ERMAC Malware Back In Action: New Threats And Attack Methods

Published: 2022-05-25 · Archived: 2026-04-05 14:33:09 UTC

Read Cyble's analysis of ERMAC 2.0, a resurfaced Android Banking Trojan that targets over 400 applications.

Cyble Research Labs came across a Twitter post wherein a researcher mentioned the distribution of ERMAC 2.0. ERMAC is an Android Banking Trojan that was first discovered in late August 2021, when it was found targeting Poland. ERMAC 1.0 was capable of stealing the credentials of **378 applications**. The Threat Actor was renting it out for **$3K/month** on a cybercrime forum.

Recently, Cyble Research Labs observed that an upgraded version – ERMAC 2.0 – has been available on underground forums for rent at **$5K/month** and targets **467 applications** for stealing credentials.

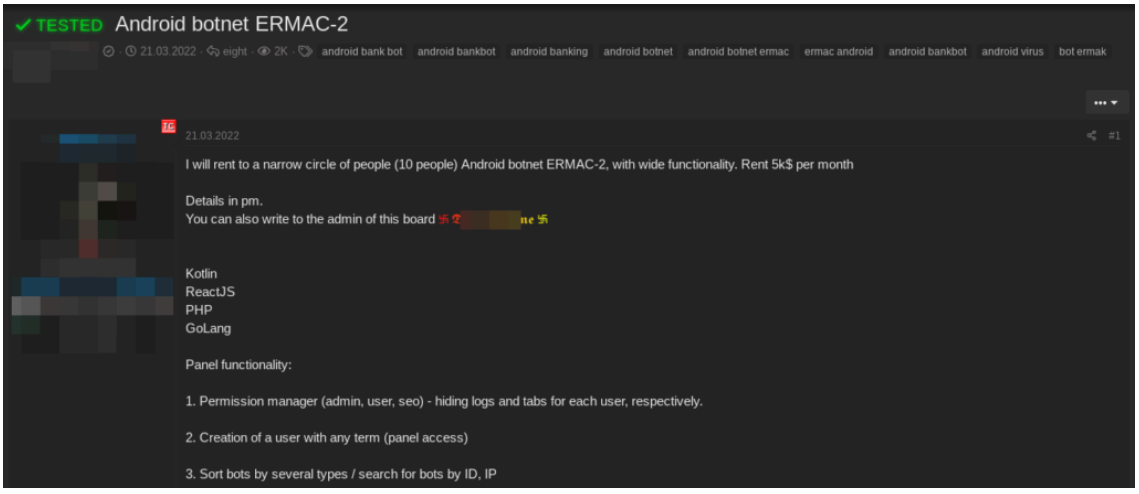
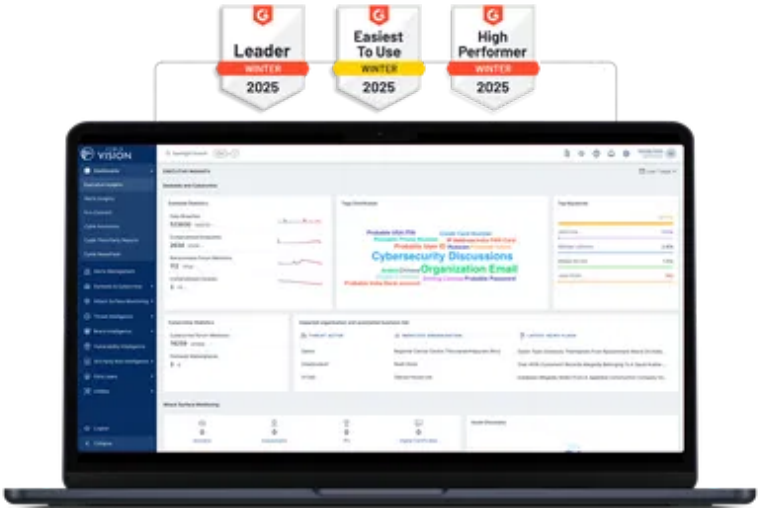


Figure 1 – Cyber Crime Forum selling ERMAC 2.0



We have observed that the ERMAC 2.0 is being delivered through fake sites. For example, via the Bolt Food site – a delivery platform that provides high-quality food delivery services. The fake app impersonates the Bolt Food Android application and targets Polish Bolt Food users.

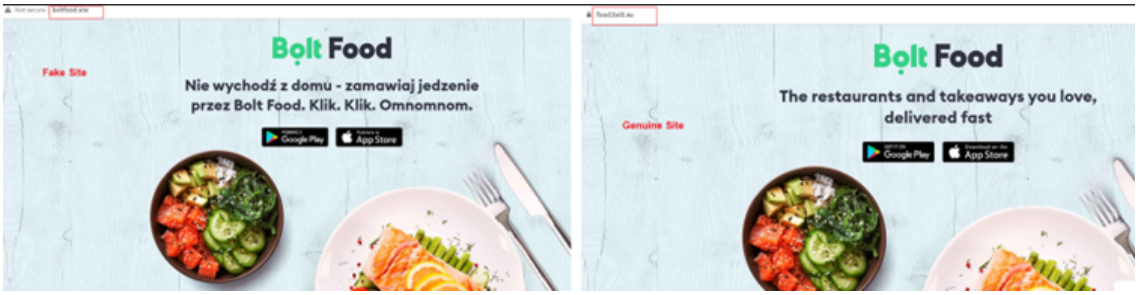


Figure 2 – Fake Bolt Food site distributing malware

Apart from the fake Bolt Food site, EMRAC 2.0 spreads through fake browser update sites, as shown in Figure 3.

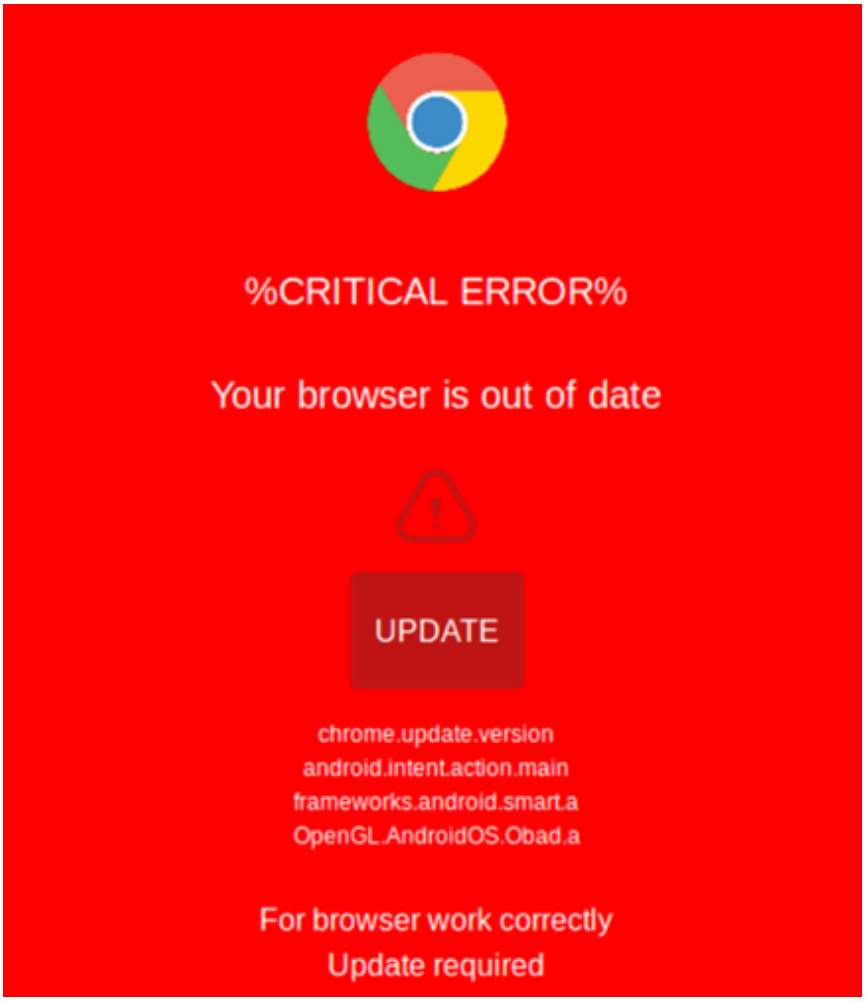


Figure 3 – Fake browser update site (Source – MalwareHunterTeam)

The below image shows the Control Panel of ERMAC 2.0 Banking Trojan. In the UI, the Threat Actor (TA) has named it “ERMVC PVNEL.

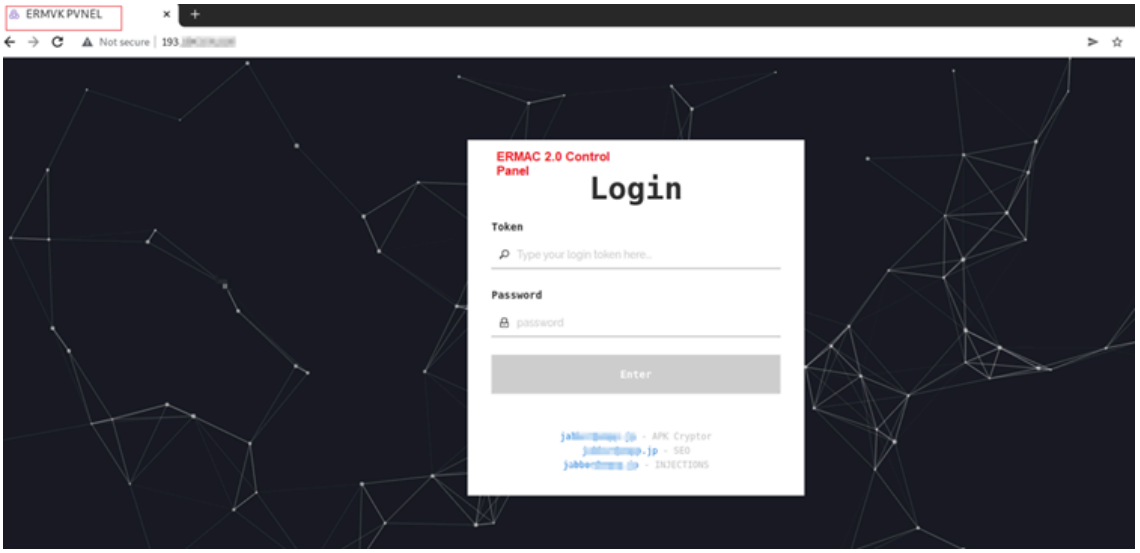


Figure 4 – Control Panel

Technical Analysis

APK Metadata Information

- App Name: **Bolt Food**
- Package Name: **com.kototomumeci.nacipiyi**
- SHA256 Hash: **2cc727c4249235f36bbc5024d5a5cb708c0f6d3659151afc5ae5d42d55212cb5**

Figure 5 shows the metadata information of an application.

Figure 5 – App Metadata Information

Manifest Description

The malicious application asks for **43** permissions, of which the TA exploits **12**. The malware's harmful permission requests are listed below:

Permission	Description
REQUEST_INSTALL_PACKAGES	Allows an application to request installing packages

CALL_PHONE	Allows an application to initiate a phone call without going through the Dialer user interface for the user to confirm the call
RECEIVE_SMS	Allows an application to receive SMS messages
READ_SMS	Allows an application to read SMS messages
SEND_SMS	Allows an application to send SMS messages
READ_CONTACTS	Allows an application to read the user's contacts data
READ_PHONE_STATE	Allows read access to the device's phone number
SYSTEM_ALERT_WINDOW	Allows an app to create windows shown on top of all other apps.
READ_EXTERNAL_STORAGE	Allows an application to read from external storage
RECORD_AUDIO	Allows an application to record audio
WRITE_EXTERNAL_STORAGE	Allows an application to write to external storage

Source Code Review

Apart from the application's subclass, the rest of the components identified from the Manifest file are missing. We can thus infer that the application is packed.

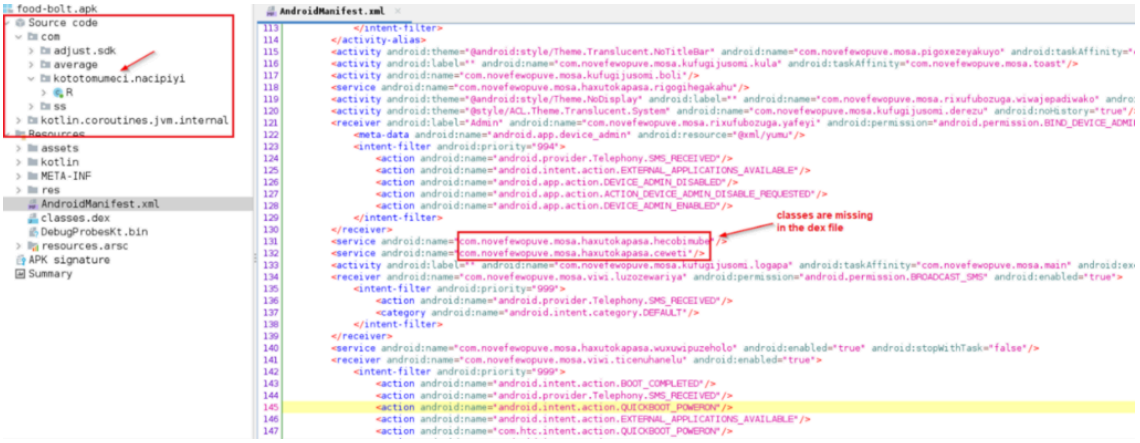


Figure 6 – Manifest File

Upon execution, the malicious application unpacks the DEX file present in the assets folder and then loads the classes.

In this case, the dropped dex file name is "*pqiRsn.json,*" which has all the missing classes

The strings present in the classes are encoded using base64 and encrypted using the AES-128-bit algorithm.

The Secret Key and IVparameter were dumped during dynamic analysis. Both Secret Key and IVparameter are used to decrypt hardcoded strings present in the file as well as encrypt the data sent to the C&C server. The below image explains the decryption process.

Figure 7 – Encryption and Decryption Technique

On installing the application, it prompts the user to turn on the Accessibility Service. When the victim grants this permission, it starts abusing services by auto-enabling overlay activity and auto-granting permissions.

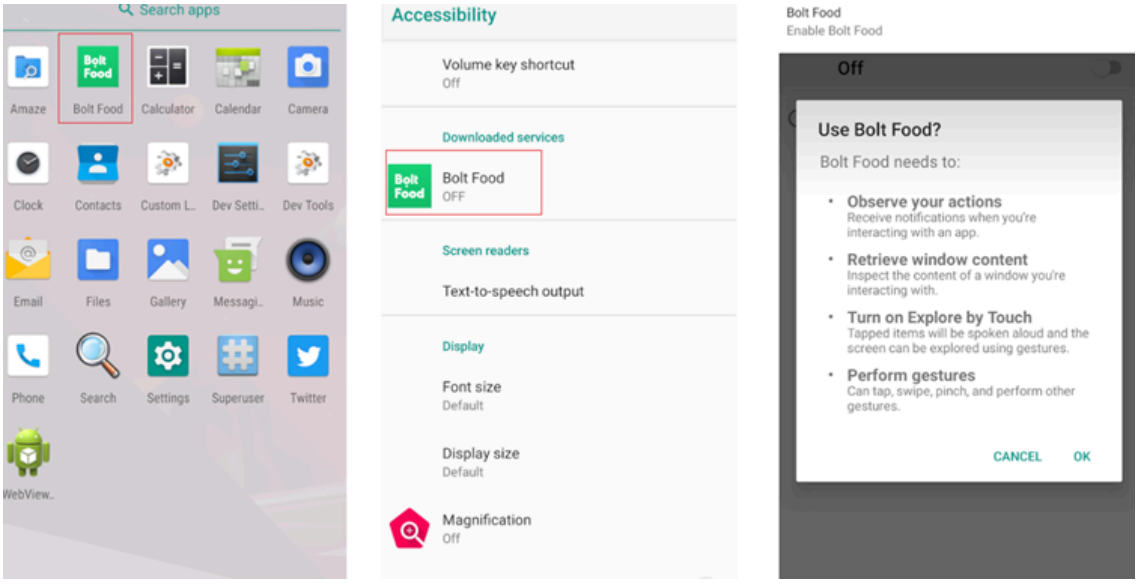


Figure 8 – Accessibility Service

After granting the Accessibility permission, the malware sends a list of installed applications on the victim's Android device to the C&C server. The malware then downloads and installs the injection modules of targeted applications based on this application list.

Figure 9 showcases the C&C communications from the victim's device, which sends the details of installed applications and receives a response, including a list of targeted applications to perform overlay activity. At the time of our analysis, it was observed that the "*Unocoin*" wallet was the targeted application by the attacker.

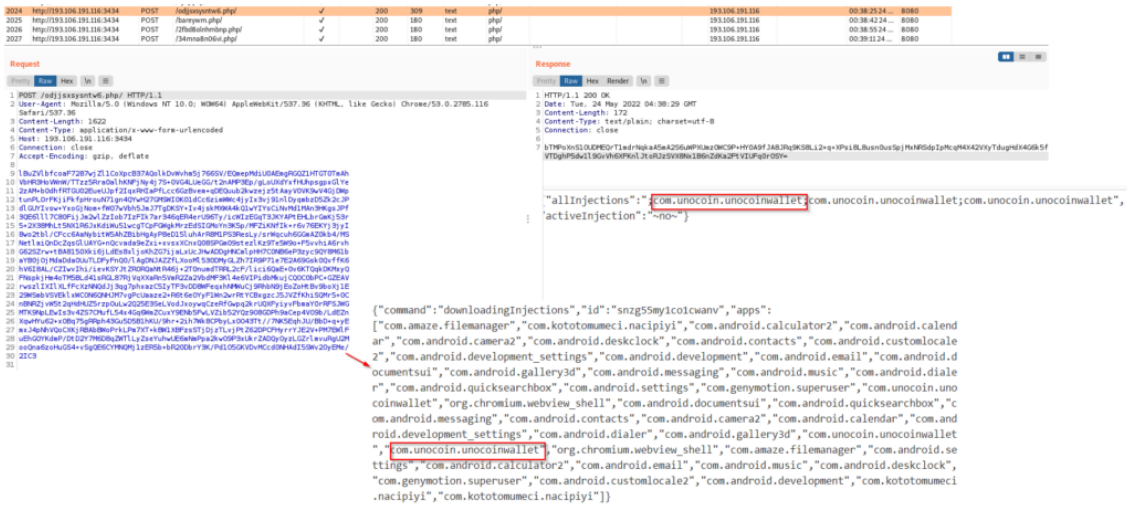


Figure 9 – Sending application list and receiving the response for the targeted app

The malware then receives an encrypted HTML phishing page, which will be decrypted and stored into the Shared Preference file named "*setting.xml*" with the action status, as shown in the figure below.

Figure 10 – Downloaded injection saved in Shared Preference file

As soon as the victim interacts with a targeted genuine application, the injected phishing page is displayed through *setting.xml* that can be used for credential harvesting.

Figure 11 shows the phishing page for "*Unocoin.*" The malware sends the *start_inject* command to the C&C server before sending the credentials.

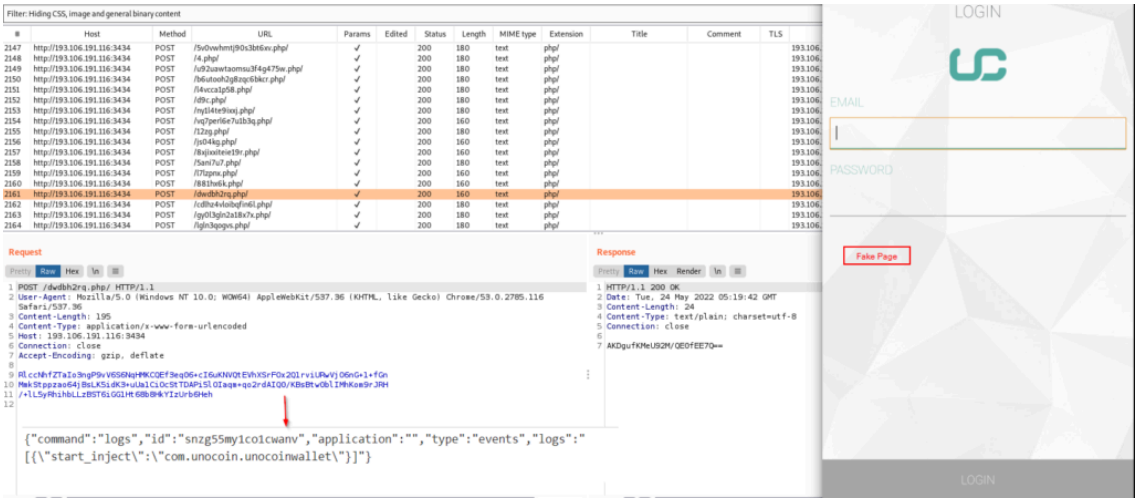


Figure 11 – Loading phishing page

The harvested credentials are sent to the same C&C server in the encrypted format, as shown in Figure 12. ERMAC 2.0 uses the command *logs* for code injection as well as data exfiltration functionalities.

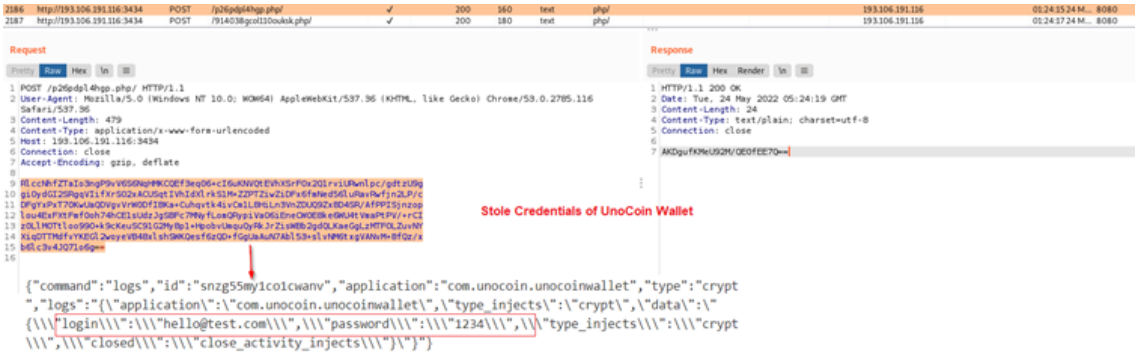


Figure 12 – Sending credentials to the C&C server

The TA can then use these credentials to steal cryptocurrency from the victim's account.

The below image shows the TA's phishing pages used to trick the victims into falling for a phishing scheme while attempting to access genuine applications.

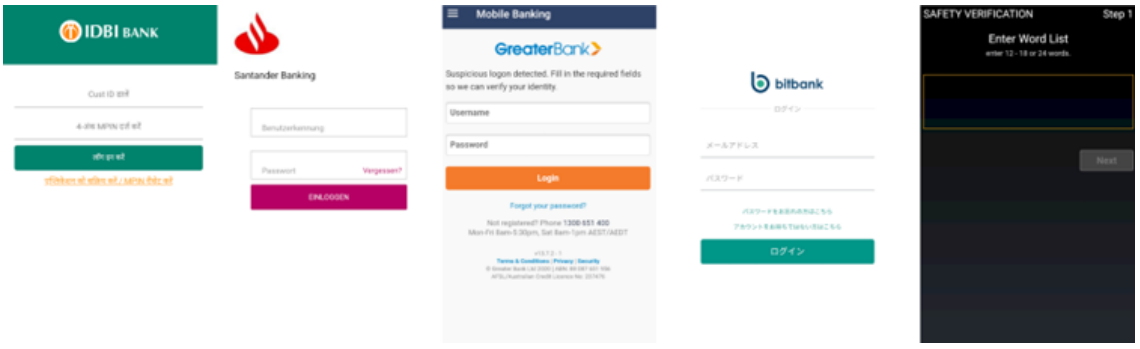


Figure 13 – Fake injected window targeting different applications

Cyble Research Labs witnessed that the malware has functionalities to target various banking applications of several banks worldwide.

The commands used by ERMAC 2.0 to execute malicious operations are:

Command	Description
downloadingInjections	Sends the application list to download injections
logs	Sends injection logs to the server
checkAP	Check the application status and send it to the server
registration	Sends device data
updateBotParams	Sends the updated bot parameters
downloadInjection	Used to receive the phishing HTML page

Conclusion

The Threat Actor behind ERMAC used the leaked code from a well-known malware variant named “Cerberus” and modified the code to sell the Android botnets in cybercrime forums. Interestingly, we observed that ERMAC 2.0 is distributed rapidly through various phishing sites, primarily targeting Polish users.

ERMAC 2.0 steals credentials from different crypto wallets and targets multiple banking applications worldwide. We foresee that the TA behind ERMAC 2.0 will continue to develop new versions with more targeted applications, new TTPs, and new delivery methods.

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 the alerts provided by Anti-viruses and Android OS and take necessary actions accordingly.

What to do when you are infected?

- Disable Wi-Fi/Mobile data and remove SIM card – 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 Mean.
Initial Access	T1444	Masquerade as Legitimate Application
Defense Evasion	T1406	Obfuscated Files or Information
Credential Access	T1412	Capture SMS Messages
Discovery	T1421	System Network Connections Discovery
Command and Control	T1571	Non-Standard Port
Command and Control	T1573	Encrypted Chanel
Collection	T1432	Access Contact List
Collection	T1507	Network Information Discovery

Indicators of Compromise (IOCs)

Indicators	Indicator Type	Description
2cc727c4249235f36bbc5024d5a5cb708c0f6d3659151afc5ae5d42d55212cb5	SHA256	Hash of the analyzed APK file

301e2ab9707abe193bb627c60f5e4b8736c86fe9	SHA1	Hash of the analyzed APK file
1e0586aef0f106031260fecb412c5cdf	MD5	Hash of the analyzed APK file
hxxp://bolt-food[.]site	URL	Malware distribution site
hxxp://193[.]106.191[.]116	URL	C&C Server
df298b0aba5aad2886ae720577557b3e48fba905055dcee0fd74336660bfd0a2	SHA256	Hash of the analyzed APK file
e2fb7981688060fc672f844c65e89d12f3e5cafe	SHA1	Hash of the analyzed APK file
1bb6da78e3c379afde1978aecfa067b8	MD5	Hash of the analyzed APK file
hxxp://boltfood[.]site	URL	Malware distribution site
df298b0aba5aad2886ae720577557b3e48fba905055dcee0fd74336660bfd0a2	SHA256	Hash of the analyzed APK file
fe4a7d079cc00e730412c7a6e0b177829ee58a73	SHA1	Hash of the analyzed APK file
65f634ef24fd686225aa4765fc63fe2b	MD5	Hash of the analyzed APK file
hxxp://apkphoto.[co].NZ	URL	Malware distribution site

hxxp://45[.]141.85[.]25	URL	C&C Server

Source: https://blog.cyble.com/2022/05/25/ermac-back-in-action/