What makes us get sick?

Looking Upstream

• Where do asthma start?
• What about obesity, diabetes and heart disease?
• Can we identify the source of depression, anxiety and high blood pressure?

Or should we rather ask… where do we *live, work, eat, sleep* and *play*?
“The Upstream Story”
The Vicious Cycle of Security Failure
And the need for Automation & Orchestration

Emerging challenge

Inefficient security team

Isolated Tactical Mode

Technology sprawl

Procure new tech

Short-lived efficacy

Unscalable complexity leaves organizations combat-ineffective

Complexity • Cost

Efficiency • Efficacy • Agility

Intel Security Public - @aboutsecurity
Treating the Symptom = Security Failure
A Pinkslipbot Story

• From **200** to **2100** affected systems in less than 48 hours – *why??*

• Pinkslipbot/Qbot – a cybercrime worm that spreads over network shares and that steals banking credentials, logged on and admin credentials, among others
Symptom without 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)

Source IP/host name

Suspicious domain
Thinking like an “Upstreamist”
Trying to address root cause

• When a system is reported as infected or is acting suspiciously:
  • Do I have enough **context** to determine what to do next?
  • Should I simply disconnect the system and re-image?
  • What if the attacker can detect my response and change tactics?
  • Is this ransomware, a non-targeted campaign or an APT like attack?
  • Is it possible that the system holds other malware that hasn’t been detected yet?
  • How can I collect enough information from these systems (**quickly** enough) to determine the **best** containment and eradication **strategy**?
  • Can they help me to profile the attacker’s techniques?
  • How can I **proactively** search for Indicators of Compromise (IoC) across my endpoints?
The Challenge: The “Chronically Ill” SOC

What was I trying to solve?

- Large healthcare agency with +50,000 endpoints centrally managed
- 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 root cause analysis = No lessons learned = No risk management!
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
Automation == Python
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?
Swimming upstream as fast as possible!

• 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)
Hunting & Smart Incident Response

WARNING: Partial scoping can lead to disaster

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

Detection & analysis cycle

FULL scope is determined
Contain
Eradicate
Lessons learned
Hunting With YARA
Simply a better GREP!

• “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](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 3\(^{rd}\) party tools
  - Easy to integrate with McAfee ePO (but also distributable via SSCM, etc.)
  - Packaged binaries available on github
Current Functionality In **Rastrea2r V0.7**

And it works on Fridays too! (if you dare…)

- **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
Triaging With Rastrea2r

Wrapper for 3rd party tools and native Windows commands

• 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

```
C:\Users\user\Desktop\rastrea2r client\rastrea2r_win32_v0.6> rastrea2r_win32_v0.6.exe triage -h
usage: rastrea2r_win32_v0.6.exe triage [-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
```
3rd Party Tools & Native Win Commands

Copy the toolset to the read-only ‘tools’ share

""" Add your list of Sysinternal / third-party / BATCH files here """

```
tool=(
    'systeminfo.cmd', # Gathers systeminfo
    'set.cmd', # Gathers Set variables
    'dir-tree.cmd', # Enumerates C:\ directory tree
    'ipconfig.cmd', # Gathers IP information
    'ip-routes.cmd', # Gathers IP routing information
    'arp.cmd', # Gathers ARP table information
    'dns.cmd', # Gathers DNS Cache information
    'users.cmd', # Gathers User/Local Admin accounts
    'shares.cmd', # Gathers local shares information
    'firewall.cmd', # Gathers local firewall information
    'hosts.cmd', # Captures Host file information
    'sessions.cmd', # Gathers Active Session information
    'nbtstat.cmd', # Gathers NetBios Sessions information
    'netstat.cmd', # Gathers Netstat with process IDs
    'services.cmd', # Gathers services information
```
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

```bash
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 With Rastrea2r
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
Hunting For IOCs In Memory With YARA

Example:

• rastrea2r_win32_v0.7.exe yara-mem localhost ransomware.yar
Example:

- rastrea2r_win32_v0.7.exe yara-disk c:\users\user localhost ransomware.yar
Rastrea2r In Action
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
**Triaging with ‘rastrea2r’ from McAfee ePO console**

Client tasks -> product deployment -> rastrea2r triage/memdump

**Modularity:**

- A specific task is created for each combination of command line switches

---

<table>
<thead>
<tr>
<th>System Properties</th>
<th>Deep Command</th>
<th>Products</th>
<th>Threat Events</th>
<th>DLP User Session</th>
</tr>
</thead>
<tbody>
<tr>
<td>Agent GUID 8738P10C-2A0E-46B6</td>
<td>Deploy Agents</td>
<td>Modify Policies on a Single System</td>
<td>Modify Tasks on a Single System</td>
<td></td>
</tr>
<tr>
<td>Communication Type</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU Serial Number</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU Speed (MHz)</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>CPU Type</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Custom 1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Custom 2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Custom 3</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Custom 4</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Drive Encryption</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Drive Encryption Go</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Tag</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Agent</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Directory Management</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Free Memory 1.544.61 MB</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Free System Drive Space 429244 MB</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

---

Intel Security Public - @aboutsecurity
Analyzing The Results (~5 Minutes)

A folder is created per system, then per collection set (timestamp)
Analyzing The Results

<table>
<thead>
<tr>
<th>File Name</th>
<th>Date</th>
<th>Size</th>
<th>Type</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>20151006204120-BHB-IS5W9LA-P01-firewall.log</td>
<td>10/6/2015</td>
<td>4 KB</td>
<td>Text Document</td>
<td>4 KB</td>
</tr>
<tr>
<td>20151006204121-BHB-IS5W9LA-P01-hosts.log</td>
<td>10/6/2015</td>
<td>1 KB</td>
<td>Text Document</td>
<td>1 KB</td>
</tr>
<tr>
<td>20151006204121-BHB-IS5W9LA-P01-sessions.log</td>
<td>10/6/2015</td>
<td>6 KB</td>
<td>Text Document</td>
<td>6 KB</td>
</tr>
<tr>
<td>20151006204133-BHB-IS5W9LA-P01-nbtstat.log</td>
<td>10/6/2015</td>
<td>1 KB</td>
<td>Text Document</td>
<td>1 KB</td>
</tr>
<tr>
<td>20151006204133-BHB-IS5W9LA-P01-netstat.log</td>
<td>10/6/2015</td>
<td>11 KB</td>
<td>Text Document</td>
<td>11 KB</td>
</tr>
<tr>
<td>20151006204134-BHB-IS5W9LA-P01-process-list.log</td>
<td>10/6/2015</td>
<td>214 KB</td>
<td>Text Document</td>
<td>214 KB</td>
</tr>
<tr>
<td>20151006204134-BHB-IS5W9LA-P01-services.log</td>
<td>10/6/2015</td>
<td>44 KB</td>
<td>Text Document</td>
<td>44 KB</td>
</tr>
<tr>
<td>20151006204135-BHB-IS5W9LA-P01-tasklist.log</td>
<td>10/6/2015</td>
<td>179 KB</td>
<td>Text Document</td>
<td>179 KB</td>
</tr>
<tr>
<td>20151006204205-BHB-IS5W9LA-P01-at-schtasks.log</td>
<td>10/6/2015</td>
<td>16 KB</td>
<td>Text Document</td>
<td>16 KB</td>
</tr>
<tr>
<td>20151006204208-BHB-IS5W9LA-P01-startup-list.log</td>
<td>10/6/2015</td>
<td>8 KB</td>
<td>Text Document</td>
<td>8 KB</td>
</tr>
<tr>
<td>20151006204214-BHB-IS5W9LA-P01-psinfo.log</td>
<td>10/6/2015</td>
<td>1 KB</td>
<td>Text Document</td>
<td>1 KB</td>
</tr>
<tr>
<td>20151006204214-BHB-IS5W9LA-P01-zRemote.log</td>
<td>10/6/2015</td>
<td>1 KB</td>
<td>Text Document</td>
<td>1 KB</td>
</tr>
<tr>
<td>20151006204216-BHB-IS5W9LA-P01-diskext.log</td>
<td>10/6/2015</td>
<td>1 KB</td>
<td>Text Document</td>
<td>1 KB</td>
</tr>
<tr>
<td>20151006204216-BHB-IS5W9LA-P01-logonsessions.log</td>
<td>10/6/2015</td>
<td>6 KB</td>
<td>Text Document</td>
<td>6 KB</td>
</tr>
</tbody>
</table>
That Sounds Highly Suspicious...

```plaintext
Caption=ConnectionCenter
Command="C:\Users\BHB-NS17N-P05\AppData\Local\Citrix\ICA Client\concentr.exe" /startup
Description=ConnectionCenter
Location=HKU\S-1-5-21-2250110424-2442967196-2465209428-110119\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
SettingID=
User=CORP\BHB-NS17N-P05

Caption=MSConfig
Command="C:\Users\BHB-NS17N-P05\mvsbsihj.exe"
Description=MSConfig
Location=HKU\S-1-5-21-2250110424-2442967196-2465209428-110119\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
SettingID=
User=CORP\BHB-NS17N-P05

Caption=RTHDVCPCL
Command=C:\Program Files\Realtek\Audio\HDA\RTHDVCPCL.exe -s
Description=RTHDVCPCL
Location=HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
SettingID=
User=Public

Caption=IMSS
Command="C:\Program Files\Intel\Intel(R) Management Engine Components\IMSS\PIconStartup.exe"
Description=IMSS
Location=HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run
SettingID=
User=Public
```
Someone Wants To Send Mail To Russia!

Record Name: mail.ru
Record Type: 15
Time To Live: 4
Data Length: 8
Section: Answer
MX Record: mxs.mail.ru
10
0

Record Name: mxs.mail.ru
Record Type: 1
Time To Live: 4
Data Length: 4
Section: Additional
A (Host) Record: 217.69.139.150

Record Name: mxs.mail.ru
Record Type: 1
Time To Live: 4
Data Length: 4
Section: Additional
A (Host) Record: 94.100.180.150

b-0.19-43000408.9851081.1644.1f0a.2f4a.410.0.9ape7qnbhhzejna5s1525sn3wb.avts.mcafee.com

Record Name: b-0.19-43000408.9851081.1644.1f0a.2f4a.410.0.9ape7qnbhhzejna5s1525sn3wb.avts.mcafee.com
Record Type: 1
Time To Live: 3061
Data Length: 4
Section: Answer
A (Host) Record: 127.129.0.128
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
- 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.
What Next?

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. Determine initial vector: root cause analysis!!

rule systemXYZ-spambot
{
    strings:
        $string0 = "fffff."
        $string1 = "AVVWSH"
        $string2 = "AWAVAUATVWSH"
        $string3 = "ffiff.";
        $string4 = ".reloc"
    condition:
        4 of them
}
Initial Vector Determination
Malicious download from personal email = root cause!

- Reconstructing the Master File Table based off the memory dump captured with rastrea2r and parsing it with Volatility’s `mftparser` plugin shows web activity related to `mail.yahoo.com` and a reference to an email attachment named "Court_Notification_00000557121.zip"

- Correlation with web proxy logs confirms the suspicious activity

<table>
<thead>
<tr>
<th>&amp;enc=auto&amp;cmd=msg.scan&amp;pid=2&amp;tnef=&amp;fn=Court_Notification_00000557121.zip</th>
<th>63.238.84.250</th>
<th>vassg142.ocsp.omniroot.com</th>
</tr>
</thead>
<tbody>
<tr>
<td>98.139.21.169</td>
<td>98.139.21.169</td>
<td>login.yahoo.com</td>
</tr>
<tr>
<td>98.139.21.169</td>
<td>login.yahoo.com</td>
<td>login.yahoo.com</td>
</tr>
<tr>
<td>login.yahoo.com</td>
<td>login.yahoo.com</td>
<td>login.yahoo.com</td>
</tr>
<tr>
<td>login.yahoo.com</td>
<td>login.yahoo.com</td>
<td>login.yahoo.com</td>
</tr>
<tr>
<td>login.yahoo.com</td>
<td>login.yahoo.com</td>
<td>login.yahoo.com</td>
</tr>
<tr>
<td>23.54.187.27</td>
<td>gn.symcd.com</td>
<td>ocs.pentrust.net</td>
</tr>
<tr>
<td>104.97.107.102</td>
<td>ocs.pentrust.net</td>
<td>ocs.digicert.com</td>
</tr>
<tr>
<td>72.21.91.29</td>
<td>ocs.digicert.com</td>
<td>westcoastcontest.com</td>
</tr>
<tr>
<td>178.255.83.1</td>
<td>rabbitheadstudios.net</td>
<td>vaska.pl</td>
</tr>
<tr>
<td>vaska.pl</td>
<td>vaska.pl</td>
<td>vaska.pl</td>
</tr>
<tr>
<td>vaska.pl</td>
<td>vaska.pl</td>
<td>luchgallery.com</td>
</tr>
<tr>
<td>vaska.pl</td>
<td>luchgallery.com</td>
<td>vaska.pl</td>
</tr>
</tbody>
</table>
The Results: The “Healthy SOC”

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

• *Root cause analysis = Lessons learned = Risk management -> Healthier SOC!*
Are you an “Upstreamist”?

Thanks!

@aboutsecurity

http://blog.ismaelvalenzuela.com

ismael.valenzuela@intel.com

The Healthy SOC
Thought Leadership

Contributing authors to all editions of Hacking Exposed

Professors and Lecturers

blog.opensecurityresearch.com

Competition Judges/Mentors

Foundstone.com

Free Tools and Whitepapers!