Securemetric Technology Sdn. Bhd. (759614-V)
Level 5-E-6, Enterprise 4, Technology Park Malaysia, Lebuhraya
Puchong-Sg. Besi, 57000 Bukit Jalil, Kuala Lumpur, Malaysia
T +603 8996 8225 F +603 8996 7225

CENTAGATE® Soft Token Security Features

CENTAGATE® soft token solution (mobile token) comprises of standard CENTAGATE mobile application and CENTAGATE mobile SDK. CENTAGATE soft token contains multiple security features to counter threats found in the Android and iOS eco-systems and are outline further below.

All security features can be turned ON and OFF. And in case of building soft token based on CENTAGATE mobile SDK, the security features may be limited in order to serve customer's specific soft token functionality.

1. Root detection and prevention

Rooting is the process of circumventing security measures of the operating system. This is usually performed by the users of a device in order to customize the device beyond of what the manufacturer allows.

Attackers can 'root' a device in order to bypass the Android application sandbox. This can allow access to data that is stored on the device which would otherwise have been access restricted. Similarly, malware can exploit known weaknesses in Android to gain elevated permissions on a device while running.

Root detection can be performed in several ways, such as checking for well-known indicators of root files, processes and other anomalies. Root detection is inherently a "cat-and-mouse" game between new rooting techniques, and rooting detection methods.

Scope of protection: CENTAGATE soft token implements several layers and levels of root detection mechanisms to handle most well-known approaches to more heuristics type indicators that are looking for symptoms of a rooted device rather than conclusive evidence. Newer versions of Android restrict the ability to detect certain types of rooting due to a stricter app security sandbox being enforced on the detection mechanism itself.

2. Jailbreak detection and prevention

Jailbreaking is the process of circumventing security measures of the operating system. This is usually performed by the users of a device in order to customize the device beyond of what the manufacturer allows.

However, attackers can also perform jailbreaking in case a device is stolen to bypass the protection mechanisms of the device in order to gain access to the data that is stored on the device. Similarly, some of the available jailbreak techniques can be used by malware to gain extended permissions on a device.

Since a jailbroken device is much more at risk of being compromised, it is important to know about it.

Jailbreak detection can be performed in different ways. Very naive approaches simply test for the existence of files in the file-system that are associated with a jailbroken device. Several 'jailbreak hider' tools are available on iOS and shows how easy it is to bypass these detection tools.

Scope of protection: CENTAGATE soft token, while also implementing these naive detection mechanisms performs detection on multiple levels ranging these well-known approaches to cutting edge low-level mechanisms that are targeted more towards detecting the essence of a jailbreak.

3. Anti-repackaging

Repackaging (or Repacking) has become a common practice on Android and iOS in the recent years and means that an attacker can obtain a copy of the application, add malicious functionality such as a keylogger to it, and then offers it to users who believe that they are using the original application.

This type of attack is made possible since there are many alternative distribution platforms in addition to the official Google Play store, and since Apple has an alternative distribution mechanism apart from the App Store called enterprise distribution.

The act of repackaging the application is also used when attempting to reverse engineer the application.

Scope of protection: CENTAGATE soft token detects, when an application has been repackaged. Part of the protection implements digital signature and/or internal structure binding, so the security features cannot be bypassed easily.

4. Code injection prevention

In order to gain control of an application, attackers may inject code into the application to control it from within its own process. This can for example be used to read encrypted SSL communication, or to intercept user input such as passwords.

This type of threat is elevated on rooted or jailbroken devices since injecting code into another application would otherwise have been prevented by the operating system sandbox.

Scope of protection: CENTAGATE soft token can be configured to detect the presence of code hooks, as well as typical code injection frameworks such as Xposed or Cydia Substrate. In some cases, also block injection of code into the process.

On iOS, there are two known ways to inject code into an application: Either during load time (which is done by tools like 'MobileSubstrate') or during runtime (which is done by tools like 'Cycript'). CENTAGATE soft token can be configured to detect load time injection and is able to block and detect runtime injection.

5. Hook detection and prevention

Scope of protection: In case the code injection protection can be bypassed by an attacker, CENTAGATE soft token can also be configured to detect actions that the attacker (usually in form of a hooking framework) performs inside the process.

6. Secure Storage

The authentication solution requires distribution and storing keys and seeds inside CENTAGATE soft token.

Scope of protection: CENTAGATE soft token uses encryption to protect user keys and seeds. Encryption keys are derived from hardware identification, user PIN.

7. Anti-cloning (device binding)

In order to mimic the real application, attackers clone a part or whole device content into a new one. Attacker can also get user cryptographic keys after cloning.

Scope of protection: CENTAGATE soft token uses binding to hardware identification, and Secure storage to prevent cloning.

8. Anti-debugging

Debuggers can be used during run-time of the application to extract sensitive information, perform code injection, alter the program flow and help attackers reverse engineer the application.

Scope of protection: CENTAGATE soft token can be configured to prevent debuggers from attaching to the application by actively blocking such debuggers. This is implemented by launching one trusted guard process and by letting this component act as soft token's own debugger.

CENTAGATE soft token supports advanced debugger protection by executing a security handshake with the guard process. If the guard component is circumvented by an attacker, this feature may block further usage of third-party debuggers.

9. Prevent access data in memory

Scope of protection: CENTAGATE soft token can mitigate the unauthorized access to data in memory in several ways:

1. Data at rest: by using Secure Storage feature.
2. Data in motion: by implementing Anti-debugging feature.

10. • Emulator detection and prevention

Emulators can be used to analyze an application to determine how it works, and used to extract sensitive information that is available while the application is being executed.

Scope of protection: CENTAGATE soft token can detect that the application is being executed in an emulator or typical virtual environment.

11. Key logger protection

Android offers its users the possibility to install custom software keyboards. These keyboards are naturally being informed about every input, the user makes on it and can be used by an attacker as a key logger.

Scope of protection: CENTAGATE soft token can be configured to white list trusted software keyboards to determine if the used software keyboard is trustworthy or not.

Most keyloggers on iOS are implemented using code injection. CENTAGATE soft token protects against these with its code injection prevention feature.

12. Screenshot protection

Applications often display sensitive information that should not be easily exfiltrated from the application.

One easy way to extract information from an application is in form of a screenshot.

Screenshots are taken in different ways: They can be initiated by the user, they can be taken by the system for various reasons and they can be taken by screen mirroring.

Scope of protection: CENTAGATE soft token can be configured to detect when the user takes a screenshot of the application, is able to block screenshots taken by the system and is able to block screen mirroring, so screen shots appear to be a black rectangle.

13. Screen reader protection

Screen readers are using the Android Accessibility APIs in order to interact with the application. These APIs were intended in order to create accessibility aids, such as text-to-speech and other types of accessibility tools.

Malware can also be installed and activated as a screen reader, which means that it potentially can interactively and potentially remotely control the device and application.

Scope of protection: CENTAGATE soft token can configured to detect if any untrusted screen reader is active, and actively block these from receiving data from the app. Trusted screen readers may be white listed so the accessibility functionality is retained for specific screen readers.

14. Code Obfuscation

Scope of protection: CENTAGATE soft token can be configured to have full app obfuscation where class names, function names and field names are modified.

Notice (for apps based on CENTAGATE mobile SDK): enabling full app obfuscation is in most cases a non-trivial task, that requires knowledge of the inner workings of the app, and all its components including any third-party libraries.

Java is a dynamic language which is used extensively with reflection by Android itself and also by many libraries. The use of reflection is commonly causing problems for obfuscation tools.