Mobile applications are uniquely exposed to hacking attacks as the application code must be released "out into the wild." Attackers can directly access, compromise, and exploit the binary code (e.g., analyze or reverse-engineer sensitive code, modify code to change application behavior, or inject malicious code).

Contents

Executive Summary....................................................................................................................................................................................................2
Technical Overview: What is the Threat?.........................................................................................................................................................3
The Business Case for Protecting Your Applications..................................................................................................................................4
The Solution: Application Hardening and Run Time Protection...........................................................................................................5
Key Characteristics of Arxan’s Application Protection...............................................................................................................................6
Build It Secure, Keep It Secure – Mobile Application Hardening and Run Time Protection in the SDLC.......................6
Executive Summary

We live in a mobile powered world, where nearly 7 billion mobile devices are expected to be in use by the end of 2014 and 108 billion mobile downloads are anticipated by 2017. Businesses that are most efficiently adapting to today’s “App Economy” are the most successful at deepening customer engagement and driving new revenues in this changing world.

However, where business opportunities abound, opportunities for hackers abound as well. Brand reputation, financial transactions, personal data and corporate intellectual property are all at serious risk since they pose lucrative and profitable targets for attackers. Hackers are increasingly targeting binary code to launch attacks on high-value mobile applications, across all platforms. To help you defend against such threats, this white paper will define the landscape of threat vectors in the rapidly-evolving world of binary code hacking, the consequences at stake, and remedies that have become available.

Unlike web applications, mobile software is uniquely exposed to binary risks, since by its nature, application code must be released "out into the wild." Attackers can directly access, compromise, and exploit binary code (e.g., analyze or reverse-engineer sensitive code, modify code to change application behavior, or inject malicious code).

Tackling sophisticated and often covert hacking attacks on code binaries requires a whole new layer of security, namely Application Hardening and Run-Time Protection, to protect exposed mobile binaries. Traditional application security practices—including safe-coding practices—are alone no longer sufficient to protect mobile assets from new binary vulnerabilities, since even “flawless code” can be reversed, modified, repackaged and distributed. The evolution toward applications that are rich in functionality and support customer service, or internal workflow processes must be balanced with heightened security—especially for high-value, sensitive transactions and client information.

Here’s a look at the two main risks that mobile applications face:

**Integrity Risk**
(Code Modification or Code Injection Vulnerabilities)
- Application binaries can be modified
- Run-time behavior of applications can be altered
- Malicious code can be injected into applications

**Confidentiality Risk**
(Reverse Engineering or Code Analysis Vulnerabilities)
- Sensitive information can be exposed
- Applications can be reverse-engineered back to source code
- Code can be lifted and reused or repackaged

As is the case with most new technologies, the development and deployment of mobile applications is moving at a faster pace than the security that’s being developed to protect them, and a majority of applications are being distributed without any form of binary protection. This provides susceptibility to malware, code modification or other forms of tampering that compromises application integrity.

In fact, 87% of the top 100 iOS and 97% of the top 100 Android applications have been turned into hacked variants, and at least 86% of mobile malware consists of legitimate applications unpacked, infected with malicious payloads, and then repackaged. Member-based data from OWASP (The Open Web Application Security Project) indicates that 86% of mobile applications that were tested lacked binary hardening, leaving them vulnerable and leading OWASP to identify “Lack of Binary Protections” as part of its Top Ten Mobile Risks announcement for 2014. Similar security advocacy is being echoed by respected analyst firms such as Gartner and Forrester on the need for binary protection via application hardening.
Technical Overview: What is the Threat?
In contrast to centralized web environments, mobile application reside on a distributed, fragmented, and unregulated mobile device ecosystem. Threats primarily come from two distinct types of attack agents:

1. **Malicious hackers** – Individuals or groups who seek to compromise the integrity and/or confidentiality of mobile applications. Hackers will typically target application binaries to access source code, steal or expose sensitive data or IP, or gain control of application functionality for malicious purposes (e.g., to modify or bypass sensitive code).

2. **Rogue application** – This is a malicious application, which shares the same execution environment and memory resources as a target application. The rogue application may be a seemingly innocuous application that the user is tricked into downloading and installing, or a genuine application that has since become infected and therefore compromised. The rogue application may contain malicious payloads that also target critical application integrity components.

These attack agents seek to compromise the application binary to gain access to source code or hijack the application and target critical code, such as security logic or key business routines for modification, analysis or theft.

Hackers are leveraging diverse and easily available (and often free) tools for their use in this endeavor. A summary of hacker tools is provided below:

---

As the business world moves ever-faster towards a mobile-first culture, the inherent complexities of such a shift require a similar change in security strategy. Companies must implement a layered mobile security framework based on enterprise risk. By doing so, they optimize their ability to harness mobile as a transformational competitive advantage. — AT&T

---

<table>
<thead>
<tr>
<th>Examples of Readily Available Hacker Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Application decryption/unpacking/conversion tools</strong> (Clutch, APKtool)</td>
</tr>
<tr>
<td><strong>Static binary analysis, disassembly, decompilation tools</strong> (IDA Pro, Hopper, JEB)</td>
</tr>
<tr>
<td><strong>Runtime binary analysis tools</strong> (GDB, ADB, LLDB)</td>
</tr>
<tr>
<td><strong>Runtime manipulation, code injection, method swizzling, patching, malware injection tools</strong> (Cydia Substrate, Dendroid, AndroRAT)</td>
</tr>
<tr>
<td><strong>Jailbreak detection evasion tools</strong> (xCon)</td>
</tr>
<tr>
<td><strong>Integrated weaponized toolsets</strong> (AppUse, iNalyzer)</td>
</tr>
</tbody>
</table>
A mobile application is exposed to many threats once it is deployed, as shown in the exhibit below.

### Mobile App Attack Vectors for Binary Hacking

**Reverse Engineering or Code Analysis**  
(Confidentiality)

- Disassembly / decompilation of native code (Obj C/C++)
- Application decryption
- Binary patching
- Method swizzling / function hooking within application
- Library and system service API hooking and swizzling

**String literal structure analysis**
**Symbol dumping and analysis**
**Static or dynamic key lifting**
**Application re-signing and re-packaging**

**Malware payload insertion**

**Code Modification or Code Injection**  
(Integrity)

The Business Case for Protecting Your Applications

When a mobile application is compromised, the consequences can be severe. Technical risks can easily translate into real business risks, and the implications of a successful attack on your mobile applications may include many of the risks shown in the exhibit below, to the right.

The impact to businesses’ brand image is being hotly debated, whenever we hear about another cyber attack. While retailers’ data breaches attract most of the publicity, because of the volume of records typically affected, nearly every type of business can be impacted by mobile application compromises. IBM’s *X-Force Threat Intelligence Quarterly Q1 2014* report found that 33% of attacks targeted Web applications in 2013. It is clear that enterprises need to not only deploy mobile security initiatives at the device platform level (such as authentication) and at the data (such as encryption), while also protecting the application layer. Base layer protection at the device and data level can be quickly and easily overturned by sophisticated attacks. Furthermore, application hardening encompasses run-time protection that endeavors to add tamper-resistance to applications and the data they process.

### Summary of Business Risks

- Brand Image Deterioration
- Damage to Customer Trust
- User Experience Damage
- Unauthorized Access and Fraud
- Identity Theft
- Privacy-Related Data Theft
- Confidential Data Theft
- Revenue Loss from Piracy
- Business Logic Bypass
- Repudiation
The Solution: Application Hardening and Run-Time Protection with Arxan

Arxan Application Protection for IBM Solutions provides mobile applications with binary protection at run-time and at rest with self-defense and tamper-resistance attributes, which are directly embedded within mobile applications themselves. For the mobile enterprise, this approach facilitates rapid and uniform deployment of comprehensive application security that is required across all the fragmented device platforms and form factors developers must tackle today.

Arxan’s patented Guard technology for Application Hardening and Run-Time Protection combines a triple Defend-Detect-React methodology, which not only protects against known types of threats but also detects if a critical asset is being compromised – regardless of the attack vector. Thus, it future-proofs applications such that they remain protected against new and evolving types of threats, as well as current known risks. Further, Guards can be programmed to react in appropriate ways when a compromise is detected, ranging from shut down to self-healing or issuing a remote alert. The protection itself is layered and networked, providing randomized and customized defense in depth for mobile applications.

“The threat of having our code reverse engineered or compromised by malware on mobile devices posed an unacceptable level of risk to our business and brand, and to our consumers. Mobile application protection is a key component of our mobile security strategy, ensuring our applications remain secure against known, unknown, and as yet undeveloped threat vectors.”

— Chief Security Officer
Major financial institution

Arxan’s Guarding System Enables Application Hardening and Run-Time Protection

<table>
<thead>
<tr>
<th>Defend</th>
<th>Detect</th>
<th>React</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Control Flow Obfuscation</td>
<td>• Run-time within the app</td>
<td>• Shutdown (Exit/Fail)</td>
</tr>
<tr>
<td>• Renaming</td>
<td>• Checksum</td>
<td>• Self-Repair</td>
</tr>
<tr>
<td>• Pre-Damage</td>
<td>• Debugger Detection</td>
<td>• Custom Reactions</td>
</tr>
<tr>
<td>• String Encryption</td>
<td>…within the package</td>
<td>• Alert/Phone Home</td>
</tr>
<tr>
<td>• Meta Data Removal</td>
<td>• Resource Verification</td>
<td></td>
</tr>
<tr>
<td>• Call Hiding</td>
<td>…between apps</td>
<td></td>
</tr>
<tr>
<td>• Resource Encryption</td>
<td>• Hook Detection</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Swizzle Detection</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Environment</td>
<td></td>
</tr>
<tr>
<td></td>
<td>• Jailbreak/Root Detection</td>
<td></td>
</tr>
</tbody>
</table>

Protected App

• Self-Defending
• Tamper Resistant
• Hardened against hacking attacks and malware exploits
Key Characteristics of Arxan Application Protection for IBM Solutions

✔ "Gold Standard" protection strength:
  • Multi-layer interconnected Guard Network for defense-in-depth, with no single point of failure
  • Breadth of static & run-time Guard types to stop diverse attack vectors
  • No binary signatures or agents
  • Risk-based, customized protection design
  • Automated variability and binary randomization or each build

✔ No source code involvement, due to unique binary-based guard injection engine; no disruption to your SDLC

✔ Broader multi-platform support: Enables standardization across applications, platforms, and languages

✔ Battle-tested credibility: Protected applications deployed on more than 300 million devices

✔ Unique Intellectual Property ownership on application protection technology with 10+ patents

✔ Integrates with companion IBM Security and mobility solutions

Build It Secure, Keep It Secure – Mobile Application Hardening and Run-Time Protection in the SDLC

The first line of protection comes from traditional practices, such as following secure-coding practices with IBM MobileFirst Platform Foundation (formerly IBM Worklight) or third-party development platforms, and code scanning with IBM MobileFirst Platform Application Scanning (formerly IBM AppScan) for known vulnerabilities or flaws. Once the code is deemed to be "clean", the final line of defense is achieved by deploying binary-level Application Hardening and Run-Time Protection to mitigate application risks from binary-based vulnerabilities, or new application risks that evolve after you deploy.
Arxan Application Protection for IBM Solutions makes it possible for mobile application developers to easily incorporate a best-practice based development process for mobile application security. Self-defense and incorporation of tamper-resistance into mobile applications is made possible during the build process, without requiring source-code changes to harden applications.

And, with AppScan, new application integrity rules can be used to identify binary-based vulnerabilities and potential points of attack, so that they can be hardened with Arxan against exploits prior to deployment. There are new customized rules available for AppScan Source that help inform security teams and developers about binary exposures, including risks such as swizzling for behavior change, security control bypass, cryptographic key replacement, exposed methods, exposed data symbols, key interception, repackaging, application decryption, and presentation layer modification.

Applications need to have the ability to defend themselves in the wild. Traditional application security practices with safe coding and vulnerability testing are still very important but won’t suffice for these new binary-based risks/threats.

Arxan Application Protection for IBM Solutions achieves hardening and run-time protection, with minimal performance impact and footprint overhead, both of which are at a premium in mobile security applications. Further, given that most organizations typically don’t have the budget or the time to fix every flaw they find —their time and resource allocations can be devoted to the most severe vulnerabilities that are found. By deploying binary protection that delivers self-defense and tamper resistance, you can rest assured that your brand, your customers, your release schedule and your competitive differentiation are all protected.

For more about Arxan Application Protection for IBM Solutions, contact IBM to learn more about the Arxan solution for your enterprise, or request a complimentary Arxan evaluation or proof of concept deployment. You can also visit us at: www.ibm.com/applicationsecurity.

---

1 International Telecommunications Union – January 2013:
http://www.siliconindia.com/magazine_articles/World_to_have_more_cell_phone_accounts_than_people_by_2014-DASD767476836.html


3 Arxan’s State of Security in The App Economy, 2014 study finds that 97% of the top 100 paid Android and 87% of the top 100 paid iOS applications are available as hacked versions on third party sites – many of these posing as legitimate variants of the original application (such as a local language translation). Even for Apple iOS, over 75% of Top 100 iOS applications were found hacked. Another recent study (Alcatel-Lucent Kindsight Security Labs Malware Report, Q4 2013) found that “it is trivial for an attacker to hijack a legitimate Android application, inject malware into it and redistribute it for consumption. There are now binder kits available that will allow an attacker to automatically inject malware into an existing application”. A study by North Carolina State University, published with IEEE Security & Privacy, found that at least 86 percent of mobile malware consists of legitimate applications unpacked, infected with malicious payloads, and then repackaged. After repackaging, mobile apps can also be re-distributed via application storefronts. The rate of mobile malware is also skyrocketing, as 92% of malware is being created for Android, and Android is reported by Juniper Research to have 80% of the smartphone market share in Q3’13.