RIIS' 3rd Annual Dating Android App Security Index

"Making the world a safer place, one dating app at a time"

Executive Summary

As part of our ongoing research into mobile security, RIIS LLC has put together a third Android Mobile Security Index for dating apps. Just before Valentine's Day 2016 we analyzed six of the most popular dating apps. We looked at each of these apps and rated them using the updated industry standard OWASP Mobile Top Ten Risks which can be found at https://www.owasp.org/index.php/OWASP_Mobile_Security_Project

Tinder and Coffee Meets Bagel came out on top as it had no major security issues. Having Facebook handle the personal information necessary for these applications resulted in no significant information being stored on the device. We were able to penetrate the security of each of the other apps.

For the third year in a row, Match.com fell short when it came to securing their application. Username and password, while encrypted, could be pulled from the device. No obfuscation meant we could decrypt the password and recover the user login credentials.

On the positive side, it was great to see that every other tested application used some form of code obfuscation to hide information from anyone reverse engineering the code.

We analyzed these apps using the following OWASP criteria:

- Weak Server Side Controls
- Insecure Data Storage
- Insufficient Transport Layer Protection
- Unintended Data Leakage
- Poor Authorization and Authentication
- Broken Cryptography
- Client Side Injection
- Security Decisions Via Untrusted Inputs
- Improper Session Handling
- Lack of Binary Protections

Android App Rank	Company	OWASP Score
1	Tinder	0
1	Coffee Meets Bagel	0
2	OkCupid	2
3	Plenty of Fish	4
3	eHarmony	4
4	Match.com	6

Key Findings

Leaders

The top ranking app was:

	Company	Score	Weak Server Side Controls	Insecure Data Storage	Insufficient Transport Layer Protection	Unintended Data Leakage	Poor Authorization and Authentication	Broken Cryptography	Client Side Injection	Improper Session Handling	Security Decisions via Untrusted Inputs	Lack of Binary Protections
	Tinder	0	0	0	0	0	0	0	0	0	0	0
	Coffee Meets Bagel	0	0	0	0	0	0	0	0	0	0	0

Tinder and Coffee Meets Bagel ranked highest in our analysis. User accounts require Facebook to login. All traffic being passed form the client to the server uses SSL Transport Layer Protection. Neither of these applications store significant information on the device and each of their binaries use full code obfuscation.

Room for improvement

The bottom ranking apps were:

	Company	Score	Weak Server Side Controls	Insecure Data Storage	Insufficient Transport Layer Protection	Unintended Data Leakage	Poor Authorization and Authentication	Broken Cryptography	Client Side Injection	Improper Session Handling	Security Decisions via Untrusted Inputs	Lack of Binary Protections
2	OkCupid	2	1	1	0	0	0	0	0	0	0	0
3	eHarmony	4	1	1	0	0	0	0	0	1	0	1
4	POF	4	1	1	0	0	1	0	0	1	0	0
5	Match.com	6	1	1	0	0	1	1	0	1	0	1

eHarmony and Match.com both save encrypted login information on the phone. Both of these apps use encryption but do not use obfuscation. If you store the user data on the phone then no matter how hard you try to protect it, all the hacker has to do is backup the app data and then restore it on a different phone. This is the trade off in mobile development; if you want to protect your user's data, you have to make them login when they use the application.

Match.com scored the worst as has insecure data storage and insufficient transport layer protection. The username and password is encrypted in the shared preferences file but can be easily recovered by analyzing the app's decompiled source code.

Plenty of Fish had no transport layer protection in place when accessing the users *edit user profile* portion of their application. A users profile could be intercepted via a Man-in-the-Middle attack.

To gain access to this user data, a hacker would need access to the person's unlocked phone so they can backup the app's runtime data and APK. We recommend enabling the screen lock if you have any of these apps on your phone.

Takeaways

1. The safest app did not store any login information or sensitive user data on the Android device.

2. It is common practice (and a fundamental security flaw) to store the username and password encrypted in a shared preferences folder with a hardcoded encryption key so that the user does not have to login after first use.

3. Backing up the app data and restoring it onto another phone allows you to gain access.

4. If a spouse finds these apps on your phone, he or she does not need to hack anything as they can simply open the apps.

5. Exploits based on these security issues are limited to a hacker backing up single devices requiring physical access to a person's phone. A more widespread attack would require a malware app that is specifically written to exploit any security issues identified.

6. These hacks do not require any of the phones to be rooted.

7. Obfuscation is now common practice.

8. If you don't ask your users to login when they use your app then all the hacker has to do is backup the app data and then restore it on a different phone. This is one of the major security tradeoffs in mobile development.

Methodology

We only analyzed apps for which we had valid user accounts.

The apps were analyzed as follows for login, sensitive information and http issues:

- Connect phone or tablet to PC using USB cable
- Backup the app's runtime data using adb backup command
- Convert the backup data into readable format using Android Backup Extractor, abe.jar
- Look for user information in shared preferences and database folders
- If encrypted data is found then search for encryption key
- Pull the APK off the phone using adb pull command
- Decompile the APK using JADX
- Look for encryption key by searching for encrypt and decrypt routines in decompiled source
- Decrypt encrypted runtime user information.
- Restore the app data onto another phone using the adb restore command
- Proxy http and SSL traffic
- Attempt man-in-the-middle attack

- Log into app on web server

We award 1 point for an identified security issue or 0.5 point for a suspected issue that we haven't been able to exploit. Lower scores mean fewer issues from the OWASP top 10 were identified. All apps were analyzed in February 2015.

The OWASP security issues are as follows:

1. Weak Server Side Controls - Backend APIs and web server are not secure.
2. Insecure Data Storage - Sensitive information is stored in cleartext and left unprotected.
3. Insufficient Transport Layer Protection - http or https traffic is not secure.
4. Unintended Data Leakage -Sensitive data is leaked in log or temp files or via 3rd party libraries, or when test data is included in the production app.
5. Poor Authorization and Authentication - Access can be achieved due to insecure tokens or poor login authorization.
6. Broken Cryptography - Encryption is incorrectly implemented or the key is visible in the SQLite database or in the decompiled APK source code.
7. Client Side Injection - Hybrid apps have SQL injection of XSS issues.
8. Improper Session Handling - The app rarely if ever asks the user to login after the initial login as the session never expires.
9. Security Decisions via Untrusted Inputs - The app does not use a minimal permissions required model.
10. Lack of Binary Protections – Java code in APK is not obfuscated.

How Does Your App Score?

Our team of mobile security professionals will audit your code and provide you with a score. We can then work together to help you improve your score and truly secure your mobile app. Contact us today.

About RIIS

RIIS is an IT consulting firm based in Troy, MI. Our primary service includes accelerated application development through visualization and automated tools for web and mobile technologies. We help companies get the applications they need, faster! Industry experience includes software, e-commerce, advertising, defense, insurance, banking/finance, and telecommunications.