delay” or “get_device_admin_delay” are unique and we have not seen it in other malware except ExobotCompact.
- The commands available on the Octo panel are similar to commands supported by ExobotCompact and do not contain any command that is not present in ExobotCompact code.

Thus, having these facts in mind, we conclude that ExobotCompact was rebranded to Octo Android banking Trojan and is rented by its owner “Architect”, also known as “goodluck”. ThreatFabric tracks this variant as ExobotCompact.D.

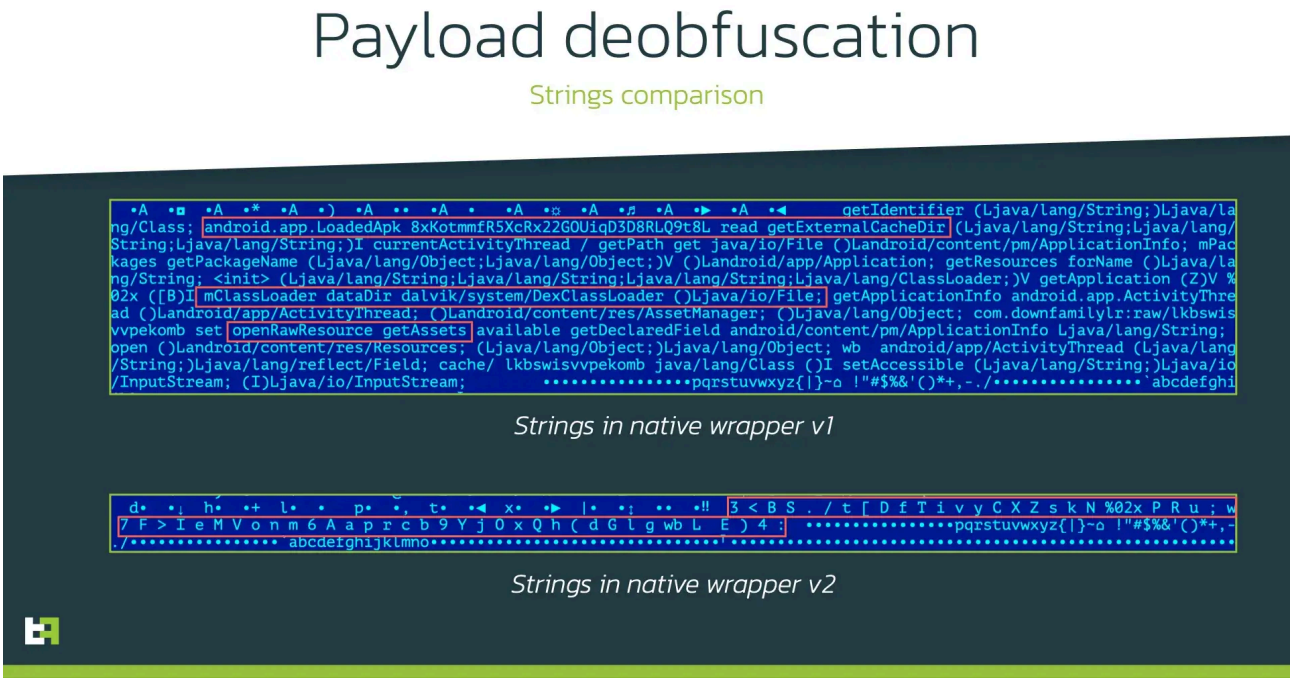
Other capabilities

As highlighted in previous section, ExobotCompact/Octo has several notable features that help it to stay under the radar and perform on-device fraud (ODF). The full list of Octo capabilities is shown hereunder:

Analyzing the current mobile threats landscape, it is hard to point out a malware family that does not use anti-detection and anti-analysis techniques. However, most of threat actors use third-party services that provide malicious payload protection (so-called “cryptors”), while ExobotCompact implements **proprietary payload protection** developed by its author. ExobotCompact.D uses a native library to decrypt and load the malicious payload, which makes it hard to analyze and detect.

Despite the fact that the idea of using **native libraries** for obfuscation is not new, the implementation is quite unique and was only seen used by ExobotCompact. The author of ExobotCompact pays attention not only to development of the new features, but also to improving the payload protection. First versions of native payload obfuscation were rather straightforward: the “decryptor” code was not obfuscated itself, making easy to read and analyze. In the latest versions of this native wrapper author took further step: native code obfuscation. Since a lot of anti-virus solutions rely on signature-based detection, this obfuscation makes it harder for them to detect the malicious activity as native code does not contain “suspicious” string signatures.

The following screenshots show strings in first versions of native wrapper compared to its latest versions:



The obfuscation trick used here is not new and widely used in desktop malware as well as in some Android banking Trojans. Strings are created dynamically during the execution of the native code by concatenating it symbol by symbol as seen in the following screenshot:

Source: https://threatfabric.com/blogs/octo-new-odf-banking-trojan.html