

Mobile App Security Best Security Practices

Arnold Kibuuka, Project Officer, ITU
dfssecuritylab@itu.int

March 2024



Overview

1. Mobile application security framework
2. DFS app security tests
 - Android App and
 - iOS Mobile Payment

Objectives of the Recommendations

- **Enhance Security:** Implement robust security measures to protect against SS7 vulnerabilities, SIM risks, and SIM swap fraud.
- **Promote Best Practices:** Encourage the adoption of best practices in mobile financial services application security.
- **Foster Collaboration:** Strengthen collaboration between Telecommunications Regulators and Central Banks through a model MOU.
- **Improve Consumer Competency:** Enhance DFS consumer competency through a structured framework.
- **Advance Financial Inclusion:** Use these recommendations as tools to advance financial inclusion by making DFS safer and more secure for all users.

Recap: DFS Security Recommendations



[Security recommendations to protect against DFS SIM related risks like SIM swap fraud and SIM recycling](#)



[Recommendations to mitigate SS7 vulnerabilities](#)



[Template for a Model MOU between a Telecommunications Regulator and Central Bank related to DFS Security](#)



[Mobile Application Security Best practices](#)



ITU DFS Consumer Competence Framework

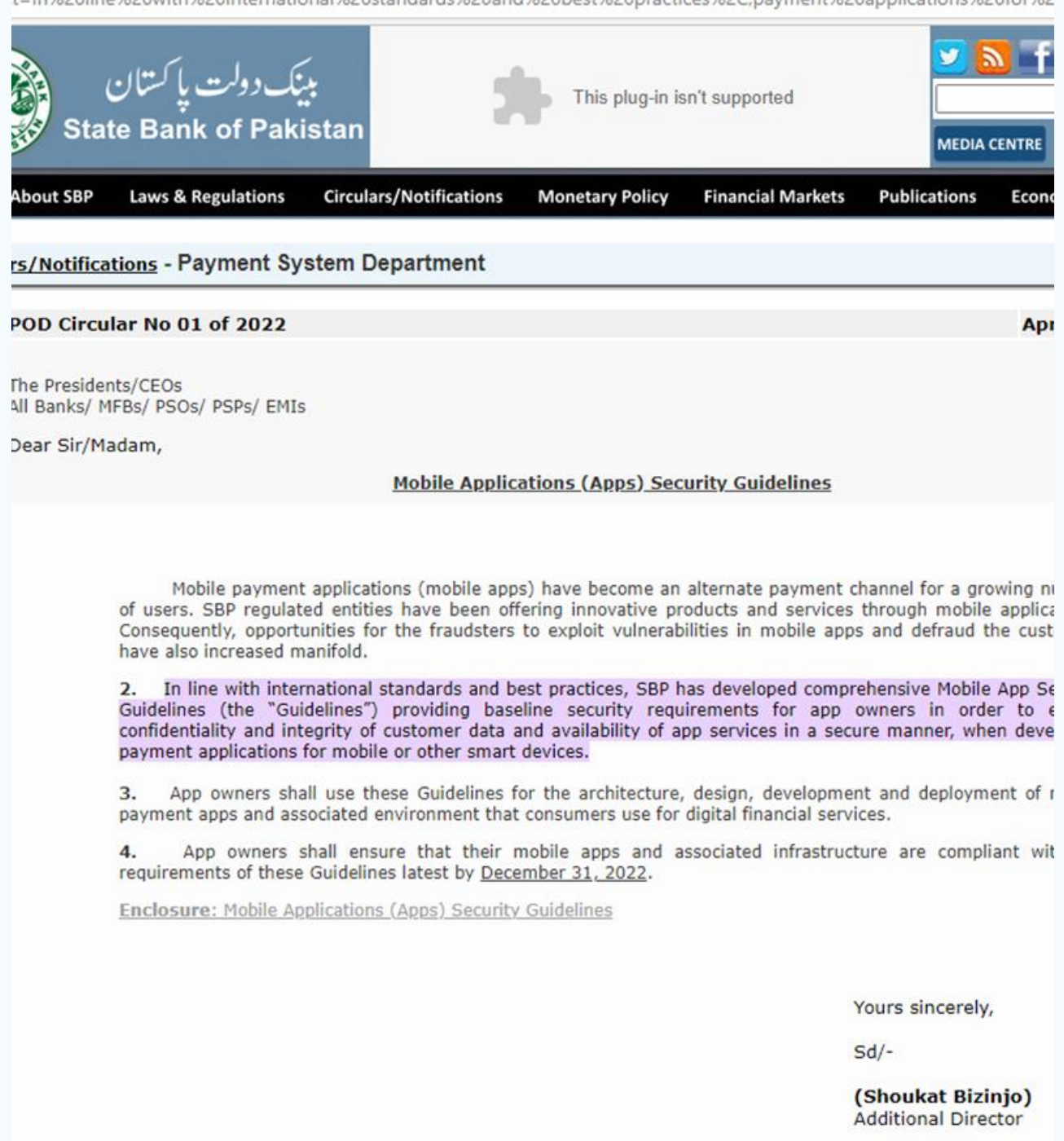
Mobile Payment App Security framework

Mobile Payment App Security Best Practices (Section 9)

- Draws upon:
 - GSMA study on mobile money best practices,
 - ENISA smartphone security development guidelines,
- Template can be used as input to an app security policy by DFS providers to provide minimum security baselines for app developers and DFS providers as well as setting criteria for verifying compliance of apps
- Template considerations:
 - i. device and application integrity.
 - ii. communication security and certificate handling.
 - iii. user authentication.
 - iv. secure data handling.
 - v. secure application development.

Application security best considerations:

- device and application integrity.
- communication security and certificate handling.
- user authentication.
- secure data handling.
- secure application development.



The screenshot shows the State Bank of Pakistan (SBP) website. The header includes the SBP logo and name in Urdu and English. A navigation menu lists various sections: About SBP, Laws & Regulations, Circulars/Notifications, Monetary Policy, Financial Markets, Publications, and Economic. The main content area is titled "Circulars/Notifications - Payment System Department" and features a circular titled "POD Circular No 01 of 2022". The circular is addressed to "The Presidents/CEOs" and "All Banks/ MFBs/ PSOs/ PSPs/ EMIs". It is dated "Apr" and is titled "Mobile Applications (Apps) Security Guidelines". The text of the circular discusses the growth of mobile payment applications and the need for security. It states that SBP has developed comprehensive Mobile App Security Guidelines (the "Guidelines") providing baseline security requirements for app owners in order to ensure confidentiality and integrity of customer data and availability of app services in a secure manner, when developing payment applications for mobile or other smart devices. The guidelines are numbered 1 through 4. The circular concludes with "Enclosure: Mobile Applications (Apps) Security Guidelines" and is signed by "Yours sincerely, Sd/- (Shoukat Bizinjo) Additional Director".

State Bank of Pakistan

This plug-in isn't supported

MEDIA CENTRE

About SBP Laws & Regulations Circulars/Notifications Monetary Policy Financial Markets Publications Economic

Circulars/Notifications - Payment System Department

POD Circular No 01 of 2022 Apr

The Presidents/CEOs
All Banks/ MFBs/ PSOs/ PSPs/ EMIs

Dear Sir/Madam,

Mobile Applications (Apps) Security Guidelines

Mobile payment applications (mobile apps) have become an alternate payment channel for a growing number of users. SBP regulated entities have been offering innovative products and services through mobile applications. Consequently, opportunities for the fraudsters to exploit vulnerabilities in mobile apps and defraud the customers have also increased manifold.

2. In line with international standards and best practices, SBP has developed comprehensive Mobile App Security Guidelines (the "Guidelines") providing baseline security requirements for app owners in order to ensure confidentiality and integrity of customer data and availability of app services in a secure manner, when developing payment applications for mobile or other smart devices.

3. App owners shall use these Guidelines for the architecture, design, development and deployment of mobile payment apps and associated environment that consumers use for digital financial services.

4. App owners shall ensure that their mobile apps and associated infrastructure are compliant with requirements of these Guidelines latest by December 31, 2022.

Enclosure: Mobile Applications (Apps) Security Guidelines

Yours sincerely,
Sd/-
(Shoukat Bizinjo)
Additional Director

Mobile Application Security best practices



Device and Application Integrity

Use platform services for integrity checks;
remove extraneous code
maintain high-integrity state server-side.



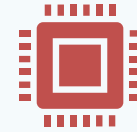
Communication Security and Certificate Handling

Standardized cryptographic libraries; strong, up-to-date TLS certificates; limit certificate lifetimes (825 days); contingency for untrusted CA; secure TLS configuration; certificate pinning; correct server certificate validation.



User Authentication

Disallow easily guessable credentials;
encourage multi-factor authentication;
prefer authenticator apps over SMS for OTPs;
secure storage of biometric information.



Secure Data Handling

Secure storage of confidential info;
trusted hardware for sensitive data;
avoid external storage;
clean caches/memory;
fine-grained permissions for data sharing;
avoid hard-coding sensitive info;
validate client input for database storage.



Secure Application Development

Adhere to secure coding practices and standards;
provide secure application updates;
regular internal or external code reviews.

Mobile Application Security best practices



Device and Application Integrity

- T1.2 Android:debuggable
- T1.4 Dangerous permissions
- T8.1 The application should refuse to run on a rooted device



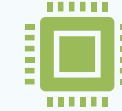
Communication Security and Certificate Handling

- T3.1 Application should only use HTTPS connections
- T3.2 Application should detect Machine-in-the-Middle attacks with untrusted certificates
- T3.3 Application should detect Machine-in-the-Middle attacks with trusted certificates
- T3.4 App manifest should not allow clear text traffic
- T5.1 The app should not use unsafe crypto primitives
- T5.2 The HTTPS connections should be configured according to best practices
- T5.3 The app should encrypt sensitive data that is sent over HTTPS



User Authentication

- T4.1 Authentication required before accessing sensitive information
- T4.2 The application should have an inactivity timeout
- T4.3 If a fingerprint is added, authentication with fingerprints should be disabled
- T4.4 It should not be possible to replay intercepted requests



Secure Data Handling

- T1.1 Android:allowBackup
- T1.3 Android:installLocation
- T2.1 Android.permission.WRITE_EXTERNAL_STORAGE
- T2.2 Disabling screenshots



Secure Application Development

- T9.1 The code of the app should be obfuscated

Android & iOS App DFS Security Tests

Introduction

The Open Web Application Security Project

A collaborative, non-for-profit foundation that works to improve the security of web applications

Also works on security of mobile applications.

OWASP Mobile Top Ten

OWASP project that aims to identify and document the top ten vulnerabilities of mobile applications

Lab methodology

18 tests on Android organized according to OWASP mobile top 10.

21 tests on iOS DFS applications

iOS and tests

- Our tests are organized according to the subjects of the OWASP Mobile Top Ten:
 - M1 Improper Platform Usage
 - M2 Insecure Data Storage
 - M3 Insecure Communication
 - M4 Insecure Authentication
 - M5 Insufficient Cryptography
 - M6 *Insecure Authorization*
 - M7 *Client Code Quality*
 - M8 Code Tampering
 - M9 Reverse Engineering
 - M10 *Extraneous Functionality*
- M6, M7, M10 out of scope because they would need access to the source code or require collaboration with the editor

DFS lab hardware and software

EQUIPMENT	QTY	COMMENT	TESTS
Desktop/laptop	2	32 GB RAM, 1TB, 4+ core 64-bit processor	All
Mobile smartphone (Android OS).	2	Google Pixel	Android
Wi-Fi router	1		All
iOS Device	2	(One of the iPhones MUST run iOS 14)	iOS
Kali Linux		Opensource	All
Wireshark		Opensource	All
Magisk		Opensource	All
Frida		Opensource	All
MobSF		Opensource	All
Androguard		Opensource	All
Burp proxy		Opensource/licenced	All
Objection			iOS
Checkra1n			iOS
Bettercap		Opensource	All
apk-mitm		Opensource	All
Personnel		Security professionals with at least 3 years technical security experience	All
DFS Apps, SIM cards to be tested.			All

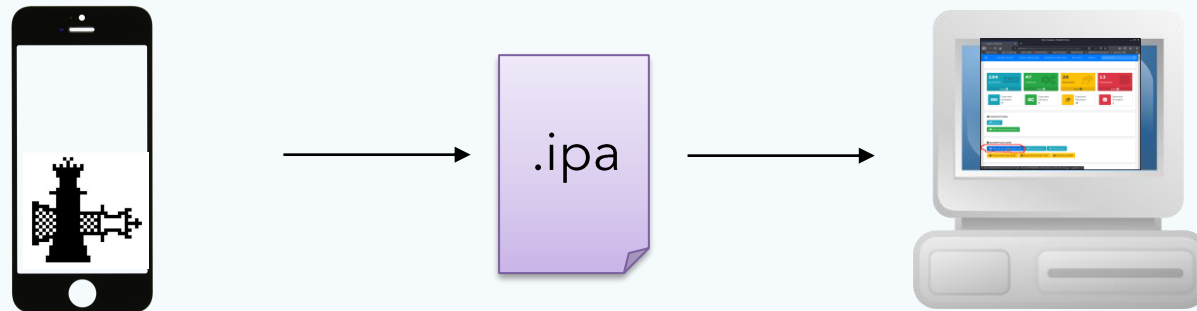


Summary of the tests

- 22 iOS and 18 Android tests organized according to OWASP mobile top ten
- Tests with jailbroken/rooted and non jailbroken/non rooted phones
- Static analysis of apps on a workstation
- Dynamic analysis with a man-in-the-middle proxy

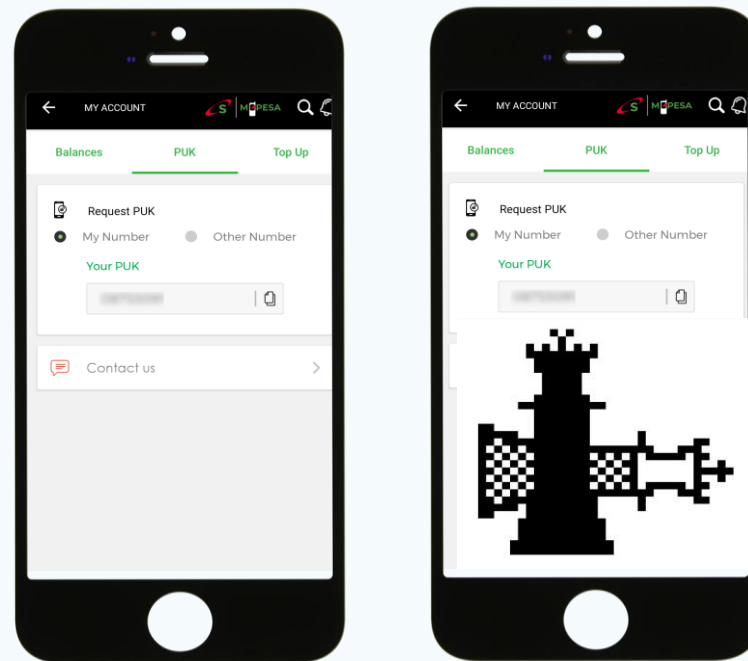
Static tests

- Extract application package (apk/ipa) from a rooted/jailbroken phone
- Analyse the package on workstation with different tools (Mobsf, Jadx)



Functionality tests

Test security features on standard phone and on rooted phone



Interception tests

- Use the workstation as man-in-the-middle between phone and server contacted by the application
 - Use Bettercap to force traffic through the workstation
 - Replace certificate on phone
 - Disable certificate pinning (on jailbroken phone only)
 - Use Burp proxy to analyse, modify, replay traffic



Test details

M1 Improper Platform Usage

The application should make correct use of the features of the platform (phone's operating system)

T1.1 Android:allowBackup

- Backup of the application and its data into the cloud should be disabled

T1.2 Android:debuggable

- Debugging features of the application should be disabled

T1.3 Android:installLocation

- The application should be installed in the internal, more secure, memory

T1.4 Dangerous permissions

- The application should not require dangerous permissions, as defined by Android.

PERMISSION	STATUS	INFO	DESCRIPTION
android.permission.ACCESS_COARSE_LOCATION	dangerous	coarse (network-based) location	Access coarse location sources, such as the mobile network database, to determine an approximate phone location, where available. Malicious applications can use this to determine approximately where you are.
android.permission.ACCESS_FINE_LOCATION	dangerous	fine (GPS) location	Access fine location sources, such as the Global Positioning System on the phone, where available. Malicious applications can use this to determine where you are and may consume additional battery power.

M2 Insecure Data Storage

```
<uses-sdk android:minSdkVersion="16" android:targetSdkVersion="28" />
<uses-feature android:name="android.hardware.telephony" android:required="false" />
<uses-feature android:name="android.hardware.telephony.cdma" android:required="false" />
<uses-feature android:name="android.hardware.telephony.gsm" android:required="false" />
<uses-feature android:name="android.hardware.camera" android:required="false" />
<uses-feature android:name="android.hardware.camera.autofocus" android:required="false" />
<uses-feature android:name="android.hardware.camera.flash" android:required="false" />
<uses-feature android:name="android.hardware.camera.front" android:required="false" />
<uses-feature android:name="android.hardware.camera.any" android:required="false" />
<uses-feature android:name="android.hardware.bluetooth" android:required="false" />
<uses-feature android:name="android.hardware.location" android:required="false" />
<uses-feature android:name="android.hardware.location.network" android:required="false" />
<uses-feature android:name="android.hardware.location.gps" android:required="false" />
<uses-feature android:name="android.hardware.microphone" android:required="false" />
<uses-feature android:name="android.hardware.wifi" android:required="false" />
<uses-feature android:name="android.hardware.wifi.direct" android:required="false" />
<uses-feature android:name="android.hardware.screen.landscape" android:required="false" />
<uses-feature android:name="android.hardware.screen.portrait" android:required="false" />
<uses-feature android:glEsVersion="0x00020000" android:required="true" />
<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
<uses-permission android:name="android.permission.ACCESS_WIFI_STATE" />
<uses-permission android:name="android.permission.VIBRATE" />
<uses-permission android:name="android.permission.WAKE_LOCK" />
<uses-permission android:name="android.permission.USE_FINGERPRINT" />
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.READ_PHONE_STATE" />
<uses-permission android:name="android.permission.READ_CONTACTS" />
<uses-permission android:name="android.permission.WRITE_CALENDAR" />
<uses-permission android:name="android.permission.CAMERA" />
<uses-permission android:name="android.permission.FLASHLIGHT" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />
<supports-screens android:largeScreens="true" android:xlargeScreens="true" />
<uses-permission android:name="com.google.android.c2dm.permission.RECEIVE" />
```

Data should be stored in a way that limits the risks in case of loss or compromise of the phone

T2.1 Android.permission.WRITE_EXTERNAL_STORAGE

- No permission to write to a removable memory card

T2.2 Disabling screenshots

- If not disabled, screen shots are done automatically to generate thumbnails for task switching

M3 Insecure Communication

Protect against eavesdropping and manipulation of traffic

T3.1 Application should only use HTTPS connections

- Test by sniffing traffic

T3.2 Application should detect Machine-in-the-Middle attacks with untrusted Certificates

- Would allow anybody to intercept traffic
- Test by intercepting traffic with proxy

T3.3 Application should detect Machine-in-the-Middle attacks with trusted certificate

- Would allow authorities to intercept traffic
- Test by installing root certificate on phone, intercept with proxy

T3.4 App manifest should not allow clear text traffic

The screenshot displays the Burp Suite interface. The top menu includes 'Burp Project', 'Intruder', 'Repeater', 'Window', 'Help', 'Logger++', and 'Backslash'. Below the menu is a toolbar with various tools like 'Errors', 'EsPReSSO', 'ExFTool', 'JSON Beautifier', 'Deserialization Scanner', 'Logger++', 'Paramalyzer', 'Versions', 'Software Vulnerability Scanner', and 'Additional Scanner Checks'. The main window shows a list of intercepted items with columns for '#', 'Host', 'Method', 'URL', 'Params', 'Edited', 'Status', 'Length', 'MIME type', 'Extension', 'Title', 'Comment', 'TLS', 'IP', 'Cookies', and 'Time'. A row is highlighted in orange, showing a POST request to '/smartphone/service/v11/orders/p2p/send'. Below this, the 'Request' tab is active, showing the raw request details: 'POST /smartphone/service/v11/orders/p2p/send HTTP/1.1', 'Accept-Encoding: gzip, deflate', 'Accept: application/json', 'Accept-Language: fr_CH', 'X-TWINT-WALLETAPP-LIB-VERSION: 15.3.0.18', 'Cookie: NavajonUNBjXYu02vvy2A1NYol+qgo/M3ThiBT8PhA94426Do/24f5NEDKkahF2VohHy0z2Kx2Uz1v0g-', 'Content-Type: application/json; charset=UTF-8', 'Content-Length: 764', 'Host: [redacted]', 'Connection: close', 'User-Agent: okhttp/3.12.0', 'ADDRESS: isMobile:true', 'ADDRESS: isAjax:true'. The body of the request is a JSON object: { "amount": { "amount": 20, "currency": "CHF" }, "certificateFingerprint": "ef[redacted]417b", "moneyReceiver": { "firstName": [redacted], "lastName": [redacted] }, "moneyReceiverMobileNumber": "+4179 [redacted]", "moneySender": { "firstName": [redacted], "lastName": [redacted] }, "orderId": "13976b6e-a57c-448a-8535-51d97f01928d", "reservationDate": "2020-07-10T08:55:12", "sendMoneyEvenIfCustomerUnknown": true, "signature": "gu1DEX3p0x+0c6vQm0cU94MnYqyb+R1HT8i24jHGou1/Jx81WV1m6W64058oJnnD8H9A:1d0mmc61/bZEJ0EPJERX/2kfrArcQNhEDU10c18wJFxx91At3HEj36yHehB0q29zTgTmZw0u8a3tzJNRpvRzic0QCx5X78Ih26A104KD047uFmKFPtHc" }

Burp Project Intruder Repeater Window Help Logger++ Backslas...

Errors EsPReSSO ExifTool JSON Beautifier Deserialization Scanner Logger++ Paramalyzer Versions Software Vulnerability Scanner Additional Scanner Checks
 Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options AuthMatrix Bypass WAF CO2

Intercept HTTP history WebSockets history Options

Filter: Hiding out of scope items

#	Host	Method	URL	Params	Edited	Status	Length	MIME type	Extension	Title	Comment	TLS	IP	Cookies	Time
148	https://[REDACTED]	GET	/iizwlm?_=1594371899392		✓	200	491	JSON				✓	[REDACTED]		11:04:54
145	https://[REDACTED]	GET	/iizwlm?_=1594371717242		✓	200	491	JSON				✓	[REDACTED]		11:01:51
144	https://[REDACTED]	GET	/iizwlm?_=1594371530169		✓	200	491	JSON				✓	[REDACTED]		10:58:40
141	https://[REDACTED]	GET	/P2PPaymentSystem/P2PInterfaceP2PLogin/V4_...		✓	200	576	JSON				✓	[REDACTED]		10:55:41
139	https://[REDACTED]	POST	/smartphone/service/v11/privateCustomers/me...		✓	200	1480	JSON				✓	[REDACTED]		10:55:21
138	https://[REDACTED]	GET	/smartphone/service/v11/privateCustomers/me...		✓	200	870	JSON				✓	[REDACTED]		10:55:20
137	https://[REDACTED]	POST	/P2PPaymentSystem/P2PInterfaceP2PLogin/V4_...		✓	200	805	JSON				✓	[REDACTED]		10:55:12
136	https://[REDACTED]	POST	/smartphone/service/v11/orders/p2p/send		✓	200	777	JSON				✓	[REDACTED]		10:55:09
135	https://[REDACTED]	GET	/P2PPaymentSystem/P2PInterfaceP2PLogin/V4_...		✓	200	576	JSON				✓	[REDACTED]		10:55:00
134	https://[REDACTED]	GET	/P2PPaymentSystem/P2PInterfaceP2PLogin/V4_...		✓	200	576	JSON				✓	[REDACTED]		10:54:47
133	https://[REDACTED]	GET	/P2PPaymentSystem/P2PInterfaceP2PLogin/V4_...		✓	200	576	JSON				✓	[REDACTED]		10:54:17
132	https://[REDACTED]	GET	/smartphone/service/v11/orders?limit=100&pa...		✓	200	18539	JSON				✓	[REDACTED]		10:53:47
131	https://[REDACTED]	POST	/smartphone/service/v11/privateCustomers/me...		✓	200	1480	JSON				✓	[REDACTED]		10:53:46
130	https://[REDACTED]	GET	/smartphone/service/v11/privateCustomers/me...		✓	200	870	JSON				✓	[REDACTED]		10:53:45
129	https://[REDACTED]	GET	/smartphone/service/v11/orders?since=1970-0...		✓	200	50014	JSON				✓	[REDACTED]		10:53:45
128	https://[REDACTED]	POST	/P2PPaymentSystem/P2PInterfaceP2PLogin/V4_...		✓	200	1340	JSON				✓	[REDACTED]		10:53:44

Request Response

Raw Params Headers Hex JSON JSON Beautifier

```

1 POST /smartphone/service/v11/orders/p2p/send HTTP/1.1
2 Accept-Encoding: gzip, deflate
3 Accept: application/json
4 Accept-Language: fr_CH
5 X-TWINT-WALLETAPP-LIB-VERSION: 15.3.0.18
6 Cookie: Navajo=UNBjXyUG2vyu2A3NYol+qgo/M3ThiBT8PhA944Z6Do/24f5NEDkkahF2VEohHy0zNKx2UuZivUg-
7 Content-Type: application/json; charset=UTF-8
8 Content-Length: 764
9 Host: [REDACTED]
10 Connection: close
11 User-Agent: okhttp/3.12.0
12 ADRUM_1: isMobile:true
13 ADRUM: isAjax:true
14
15 {
  "amount": {
    "amount": 20,
    "currency": "CHF"
  },
  "certificateFingerprint": "ef[REDACTED]417b",
  "moneyReceiver": {
    "firstName": [REDACTED],
    "lastName": [REDACTED]
  },
  "moneyReceiverMobileNumber": "+4179[REDACTED]",
  "moneySender": {
    "firstName": [REDACTED],
    "lastName": [REDACTED]
  },
  "orderId": "13976b6e-a57c-448a-8535-51d97f01928d",
  "reservationDate": "2020-07-10T08:55:12",
  "sendMoneyEvenIfCustomerUnknown": true,
  "signature": "gu2DEXJ5pqGx+0c6vQm0cU04MmYqyb+RIHT8iZ4jHGcul/Jx8iIwV1m6WU64G58oJnnEGH8WAr1dOmmc61/bZEjOEF3fRXR/2kffAreQNhE01Uc18sJFxx96iAt3Hfe336yHehB0qZ9zTKgtMZwGu8s3tzJNRpvrRszio2QCk5X7SIh26Ai04KD047uFmKEPThC

```

M4 Insecure Authentication

Prevent unauthorized access to the application

T4.1 Authentication required before accessing sensitive information

- Application must require PIN or fingerprint

T4.2 The application should have an inactivity timeout

T4.3 If a new fingerprint is added, authentication with fingerprints should be temporarily disabled

- User should provide PIN to enable fingerprints again
- Prevents attacks where an attacker adds their fingerprint to access the application

T4.4 It should not be possible to replay intercepted requests (e.g. a money transfer)

- An attacker intercepting a request for a money transfer could replay it to steal money from the victim.

M5: Insufficient Cryptography

```
112.     }
113.
114.     @TargetApi(8)
115.     public static File b(Context context) {
116.         if (bl.a()) {
117.             return context.getExternalCacheDir();
118.         }
119.         return new File(Environment.getExternalStorageDirectory().getPath() + "/");
120.     }
121.
122.     public static String b(String str) {
123.         try {
124.             MessageDigest instance = MessageDigest.getInstance("SHA-1");
125.             instance.update(str.getBytes());
126.             return a(instance.digest());
127.         } catch (NoSuchAlgorithmException unused) {
128.             return String.valueOf(str.hashCode());
129.         }
130.     }
131.
132.     @TargetApi(9)
133.     public static boolean b() {
134.         if (bl.b()) {
135.             return Environment.isExternalStorageRemovable();
136.         }
137.     }
138. }
```

Cryptography can only protect confidentiality and integrity of data if correctly implemented

T5.1 The app should not use unsafe crypto primitives

- E.g., MD5, SHA-1, RC4, DES, 3DES, Blowfish, ECB
- Search for these in the code
- Detection of these primitives does not imply that they are used for protecting critical information!

T5.2 The HTTPS connections should be configured according to best practices

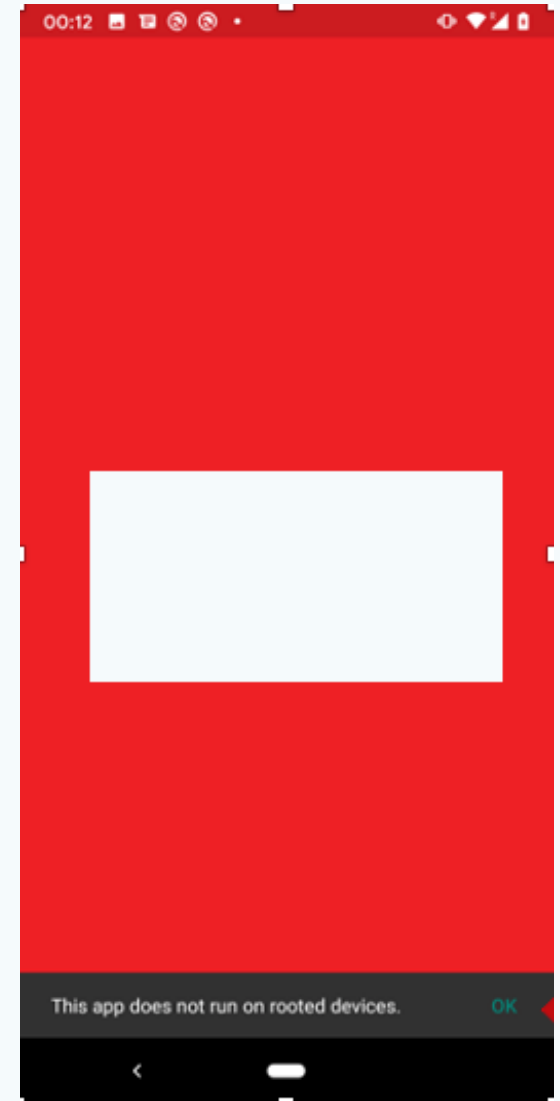
- Watch where the app connects to, use Qualys SSL labs to evaluate configuration, expect a grade of B or more

M8: Code Tampering

Prevent an attacker from tampering the code on the telephone

T8.1 The application should refuse to run on a rooted device

- On a rooted device, users can manipulate the code of the application



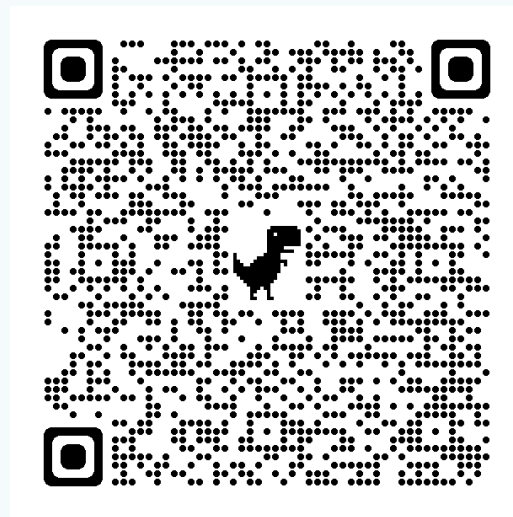
M9 Reverse engineering

Prevent attackers from analyzing the logic of the application

```
125.         instance.update(str.getBytes());
126.         return a(instance.digest());
127.     } catch (NoSuchAlgorithmException unused) {
128.         return String.valueOf(str.hashCode());
129.     }
130. }
131.
132. @TargetApi(9)
133. public static boolean b() {
134.     if (bl.b()) {
135.         return Environment.isExternalStorageRemovable();
136.     }
137.     return true;
138. }
139.
140. public Bitmap a(String str) {
141.     dt<String, Bitmap> dtVar = this.d;
142.     if (dtVar != null) {
143.         return dtVar.a(str);
144.     }
145.     return null;
146. }
147.
148. public void a() {
149.     synchronized (this.g) {
150.         if (this.c == null || this.c.a()) {
151.             File file = this.f.c;
152.             if (this.f.g && file != null) {
153.                 if (!file.exists()) {
154.                     file.mkdirs();
155.                 }
156.             }
157.         }
158.     }
159. }
```

T9.1 The code should be obfuscated

- When the code is obfuscated, it is much more difficult to understand the logic of the code
- This makes it more difficult to manipulate the code or to find potential vulnerabilities
- Decompile the code and assess its readability



<http://www.itu.int/go/dfssl>

Contact: dfssecuritylab@itu.int

Thank you!