Forensic Memory Analysis - Files Mapped In Memory

By

Ruud van Baar, Wouter Alink, Alex van Ballegooij

Presented At

The Digital Forensic Research Conference

DFRWS 2008 USA  Baltimore, MD (Aug 11th - 13th)

DFRWS is dedicated to the sharing of knowledge and ideas about digital forensics research. Ever since it organized the first open workshop devoted to digital forensics in 2001, DFRWS continues to bring academics and practitioners together in an informal environment. As a non-profit, volunteer organization, DFRWS sponsors technical working groups, annual conferences and challenges to help drive the direction of research and development.

http://dfrws.org
Forensic Memory Analysis: Files mapped in memory

Ruud van Baar
Wouter Alink
Alex van Ballegooij
Netherlands Forensic Institute

August 12, 2008
DFRWS 2008 Baltimore
Introduction

• Why analyse memory dumps?
• Files mapped in memory
• Implementation
• Results
• Demonstration
Why analyse memory?

• Interesting data [Walters 2007]
  • Processes
  • Network information
  • Passwords
  • Cryptography keys
  • …

• Can provide information about recent activities on a system
• Mapped files…
Files mapped in memory

- Link files to a user or process
- Point out recently used data
- Identify data in a memory dump to reduce search space for other information, e.g. passwords
Fragmentation

Problem:
High degree of fragmentation of files in memory
Approach

- Recognize pool structures used by the Windows Memory manager
  - Process structures [Schuster 2006]
  - File structures [Dolan-Gavitt 2007]
- Link these structures if possible
- Carve for unidentified structures
Different methods of identification

- Find process structures
- Identify private and shared files
- Carve for specific file-mapping structures
  - Control areas (MmCa)
  - Page tables (MmSt)
File information

• Original path in file system
• Not always available; Backup:
  • Use known mapping to eliminate fragmentation
  • Header/footer information
Implementation

• Proof of concept
• Currently only Windows XP SP2
• File size not always known
• Needs more testing
  • Different configurations (PAE, 64 bit)
  • Larger memory dumps
# Results by extension

<table>
<thead>
<tr>
<th></th>
<th>DFRWS 2005 -1</th>
<th>DFRWS 2005 -2</th>
<th>Windows XP</th>
</tr>
</thead>
<tbody>
<tr>
<td>.dll</td>
<td>202</td>
<td>.dll</td>
<td>.dll</td>
</tr>
<tr>
<td></td>
<td></td>
<td>211</td>
<td>298</td>
</tr>
<tr>
<td>.pnf</td>
<td>82</td>
<td>.exe</td>
<td>.gif</td>
</tr>
<tr>
<td></td>
<td></td>
<td>28</td>
<td>86</td>
</tr>
<tr>
<td>.exe</td>
<td>29</td>
<td>.mmf</td>
<td>.ttf</td>
</tr>
<tr>
<td></td>
<td></td>
<td>11</td>
<td>52</td>
</tr>
<tr>
<td>.mmf</td>
<td>11</td>
<td>.ttf</td>
<td>.ini</td>
</tr>
<tr>
<td></td>
<td></td>
<td>8</td>
<td>31</td>
</tr>
<tr>
<td>.png</td>
<td>11</td>
<td>.log</td>
<td>.jpg</td>
</tr>
<tr>
<td></td>
<td></td>
<td>5</td>
<td>30</td>
</tr>
</tbody>
</table>
Results by method

Number of identified blocks

Method

vadwalk  objecttables  mmca/mmci  mmst
Conclusion

• Possible to identify 25% of pages as part of mapped files
• 40% of these pages can be linked to the relating process or processes
• Less informative methods result in more pages identified
Demonstration

- Test memory dump of a VMWare system running Windows XP SP2 with 256MB of RAM.
- VirtualBLOB [XIRAF 2006]
Demonstration
Contact

Ruud van Baar  ruud@holmes.nl
Wouter Alink
Alex van Ballegooij
Netherlands Forensic Institute