Cracking the Code of Mobile Application

- Sreenarayan A
Paladion Mobile Security Team
Take Away for the day

• Why Mobile Security?
• Purpose of Decompiling Mobile Applications?!
• Methodology of Decompilation
• Live Demo’s:
  – Windows Phone App
  – Android App
  – iOS (iPhone / iPad App)
  – Blackberry Apps / Nokia App [Jar Files]
  – Blackberry Apps [COD Files]
Why is security relevant for Mobile Platform?

• 400% Increase in the number for Organizations Developing Mobile Platform based applications.
• 300% Increase in the no of Mobile Banking Applications.
• 500% Increase in the number of people using the Mobile Phones for their day to day transactions.
• 82% Chances of end users not using their Mobile Phones with proper caution.
• 79% Chances of Mobile Phone users Jail Breaking their Phones.
• 65% Chances of Mobile Phone users not installing Anti-virus on their Mobile Phones.

• **71% Chances of any application to get misused.**
• 57% Chances of a user losing his sensitive credentials to a hacker.
Market Statistics of Mobile Users

**MARKET SHARE OF SMARTPHONE SUBSCRIBERS BY PLATFORM**

- **46.9%** Google
- **28.7%** Apple
- **16.6%** RIM
- **5.2%** Microsoft
- **1.5%** Symbian

**KEY DATA COMMUNICATIONS INTERCEPTION FINDINGS**

- Wi-Fi hotspots expected to grow 350 percent by 2015
- Widely available tools make it simple to hijack users’ credentials from Wi-Fi networks

**Table 2**

<table>
<thead>
<tr>
<th>Operating System</th>
<th>1Q12 Units</th>
<th>1Q12 Market Share (%)</th>
<th>1Q11 Units</th>
<th>1Q11 Market Share (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Android</td>
<td>81,067.4</td>
<td>56.1</td>
<td>36,350.1</td>
<td>36.4</td>
</tr>
<tr>
<td>iOS</td>
<td>33,120.5</td>
<td>22.9</td>
<td>16,883.2</td>
<td>16.9</td>
</tr>
<tr>
<td>Symbian</td>
<td>12,466.9</td>
<td>8.6</td>
<td>27,598.5</td>
<td>27.7</td>
</tr>
<tr>
<td>Research In Motion</td>
<td>9,939.3</td>
<td>6.9</td>
<td>13,004.0</td>
<td>13.0</td>
</tr>
<tr>
<td>Bada</td>
<td>3,842.2</td>
<td>2.7</td>
<td>1,862.2</td>
<td>1.9</td>
</tr>
<tr>
<td>Microsoft</td>
<td>2,712.5</td>
<td>1.9</td>
<td>2,582.1</td>
<td>2.6</td>
</tr>
<tr>
<td>Others</td>
<td>1,242.9</td>
<td>0.9</td>
<td>1,495.0</td>
<td>1.5</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>144,391.7</strong></td>
<td><strong>100.099,775.0</strong></td>
<td><strong>100.0</strong></td>
<td><strong>100.0</strong></td>
</tr>
</tbody>
</table>

Source: Gartner (May 2012)
Mobile Market Trends

Figure 1: Category of apps used in the past 30 days

- Games: 61% (Smartphone), 52% (Feature phone)
- Weather: 55% (Smartphone), 39% (Feature phone)
- Maps/Navigation/Search: 50% (Smartphone), 30% (Feature phone)
- Social Networking: 49% (Smartphone), 32% (Feature phone)
- Music: 42% (Smartphone), 36% (Feature phone)
- News: 36% (Smartphone), 36% (Feature phone)
- Entertainment: 33% (Smartphone), 22% (Feature phone)
- Banking/Finance: 28% (Smartphone), 15% (Feature phone)
- Dining/Restaurant: 25% (Smartphone), 14% (Feature phone)
- Sports: 24% (Smartphone), 15% (Feature phone)
- Productivity: 22% (Smartphone), 10% (Feature phone)
- Shopping/Retail: 21% (Smartphone), 11% (Feature phone)
- Video/Movies: 21% (Smartphone), 16% (Feature phone)
- Travel: 20% (Smartphone), 13% (Feature phone)
- Food/Drink: 19% (Smartphone), 11% (Feature phone)
- Communication (e.g., IM, VoIP): 18% (Smartphone), 11% (Feature phone)
- Lifestyle/Health: 14% (Smartphone), 10% (Feature phone)
- Household/Personal Care: 5% (Smartphone), 7% (Feature phone)

Source: The Nielsen Company
Different Types of Mobile Applications

• Mobile Browser based Mobile Applications
• Native Mobile Applications
• Hybrid Mobile Applications
Different Types of Mobile Applications
Different Types of Mobile Architecture

- Browser App
- Hybrid App
Why did we learn the above types??

• Which applications can be Decompiled?
  – Browser based Mobile Applications ?
  – Native Mobile Applications ?
  – Hybrid Mobile Applications ?

• We have to get to know of the basics!
Cracking the Mobile Application Code
Cracking the Mobile Application Code

• What do you mean by Decompileation? -> What is Compilation?
• What do you mean by Reverse Engineering?

Questions to be answered ahead:
• What are the goals/purpose of Cracking the code?
• What is the methodology of Decompileation?
• What the tools which can be used to Decompile?

• Can Decompileation be done on all platforms?
  1. WINDOWS PHONE / WINDOWS MOBILE?
  2. ANDROID?
  3. iPHONE / iPAD?
  4. BLACKBERRY?
  5. NOKIA?
Goal of Cracking the Mobile Application Code
Goals of Cracking the Source Code

• “UNDERSTAND THE WORKING OF THE APPLICATION AND TO FIGURE OUT THE LOOPHOLES!”

• To find Treasure Key Words like: password, keys, sql, algo, AES, DES, Base64, etc

• Figure out the Algorithms Used and their keys.

• By-passing the client side checks by rebuilding the app.

• E.g. Password in Banking Application (Sensitive Information)
  • E.g. Angry Birds Malware (Stealing Data)
  • E.g. Zitmo Malware (Sending SMS)

• We have understood the goals, how to achieve them? Methodology.
Methodology of Cracking
Methodology / Study

Step 1: Gaining access to the executable (.apk / .xap / .jar / .cod / .jad ..)

Step 2: Understanding the Technology used to code the application.

Step 3: Finding out ways to derive the Object Code from the Executable.

Step 4: Figuring out a way to derive the Class Files from the Object Code.

Step 5: Figuring out a way to derive the Function Definitions from the Object Code
JUMP TO DEMO’s

Lets us understand the methodology in all platforms..
Demo - Reverse Engineer the Windows Phone Application

**Tools used:**
- De-compressor (Winrar / Winzip / 7zip)
- .Net Decompiler (ILSpy)
- Visual Studio / Notepad

**Steps**
1. .xap -> .dll
2. .dll -> .csproject

**Demo**

**Mitigation**
2. Dotfuscator (program flow): [Link](http://example.com)
Demo - Reverse Engineer the Android Application

• **Tools** used:
  - De-compressor (Winrar / Winzip / 7zip)
  - Dex2jar Tool (Command Line)
  - Java Decompiler / Jar decompiler (JD-GUI, etc)

• **Steps**
  1. .apk -> .dex
  2. .dex -> .jar
  3. .jar -> .java

• Demo

• **Mitigation**
Demo - Reverse Engineer the Blackberry Application

• **Tools** used:
  - JD – GUI (Java Decompiler)
  - Notepad

• There are two types of Application files found in Blackberry:
  1. .Jar (.jad -> .jar)
  2. .Cod (.jad -> .cod (Blackberry Code Files))

• **Steps**
  1. .jar -> .java (JD-GUI) -> Notepad
     Or
  1. .cod -> codec Tool -> Notepad

• **Demo**

• **Mitigation**
Demo - Reverse Engineer the iOS Application

**Tools** used:
- iExplorer
- Windows Explorer
- oTool
- Class-dump-z

**Steps**
1. .app -> Garbage (Object Code) (DVM)
2. Object Code -> Class definitions

**Demo**

**Limitations:** Apple changes the IDE every release leading to challenges.

**Mitigation**
Palisade Articles

• iOS vs Android Testing
• Mobile Data Encryption
• Mobile Application Security Testing
• Demystifying the Android Malware
• And …

• Website link: palizine.plynt.com
• Questions and Answers
• Quiz
• Feedback
Thank You

Sreenarayan.a@paladion.net

Twitter: Ace_Sree