Everisk Technology

Mobile App Threat & Risk detection and monitoring

Table of Contents

1 Overview

Today rapid adoption of mobile devices and the explosion of mobile apps created a significant security challenge for developers or organizations. IT security teams are now responsible for mobile app security but often don't have the resources and skills to thoroughly assess the risks. In addition, mobile apps are an easy target for hackers, putting your customers' private data at risk.

1.1 Objective

Everisk developed to detect, monitor & block the mobile application threats for financial services institution, mobile game developer or publisher, government and enterprises. Everisk provide relevant threat information & statistic for user to analyze mobile application or threat data, and precisely identified sources of threat events.

Three key functions in Everisk as below,

- Threat Analysis: Determine the threat event, the attack target, process and the source of the attack
- Threat Control: Attack event management based on application, version, device and etc.
- Threat Statistics: Attack event statistic based on location, application version, system, date & time and etc.

2 Product

2.1 Real-time Threat Alert and Protection

If a Hacker successful crack the mobile application or devices, they will beneficial from the sensitive data retrieved before network security team discovers the intrusion and response to the incident. Everisks is a platform to provide real-time mobile application or device threat monitoring to discover the hacker's activities immediately and alert network security team for incident handling.

2.2 Comprehensive Threat Analysis

Everisk threat analysis is carried out on system layer, application layer and business layer where attack analysis will focus on the sources of attack (inclusive of network information, device information, application information, system information & etc.), attack target (refer to application information), time of attack event, attack flow and attack techniques for threat analysis and statistic.

The combination of Everisk threat perception sensor and business model analysis deliver an end to end threat analysis platform to monitor, detect & blocking threats in real-time.

3 Product Functions

3.1 Threat Detection and Analysis

Typical business owners focus their security on the data analysis of server, web application security but not for mobile terminal. Threat Detection & Analysis collect all the information and present it as analyzed data, provide full visibility toward the threats and risk on your mobile application environment. Furthermore, Threat Detection & Analysis enhance the security by analyzed business application, system, network, configuration, devices & etc to discover the potential risks and isolates the threat from the environment.

3.1.1 Product Components

- **Risk Detection**: Detect potential risks on mobile application environment in real-time and covered the following layers: application layer, system layer, network layer, configuration layer and hardware layer, etc. Everisk send an alert to administrator when risk detected and response to the alert base on the strategy configured.
- **Threat Analysis**: the threats is precisely located through behavior analysis in combination of risk perception results, and threat details such as attack source, attack method, attack target, time and place, etc will be feedback in real time. There is several response modes provided to block the threat.

3.1.2 Product Features

3.1.2.1 Emulator Analysis

Emulator frequently used by hacker when they need to change the hardware information to automatically running app-related cheating tools. Take an example, running cheating tools and also mobile game in Bluestacks, Droid4x and other simulator environments, or modification of customized version of simulator.

Detection mechanisms as below,

- **Configuration-based detection**: detection base on configuration information of emulator, e.g. configuration of Wi-Fi link, SMS function,

SD card access, and etc. This detection is not applicable to customized emulators.

- **System-based detection**: detection base on specified emulators, e.g. identification of the name of operating system, version information, etc. This detection is not applicable to customized emulators.
- **Low-level instruction based detection**: detection base on low-level instructions sent to the system and information response, e.g. CPU instruction set and cache mechanism. This detection is applicable to all type of emulators.
- **Hardware characteristics-based detection**: detection base on the characteristics of hardware, and information response, e.g. is hardware consist of SD card slot, SIM card slot, Wi-Fi device and sensor. This detection is applicable to all type of emulators.
- **Detection base on big data analysis**: detection base on suspicious system characteristics and discovered the issues on big data analysis, take an example IMEI, IMSI, MAC, BluetoothMac, etc. is where we can distinguish emulator and real machine. This detection is applicable to all type of emulators.

Low-level instructions set and hardware characteristics are the main differences of emulator and device terminal, this will makes the detection of emulator is more precise.

3.1.2.2 Attack Framework Analysis

Hacker modifies app with attack framework such as xposed, substrate, WSM, etc. and typically app is modified by zygote process injection.

Detection mechanisms as below,

- **Zygote-based detection**: detect the change of zygote during the initialization process to identify the attack framework. This detection is applicable to all zygote attacks.
- **Detection of memory modification**: detect if key information in the memory is modified. This detection is applicable to all attack frameworks.

3.1.2.3 Geo-location Fraud Analysis

Location spoofing is one of the techniques used by hacker to perform the geo-location fraud; hacker modified geographic location and relevant information through various ways and the fake location will be presented.

Detection mechanisms as below,

- **Hijack-based detection**: detect if key geographic location information is hijacked and applicable to all GPS hijacking.
- **Tamper-based detection**: detect if key geographic location information is tampered and it is applicable all fraud cause by location data tampering.

3.1.2.4 Domain name Scam Analysis

- **Detection base on local host information**: analyze local host configuration information and determined if the host is being hijacked or tampered.
- **Domain name white/blacklist**: user can configure blacklisted or white-listed domain name based on business needs.

3.1.2.5 Device Repurposing Analysis

Hacker able to repurposing a device to another device by modifies device information through various ways and resulting in the fake device will be used for legitimate proposes, e.g. tampering of IMEI, IMSI, MAC, etc.

Detection mechanisms as below,

- **Detection base on device fingerprint**: an alert will be send when there are multiple device hardware information, software information and system information detected on the same device fingerprint.
- **Cache-based detection**: determine if the device is repurposed by record cache module characteristic and detect the characteristic changes.
- **Visualized correlation analysis detection**: correlation analysis conducted for defective device and alert will be send.

3.1.2.6 Memory Space Protection

Detect if memory space is read and written by unauthorized party to ensure the completeness and controllability of App memory space.

3.1.2.7 Debugging Behavior Analysis

Debugging is the famous technique used by hacker to retrieved data or information and hacker can debug, modify and inject code into the process through various ways. Common ways used by hacker is using ptrace attach to the process and affect the memory process by modifying files or data in memory.

Detection mechanisms as below,

- **Detection base on hijack activities**: detect if system functions are hijacked. This detection is applicable to all hijack activities.
- **Detection base on memory modification**: detect if mapping files and key information in the memory are modified. This detection is applicable to all memory modification and injection process.
- **Detection base on debugging activities**: detect if the application has been debugged. This detection is applicable to all debugging activities.

3.2 Transaction Fraud Detection

Fraud prevention in financial services institution is generally focus on the security of transaction environment is analyzed and identity authentication by using user's persona to secure the entire transaction process.

3.2.1 Security Components

- **Environmental threat index**: evaluated the environmental threats encountered by the application based on system environment security risk level indexes were executed by the application.

- **Application threat index**: index obtained through the evaluation of threats encountered by the application, and threat level represent the application may be hijacked or tampered.
- **Identity authentication**: user persona created by device configuration and user behavior such as the behavior of click interval, sliding speed and other information. Identity authentication base on user persona will be enforced on user registration, login, payment, etc.

3.2.2 Product Features (User Personas)

3.2.2.1 Sliding Behavior Persona

Sliding characteristics such as sliding distance, track, speed and other dimensions are calculated, and determine if the operation is perform by the same user by comparing current sliding characteristics with historical sliding data.

3.2.2.2 Clicking Behavior Persona

Separate screen or keypad into several areas, and click persona is recorded based on click behavior on the areas. Historical data used to determine if it is operated by the same user.

3.2.2.3 Touch Behavior Persona

Touching areas by the user for business process will be recorded and historical data is compared to determine if it is operated by the same user.

3.2.2.4 Input Interval

Detection on the input time interval during user enters sensitive information such as username, password, bank account number or credit card number. User's input speed is obtained through input time interval, starting and ending time interval and other information, compare with historical data to determine if it is operated by the same user.

3.2.2.5 Gyroscope Information

Real-time fluctuation change data of gyroscope is obtained to analyze the user's operation habit during perform business transaction such as inclination of device, fluctuation of gyroscope during click operation, fluctuation of gyroscope during sliding operation, and establish the operation habit model based on business transactions, and compare with historical data. An alert will be send when operation habit is detected abnormal.

3.2.2.6 Handedness

Sliding data, gyroscope data, click data and other information collected to conduct multi-dimension analysis, and user's handedness persona will be created to determine change of users to some extent.

3.2.2.7 Light Sensation Information

Detection based on the information of light sensor, and optical data analysis is conducted to determine if the environment where user located is normal or abnormal.

3.2.2.8 Conventional Network Environment

Detect the network environment where the user is located is frequently used network environment, e.g. whether it is resident area (by location) and used Wi-Fi consistently, and etc.

3.2.2.9 Conventional Geographic Area

Geographic location data is used to determine if the user is in frequently used geographic area, e.g. resident area within working hours, resident area beyond working hours, etc.

3.3 Online Scalping Detection

Ticket Bot operators use software program to purchase the in-demand train, flight, sporting or music event tickets from ticket selling sites on bulk and these tickets are sold on secondary ticket reseller or end user at an obnoxiously high price. Online scalping detection is design specifically to prevent against these illegal operations.

3.3.1 Security Components

- **Fake device detection**: emulator or unauthorized procedure (such as scripting and tampering on client software) identification by detecting changes on hardware configuration, device system and operation behavior.
- **Repurposing device detection**: detect if device system being debugged and modified.
- **Behavioral dynamic detection**: detect ticket scalping on module call, API call, and function call and form a specific rules for dynamic detection.
- **Transaction environmental security detection**: detect threats, identify malicious application, root environment and other risks in the app executed environment and ensure the reliability for the transaction environment.

3.3.2 Application Crash Detection

It is difficult to determine an encrypted application crashed is happen on encryption initialization stage or running stage, therefore a new loader will be generated in the device to monitor the whole process and collection application crash information.

4 Everisk Self-Protection

Everisk using White-box cryptography solution (WhiteCrypto) to protect their communication channel by encrypts the cryptography keys.

WhiteCrypto is widely used for DRM (digital rights management) on commercial software protection and embedded system software protection. A very successful use case is Skype successfully protected its communication protocol by using white-box cryptography solution to prevent hacker write some software to communicate with Skype and selling it on lower price, and maintain its leading position in VoIP industry.

4.1 WhiteCrypto Key Store

WhiteCrypto generates the white-box library and crypto-key for user to integrate and use. At about 200 bytes, the white-box crypto-key is very small, hence updating the crypto-key becomes extremely convenient, and a dynamic crypto-key update mechanism is supported. The user can achieve each device one key, one key each time for crypto-key management.

Based on the white-box crypto-key generation request submitted by the user, WhiteCrypto will generate different white-box crypto-keys for the same original crypto-key for two different requests. This increases the diversity of the white-box crypto-keys, and offers better protection for the original crypto-key.