Introduction

- Traditional security mechanisms have a hard time against sophisticated threats.
- Presentation and analysis of five (5) Advanced Persistent Threats, (APT).
- Work highlights characteristics and identifies common patterns and techniques.
- Issues that enabled the malware to evade detection from security solutions.
- Proposition of technical countermeasures for strengthening defenses against similar threats.

Stuxnet

- Targeted the Iranian Nuclear Program.
- Interfered with Industrial Control Systems (ICS) on Windows systems.
- “Fire and forget weapon”.
- Infected removable drives. Zero-day vulnerability (MS10-046).
- Modular malware, using compromised certificates to sign components.
- Rootkit code to hide its binaries on windows systems.
- XOR encryption with a static key to decrypt payload.
- Rootkit module to hide its files - Payload injection to evade detection.

MiniDuke

- Targeted government bodies in 23 countries, mainly in Europe.
- Combined exploitation of Adobe’s PDF sandbox and assembly code for its payload.
- Sophisticated, layered communication: Twitter accounts with encrypted URL’s.
- Payload obfuscated as GIF images.
- Evasion techniques using process detection and idle state.

Duqu

- Developed for espionage: Key logging.
- Modular malware, using compromising certificates to sign components.
- Infection and propagation using Microsoft Word files.
- Rootkit module to hide its files - Payload injection to evade detection.
- AES-CBC and XOR encryption.
- Strong connections to Stuxnet development.

Flame

- Uncommon size: 20 megabytes, including all modules.
- Widespread Information stealing malware.
- USB infection but did not replicate on its own – Used two zero-day vulnerabilities.
- Complex cryptanalytic attack against Microsoft’s Terminal Services.
- Rootkit functionality - .ocx binaries to avoid detection.

Red October

- Information gathering, targeting diplomatic, governmental and scientific agencies.
- Minimalistic architecture allowed it to remain undetected: One component downloaded modules (at least 1000 different modules have been identified).
- Targeted emails containing malicious Word and Excel documents.
- Plugin for Office and Adobe reader.

<table>
<thead>
<tr>
<th>COMMON CHARACTERISTICS</th>
<th>SUGGESTED COUNTERMEASURES</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Targeted system and architecture.</td>
<td>• Patch Management: Stop further exploitation.</td>
</tr>
<tr>
<td>• Initial attack vectors.</td>
<td>• Network Segregation: Workstation isolation blocks malware.</td>
</tr>
<tr>
<td>• Escalation of privileges.</td>
<td>• Relaxed Whitelisting: Block malicious connection attempts.</td>
</tr>
<tr>
<td>• Network Access.</td>
<td>• Dynamic content execution: Filter dynamic content in traffic.</td>
</tr>
<tr>
<td>• Network IDS and endpoint antivirus products.</td>
<td></td>
</tr>
<tr>
<td>• Use of Encryption / Obfuscation.</td>
<td></td>
</tr>
<tr>
<td>• Exploitation of digital signatures.</td>
<td></td>
</tr>
</tbody>
</table>

- We provide:
  - Technical comparison of malware, focusing on behavioral/dynamic analysis of the samples in a controlled environment, as well as on published technical reports.
  - Identification of common attack patterns in samples to identify why current security solutions failed.
  - A proposition of effective countermeasures for strengthening our defenses against similar threats.

<table>
<thead>
<tr>
<th>Threats</th>
<th>Stuxnet</th>
<th>Duqu</th>
<th>Flame</th>
<th>Red October</th>
<th>MiniDuke</th>
<th>Evasion</th>
</tr>
</thead>
<tbody>
<tr>
<td>PE Type</td>
<td>DLL</td>
<td>OCX</td>
<td>EXE</td>
<td>EXE</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Initial infection</td>
<td>Unknown</td>
<td>MS Word</td>
<td>Unknown</td>
<td>MS Excel / Word, Java</td>
<td>PDF</td>
<td></td>
</tr>
<tr>
<td>Replication</td>
<td>Removable drives, network</td>
<td>Manual replication only</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Rootkit module</td>
<td>Yes</td>
<td>No</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Key logging</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Evasion</td>
<td>Yes</td>
<td>No</td>
<td>Yes</td>
<td>No</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Encryption</td>
<td>XOR</td>
<td>XOR, AES-CBC</td>
<td>XOR, Substitution, RC4</td>
<td>XOR Unique per victim, XOR, ROL</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Target</td>
<td>Sabotage</td>
<td>Information gathering</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Table 1: Advanced Persistent Threats Comparison

Conclusions

- Widely accepted best countermeasures would have reduced the impact of APT malware.
- Even in critical infrastructures, a small subset of protection mechanisms was enforced.
- Need to shift our focus on more robust and transparent solutions, since traditional security solutions, have failed repeatedly.
- Limiting and controlling the software that is allowed to be installed and executed on a system, would significantly reduce the impact of APT attacks.
- Combines with the research carried out by our Laboratory, in particular the one focused on security-critical information systems (e.g. health information systems).

References