Abstract

Any type of software, from desktop to mobile applications, is prone to contain defects that can lead to vulnerabilities. These vulnerabilities, when exploited, may put in risk the integrity, confidentiality and availability of the software. Security auditing methodologies help to reduce at some level of confidence these risks. With the explosion of mobile applications for daily activities like checking email, news, social networks, or even managing bank accounts, guaranteeing an acceptable level of application security becomes critical for the usage and trust of mobile services. In this paper, we review and classify OWASP 2014 Top Ten Mobile risks in analysis blocks. Based on the blocks classification, we propose a methodology to security audit mobile software applications. We demonstrate the effectiveness of the proposed methodology by auditing the same mobile application in Google’s Android and Apple’s iOS platforms surfacing multiple vulnerabilities.

Analysis Blocks to Identify Mobile Risks


- (M1) Weak Server Side Controls
- (M2) Insufficient Data Storage
- (M3) Insecure Transport Layer Protection
- (M4) Unauthorized Data Leakage
- (M5) Poor Authorization and Authentication
- (M6) Improper Session Handling
- (M7) Security Decisions via Untrusted Inputs
- (M8) Improper Data Storage
- (M9) Lack of Binary Protection
- (M10) Broken Cryptography

Analysis blocks proposed

- Environment Analysis
  - Firmware, developer, backend server, etc.
- Connections
  - GRPS, Wi-Fi, IRDA, Bluetooth, or NFC
- Sensitive Data
  - UDID, MAC, IMEI, etc.
- Application Own Data
  - XML, PList, SQLite, etc.
- Application Structure
  - Design, implementation

Environment Analysis

<table>
<thead>
<tr>
<th></th>
<th>M1</th>
<th>M2</th>
<th>M3</th>
<th>M4</th>
<th>M5</th>
<th>M6</th>
<th>M7</th>
<th>M8</th>
<th>M9</th>
<th>M10</th>
</tr>
</thead>
<tbody>
<tr>
<td>OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>OK</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Case Study: eCommerce app (Android and iOS platforms)

- On Pre-Runtime Phase
  - CVE-2004-0230 vulnerability in server side [8]
  - Weak ciphers in server side (RC4-MD5 and RC4-SHA)
  - Login passwords stored in MD5
  - Login username (i.e., email) in plain text (SQLite3 file)
  - Excess of permissions in both platforms
- On Runtime Phase
  - Login requests (plain-text username, MD5 password) sent via HTTP
  - Market retargeting sending relevant user data
  - Purchase requests sent via HTTPS but without certificate pinning
    - Card number, issuer, holder, expiration, and CVV code
  - On Post-Runtime Phase
    - Files contain login credentials in plain text
      - Android: Email in plain text, password in MD5
      - iOS: Email and password in plain text
    - Cookies stored in plain text

Methodology

Pre-Runtime

- Preliminary Analysis
- NO

Runtime

- Static Analysis
- NO
- Dynamic Analysis
- YES

Post-Runtime

- Forensic Analysis
- YES

Conclusions

- Complete security auditing methodology developed
- Five analysis blocks defined over OWASP 2014 Top Ten Mobile Risks
- Allows to find vulnerabilities and to detect suspicious behaviours
- Validated through a real case study, finding several vulnerabilities
  - Spanish outlet app

References


Contact Information

- Corresponding author: alvaro.botas@unileon.es