

85% of App Store Apps Fail OWASP Mobile Top 10:

Are you exposed?



NowSecure™

AGENDA & SPEAKERS

- Introduction
- Inside OWASP Mobile Top 10
- Large Scale Analysis of 3rd Party Apps
- Recommendations
- Q & A



Tony Ramirez







Mobile Security Analyst







MOBILE APPS ARE TRACKING YOU

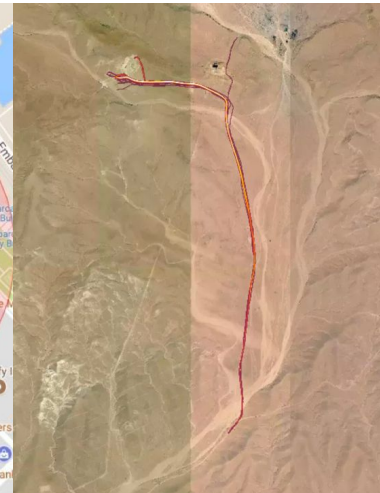
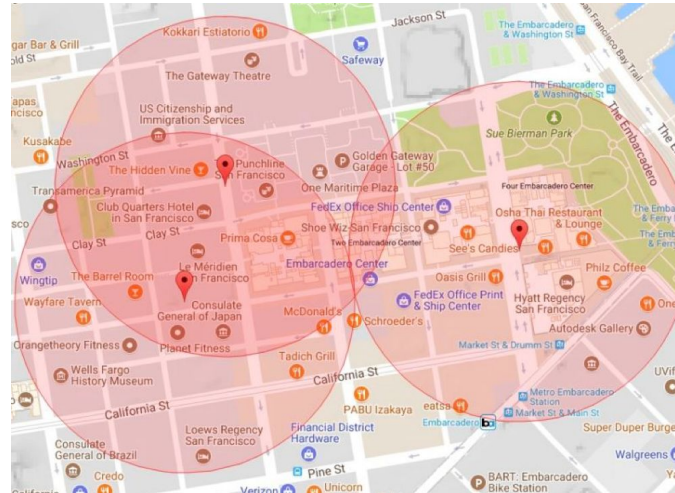
STRAVA™

HARVESTED DATA OF
APP USERS GEO

HARVESTED DATA OF
MILITARY GEO

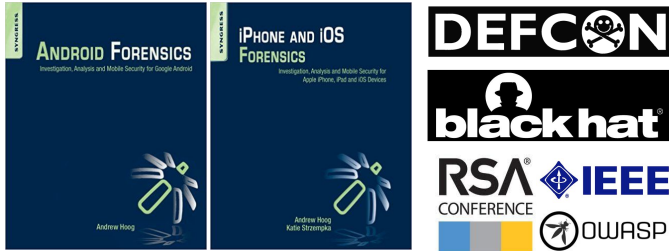
Mobile App
 Striver
 Straza Mate
 Run An Empire
 IpBike
 Fitness AR
 Teaming

Training
 Clliiimb
 Xert
 Tacit Training
 movecoach
 WattsBoard
 myWindsock.com



DEEP MOBILE SECURITY EXPERTISE

Books & Speaking



Mobile threat research is in our DNA

- Dream team of security researchers
- Every waking moment spent:
 - Discovering critical vulns
 - Identifying novel attack vectors
 - Creating/maintaining renowned open-source mobile security tools/projects

Open source

FRIDA



The NowSecure Mission

- Save the world from unsafe mobile apps
- Educate enterprises on the latest mobile threats
- Maximize the security of apps enterprises develop, purchase and use

INSIDE THE MOBILE APP ATTACK SURFACE

- Data caching
- Data stored in application directory
- Decryption of keychain
- Data stored in log files
- Data cached in memory/RAM
- Data stored in SD card
- OS data caching
- Passwords & data accessible
- No/Weak encryption
- TEE/Secure Enclave Processor
- Side channel leak
- SQLite database
- Emulator variance

DATA AT REST

CODE FUNCTIONALITY

- GPS spoofing
- Buffer overflow
- allowBackup Flag
- allowDebug Flag
- Code Obfuscation
- Configuration manipulation
- Escalated privileges
- Android rooting/iOS jailbreak
- User-initiated code
- Confused deputy attack
- Multimedia/file format parsers
- Insecure 3rd party libraries
- World Writable Files
- World Writable Executables
- URL schemes
- GPS spoofing
- Integrity/tampering/repacking
- Side channel attacks
- App signing key unprotected
- JSON-RPC
- Automatic Reference Counting
- Dynamic runtime injection
- Unintended permissions
- UI overlay/pin stealing
- Intent hijacking
- Zip directory traversal
- Clipboard data
- World Readable Files



- Wi-Fi (no/weak encryption)
- Rogue access point
- Packet sniffing
- Man-in-the-middle
- Session hijacking
- DNS poisoning
- TLS Downgrade
- Fake TLS certificate
- Improper TLS validation
- HTTP Proxies
- VPNs
- Weak/No Local authentication
- App transport security
- Transmitted to insecure server
- Zip files in transit
- Cookie “httpOnly” flag
- Cookie “secure” flag

DATA IN MOTION

API BACKEND

- Platform vulnerabilities
- Server misconfiguration
- Cross-site scripting
- Cross-site request forgery
- Cross origin resource sharing
- Brute force attacks
- Side channel attacks
- SQL injection
- Privilege escalation
- Data dumping
- OS command execution
- Weak input validation
- Hypervisor attack
- VPN

A grayscale image of a stack of books with a graduation cap (mortarboard) resting on top. The books are stacked in a slightly irregular manner, with some pages visible. The graduation cap is positioned centrally on the stack. The entire scene is set against a plain, light background. The text is overlaid in the center of the image.

Inside OWASP & OWASP Mobile Top 10

OWASP MOBILE TOP 10

2011 OWASP Group determined Mobile is Different

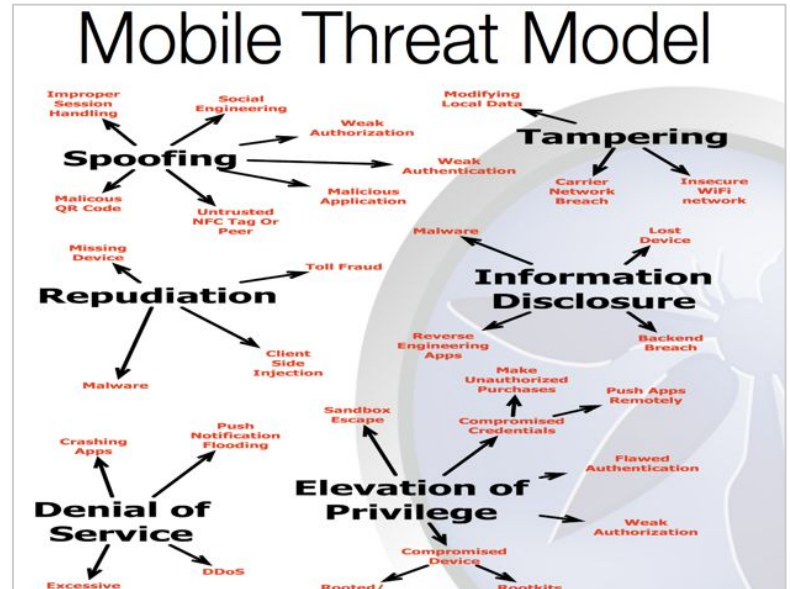
- Mobile OS Platforms vary widely
- Mobile apps very different from traditional web app model due to wildly varying use cases and usage patterns

Must consider more than the “Apps”

- Remote web services
- Platform integration (iCloud, GCM)
- Device (in)security considerations

Intended to be platform-agnostic

- Focused on areas of risk rather than individual vulnerabilities
- Weighted utilizing the OWASP Risk Rating Methodology



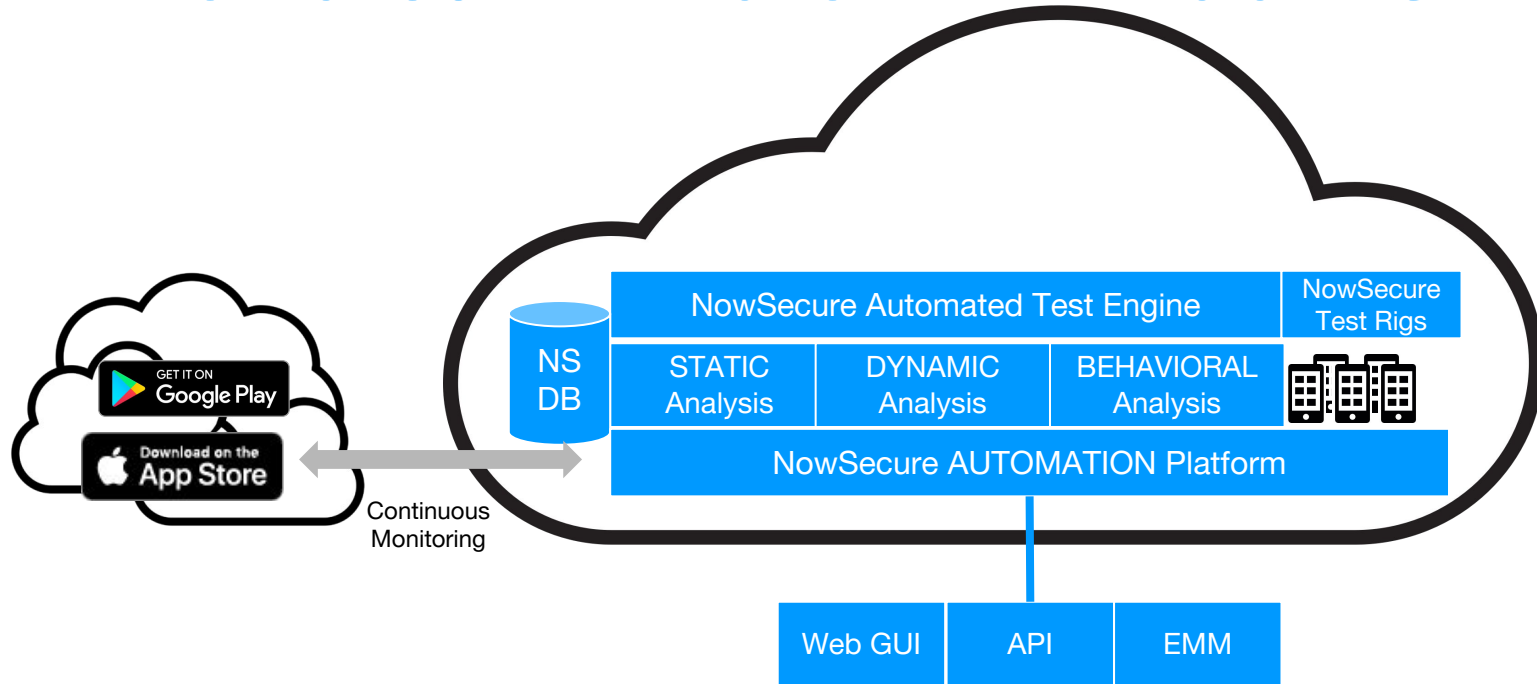
OWASP MOBILE TOP 10

M1 - Improper Platform Usage	Misuse of features like Touch ID, permissions, Keychain	
M2 - Insecure Data Storage	Data Leakage, client-side injection, weak server-side controls	
M3 - Insecure Communication	Poor handshake, SSL/TLS/Cert issues, transfer in clear text	
M4 - Insecure Authentication	Improper identity mgmt, weak session mgmt	
M5 - Insufficient Cryptography	Lack of crypto, improper crypto use	
M6 - Insecure Authorization	Improper local auth, forced browsing	
M7 - Client Code Quality	Code mistakes eg. Buffer overflows, format string vulns	
M8 - Code Tampering	Binary patching, method hooking/swizzling, memory mods	
M9 - Reverse Engineering	Exposure to attacker reversing tools	
M10 - Extraneous Functionality	Dev/QA inadvertent disabling security, hidden backdoors	



Analysis of Mobile App Store Apps for OWASP Mobile Top 10

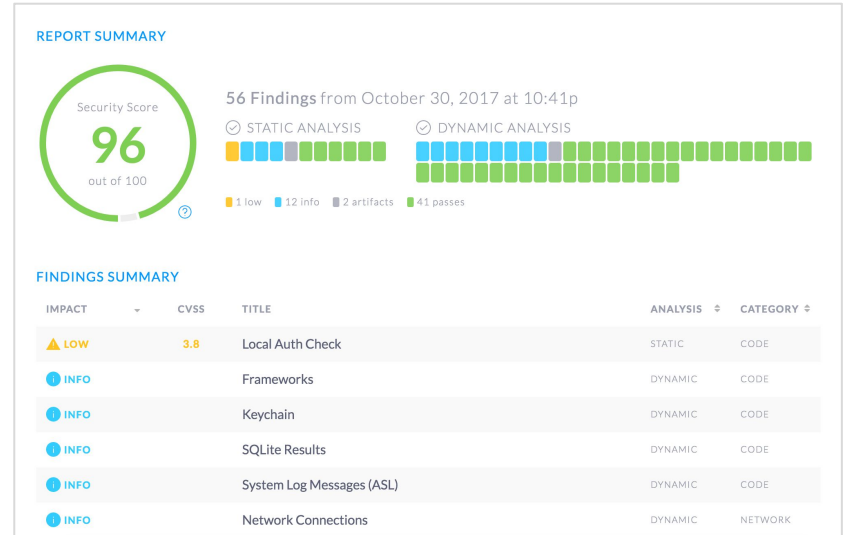
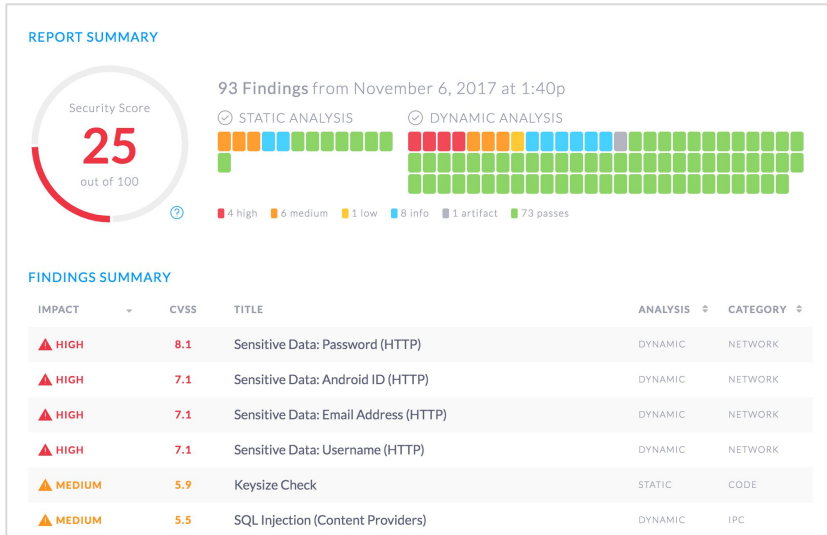
NOWSECURE APP STORE ANALYSIS ENGINE



NowSecure INTEL

*AlwaysOn AppStore Cloud Analysis
for EMM & Security teams*

INSIDE MOBILE APP RISK SCORING



INDUSTRY STANDARD CVSS SCORES



INDUSTRY REGULATORY COMPLIANCE

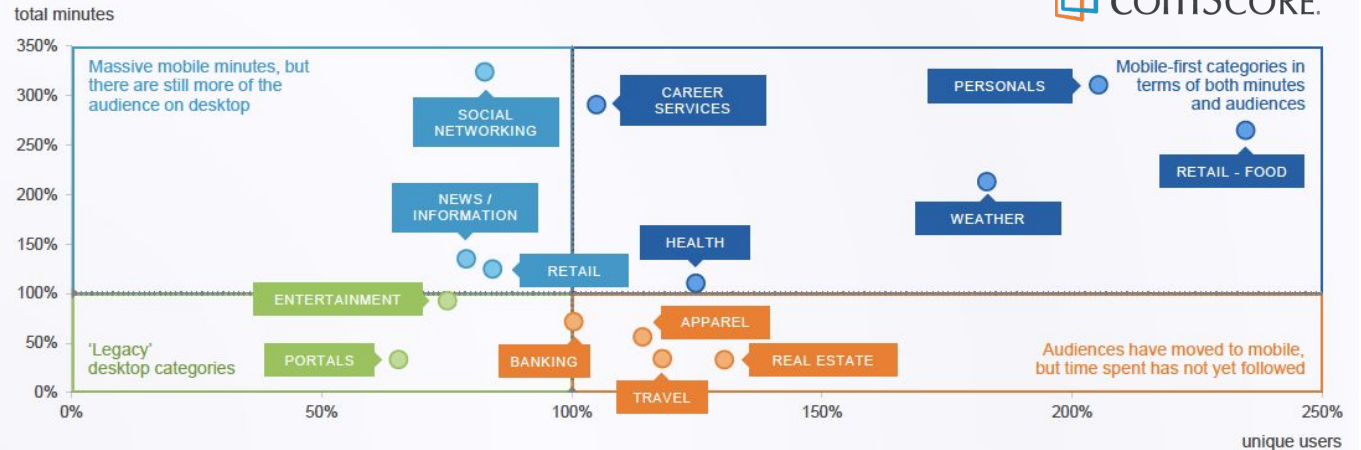
- OWASP: [Mobile Top 10: M3-Insecure Communication](#)
- GDPR: [Risks violating Article 25](#)
[Risks violating Article 32](#)
- FFIEC: [May violate D3.PC.Am.Int.7](#)
- PCI: [May violate requirement 4.1](#)

NOWSECURE APP STORE ANALYSIS PROJECT



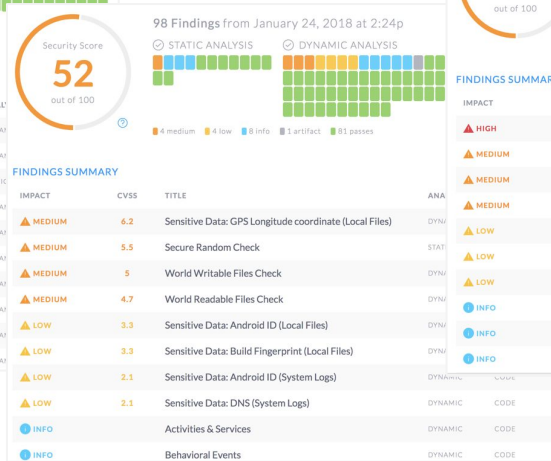
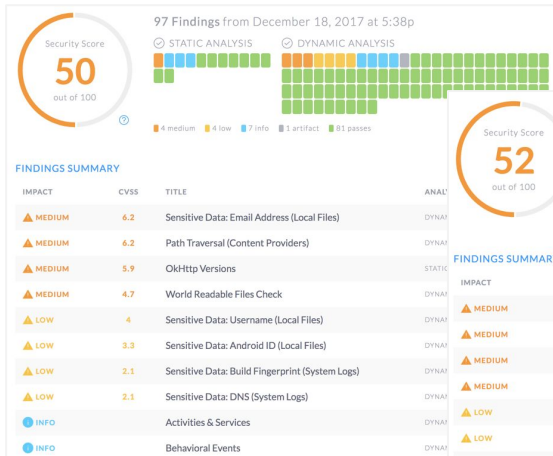
- 45,000 public apps posted to Apple App Store and Google Play store
- Broad distribution of categories

Category mobile users & minutes as a % of desktop



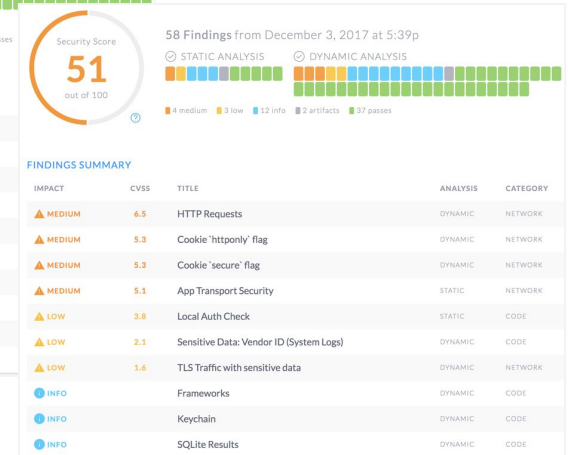
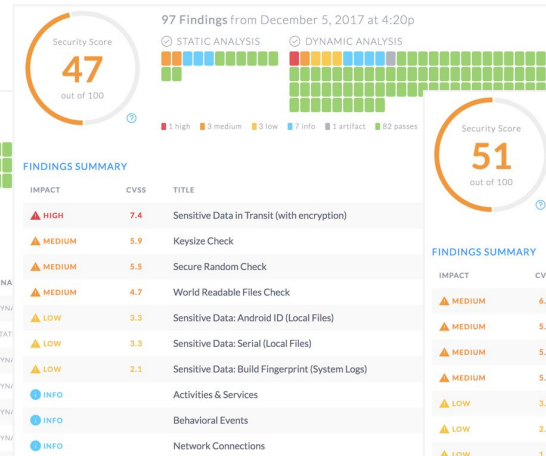
RISKY APPS THAT YOU MIGHT BE USING

POPULAR BUSINESS EMAIL



POPULAR BUSINESS CHAT APP

POPULAR BUSINESS CRM



POPULAR BUSINESS TRAVEL APP

OWASP MOBILE TOP 10 - 3rd PARTY ANALYSIS [TOP 7]

M1 - Improper Platform Usage	Misuse of features like Touch ID, permissions, Keychain	
M2 - Insecure Data Storage	Data Leakage, client-side injection, weak server-side controls	50% Fail

TESTING FOR RISK -- DATA AT REST

	Android	iOS	Total
M2-Insecure Data Storage	85%	16%	50%

- Local log/file data
 - Account Credentials
 - PII
 - Email
 - Geolocation
 - IMEI/Serial Number
 - WiFi
- World Writable Executables
 - 52% of Android Apps
- External storage
 - Risk depends on your policy

⚠ Writable Executable Check CVSS: 7.7 DYNAMIC PERMISSIONS 1 / 92

Writable executable files were discovered in the application.

DESCRIPTION
Checks for calls within the application using writable executable permissions. The table below will show any violations that were detected.

REGULATORY
NIAP: [FMT_CFG_EXT.1.2FPT_AEX_EXT.1.2FPT_AEX_EXT.1.4](#)
OWASP: [Mobile Top 10: M2-Insecure Data Storage](#)
FFIEC: [May violate D3.1](#)
PCI: [May violate requirement 3.1 through 3.4](#)
HIPAA: [May violate §164.312\(a\)\(1\): Standard: Access control](#)

⚠ Sensitive Data: Password (Local Files) CVSS: 6.2 DYNAMIC DATA STORAGE 3 / 93

Password has been found within local application folders or external storage locations on the device. You can review the output below to see where each instance was found.

DESCRIPTION
Local application files and external storage locations are inspected for sensitive user/application data. For this check, instances of the Password were searched.

REGULATORY
OWASP: [Mobile Top 10: M2-Insecure Data Storage](#)
FFIEC: [May violate D3.PC.Am.A.1](#)
PCI: [May violate requirement 3.1 through 3.4](#)
HIPAA: [May violate §164.312\(a\)\(1\): Standard: Access control](#)

SEARCH TERM	ENCODING	LOCATION(S)
password	original	/data/data/acc.app/files/settings.prope...

📁 Files Stored on SD Card DYNAMIC

Files were found to be stored in an external storage location which allows any app to access and read files in external storage. Under certain circumstances, allows apps to securely store data. <http://source.android.com/devices/tech/storage/>

DESCRIPTION
As the application is running, we monitor external storage for files that are stored in the application's external storage.

FILES

```
/sdcard/Android/data/com.facebook.orca/cache  
/sdcard/Android/data/com.facebook.orca/files  
/sdcard/Android/data/com.facebook.orca/files/Download  
/sdcard/Android/obb/com.facebook.orca
```

OWASP MOBILE TOP 10 - 3rd PARTY ANALYSIS [TOP 7]

M1 - Improper Platform Usage	Misuse of features like Touch ID, permissions, Keychain	
M2 - Insecure Data Storage	Data Leakage, client-side injection, weak server-side controls	50% Fail
M3 - Insecure Communication	Poor handshake, SSL/TLS/Cert issues, transfer in clear text	48% Fail

TESTING FOR RISK -- DATA IN MOTION

	Android	iOS	Total
M2-Insecure Communication	20%	76%	48%

- Assume that the network layer is not secure and is susceptible to eavesdropping
- Frequent lack of proper iOS ATS and cross-platform SSL implementations
- Unencrypted data OTA
 - Account Credentials
 - PII
 - Email
 - Geolocation
 - IMEI/Serial Number
- 30% of iOS apps use HTTP (not HTTPS)

⚠ Sensitive Data in Transit (with encryption) CVSS: 7.4 DYNAMIC NETWORK 2 / 92

One or more sensitive values were intercepted in transit, due to a lack of proper certificate validation. This is a high risk vulnerability as it is possible for an attacker on the same network to easily retrieve this information. It is encouraged to review the table below, which displays the type of data that was intercepted, whether it is sent in plain text or a special encoding, the actual value that was recovered, the related URL, and some additional context around this violation.

DESCRIPTION
This test determines whether the proper cert validation could result in decrypting the application's traffic. Mac Address, IMEI, Device Ser

REGULATORY
OWASP: [Mobile Top 10; M3-Insec](#)
FFIEC: [May violate D3.PC.Am.In](#)
PCI: [May violate requirement](#)
HIPAA: [May violate §164.312\(e\)](#)

TYPE	FORMAT	VALUE
US		
pb		

⚠ Broken SSL Check CVSS: 7.4 DYNAMIC NETWORK

The application does not have proper certificate validation implemented. Broken SSL/TLS was detected for the

DESCRIPTION
Determines whether the application is performing proper certificate validation and hostname verification. certificate validation could result in sensitive data being intercepted by a man-in-the-middle attack.

REGULATORY
CWE: [319](#)
OWASP: [Mobile Top 10; M3-Insecure Communication](#)
FFIEC: [May violate D3.PC.Am.Int.7](#)
PCI: [May violate requirement 4.1](#)

⚠ Sensitive Data: Password (HTTP) CVSS: 8.1 DYNAMIC NETWORK 1 / 93

Password was intercepted over HTTP traffic. The table below provides the context of how the data was transmitted (plain-text or encoded) and the URL of this violation.

DESCRIPTION
Traffic is analyzed to determine if any sensitive data is transmitted insecurely over the network without encryption. For this check, instances of the Password were searched across any intercepted traffic.

REGULATORY
NIAP: [FTP DIT EXT.1.1FCS TLSC EXT.1.2FCS TLSC EXT.1.3FCS TLSS EXT.1.5FCS DTLS EXT.1.3](#)
OWASP: [Mobile Top 10; M3-Insecure Communication](#)
FFIEC: [May violate D3.PC.Am.Int.7](#)
PCI: [May violate requirement 4.1](#)
HIPAA: [May violate §164.312\(e\)\(1\); Standard: Transmission security](#)

OWASP MOBILE TOP 10 - 3rd PARTY ANALYSIS [TOP 7]

M1 - Improper Platform Usage	Misuse of features like Touch ID, permissions, Keychain	
M2 - Insecure Data Storage	Data Leakage, client-side injection, weak server-side controls	50% Fail
M3 - Insecure Communication	Poor handshake, SSL/TLS/Cert issues, transfer in clear text	48% Fail
M4 - Insecure Authentication	Improper identity mgmt, weak session mgmt	5% Fail
M5 - Insufficient Cryptography	Lack of crypto, improper crypto use	
M6 - Insecure Authorization	Improper local auth, forced browsing	2% Fail
M7 - Client Code Quality	Code mistakes eg. Buffer overflows, format string vulns, 3rd Party	32% Fail

TESTING FOR RISK -- CODE & 3rd PARTY

	Android	iOS	Total
M7-Client Code Quality	59%	4%	32%

- iOS clearly has strong code quality practices
- Nearly all apps have 3rd party/OSS libraries
 - Open source often untested/unvetted
 - Inconsistent pattern of upgrading to latest more secure library versions
- Android app challenges
 - 1465 arbitrary code injection
 - 1133 SQL injection
 - 112 Debug flag on

Stack Smashing Check CVSS: 9.8 STATIC CODE 2/56

Stack smashing protection has not been implemented in your application.

DESCRIPTION
When an application is compiled with stack smashing protection, a known value or "canary" is placed on the stack directly before the local variables to protect the saved base pointer, saved instruction pointer, and function arguments. The value of the canary is determined by a heuristic to intelligently choose a value that is unlikely to appear in arrays. This test checks if the application is vulnerable to stack smashing attacks.

REGULATORY
NIAP: [FPT AEX EXT.1.5](#)
OWASP: [Mobile Top 10: M7-Client Code Quality](#)
FFIEC: [May violate D3.PC.Im.1.6](#)
PCI: [May violate requirement 6.5](#)

Heartbleed Check CVSS: 7.4 STATIC CODE 19/57

Your application was not found to be vulnerable to the Heartbleed vulnerability.

DESCRIPTION
This test checks to see if your application is vulnerable to the Heartbleed vulnerability. This serious issue is caused by a vulnerable version of library called OpenSSL (1.0.1 with heartbeats support enabled). In this version, the `tls1_process_heartbeat` function does not validate its input properly and can lead to information disclosure due to the inclusion of memory that contains sensitive information like credentials or other data.

AFNetworking Implementation CVSS: 7.1 DYNAMIC CODE 1/57

Your application was found to be using an outdated version of the AFNetworking library. This vulnerability was patched as of version 2.5.2, however, if an older version is used, it allows all the SSL traffic to be intercepted and decrypted in a standard man-in-the-middle environment.

DESCRIPTION
Checks the security of the AFNetworking library's implementation setting, which allows developers to add networking functionality to their applications.

REGULATORY
OWASP: [Mobile Top 10: M3-Insecure Communication](#)
FFIEC: [May violate D3.PC.Im.1.6](#)
PCI: [May violate requirement 6.5](#)
HIPAA: [May violate requirement 6.5](#)

SQL Injection (Content Providers) CVSS: 8.5 DYNAMIC IPC 2/95

The application was found to be vulnerable to SQL Injection targeting its content provider(s). Context of the tests run, the vulnerable uri(s), and leaked data can be referenced in the table below. This means that any application, including proof-of-concept ones, that simply require this permission in its Android Manifest will be able to target the content provider and retrieve potentially leaked data.

DESCRIPTION
Android applications may use untrusted input to construct SQL queries and do so in a way that's exploitable. The most common case is when applications do not sanitize input for any SQL and do not limit access to content providers. Any vulnerable content providers will be listed in the table below.

REGULATORY
CWE: [926](#)
OWASP: [Mobile Top 10: M7-Client Code Quality](#)
FFIEC: [May violate D3.PC.Im.1.6](#)
PCI: [May violate requirement 6.5](#)
HIPAA: [May violate §164.312\(a\)\(1\): Standard: Access control](#)

OWASP MOBILE TOP 10 - 3rd PARTY ANALYSIS [TOP 7]

M1 - Improper Platform Usage	Misuse of features like Touch ID, permissions, Keychain	
M2 - Insecure Data Storage	Data Leakage, client-side injection, weak server-side controls	50% Fail
M3 - Insecure Communication	Poor handshake, SSL/TLS/Cert issues, transfer in clear text	48% Fail
M4 - Insecure Authentication	Improper identity mgmt, weak session mgmt	5% fail
M5 - Insufficient Cryptography	Lack of crypto, mproper crypto use	
M6 - Insecure Authorization	Improper local auth, forced browsing	2% Fail
M7 - Client Code Quality	Code mistakes eg. Buffer overflows, format string vulns	32% Fail
M8 - Code Tampering	Binary patching, method hooking/swizzling, memory mods	
M9 - Reverse Engineering	Exposure to attacker reversing tools	32% Fail
M10 - Extraneous Functionality	Dev/QA inadvertent disabling security, hidden backdoors	47% Fail

TESTING FOR RISK -- TAMPERING

	Android	iOS	Total
M9-Reverse Engineering	64%	0%	32%
M10- Extraneous Functionality	92%	2%	47%

- Obfuscation insufficiently used by Android developers
- 90% of Android apps allow backup of data
- 1465 Android apps allow arbitrary code execution

Obfuscation Check CVSS: 4 STATIC CODE 7 / 94

The source code does not appear to have been obfuscated. Your intellectual property is at risk because your app can easily be reverse-engineered.

DESCRIPTION
Checks if the source code has been obfuscated either by Proguard or Dexguard in order to make class identification less obvious.

REGULATORY
OWASP: [Mobile Top 10: M9-Reverse Engineering](#)

Javascript Interface Check CVSS: 2 H STATIC CODE 10 / 94

Your application is using `addJavaScriptInterface()`.

DESCRIPTION
Checks for the usage of `addJavaScriptInterface()`. This can be used to intercept network traffic that's being sent and interact with the javascript interface.

REGULATORY
CWE: [501](#)
OWASP: [M10-Extraneous Functionality](#)
FFIEC: [N/A](#)

Arbitrary Code Execution (Probable) CVSS: 7.7 DYNAMIC PERMISSIONS 1 / 94

A world_writable file was found, and certain behaviors were detected which indicates that the app might have a code path that executes this file (eg a lua or a python script on the SD card). This combination of behaviors could allow other apps to cause this one to execute their own code.

DESCRIPTION
Checks for arbitrary code execution. When executable code is world writable, another app could swap the file and gain code execution in the context of another app.

REGULATORY
OWASP: [Mobile Top 10: M8-Code Tampering](#)
FFIEC: [May violate D3.PC.Im.1.6](#) [May violate D3.PC.Se.B.1](#)
PCI: [May violate requirement 6.5](#)
HIPAA: [May violate §164.312\(c\)\(1\); Standard: Integrity](#) [May violate §164.312\(a\)\(1\); Standard: Access control](#)

Allow Backup Check CVSS: 4.8 STATIC CODE

Your application is declaring the `allowBackup` flag as true (which is true as well).

This can allow an attacker to backup your application folder.

DESCRIPTION
Checks to determine whether the `allowBackup` flag is enabled, it could allow easier access to the application data.

REGULATORY
CWE: [538359](#)
NIAP: [FPT_AEX_EXT.1.3](#)
OWASP: [Mobile Top 10: M10-Extraneous Functionality](#)
FFIEC: [May violate D3.PC.Am.B.10](#)
PCI: [May violate requirement 6.4.4](#) [May violate requirement 3.1 through 3.4](#)
HIPAA: [May violate §164.312\(c\)\(1\); Standard: Integrity](#) [May violate §164.312\(a\)\(1\); Standard: Access control](#)

TESTING FOR RISK -- PERMISSIONS & ENTITLEMENTS

- Risk Dependent on your corporate policies
- Sample potentially risky permissions
 - Contact list access
 - Write external storage
 - Calendar
 - Send SMS
 - NFC

Activities & Services DYNAMIC CODE 8 / 93

DESCRIPTION
The activities called for by an app are an important part of understanding the application's lifecycle, from the initial main activity launch to the final activity shutdown. The main activity is the main entry point into the application's user interface.

Permissions	
	android.permission.INTERNET
	com.android.launcher.permission.INSTALL_SHORTCUT
	com.android.launcher.permission.UNINSTALL_SHORTCUT
	android.permission.READ_CONTACTS
	android.permission.RECEIVE_BOOT_COMPLETED
	android.permission.USE_CREDENTIALS
	android.permission.ACCESS_COARSE_LOCATION
	android.permission.READ_PHONE_STATE
	android.permission.READ_CALL_LOG
	android.permission.ACCESS_NETWORK_STATE
	android.permission.READ_LOGS
	android.permission.VIBRATE
	android.permission.GET_ACCOUNTS
	android.permission.WRITE_EXTERNAL_STORAGE
	android.permission.CALL_PHONE
	android.permission.READ_CALENDAR
	android.permission.NFC
	android.permission.EXPAND_STATUS_BAR
	android.permission.SEND_SMS
	com.todoroo.astrid.READ
	android.permission.ACCESS_WIFI_STATE
	com.anydo.permission.C2D_MESSAGE
	com.google.android.c2dm.permission.RECEIVE

TESTING FOR RISK -- IP ADDRESSES

- Risk Dependent on your corporate policies
- 3rd party libraries, SDKs are common culprits
Ad networks frequently uniquely identify users and geo-locate them insecurely
- Apps frequently have 100s of connections (this one had 250)

Network Connections

DYNAMIC NETWORK 11/93

Network connections were detected during dynamic analysis of the application. Any unexpected connections should be flagged for further analysis, as many third party libraries, especially those linked to advertising or analytic SDKs may send data to unexpected locations.

DESCRIPTION
As the application is running, we monitor the network communications in order to understand where the application is sending its data.

PAGE SIZE: 10

SEARCH:

DOMAIN	HOST	IP	ORGANIZATION	LOCATION
amazon.com	api.kiip.me	52.72.58.52	Amazon Technologies Inc.	Ashburn, Virginia, US
amazon.com	api.kiip.me	54.84.215.79	Amazon Technologies Inc.	Ashburn, Virginia, US
amazon.com	api.kiip.me	54.165.235.126	Amazon Technologies Inc.	Ashburn, Virginia, US
amazon.com	api.kiip.me	52.44.137.112	Amazon Technologies Inc.	Seattle, Washington, US
amazon.com	api.kiip.me	52.2.19.56	Amazon Technologies Inc.	Ashburn, Virginia, US
amazon.com	api.kiip.me	54.209.4.42	Amazon.com Inc.	Ashburn, Virginia, US
google.com	ssl.google-analytics.com	172.217.4.232	Google Inc.	Mountain View, California, US

OWASP MOBILE TOP 10

M1 - Improper Platform Usage	Misuse of features like Touch ID, permissions, Keychain	
M2 - Insecure Data Storage	Data Leakage, client-side injection, weak server-side controls	50% Fail
M3 - Insecure Communication	Poor handshake, SSL/TLS/Cert issues, transfer in clear text	48% Fail
M4 - Insecure Authentication	Improper identity mgmt, weak session mgmt	5% Fail
M5 - Insufficient Cryptography	Lack of crypto, mproper crypto use	
M6 - Insecure Authorization	Improper local auth, forced browsing	2% Fail
M7 - Client Code Quality	Code mistakes eg. Buffer overflows, format string vulns	32% Fail
M8 - Code Tampering	Binary patching, method hooking/swizzling, memory mods	
M9 - Reverse Engineering	Exposure to attacker reversing tools	32% Fail
M10 - Extraneous Functionality	Dev/QA inadvertent disabling security, hidden backdoors	47% Fail

A Newton's cradle with five silver spheres. The leftmost sphere is in motion, having just struck the others or about to. The other four spheres are in a row, slightly offset from their resting position. The background is a dark, solid color.

Recommendations & Next Steps

BEST PRACTICES RECOMMENDATIONS

FOR ENTERPRISES

1. Recognize the risks of 3rd party apps on BYOD and COPE devices
 - **Assume all are untrusted until validated**, no matter who the developer
2. Put controls and processes in place to analyze and monitor 3rd party app risk
 - Inventory & analyze your existing mobile apps leveraging EMM/MDM
 - Adapt processes to review and approve all new mobile apps before introduction
 - Leverage automated tools for in depth testing and continuous monitoring
3. Find a reputable source to stay up to date on the latest threats
 - Sign up for Nowsecure #MobSec5 at www.nowsecure.com/go/subscribe
 - Read our blog at www.nowsecure.com/blog

FOR APP DEVELOPERS

1. Train developers on secure coding best practices & fully vet 3rd party libraries
 - Leverage the NowSecure Guide to Secure Mobile App Development Best Practices
2. Ensure all mobile app releases are properly security pen tested
 - Leverage automated mobile appsec testing tools in SDLC lifecycle
 - Leverage 3rd party expert mobile app Pen Testing

GET A FREE MOBILE APP SECURITY REPORT

- Free for OWASP Members
- Delivered by NowSecure Mobile App Security Experts
- Choose a 3rd Party Mobile app used in your business
- Surf to request: <http://bit.ly/2BB8sAk>



NowSecure

LIMITED TIME OFFER - ACT FAST!

Get Your FREE NowSecure INTEL Mobile App Security Report

Corporate Document Storage App

83

Mobile Chat App

50

Don't fly blind on 3rd-party mobile app risk!

Thousands of devices within the average enterprise and hundreds of apps per device spawn millions of potential risks seeping through the disintegrating mobile perimeter.

In testing millions of the top Apple® App Store® and Google Play™ store apps, we find a surprising number of problems including:

- 49% of apps harbor **at least one high-risk flaw**
- 17% of apps **reveal sensitive user data**
- 29% of apps **send unencrypted traffic**

NowSecure INTEL empowers enterprise mobility and security teams to make swift, smart decisions for mobile app risk management – including blacklist/whitelist – based on the world's highest quality,

Get Deep Security Insight Into One 3rd-Party Mobile App - FREE

* First Name

-

reedontherun@gmail.com

* Direct Phone Number

-

NOWSECURE COMING ATTRACTIONS



WEBINAR: “Top OSS for Mobile AppSec Testing: The Latest on R2 & FRIDA”

Delivered by the creators of R2 & FRIDA from NowSecure Research Team

Tomorrow: Weds Feb 21, 2019

Register Now: <https://www.nowsecure.com/events/>

RSA[®]
Conference
2018

RSA 2018

April 16-20, 2018

Meet us at **booth 3229 (North Expo)**
in San Francisco, CA!



Open Q&A

Tony Ramirez, Security Analyst

NowSecure

+1 312.878.1100

@NowSecureMobile

www.nowsecure.com

Subscribe to #MobSec5

A digest of the week's mobile security news that matters

<https://www.nowsecure.com/go/subscribe>