Malware analysis techniques

♣ Dynamic Analysis and Static Analysis

- Static Analysis is harder and takes more time than Dynamic Analysis
- Static Analysis is more exact than Dynamic Analysis
- Performing static analysis when dynamic analysis is not sufficient

<table>
<thead>
<tr>
<th></th>
<th>Dynamic Analysis</th>
<th>Static Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Method</td>
<td>Execute and trace actions</td>
<td>Read assembly codes</td>
</tr>
<tr>
<td>Accuracy</td>
<td>Not accurate</td>
<td>(Theoretically) perfect</td>
</tr>
<tr>
<td>Time</td>
<td>several minutes (or several hours)</td>
<td>More than 1 week</td>
</tr>
<tr>
<td>Performed by</td>
<td>Automatically</td>
<td>Manually</td>
</tr>
</tbody>
</table>

- If necessary, a researcher must do static analysis with too much time.
Our proposal

♣ Extracting the difference of two malware programs
  ● To reduce the cost of Static Analysis
  ● We use the software named "BinDiff" which is developed by H. Flake

♣ BinDiff
  ● A program is divided into some functions (This is called Call Graph)
  ● A function is divided into some basic blocks (This is called Control Flow Graph)
  ● Compare Call Graph and Control Flow Graph between two malware programs
Demonstration of BinDiff

Difference between two control flow graphs

1. Perfectly the same (Green)
2. Partly the same (Yellow)
3. Exists in only No.1 (Red)
4. Exists in only No.2 (Blue)
Latest threats of malware

♣ Massive malware programs have been discovered
  • McAfee has detected over 80 million malware so far. (i.e. one per 1.5 sec)

![Graph showing total malware samples in the database]

McAfee Threats Report: First Quarter 2012

• The techniques of generating malware variants.
  (e.g. metamorphic, polymorphic, frequency maintenance)
Recent tendency of malware damage

♣ Enhanced technique of infection
  ● Our social infrastructure systems are under targeted-attack

♣ The incident of Mitsubishi Heavy Industries (MHI)
  ● In September 2011, 83 servers and PCs have been infected
  ● They were infected by famous malware such as Gumblar and SpyEye
  ● Forensic experts have researched and concluded that there was no leaks of important data

---

MHI News Release, Nov 18, 2011

---

Bulletin Board Notice re Current Status of Investigation on Virus Infections (4)

MHI has investigated possible data leakage from the computers and servers that were suspected of being infected by a new type of virus. MHI has completed a thorough investigation into the matter involving nuclear power-related data, and has concluded that the incident led to no leaks of nuclear power-related data requiring protection.

The company will continue to investigate into the incident relating to its other computer systems, and communications under way by the police authorities.

Considering the importance of its products and technologies, the company has decided to ensure a high level of information security. Taking special notice of the latest incident, the company will pursue further strengthening of information security measures.

November 18, 2011
Mitsubishi Heavy Industries, Ltd.
Proposed architecture

① Obtaining malware sample and extracting assembly codes
② Searching malware which is similar to malware sample
③ Extracting difference between two malware programs. This difference is the part to analyze
④ Static Analysis (in manually)
⑤ Feedback
Evaluation

♣ Evaluation target

Infected PC → Unpack → Disassemble → Extract difference (BinDiff) → Search DB for similar malware → Static Analysis

Feedback

♣ Evaluation measure

- How many functions can we remove?
- The aim of this architecture is to help with the manual analysis.
Result 1

♣ “SpyEye” malware
   ● which leaks passwords and credit card information

<table>
<thead>
<tr>
<th>Sample</th>
<th>Number of Function</th>
<th>MD5 Hash</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample1</td>
<td>523</td>
<td>9D2A48BE1A553984A4FDA1A88ED4F8EE</td>
</tr>
<tr>
<td>Sample2</td>
<td>139</td>
<td>D64CA15261C53279A7288616B3CB1A92</td>
</tr>
</tbody>
</table>

● Compare two malware programs and extract the difference

Sample1

Sample2

Extracting Difference (BinDiff)

● Result of comparison

<table>
<thead>
<tr>
<th>Function</th>
<th>Number of function</th>
</tr>
</thead>
<tbody>
<tr>
<td>Common functions in sample1 and 2</td>
<td>53</td>
</tr>
<tr>
<td>Only sample1</td>
<td>470</td>
</tr>
<tr>
<td>Only sample2</td>
<td>58</td>
</tr>
</tbody>
</table>

Static analysis of this function can be done efficiently
Several “SpyEye” malware

- Add malware sample 3 and 4 to database

<table>
<thead>
<tr>
<th>Sample</th>
<th>Number of function</th>
<th>MD5 Hash</th>
<th>Role</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sample1</td>
<td>523</td>
<td>9D2A48BE1A553984A4FDA1A88ED4F8EE</td>
<td>analysis target</td>
</tr>
<tr>
<td>Sample2</td>
<td>139</td>
<td>D64CA15261C53279A7288616B3CB1A92</td>
<td>in the database</td>
</tr>
<tr>
<td>Sample3</td>
<td>609</td>
<td>DF04C2CD2B5F7E471CB0435FDB9B3014</td>
<td>in the database</td>
</tr>
<tr>
<td>Sample4</td>
<td>218</td>
<td>42DACFBE2E5AF0C43D17356CA76F0271</td>
<td>in the database</td>
</tr>
</tbody>
</table>
Result 2

♣ Result of comparison

<table>
<thead>
<tr>
<th>Common functions in sample3</th>
<th>Sample2</th>
<th>Sample3</th>
<th>Sample4</th>
<th>Sample4, 5, 6</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>53</td>
<td>78</td>
<td>85</td>
<td>135</td>
</tr>
</tbody>
</table>

\[(\frac{135}{523}=25.8\%\)\]

- Using multiple malware programs, number of common functions are improved
In summary

- We proposed new architecture which makes static analysis more efficient
- One of the key components in this system is a similarity analysis function which compares disassembly code of the target malware with already known malware in the database
- We think cloud system is useful to construct the malware database to share the analysis result all over the world