Table of contents

1 Overview
1 Report Findings
2 Research Findings
2 Privacy Issues
3 Lack of Binary Protection
3 Insecure Data Storage
3 Transport Security
4 Weak Server Side Controls
4 Conclusion
5 Methodology
Overview

As computing becomes borderless, adversaries are increasingly bypassing perimeter security and taking advantage of vulnerabilities brought on by the growing number of applications and their unsecured entry points. At the same time, business managers are dramatically increasing the rate of deployment of often rushed-to-market mobile applications, many of which are, by necessity, developed by third parties. As a result, mobile applications represent a real security threat, emphasizing the need for a mobile application security strategy that enables businesses to go from “fast-to-market” to “secure-and-fast-to-market.”

Report Findings

Enterprise mobile applications sit side by side with dozens, if not hundreds, of consumer mobile apps and stored personal information. This introduces unnecessary exposure that can be easily resolved—if the vulnerabilities are identified and addressed prior to releasing an application for deployment. HPE Security Research leveraged HPE Fortify on Demand (FoD) Mobile to scan more than 2,000 mobile applications from more than 600 companies, revealing alarming realities regarding the sheer number of applications vulnerable to attack, as well as the most common and easily addressable vulnerability errors.

Statistics on apps:

- Analyzed 2,107 applications published by companies on Forbes list of Global 2000
- Applications from 601 different companies
- Companies located in 50 countries
- Companies operating in 76 industries
- Applications from 22 categories, including productivity and social networking
Privacy Issues

With dozens of consumer applications, personal information, and enterprise mobile apps on the same device, they appear to act independently; however, without proper security built in to mobile applications, hidden integration and communication may exist. Should the newest version of the game Mad Mallards have access to text your contact list or sent email? Or actually be able to call a telephone number? What if it wanted to send your contact list to a third-party website? We found that a whopping 97% of applications had access to and were able to share this type of data. Worst of all, most of this data is sent off to third-party companies over HTTP. In our research, we found banking apps that integrated with social media, chat apps that sent chat logs to be analyzed for future purchasing trends, and many, many applications that track you via geo-location.

OWASP Mobile Top Ten Categories: M4—Unintended Data Leakage & M1—Weak Server Side Controls

Research Findings

HPE Research tested more than 2,000 mobile applications from 600+ companies.

- 97% of applications tested access at least one private information source of those applications.
- 86% of applications failed to use simple binary hardening protections against modern-day attacks.
- 75% of applications do not use proper encryption techniques when storing data on a mobile device.
- 18% of applications sent user names and passwords over HTTP; (of the remaining 82%) 18% implemented SSL/HTTPS incorrectly.
- 71% of vulnerabilities resided on the Web server.
- 86% of applications sent user names and passwords over HTTP; (of the remaining 85%) 18% implemented SSL/HTTPS incorrectly.
Lack of Binary Protection

Binary protections are a class of easy-to-implement security protections. In fact, most of them are simple checkboxes you just select before compiling your application and sending it to Apple. Binary protections add more security against modern day exploit and overflow attacks. They also do things like protect your application against attackers who might want to reverse engineer it for piracy. In other cases, they are simple best practices that reduce the information your application might normally share. Things like Position-Independent Executable (PIE), stack smashing protection, symbol stripping, code obfuscation, path disclosure, jailbreak detection, and so forth, are all lumped into this category. In this study, we found an alarming number of applications did not implement these easy-to-use security protections.

OWASP Mobile Top Ten Categories: M10—Lack of Binary Protections

Insecure Data Storage

Any stored data that is accessible from a mobile application can be a target if the data is stored without encryption. Such data includes passwords, personal information, session tokens, documents, chat logs, photos, and so forth. Most consumers think that data is protected by their pin number, but that is a false assumption. This 75 percent represents data that is accessible to anyone who has an unlocked powered-on phone in their possession. Unencrypted data that is seen and used for malicious purpose by an attacker can violate numerous policies in a corporation’s governance, as well as compromise the reputation of the enterprise if sensitive trade secrets are leaked to competitors, or the media. The simple fact is, losing your phone is equal to losing your high-valued data.

OWASP Mobile Top Ten Categories: M2—Insecure Data Storage & M4—Unintended Data Leakage

Transport Security

We found too many mobile applications transported things like passwords and registration data over HTTP. This is a very insecure process further compounded by the fact that these credentials are not just for the mobile application but are often also used by their Web application counterparts. Not only were the application’s credentials themselves at risk, but any social media sites they integrated with often used that social media sites’ HTTP endpoint rather than the HTTPS one. What does this mean? Anyone with a malicious mind on your same network (think coffee shop, work Wi-Fi, airport, or any server between you and a very far away website) can sniff your data. What’s even more interesting is that many of those who did use SSL/HTTPS managed to use it incorrectly, allowing certificate vulnerabilities that made it just as possible to perpetrate a man-in-the-middle attack like in the scenario above.

OWASP Mobile Top Ten Categories: M3—Insufficient Transport Layer Protection
Weak Server Side Controls

It is our earnest belief that the pace and cost of development in the mobile space have hampered security efforts. As a community (especially organizations like OWASP), we have been fighting Web-based vulnerabilities for the last 15 years. With the advent of mobile flexibility, we have lost sight that these mobile apps have Web back ends. We are forgetting these servers need security attention as well and as a result, we see the most critical flaws existing on these mobile sites/API’s/Web services. We also see a resurgence of a lack of knowledge when it comes to Web service or API security, which we think is correlated to the use of frameworks or development shops that have no security incentives.

OWASP Mobile Top Ten Categories: M1—Weak Server Side Controls

Conclusion

As these statistics show, mobile application security is still in its infancy. Just like with their Web counterparts, first comes the technology, and then comes the security. There are certain actions organizations can take immediately, though, to catch up to the pace of innovation.

• Scan your applications—This is important for a variety of reasons. First, it’s simple to implement. When you have no existing security program in place, this is where to start. Secondly, automated scanning casts a wide net, which is necessary when both simple and complex mobile application security mistakes are being made. Automated scanning solutions are capable of submitting far more tests in a far shorter time frame than humans could hope to do. Finally, automated scanning solutions have security expertise built in, something not always easy for organizations to acquire.

• Implement penetration testing—By design, security scanners look for as many vulnerabilities as possible. That does create false positives, or things that aren’t truly vulnerabilities. That’s simply the nature of scanners, and a far preferable behavior to False Negatives (missing an existing vulnerability entirely). Penetration testing helps separate the wheat from the chaff of automated scanning solutions. By manually reviewing results and examining “interesting” findings in an in-depth manner automated scanners can’t match, penetration testing provides verified findings and much more reliable results. As well, many low-level vulnerabilities are “stepping stones” to escalate methods of attacks. Penetration testers can quickly determine whether vulnerabilities found by scanners are really low level or need immediate attention.

• Adopt a Secure Coding Development Lifecycle (SDLC) approach—There are several approaches that try to “protect” around mobile application vulnerabilities such as Mobile Device Management (MDM), Mobile Application Management (MAM), Mobile Information Management (MIM), and so forth. However, the only true solution is to find and fix the problems in the code itself. These solutions can be a great extra layer of defense, sure, but absolutely shouldn’t be used in the stead of remediating the underlying issues. While implementing secure coding practices takes the longest to implement, this step ultimately bears the most fruit. Long story short, it’s exponentially less expensive to build security into the development process than adding it to mobile applications already in production. Ultimately, it’s about baking security in, not brushing it on.
Methodology

The analysis run in this study featured new technologies created by Fortify on Demand engineers to quickly assess the security posture of a mobile application. This automated Binary and Dynamic analysis engine is employed in Fortify on Demand’s Mobile. FoD Mobile is designed to give corporations a look into any mobile apps privacy and security flaws, while remaining low cost, requiring less time, and not requiring source code. Whether you have a new app you’ve developed, an app you contracted out, or an app you are thinking about allowing into your organization, FoD Mobile gives you the information to make security decisions quickly and easily.

The data polled for the study only used a subset of data gathered from FoD Mobile. Findings such as SQL injection, specifics of files using bad encryption, dangerous logging, several caching vulnerabilities, inter-app communication vulnerabilities, Unique Device Identifier (UDID) leakage, weak cryptography implementations, and so forth, were withheld as not to reveal specific egregious security flaws.

While the scope of the FoD Mobile technology that used these analyses is mostly client based, Fortify on Demand used its 2013 Premium Assessment data to run statistical analysis on mobile server side issues. Premium Mobile Assessments offer an in-depth look into the source code, running application, and Web server domains. This service level is akin to manual penetration testing and powered by some of the brightest mobile security minds in the industry.

Learn more at go.saas.hpe.com/fortifymobile