Collecting and Hunting for Indications of Compromise with Gusto and Style!

SANS Threat Hunting & IR Summit – New Orleans, April 12, 2016

Ismael Valenzuela, GSE #132
(Coder + Pentester + Forensics + IR) in a SOC=?
{ about me };  
Twitter: @aboutsecurity  

• +15 years of experience in InfoSec  
• Leading the Incident Response and Forensics team at Intel Security  
• SANS Instructor  
• GSE #132 + 10 other GIAC certificates  
• (lazy) coder at heart and collector of infosec roles: pentester, architect, forensics analyst, incident responder and threat researcher.  
• http://blog.ismaelvalenzuela.com
it all started a snowy winter morning
What was I trying to solve?

• Working as threat researcher for a large healthcare agency
• +50,000 endpoints centrally managed by McAfee ePO
• 700 events/second (17 million events/day)
• Little or no forensics in place. Triage, evidence collection, and analysis was manual and reactive
• Results of the triage process were difficult to validate and process
• Initial vector of compromise was difficult to determine
• Attacks were difficult to classify based on impact to the business
• Containment was the same on all cases (AV scan and reimage)
• No lessons learned
And why do you need more context

- **Traffic blocked to a suspicious IP**
  - What process is generating this traffic on the endpoint? And **why**?
  - Is there any other malicious activity on this host that is not being detected?
  - How do I respond to this? What should I do next?

Threat feeds loaded into SIEM (collective intelligence framework)
prototyping a solution: quick & dirty

Requirements

1. Collect host artifacts required for triage (event analysis) from all endpoints
2. Without deploying additional agents to the endpoint
3. Cross-platform
4. Must support automated, remote response
5. With minimal footprint, adhering to forensically sound procedures
6. Triage using both native and third party tools
7. Support proactive hunting for any given IoCs
8. Extensible to integrate with well known IR and forensics open source frameworks
And why you should love Flying Circus too

1. Tons of libraries created related to security and computer forensics
2. Ideal to create quick prototypes of applications or algorithms
3. Supports the design of RESTful interfaces (client-server services)
4. Python lets me focus on concepts rather than code
5. With Pyinstaller I can freeze (package) Python programs into small stand-alone executables
6. Also… it was named after the great Monty Python!
what do we need to automate?

Automated smart “triage”

• Malware doesn’t exist in a vacuum:
  • They need to run
  • They need to communicate
  • They need to be persistent
• How do we find “evidence”?
• Traditional forensics techniques are too time consuming
• Triaging can be used to identify relevant evidence quickly and guide the IR process

• Live forensic analysis
  • Volatile data
    • Processes
    • Network connections, etc.
  • Non volatile data
    • Program execution (prefetch)
    • Autorun locations
    • Master File Table (MFT), etc.
• Dump and examine memory
• Scan with Indicators of Compromise (IOCs)
And why you shouldn’t hunt on a Friday

1. Hunt for IOCs
2. Triage systems
3. Modify/create new IOCs

Detection & analysis cycle

FULL scope is determined
Contain
Eradicate
Lessons learned

Intel Security Public - @aboutsecurity
"The pattern matching Swiss knife for malware researchers (and everyone else!)

http://plusvic.github.io/yara

Pattern matching:
- Strings, regular expressions and binary patterns (hex strings)

Classification:
- On input: combination of strings & logic, stored in a YARA rule
- On output: tags, metadata
- Can be integrated in your Python projects (bindings)
- Great repository: http://yararules.com/
  @YaraRules
{ when you are told that you need to deploy an additional agent };
so how do we do all this? – rastrea2r

Triage and Hunting for IOCs with ‘gusto’ and style 😊

• Rastrea2r (pronounced rastreador):
  • https://github.com/aboutsecurity/rastrea2r (opensource!)
  • Command line tool (coz command line is SEXY!)
  • Python / Multiplatform (win32/64, linux and osx)
  • Uses a REST API to report YARA scans
  • Wrapper to sysinternal, system command and 3rd party tools
  • Easy to integrate with McAfee ePO (but also distributable via SSCM, etc.)
  • Packaged binaries available on github
Triage and Hunting for IOCs with ‘gusto’ and style 😊

- **yara-disk**: Yara scan for file/directory objects on disk
- **yara-mem**: Yara scan for running processes in memory
- **memdump**: Acquires a memory dump from the endpoint **Win only**
- **triage**: Collects triage information from the endpoint **Win only**

Obtaining HELP:

```
rastrea2r_win32_v0.7.exe –h
```
{ triage + hunting on steroids = rastrea2r }
Forensically sound architecture and communication flows
• Example:
  
  ```
  rastrea2r_win32_v0.7.exe triage tools.myserver.com data.myserver.com
  ```

  *** tools.myserver.com - has a read only shared folder called TOOLS
  *** data.myserver.com - has a write only shared folder called DATA

  Wrapper for 3rd party tools and native Windows commands
{ 3\textsuperscript{rd} party tools & native win commands };
Copy the toolset to the read-only ‘tools’ share

""" Add your list of Sysinternal / third-party / BATCH files here """
{ 3rd party tools & native win commands };
memory dumps with rastrea2r;

Only a click away from your web console

- Example:
  - `rastrea2r_win32_v0.7.exe memdump tools.myserver.com data.myserver.com`

  *** tools.myserver.com -> has a read only shared-folder called **TOOLS**
  *** data.myserver.com -> has a write only shared-folder called **DATA**

```
C:\Users\user\Desktop\rastrea2r client\rastrea2r_win32_v0.6> rastrea2r_win32_v0.6.exe memdump -h
usage: rastrea2r_win32_v0.6.exe memdump [-h] [-s] BIN_server DATA_server

positional arguments:
  BIN_server  Binary tool server (SMB share)
  DATA_server Data output server (SMB share)

optional arguments:
  -h, --help   show this help message and exit
  -s, --silent Suppresses standard output
```
Memory dumps of any managed host piped over SMB using winpmem

Full memory dump on RAW format:

- Ready to be parsed with memory analysis tools like volatility
starting the rastrea2r server;
Listening on all interfaces on port 8080
hunting with rastrea2r

IOC hunting in memory and disk a click away from your web console

*** Client / server architecture using a RESTful API

*** YARA rules must be stored on the same directory as the server

```bash
C:\Users\user\Desktop\rastrea2r client\rastrea2r_win32_v0.6> rastrea2r_win32_v0.6.exe yara-mem -h
usage: rastrea2r_win32_v0.6.exe yara-mem [-h] [-s] server rule

positional arguments:
  server    rastrea2r REST server
  rule      Yara rule on REST server

optional arguments:
  -h, --help    show this help message and exit
  -s, --silent  Suppresses standard output

C:\Users\user\Desktop\rastrea2r client\rastrea2r_win32_v0.6> rastrea2r_win32_v0.6.exe yara-disk -h
usage: rastrea2r_win32_v0.6.exe yara-disk [-h] [-s] path server rule

positional arguments:
  path     File or directory path to scan
  server   rastrea2r REST server
  rule     Yara rule on REST server

optional arguments:
  -h, --help    show this help message and exit
  -s, --silent  Suppresses standard output
```
{ hunting for iocs in memory with yara }

Example:

- rastrea2r_win32_v0.7.exe yara-mem localhost ransomware.yar
{ hunting for IOCs on disk with yara: }

Example:

- `rastrea2r_win32_v0.7.exe yara -disk c: \users \user localhost ransomware.yar`
Let's revisit our event…

• Traffic blocked to a suspicious IP
  • What process is generating this traffic on the endpoint? And why?
  • Is there any other malicious activity on this host that is not being detected?
  • How do I respond to this? What should I do next?

Threat feeds loaded into SIEM (collective intelligence framework)

Source IP/host name

Suspicious domain
Client tasks -

- product deployment
- rastrea2r triage/memdump

{ triaging with 'rastrea2r' from McAfee ePO console; };

Modularity:
• A specific task is created for each combination of command line switches

Reporting
Dashboard ePO Summary

<table>
<thead>
<tr>
<th>Repository Package Status</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Package Name</strong></td>
</tr>
<tr>
<td>DAT</td>
</tr>
<tr>
<td>Engine</td>
</tr>
<tr>
<td>Linux Engine</td>
</tr>
<tr>
<td>HEV for ePO</td>
</tr>
<tr>
<td>Mac Engine</td>
</tr>
</tbody>
</table>

 Systems per Top-Level Group

<table>
<thead>
<tr>
<th>1st Level Group</th>
<th>Number of Managed Systems</th>
</tr>
</thead>
<tbody>
<tr>
<td>My Organization</td>
<td>2</td>
</tr>
<tr>
<td>Total</td>
<td>2</td>
</tr>
</tbody>
</table>

Malware Detection History

No Data Found
analyzing the results (~5 minutes)

A folder is created per system, then per collection set (timestamp)
{ analyzing the results };
{ that sounds highly suspicious... };
### Email Analysis

**Send Mail to Russia Alert**

<table>
<thead>
<tr>
<th>Record Name</th>
<th>mxs.mail.ru</th>
</tr>
</thead>
<tbody>
<tr>
<td>Record Type</td>
<td>1</td>
</tr>
<tr>
<td>Time To Live</td>
<td>4</td>
</tr>
<tr>
<td>Data Length</td>
<td>4</td>
</tr>
<tr>
<td>Section</td>
<td>Additional</td>
</tr>
<tr>
<td>A (Host) Record</td>
<td>217.69.139.150</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Record Name</th>
<th>b-0.19-43000408.9851081.1644.1f0a.2f4a.410.0.9ape7qnbhhejna5s1525sn3wb.avts.mcafee.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>Record Type</td>
<td>1</td>
</tr>
<tr>
<td>Time To Live</td>
<td>3061</td>
</tr>
<tr>
<td>Data Length</td>
<td>4</td>
</tr>
<tr>
<td>Section</td>
<td>Answer</td>
</tr>
<tr>
<td>A (Host) Record</td>
<td>127.129.0.128</td>
</tr>
</tbody>
</table>

**Risk Assessment**

- **High Risk**
- **Medium Risk**
- **Unverified**
- **Minimal Risk**

This page shows details and results of our analysis on the domain `mxs.mail.ru`.

---

{ someone wants to send mail to Russia! }
what about that funky binary?

Always check hashes. Bad guys use Virustotal too!

**Mvsbsihj.exe**

- Low AV detection at the time of submission (checked with PEStudio)
- Injects itself into svchost.exe
- Checks network status
- Downloads a second payload with spambot capabilities
- In addition to sending spam it can download additional plugin components from C&C servers:
  - DDoS attacks
  - Sniff traffic and steal credentials
  - Read messages on Twitter, Skype, Facebook, etc.
  - Bitcoin mining, etc.

---

<table>
<thead>
<tr>
<th>Type</th>
<th>Size</th>
<th>Blacklisted (S)</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ascii</td>
<td>14</td>
<td>x</td>
<td>VirtualAllocEx</td>
</tr>
<tr>
<td>ascii</td>
<td>18</td>
<td>x</td>
<td>CreateRemoteThread</td>
</tr>
<tr>
<td>ascii</td>
<td>11</td>
<td>x</td>
<td>SendMessage</td>
</tr>
<tr>
<td>ascii</td>
<td>8</td>
<td>x</td>
<td>VBA6.DLL</td>
</tr>
<tr>
<td>ascii</td>
<td>15</td>
<td>x</td>
<td>DllFunctionCall</td>
</tr>
<tr>
<td>ascii</td>
<td>40</td>
<td>-</td>
<td>!This program cannot be run in DOS mode.</td>
</tr>
<tr>
<td>ascii</td>
<td>5</td>
<td>-</td>
<td>.text</td>
</tr>
<tr>
<td>ascii</td>
<td>6</td>
<td>-</td>
<td>`.data</td>
</tr>
<tr>
<td>ascii</td>
<td>5</td>
<td>-</td>
<td>.rsrC</td>
</tr>
<tr>
<td>ascii</td>
<td>12</td>
<td>-</td>
<td>MSVBVM60.DLL</td>
</tr>
<tr>
<td>ascii</td>
<td>5</td>
<td>-</td>
<td>`2@x;</td>
</tr>
<tr>
<td>ascii</td>
<td>24</td>
<td>-</td>
<td>= Filmwissenschaften</td>
</tr>
</tbody>
</table>
Hunting with YARA and rastrea2r

1. Create a new YARA rule based on this sample
2. Scan the environment and triage infected systems
3. Tune the YARA rule based on findings
4. Repeat, rinse and stop once the FULL scope is determined
5. Contain & eradicate
6. Conduct memory analysis and full disk forensics to determine root cause

```
rule systemXYZ-spambot
{
  strings:
    $string0 = "ffff."  
    $string1 = "AVVWSH"
    $string2 = "AWAVAUATVWSH"
    $string3 = "ffff."  
    $string4 = ".reloc"
  condition:
    4 of them }
```
Where are we now?

- Ability to triage and collect evidence from thousands of endpoints centrally managed by McAfee ePO in minutes
- Triage and evidence collection is automated, with proactive hunting of IOCs based on FBI TLPs, vendor reports, internal IR investigations and other threat intelligence feeds
- Results of the triage process can be validated by trained analysts
- Working on automating evidence/artifacts processing and analysis
- Initial vector of compromise can be determined on most cases
Thank You!

@aboutsecurity
http://blog.ismaelvalenzuela.com
ismael.valenzuela@intel.com
Thought Leadership

Contributing authors to all editions of Hacking Exposed

Professors and Lecturers

Open Security Research sponsored by Foundstone
blog.opensecurityresearch.com

Competition Judges/Mentors

Foundstone.com
Free Tools and Whitepapers