First seen in the wild – Malware uses Corporate MDM as attack vector

By etal

Published: 2020-04-29 · Archived: 2026-04-05 21:22:34 UTC

Research by: Aviran Hazum, Bogdan Melnykov, Chana Efrati, Danil Golubenko, Israel Wernik, Liav Kuperman, Ohad Mana

Overview:

Check Point researchers discovered a new Cerberus variant which is targeting a multinational conglomerate, and is distributed by the company's Mobile Device Manager (MDM) server. This malware has already infected over 75% of the company's devices. Once installed, this Cerberus variant can collect large amounts of sensitive data, including user credentials, and send it to a remote command and control (C&C) server.

General:

This is the first time we have a reported incident of lateral movement inside a corporate network that utilizes the MDM server as a means of spreading.

Malicious actors keep upgrading their tactics and techniques, becoming more and more complex. Cisco's TALOS has reported in the past a campaign using actor-owned MDM to control victim's devices, and this campaign takes it to the next level – compromising a corporate-owned MDM and spreading malware to more than 75% of the corporate's devices via the compromised MDM.

This incident underscores the importance of distinguishing between managing and securing mobile devices. Managing a mobile device means installing applications, configuring settings, and applying policies on multiple devices at once. Securing a mobile device means protecting it from malware threats and attacks.

MDM's most prominent feature, arguably the reason for its existence, is also its Achilles' heel – a single, central control for the entire mobile network. If that platform is breached, so is the entire mobile network.

In this case, we can divide the company's devices into two categories – those with SandBlast Mobile, and those without. The protected device owners did not have access to corporate resources, while the owners of non-protected did – along with an installed malware that allowed the threat actor to control the device remotely.

Encountering the Malware:

On February 18, 2020, we detected two malicious applications installed on a large number of the customer's devices. As all the malicious applications were installed in a very short window of time, we assume that there is some automation involved. Two possibilities came to mind immediately – the first is a malware with lateral movement capabilities, and the second is that the customer's MDM was breached. Later, the customer confirmed that their MDM was indeed breached.

Figure 1*: Two spikes showing a large number of new malicious applications were installed.*

This suggests a targeted attack against the company. After gaining access to the customer's MDM, the attacker utilized the MDM's ability to install applications remotely to install malware on more than 75% of the company's devices.

We started collecting data on the attack itself. We checked the C&C and it is HTTP only. It listens on port 8888, and there is no hostname, just a Russian IP address. The On-device Network Protection's verdict on the C&C is 'Malicious / Infecting Website.' Under the right policy, with On-device Network Protection, all communications with the C&C could have been blocked.

After some initial research, we concluded that this malware is a new variant of the Cerberus Banking Trojan for Android, a known Malware-as-a-Service (Maas) that allows anyone to rent its services to build your own payload, and configure, command and control its devices. But this new variant is equipped with more than the average banker – it has Mobile Remote Access Trojan (MRAT) capabilities.

These capabilities include logging all keystrokes on the device (credentials included), stealing Google Authenticator data and any SMS received (2FA included), and commanding the device remotely via TeamViewer.

We took a closer look at those two samples, and it became very clear that their abilities were almost identical. From this stage onward, we refer to those samples as 'the main module' of the malware. We explore the technical aspect of the malware in the 'Technical Analysis' section.

Technical Analysis:

Malware Capabilities:

Though this malware belongs to the Cerberus Banker family, as we said previously, this campaign has MRAT capabilities and can steal users' information such as call logs, SMS, credentials, and installed applications. This malware consists of two parts – the main application, and a DEX file that is received as a payload from the remote C&C server.

The malware starts by showing a window that masquerades as an update for the Accessibility service. If dismissed, the window keeps popping up until the user accepts the update. The malware then uses the permissions to utilize the accessibility service at a later stage and automatically clicks on menu options and bypasses user interaction.

***Figure 2**: Popup window asking the user to update Accessibility Service.*

The malicious application registers a receiver on various events, and when triggered, starts the execution of the malicious flow.

First, the malware starts a service that sends the C&C its bot-id, lock-screen protection status, top application package name, and the storage write access. Next, the application receives a list of commands to perform – which are configurable by the actor. Then, the application registers a receiver for SMS, and uses the ability to collect incoming SMS messages and send them to the C&C server at a later stage. More information is sent including the phone number, default SMS app, and locale. Once the appropriate command is received, the malware downloads an encoded DEX file, and saves it on the device's external storage as 'ring0.apk'. From this point on, we refer to this as 'the payload module.'

***Figure 3**: Registering receiver to collect SMS.*

***Figure 4**: Download ring0.apk payload.*

After downloading the payload module, the malware creates a JSON object containing various parameters to be used to communicate with the payload module. It then calls an entry point from the DEX payload using reflection.

***Figure 5**: JSON commands used for communication between the main and payload modules.*

The major functionalities of the malware's main module are related to data and file exfiltration as well as taking control of the device.

Malware main module:

The main module can use the accessibility service to steal Google authenticator credentials, Gmail passwords and phone unlocking patterns. This module can send the C&C a list of files and installed applications and can even upload a specific file upon request from the C&C server. In addition, all the user's keystrokes are logged and sent to the server, showing the actor all activities being performed on the device. The malware waits for the Google Authenticator application to be accessed, at which point all available information is read and stored to be sent to the C&C.

***Figure 6**: Uploading a file to the C&C server.*

The main module can also run the TeamViewer remote access application, giving the actor complete remote control of the device. To achieve this, the malware receives connection parameters from the C&C and enters them in the appropriate fields using the Accessibility Service. After the evasion parameters are configured, the TeamViewer application can be run with the malware keeping the device unlocked. When running TeamViewer on Samsung devices, the malware utilizes Samsung KNOX to automatically grant permission.

***Figure 7**: Permission to run TeamViewer in Samsung KNOX.*

To maintain control of the device, the malware can block attempts to uninstall TeamViewer and prevent the user from using this application so as not to interfere with the attacker's actions on the device.

***Figure 8**: TeamViewer related commands.*

The module can also stop the RAT service responsible for receiving commands from the server. This service can later be restarted by the ring0.apk payload module.

The malware maintains persistence using a few techniques. The malware gives itself administrative privileges, complicating the application uninstallation. The malware also blocks attempts to remove the application by automatically closing the App Detail page every time the user tries to access it. In addition, Google Play Protect is disabled, using the Accessibility Service, to prevent detection and removal of the malware.

***Figure 9**: Preventing the malware removal.*

***Figure 10**: Disabling Google Play Protect.*

To keep MIUI (MI user interface) optimizations from interfering with its execution, the malicious application enables Auto Start in the permission center and configures the power keeper so it does not suspend or terminate the application.

***Figure 11**: Configuring MIUI powerkeeper.*

Payload Module:

The downloaded ring0.apk module has various functionalities for stealing data and credentials, performing assorted actions on the device, and functionalities for maintaining the malware itself.
The ring0.apk module can collect all contacts, SMS and installed applications and send it to the C&C. This module also can perform phone-related actions such as sending specific SMS messages, making calls and sending USSD requests. In addition, this module can show notifications, install or uninstall applications and open popup activities with URLs

***Figure 12**: HTML popup to steal user's Gmail credentials.*

***Figure 13**: Stealing Google Authenticator credentials with Accessibility Service.*

The ring0.apk module is also responsible for several self-maintaining changes needed for the malware to function properly. These include granting itself permissions and enabling the service responsible for receiving commands from the C&C. This module can also erase the payload file so an updated payload file can be received at a later stage, remove itself from the device's administrators list, and completely remove the malware from the device.

***Figure 14**: Deleting the ring0.apk payload file.*

The following are the commands the main malware application can receive from the C&C:

	Upload a specific file to the C&C.
open_folder	Send list of files in an external storage to the C&C.
get_apps	Send list of installed applications to the C&C.
rat_disconnect	Stop the RAT service.
device_unlock	Keep the screen unlocked.

open_teamviewer	• Configure evasion parameters for the TeamViewer application. • Start the TeamViewer application
connect_teamviewer	• Store credentials received from the C&C. • Configure evasion parameters for the TeamViewer application. • Connect the TeamViewer application to the attacker's infrastructure.
send_settings	• Configure evasion parameters for the TeamViewer app.

The following are the commands the payload module can receive from the C&C:

patch_update	Delete the payload file.
rat_connect	Enable the RAT service.
remove_bot	Remove itself from device.
run_admin_device	Remove the malware's admin from the administrators list.
request_permission	Allow the malware to grant itself permissions.
grabbing_google_authenticator2	Intercept Google authenticator credentials via the Accessibility Service.
grabbing_pass_gmail	Intercept Gmail passwords by showing a fake window.
grabbing_lockpattern	Intercept the screen unlock pattern by showing a fake window.
get_data_logs	Collect installed applications, contacts and SMS .
send_sms	Sends SMS to a specified number.
ussd	Send USSD request.
call_forward	Call a specific number.
notification	Show notifications.
url	Open popup activity with a specific URL.
run_app	Launch a specific application.
remove_app	Uninstall the application.

Corporate Damage:

Although ONP (On-device Network Protection) was not active, and we cannot be sure which specific pieces of data were leaked, we do know the malware's capabilities. We know that every credential used from an unprotected device was reported to the C&C server. We also know that every SMS message was intercepted. This makes it possible for us to speculate regarding the potential damage for the affected company.
If one unprotected device was used by an administrator who then tried to access corporate resources with his credentials, those credentials, along with any 2FA SMS codes, are compromised.

Due to the extent of this malware's capabilities, the company decided to 'factory-reset' all devices, just to be sure that no residue of the malware, or the actor, remained on the unprotected devices. However, this type of response is extremely costly, both in conducting the damage assessment, and re-establishing the entire mobile network after the factory-reset.

Conclusion:

This campaign demonstrates the importance of understanding the difference between managing and securing mobile devices. While MDM offers an easy way to manage those devices, security cannot be ignored. Mobile devices are an integral part of the way we work, how we communicate, and how our businesses operate. They need to be protected as any other endpoint as they offer a tempting target.
In recent months, we have observed malicious actors running campaigns that are embracing techniques and methodologies from the general threat landscape, using lessons learned from operating on different platforms to enhance operations in the "young" threat landscape of the mobile world.

Appendix #1 – IOC's:

C&Cs

91.210.169[.]114

Package Name	Application Name	sha256
com.wjnjrmigikmpher.efaunxm	Google Play 1.0	4254670ea5f353263570792a8ff4a1e6ea35999c2454fa1ec040786d7be33b69
com.dfxsdgr.qvoor	Google 1.0	6291192d0c2f6318f9a4f345203b35cfe140be53889f9fefdd8e057a4f02e898
com.sakkkwyl.ncceberwpdhfq	GTA V 1.0	3ef8349d4b717d73d31366dfbe941470e749222331edd0b9484955a212080ad8

Source: https://research.checkpoint.com/2020/mobile-as-attack-vector-using-mdm/