

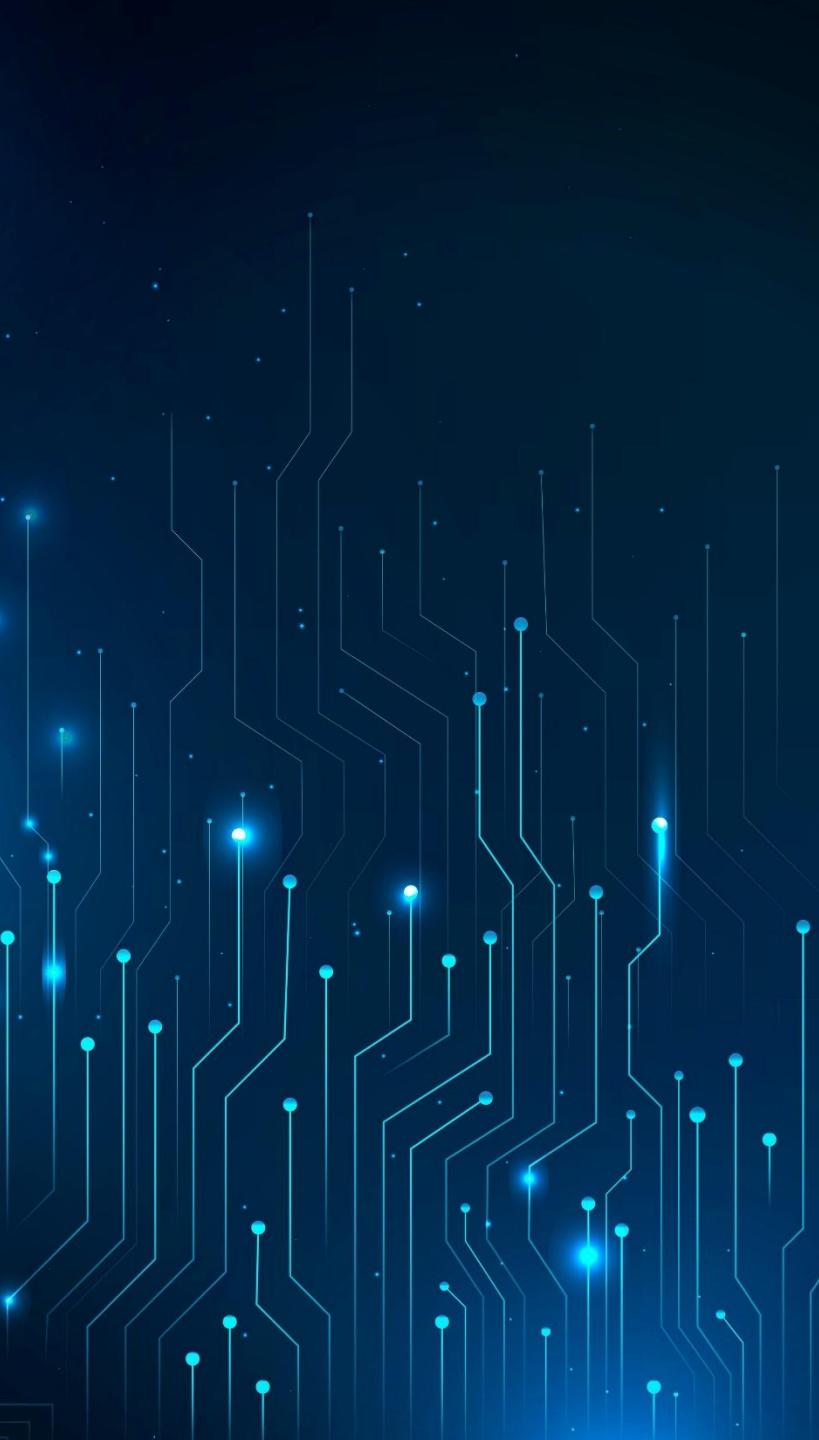
SECIRON

Mobile Application Protection Solution

Presented by:

Haekal Aufar Amriel

Technical Consultant Manager Indonesia Region



Overview

- Introduction
- Mobile App Protection Solution
- Benefit & Value Proposition



1.0 Introduction

Seclron Footprints

Seclron is a leading global provider in the mobile security industry. We are committed to bringing **mobile app security** to everyone and building security guarantees for the Smart World.



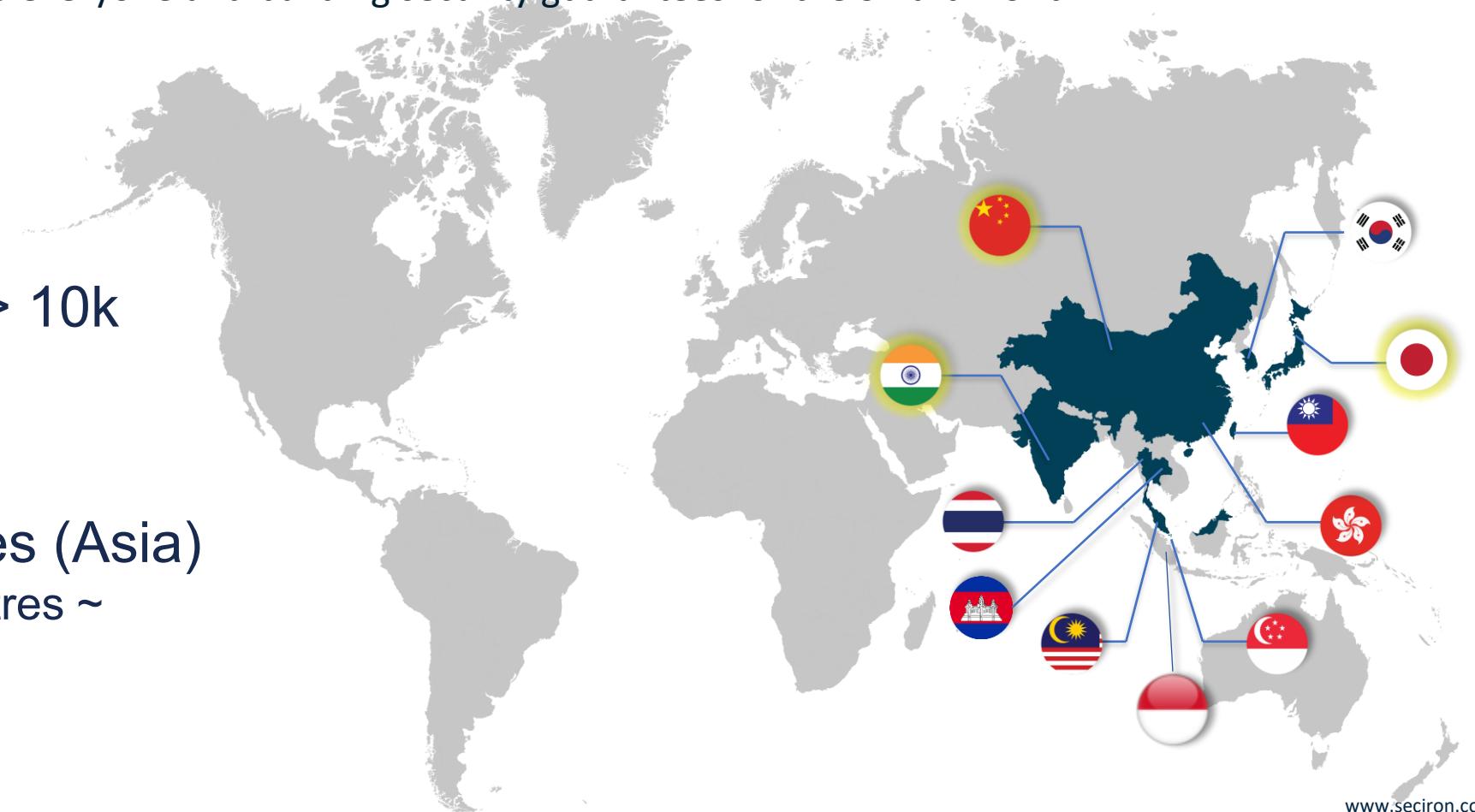
2010



Protected > 10k
apps



11 Countries (Asia)
~ 3 R&D Centres ~



Global References

SECIRON



> 60 Satisfied Clients
95% Customer Retention

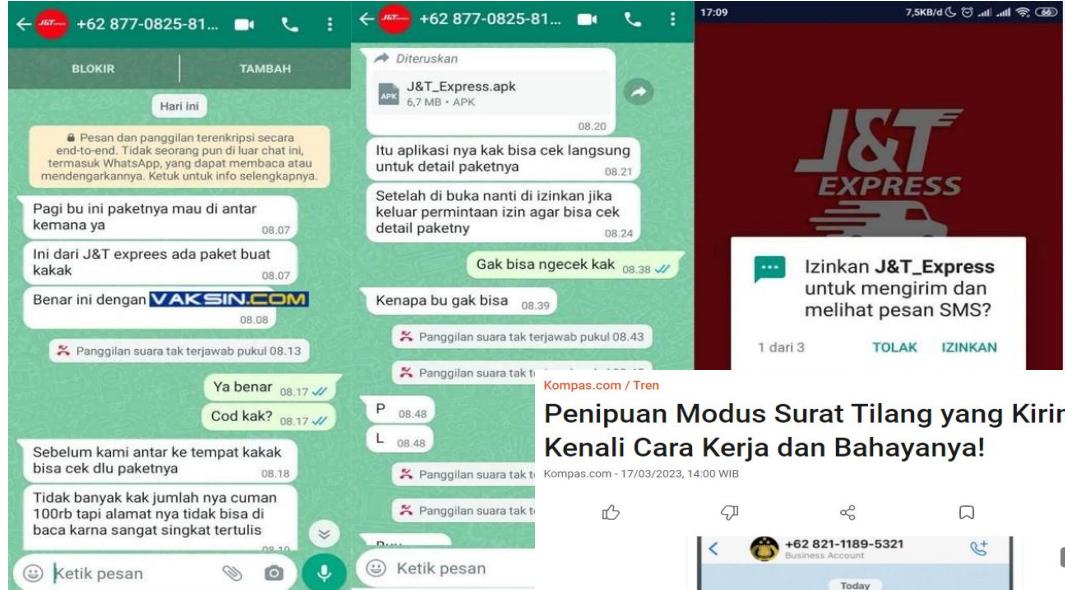
Gartner®
Peer Insights™



www.seciron.com

Fake Apps Incident

– How about if your app becomes a Victim?

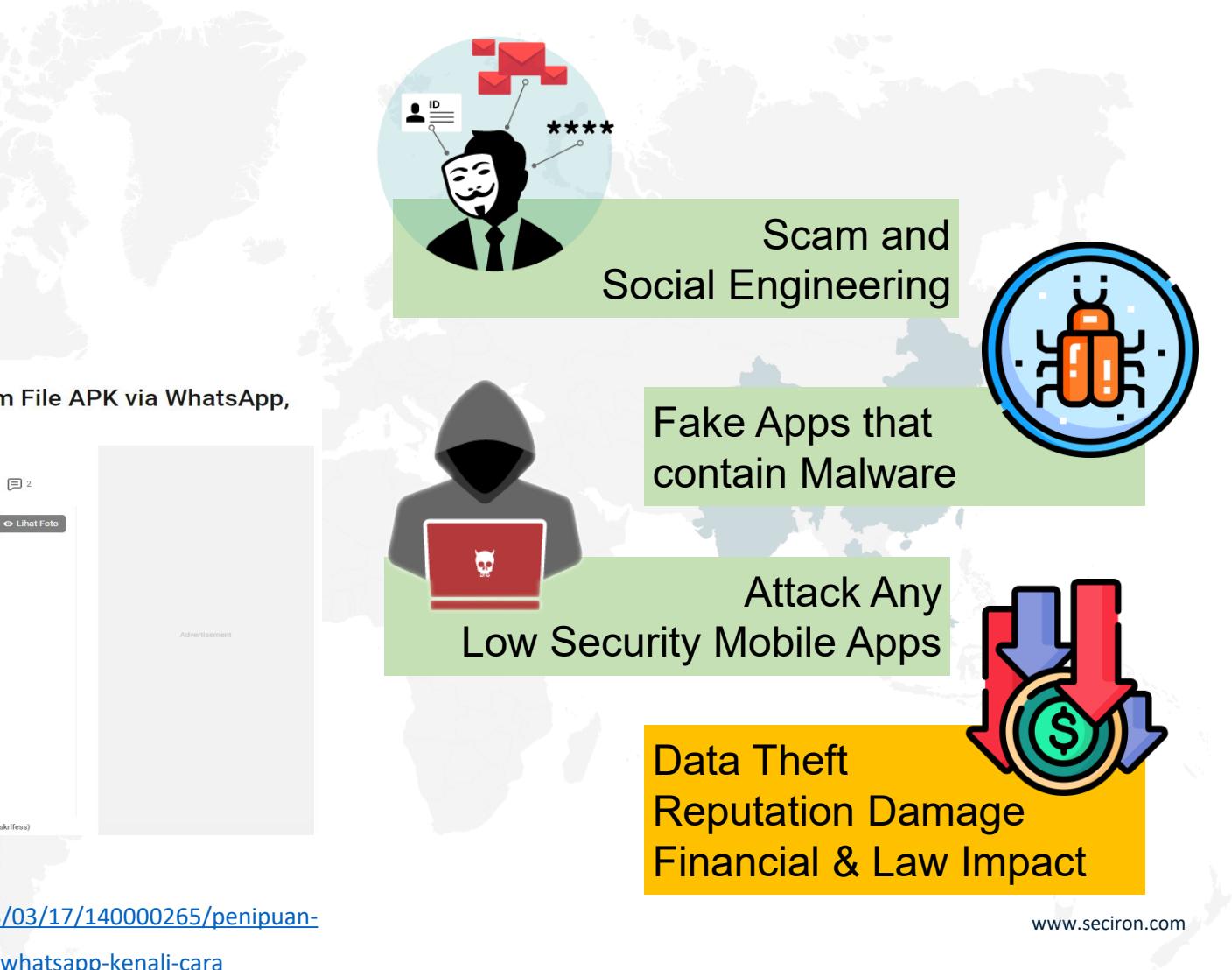


Source:

<https://kumparan.com/kumparantech/waspada-penipuan-kurir-palsu-minta-instal-aplikasi-curi-otp-bobol-m-banking-1zNzuyJbrgE/full/gallery/2>

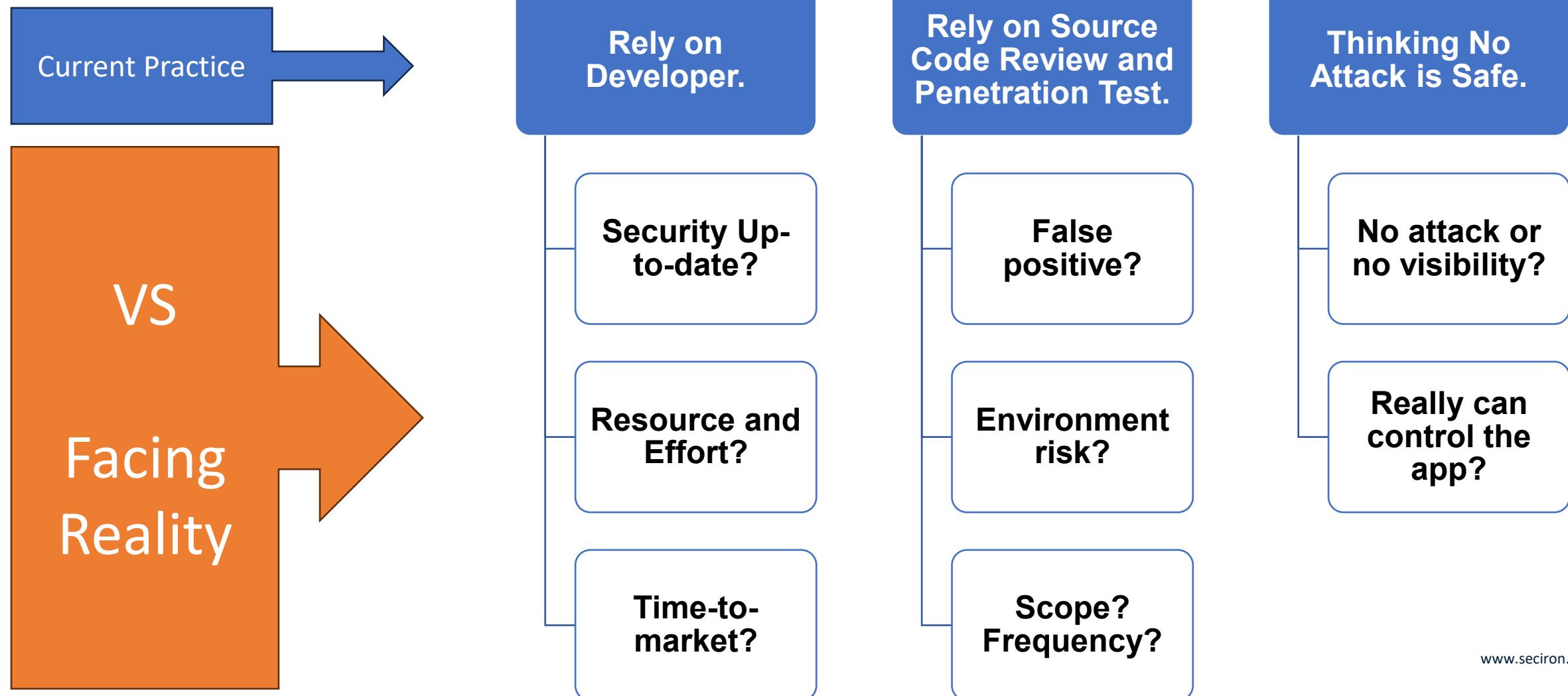
Source:

<https://www.kompas.com/tren/read/2023/03/17/140000265/penipuan-modus-surat-tilang-yang-kirim-file-apk-via-whatsapp-kenali-cara>



Current Practice for Mobile App Security

– Try to think.





2.0

Mobile App Protection Solution

SEC^SIRON

Seclron – Mobile Application Security Solution

SECIRON



IronSCAN – Mobile Application Security Testing

- Detect vulnerabilities with recommendations to rectify.



IronWALL – Application Hardening Platform

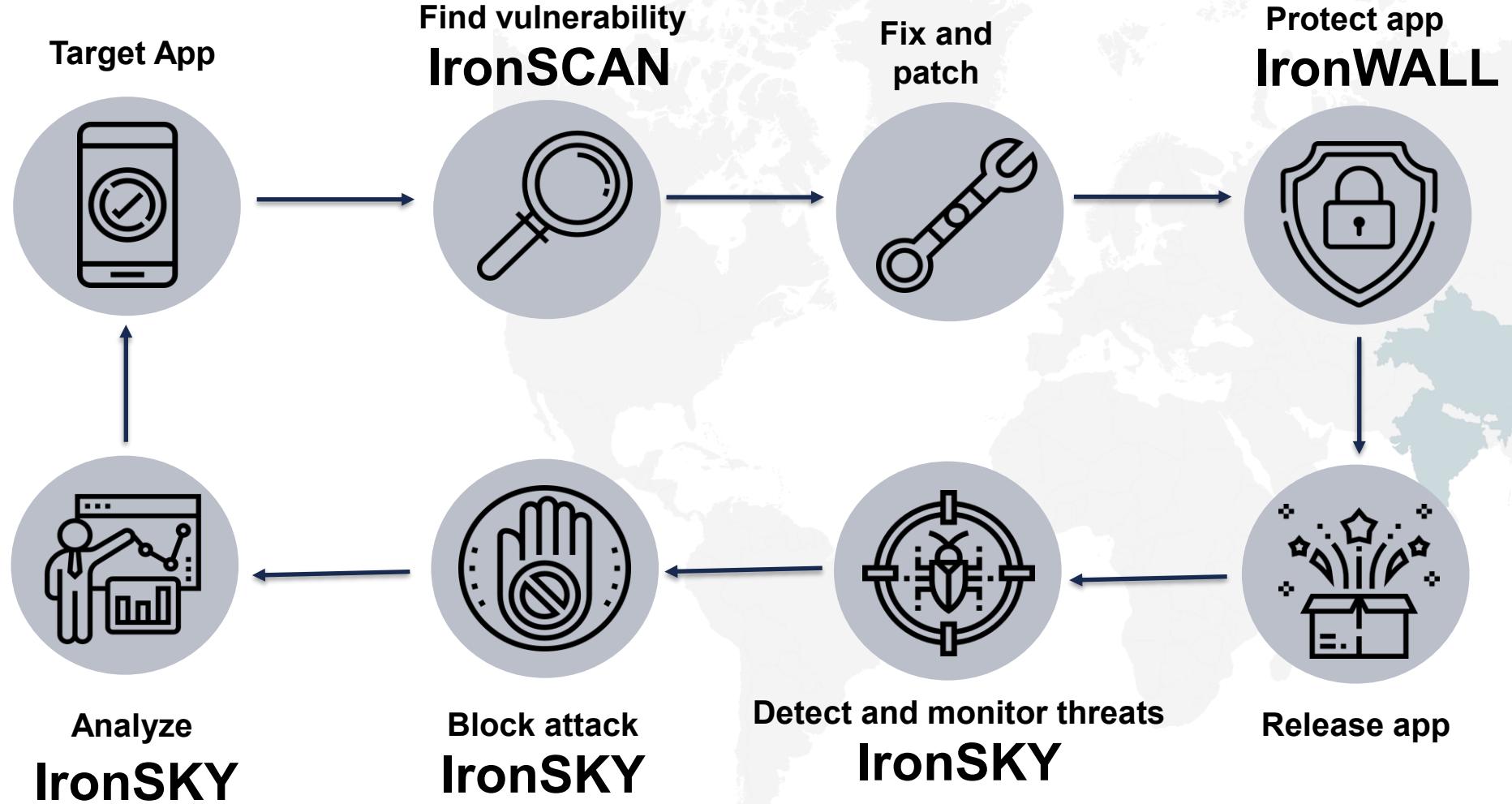
- Protect application from reverse-engineering and attacks



IronSKY – Real-time Threats Monitoring System

- Real-time monitoring, detection and response to threats.

Mobile App Security Lifecycle



Seciron Platform Preview

SECIRON

SaaS or On-Premises

SECIRON

Login to Seciron

Username
demo@seciron.com

Password
.....

Verification Code


Continue

SECIRON

Dashboard

SECURITY ASSESSMENT

- Android
- iOS

TASK TEMPLATE

- Android
- iOS

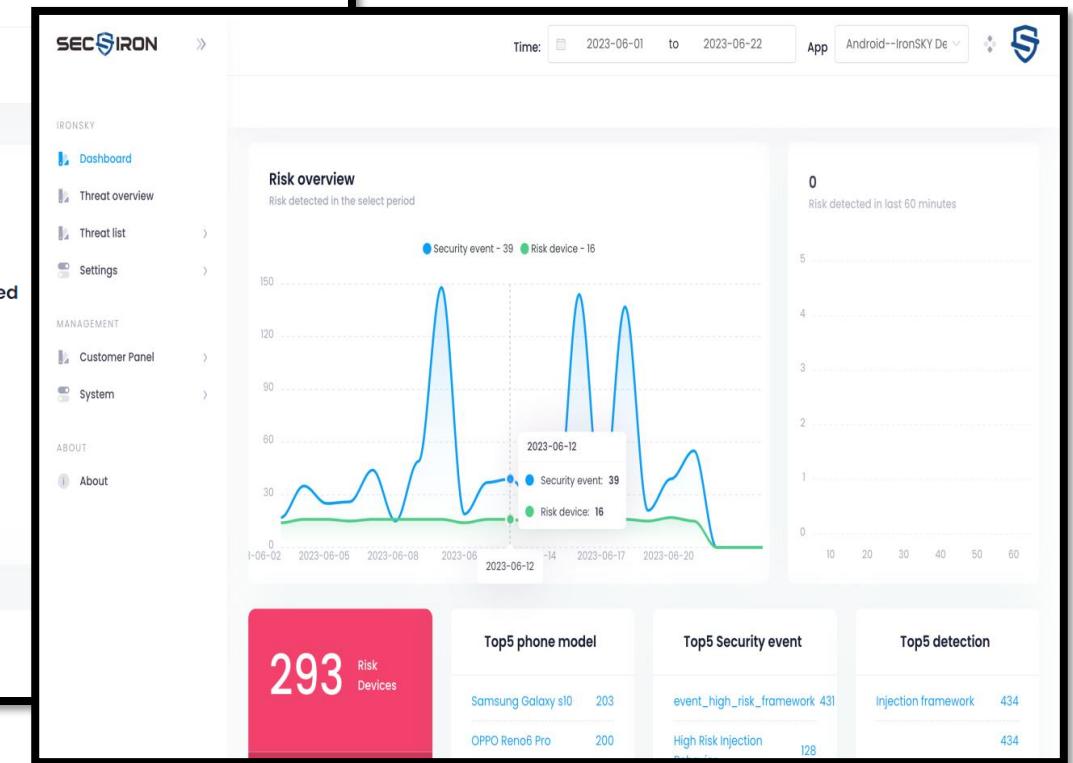
HARDENING

- Android
- Android SDK
- iOS

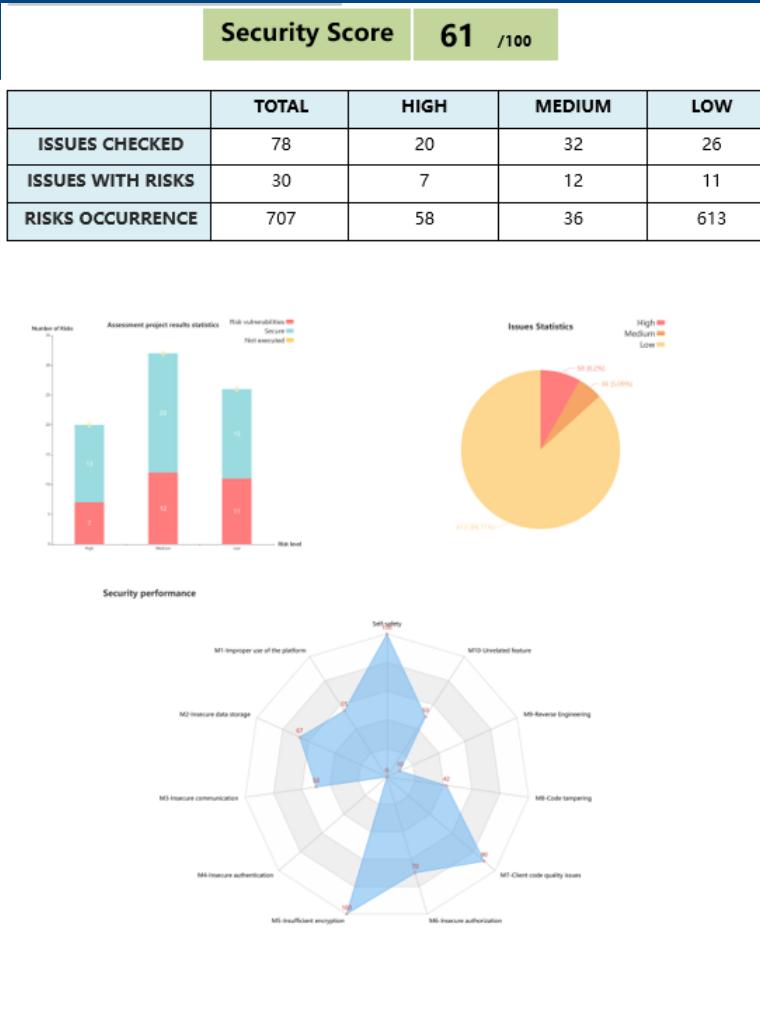
Activity Summary

APP/SDK name	Date created/modified
Sieve	2023-06-01 10:00:00
FileProviderModule	2023-06-01 10:00:00
FileProviderModule	2023-06-01 10:00:00

Dashboard



Sample of IronSCAN Reports



1.3.Issues Summary

	ISSUE	RISK LEVEL	RESULT
Self-safety(4 items)			
1	Permission information	Low	Pop-up message (27 found)
2	Behavior information	Low	Pop-up message (390 found)
3	Virus scanning	High	Secure
4	Apk file in resource file	Low	Secure
M1-Improper use of the platform(10 items)			
5	Janus signature mechanism vulnerability	High	Secure
6	Start hidden service risk	Low	Secure
7	Insecure application signature algorithm risk	Low	Secure
8	Dynamic registration of Receiver risks	Medium	Risk exist(found 5 risks)
9	PendingIntent misuse of Intent risk	Low	Risk exist(found 5 risks)
10	Intent component implicit call risk	Low	Risk exist(found 20 risks)
11	Denial of service attack vulnerability	Medium	Risk exist(found 1 risk)
12	Loading dex risk from sdcard	Medium	Secure
13	Loading so risk from sdcard	Medium	Secure
14	Insecure browser call vulnerability	Medium	Risk exist(found 5 risks)
M2-Insecure data storage(11 items)			
15	Webview plaintext password storage risk	High	Risk exist(found 6 risks)
16	Webview File same origin policy bypass vulnerability	High	Risk exist(found 5 risks)
17	Plaintext digital certificate risk	Medium	Secure
18	Database injection vulnerability	High	Secure

2.3.M2-Insecure data storage

2.3.1.Webview plaintext password storage risk

Purpose of assessment	Check whether the user name and password are saved in plaintext in WebView component of APP.
Risk Level	High
Description	The Android Webview component has the function of prompting the user to save the password by default. If the user chooses to save, the username and password will be stored in plaintext in the application directory databases/webview.db. The user name and password stored in local plaintext will not only be browsed by the application, other malicious programs may also access the webview database of the application through privilege escalation or root to steal user's login username and password that the user has logged in to.
Assessment result	Risk exist(found 2 risks)
Assessment result description	The WebView component of this app does not set to turn off the function of auto save password , user name and password may be stored in plaintext
Assessment details	<ol style="list-style-type: none"> [file]: com/ipay/IPayHActivity\$2 [method]: public onCreateWindow(Landroid/webkit/WebView;ZZLandroid/os/Message;)Z [file]: com/ipay/IPayHActivity\$2\$1 [method]: public
Solution	Developer self-check: Turn off the save password function of the webview component by setting WebView.getSettings().SetSavePassword (false).

Highlight IronWALL Features

- Protect mobile application running on an untrusted/compromised environment
- Addressing risk of:
 - Malware attack
 - Fake account opening
 - Data tampering for fraud activities
 - Security defense breached
 - Cloned app for phishing
- No source code changes required
- Quick implementation; available on SaaS and on-premise
- Suitable to use for Consumer apps related to financial or credential data; or enterprise app in BYOD environment



Anti-Reverse Engineering

3 layers (Hide, Encrypt, Transform) to protect source code from reverse engineer.



Anti-Dynamic Attack

Prevent debugging, tampering, injection attack, screen hijacking, rooted and emulators



Anti-Clone App

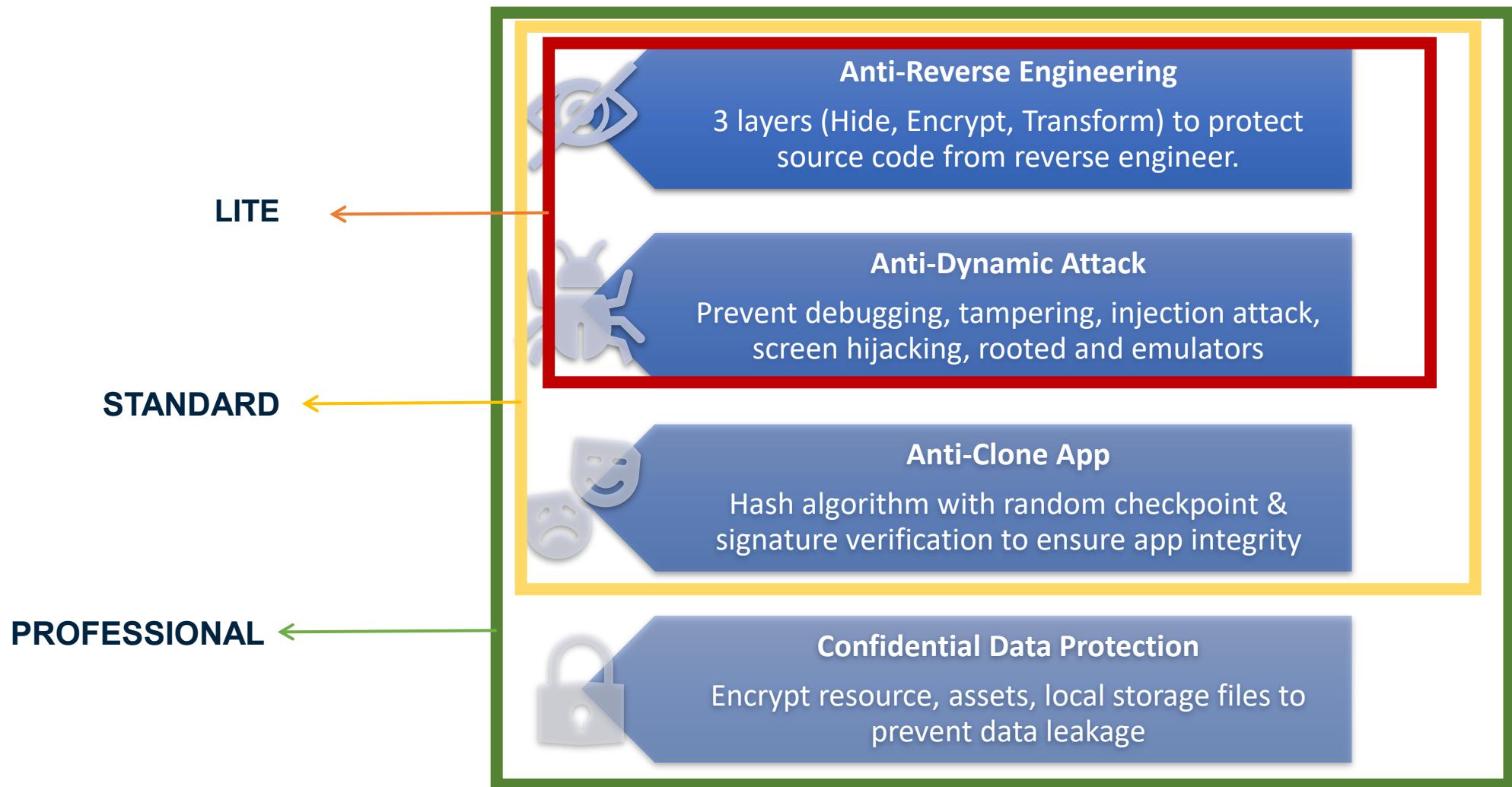
Hash algorithm with random checkpoint & signature verification to ensure app integrity



Confidential Data Protection

Encrypt resource, assets, local storage files to prevent data leakage

Highlight IronWALL Features



Example: Anti-reverse Protection on Android (IronWALL)

- Already available from IronWALL Lite Version

Original Code

This screenshot shows the JD-GUI interface with the file "no-proguard.apk" open. The left sidebar shows the project structure with "MainActivity" selected. The main pane displays the original Java code for MainActivity:package com.seciron.javasample;
import android.content.IntentFilter;
import android.os.Bundle;
import android.util.Log;
import androidx.appcompat.app.AppCompatActivity;
import androidx.navigation.NavController;
import androidx.navigation.Navigation;
import androidx.navigation.ui.AppBarConfiguration;
import androidx.navigation.ui.NavigationUI;
import com.seciron.javasample.databinding.Ac
import com.test.everisk.core.CallBack;
import com.test.everisk.core.RiskStubAPI;
import com.test.everisk.core.Type;
import org.json.JSONObject;

/* loaded from: classes.dex */
public class MainActivity extends AppCompatActivity {
 private ActivityMainBinding binding;

 @Override // androidx.fragment.app.FragmentActivity
 protected void onCreate(Bundle bundle) {
 super.onCreate(bundle);
 registerSecurityEventCallback();
 initializeRiskStubAPI();
 ActivityMainBinding inflate = Activi
 this.binding = inflate;
 setContentView(inflate.getRoot());
 AppBarConfiguration build = new AppB
NavController findNavController = Na
NavigationUI.setupActionBarWithNavController(NavigationUI.setupWithNavController(
 }
}

Standard Obfuscation

This screenshot shows the JD-GUI interface with the file "proguard-jadx-gui.apk" open. The left sidebar shows the project structure with "MainActivity" selected. The main pane displays the obfuscated Java code for MainActivity:import q1.c;
import q1.f;

/* loaded from: classes.dex */
public class MainActivity extends AppCompatActivity {

 /* renamed from: z */
 public i f4383z;

 @Override // androidx.fragment.app.FragmentActivity, androidx.activity.C
 public final void onCreate(Bundle bundle) {
 View findViewById;
 super.onCreate(bundle);
 TestAPI.initResponse(this);
 TestAPI.reg_api(new a(), Type.RISKEVENT, 0.5d);
 TestAPI.dummyInitKey(this, "a88XBOU/bvfk6aL7SliehQSw7/ILeqPG55Vur
 View inflate = getLayoutInflater().inflate(R.layout.activity_main, (ConstraintLayout) constraintLayout.inflate;
 BottomNavigationView bottomNavigationView = (BottomNavigationView) t
 if (bottomNavigationView == null) {
 throw new NullPointerException("Missing required view with ID: "
 }
 this.f4383z = new i(constraintLayout, bottomNavigationView);
 setContentView(constraintLayout);
 int[] iArr = {R.id.navigation_home, R.id.navigation_dashboard, R.id.
 HashSet hashSet = new HashSet();
 for (int i = 0; i < 3; i++) {
 hashSet.add(Integer.valueOf(iArr[i]));
 }
 mVar = new m(18, hashSet);
 if (Build.VERSION.SDK_INT >= 28) {
 findViewById = (View) b.(this, R.id.nav_host_fragment_activity_
}
 else {
 findViewById = findViewById(R.id.nav_host_fragment_activity_main);
 if (findViewById == null) {
 throw new IllegalArgumentException("ID does not reference a
}
 }
 g.d(findViewById, "requireViewById<View>(activity, viewId)");
 aVar = new c(new n1() {
 i1 r0; findViewById C0084h f1817k);
 aVar.setC0084h(f1817k);
 }
}

Next Gen Hardening

This screenshot shows the JD-GUI interface with the file "seciron-protection.jadx-gui.apk" open. The left sidebar shows the project structure with "MainActivity" selected. The main pane displays the hardened Java code for MainActivity:package com.seciron.javasample;

/* JAD INFO: This class is generated by JADX */
public final class R {

 public static final class anim {
 public static final int abc_fade_in = 0x7f010000;
 public static final int abc_fade_out = 0x7f010001;
 public static final int abc_grow_fade_in_from_bottom = 0x7f010002;
 public static final int abc_popup_enter = 0x7f010003;
 public static final int abc_popup_exit = 0x7f010004;
 public static final int abc_shrink_fade_out_from_bottom = 0x7f010005;
 public static final int abc_slide_in_bottom = 0x7f010006;
 public static final int abc_slide_out_top = 0x7f010007;
 public static final int abc_slide_out_bottom = 0x7f010008;
 public static final int abc_slide_out_top = 0x7f010009;
 public static final int abc_tooltip_enter = 0x7f01000a;
 public static final int abc_tooltip_exit = 0x7f01000b;
 public static final int btn_checkbox_to_checked_box_inner_merged_an
 public static final int btn_checkbox_to_checked_box_outer_merged_an
 public static final int btn_checkbox_to_checked_icon_inner_merged_an
 public static final int btn_checkbox_to_checked_icon_outer_merged_an
 public static final int btn_checkbox_to_unchecked_box_inner_merged_an
 public static final int btn_checkbox_to_unchecked_icon_inner_merged_an
 public static final int btn_radio_to_off_mtrl_dot_group_animation =
 public static final int btn_radio_to_off_mtrl_ring_outer_animation =
 public static final int btn_radio_to_on_mtrl_dot_group_animation =
 public static final int btn_radio_to_on_mtrl_ring_outer_animation =
 public static final int design_bottom_sheet_slide_in = 0x7f010018;
 public static final int design_bottom_sheet_slide_out = 0x7f010019;
 public static final int snackbar_in = 0x7f01001a;
 public static final int snackbar_out = 0x7f01001b;
 public static final int fragment_fast_out_extra_slow_in = 0x7f01001c;
 public static final int linear_indefinite_line1_head_interpolate
 public static final int linear_indefinite_line1_tail_interpolate
 public static final int linear_indefinite_line2_head_interpolate
 public static final int linear_indefinite_line2_tail_interpolate
 public static final int m3_bottom_sheet_slide_in = 0x7f010021;
 public static final int m3_bottom_sheet_slide_out = 0x7f010022;
 public static final int m3_motion_fade_enter = 0x7f010023;
 public static final int m3_motion_fade_exit = 0x7f010024;
 public static final int m3_side_sheet_enter_from_left = 0x7f010025;
 public static final int m3_side_sheet_enter_from_right = 0x7f010026;
 public static final int m3_side_sheet_exit_to_left = 0x7f010027;
 public static final int m3_side_sheet_exit_to_right = 0x7f010028;
 public static final int mtrl_bottom_sheet_slide_in = 0x7f010029;
 public static final int mtrl_bottom_sheet_slide_out = 0x7f01002a;
 public static final int mtrl_card_lowers_interpolator = 0x7f01002b;
 public static final int nav_default_enter_anim = 0x7f01002c;

After hardening by Seclron, the source code is not only obfuscated but also hidden and the classes are encrypted.

Support CI/CD Integration

Example:

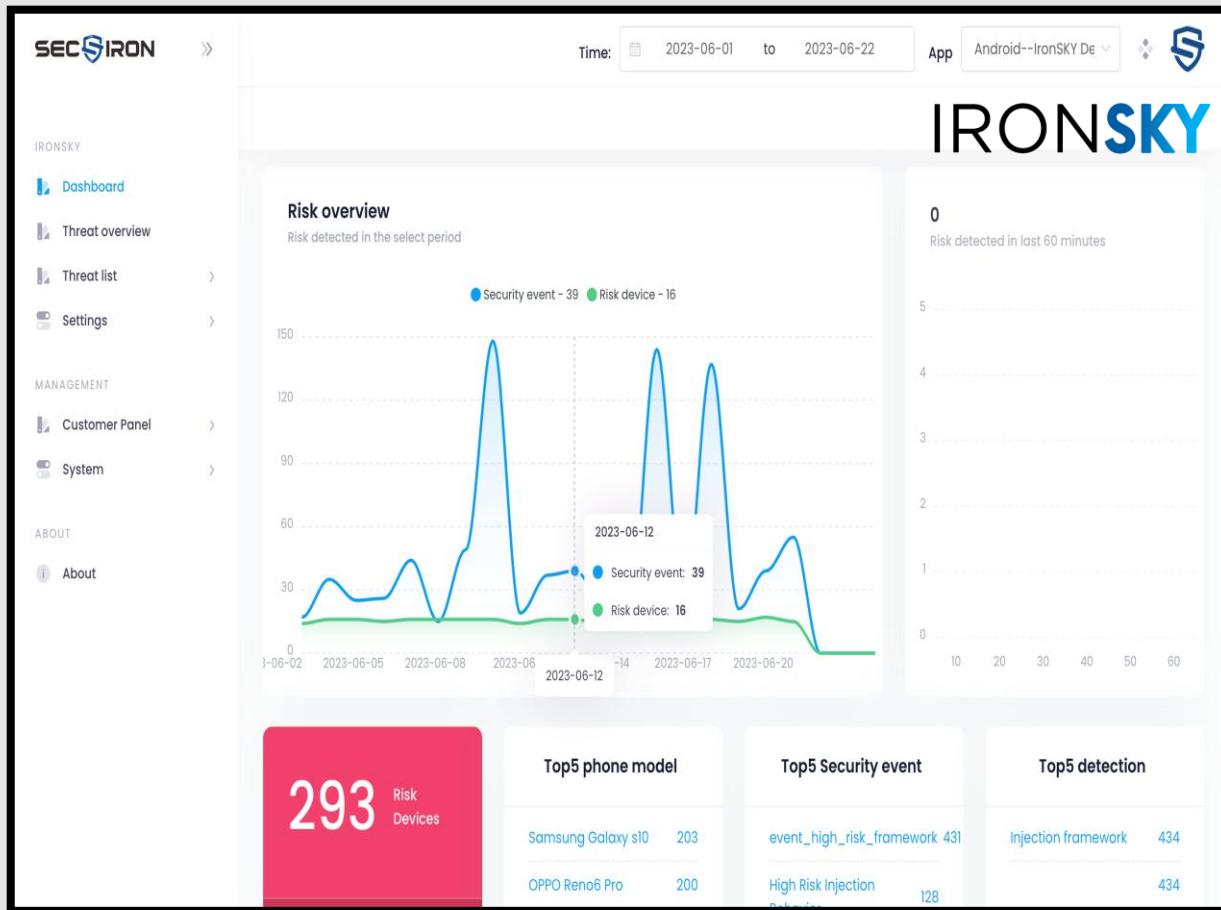
```
java -jar secapi-4.2.5.2.jar -i http://192.168.0.6:8090 -u lzz --password 1234566 -a 98918240-8a85-41a8-8002-c35457255094 -c 2ebe1900-ed3b-41d5-8183-5ecd3b1426ba -f 0 -t 458 -p D:\app-debug.apk -d G:\ --action ud --ks 1824004939987288065 --ms 1824004776086470657 --lp
```

Parameter	Description (MANDATORY)
-i	Service IP address which needs to include port number. If it is a domain name, configure it as given - https://portal.seciron.com
-u	Username: created user on the hardening service platform
--password	Password Username of Platform
-a	Api_key: obtained through the hardening service platform or from SecIron
-c	API secret: obtained through the hardening service platform or from SecIron
-f	Hardening product type: • 0: Android app hardening (APK or AAB) • 7: iOS app hardening (iPA or Xcarchieve)
-t	Hardening policy ID
-p	Upload file directory or OR file (required)
-d	Download file directory: Prior to hardening, a folder must exist
--action	• ud: upload and download at the same time • u: upload hardening package • d: download hardening package • l: view logs • s: view status

Parameter	Description (OPTIONAL)
--id	The hardening task ID to be downloaded
--dpre	The prefix of the hardened file name can be set
--dsuf	The suffix of the hardened file name can be set
-q	Timeout time: Exit automatically after timeout, in seconds Optional: The default of downloading process is 7200 seconds by default. When uploading and downloading at the same time, please excluding uploading process.
-D	Specify the information file and download all tasks contained within it, regardless of whether they have previously been downloaded.
--tmp	Script temporary file storage directory
--ks	Keystore id
--ms	SDK id
--lp	Add the queries segment, but do not add it if it's not included. If QUERY_ALL_PACKAGES has already been requested, please ignore it.

IronSKY – Real-time Threats Monitoring System

SECIRON



Environment Risk

- Injection framework
- Emulator
- VPN Proxy
- Root certificate error
- Root/Jailbreak
- Sensitive settings
- Accessibility permission
- Blacklist/whitelist
- Side-load detection
- Unofficial Source
- Risk/Malware App

Behaviors

- HTTPS hijacking
- HTTP Proxy
- Injection attack
- Debugging behavior
- Domain fraud
- Memory tampering
- Multi-boxing
- Location fraud
- Screen mirroring
- Remote Access Detection
- Screen Hijacking Detection
- Auto Clicker Detection
- Keylogger Detection

1. False positives and false negatives may occur in single-point detection
2. In order to trace the source, provide a 24-hour query of the number of startups

HYBRID SDK INTEGRATION



- Static Protection
- Integrity Protection
- Dynamic Protection
- Confidential Data Encryption



1. IronSKY SDK Integration
(Manual/Auto - Optional)

2. Harden App with
IronWALL & Config
Configuration

3. Go Live with End-
to-End Hardened
Mobile App

A large, dark blue-toned photograph of a city skyline at night, showing numerous lit-up skyscrapers and buildings. The image is partially obscured by a large teal diagonal shape on the right side.

Benefit & Value Proposition

Prevent Before It's Late.





Benefit Implementing SecIron Solution



Flexibility of Deployment



Easy Integration



Wide Range of Supported Platforms



3 Layer of Code Protection



Have Complete Features for Static & Dynamic Protection



Simplicity on License Scheme



Less Impact to User Performance



Comprehensive Support in Indonesia



Thank You!

Office: Shibuya1-11-1, COI Nishi-Aoyama Bldg 3F
Shibuya-ku, Tokyo 150-0002