Malware Analysis as a Hobby

Michael Boman - Security Consultant/Researcher, Father of 5
Why the strange hobby?
1. Start virtual environment  
2. Copy sample  
3. Start logging facilities  
4. Execute sample  
5. Stop logging facilities  
6. Analyze logs
Drawbacks

• Time consuming
• Boring in the long run (not all malware are created equal)
Choose any two....
Choose any two?
Why not all of them?

I can do it cheaply (hardware and license cost-wise). Human time not included.

I can do it quickly (I spend up to 3 hours a day doing this, at average even less).

I get pretty good results (quality). Where the system lacks I can compensate for its shortcomings.
Automate everything!

Automate

Engineer yourself out of the workflow
Birth of the MART Project

Malware Analyst Research Toolkit
• Public & Private Collections
• Exchange with other malware analysts
• Finding and collecting malware yourself
  • Download files from the web
  • Grab attachments from email
  • Feed BrowserSpider with links from your SPAM-folder
BrowserSpider

- Written in Python
- Using the Selenium framework to control REAL browsers
  - Flash, PDFs, Java applets etc. executes as per normal
  - All the browser bugs exists for real
- Spiders and follows all links seen
Sample Analysis

- Cuckoo Sandbox
- VirusTotal
A day's work for a Cuckoo

1. Fetch a task
2. Prepare the analysis
3. Launch analyzer in virtual machine
4. Execute an analysis package
5. Complete the analysis
6. Store the result
7. Process and create reports
DEMO: Submit sample for analysis
New Analysis  use this form to add a new analysis task

File to upload  Choose File  No file chosen

Package to use

Options

Timeout

Priority  Low

Submit  Cancel
Sample Reporting

Results are stored in MongoDB (optional, highly recommended)
Accessed using a analyst GUI
# File Details

<table>
<thead>
<tr>
<th>File name</th>
<th>MART-app.exe</th>
</tr>
</thead>
<tbody>
<tr>
<td>File size</td>
<td>21504 bytes</td>
</tr>
<tr>
<td>File type</td>
<td>PE32 executable (console) Intel 80386, for MS Windows</td>
</tr>
<tr>
<td>CRC32</td>
<td>561F1BFA</td>
</tr>
<tr>
<td>MD5</td>
<td>18b2708009f0efb6b12a39876bb4f87a</td>
</tr>
<tr>
<td>SHA1</td>
<td>149ce9c781d9b1049d5a2e7f321c0f34c7e9c7b</td>
</tr>
<tr>
<td>SHA256</td>
<td>dc9cde3ec7dddb2eef1e9bfe61e6891de945cc42d2a9ec8bb2f6f1380c7ff645ddd</td>
</tr>
<tr>
<td>SHA512</td>
<td>07d4ef4575c10d371053ea49e37f9705bbaf4dd1e0dafd571e6778f155e3de4e29d26d771aeede6be57b9fd790a044f17ef6e23abe20bdc58bf6c330e990cc6</td>
</tr>
<tr>
<td>Sdeep</td>
<td>None</td>
</tr>
</tbody>
</table>

**PEID Signatures:**
- Pelles C 3.00, 4.00, 4.50 EXE (X86 CRT-LIB)

**Yara Signatures:**
- None matched

**Antivirus Results:**
- File not found on VirusTotal
Signatures matched cuckoo signatures

Creates a empty file

Screenshots pictures of the desktop during execution

Static Analysis binary details

Sections
Imports

Dropped Files files created or deleted by the malware

ntfs.txt
text.txt

Network Analysis network activity performed during analysis

Hosts Involved
DNS Requests
HTTP Requests

Behavior Analysis details on the malware execution
Behavior Analysis  details on the malware execution

Summary

Files
- text.txt
- ntfs.txt:ntfs
- ntfs.txt

Mutexes
Nothing to display.

Registry Keys
Nothing to display.

Processes
MART-app.exe  PID: 3824, Parent PID: 3804
Data Mining
Where Virtual Machine analysis fails

And what to do about it
Problems

• Cuckoo is easily bypassed
• User-detection
• Sleeping malware
Problems

- VM or Sandbox detection
- The guest OS might not be sufficient enough
- Any multistage attack
Iterating automatiation

Sort out clearly non-malicious and obviously malicious samples

Devide the samples into categories

Do brief static analysis

<table>
<thead>
<tr>
<th>Known Good</th>
<th>Known Bad</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unknown</td>
<td></td>
</tr>
</tbody>
</table>
Iterating automation

- Sort out clearly non-malicious and obviously malicious samples
- Devide the samples into categories
- Do brief static analysis

- Does not do anything
- Detects environment
- Encrypted segments
- Failed execution
Iterating automation

- Sort out clearly non-malicious and obviously malicious samples
- Divide the samples into categories
- Do brief static analysis

- Run longer
- Environment customization
SHUT UP AND
TAKE MY MONEY!
Budget

- Computer: €520
- MSDN License: €800 (€590 renewal)
- Year 1: €1320
- Year N: €590
- Money saved from stopped smoking (yearly): €2040
Malware Lab
MART Hardware (overview)
MART Hardware (mounts)
MART Hardware (HDD)

Minimum Read Rate: 72.3 Mb/s
Maximum Read Rate: 144.8 Mb/s
Average Read Rate: 116.3 Mb/s
Average Write Rate: –
Minimum Write Rate: –
Average Access Time: 13.6 ms

Last Benchmark: 83 hours ago

Start Read-Only Benchmark
Measure read rate and access time

Start Read/Write Benchmark
Measure read rate, write rate and access time

Close
MART Hardware (SSD)

Minimum Read Rate: 270.1 Mb/s
Maximum Read Rate: 282.0 Mb/s
Average Read Rate: 278.8 Mb/s
Last Benchmark: 83 hours ago

Minimum Write Rate: –
Maximum Write Rate: –
Average Write Rate: –
Average Access Time: 0.2 ms

Start Read-Only Benchmark
Measure read rate and access time

Start Read/Write Benchmark
Measure read rate, write rate and access time

Close
Next steps

- Barebone on-the-iron malware analysis
- Android platform support
- OSX platform support
- iOS platform support
Proof of Concept hardware

Prototype Shield

Arduino Duemilanove

Arduino 4-Channel Relay Shield

Ethernet Shield
Questions?

Michael Boman
michael.boman@2secure.se
http://www.2secure.se

Michael Boman
michael@michaelboman.org
http://michaelboman.org
@mboman