INVESTIGATING INTRUSIONS AT ADVERSARY SPEED

CHRISTOPHER WITTER, SENIOR MANAGER FALCON OVERWATCH
Intro
The Problem
Example Intrusion Timeline
The Keys to Success
The Technologies
Winning!
INTRO

- DFIR since early 2000s
  - Service Providers
  - Financial Institutions
  - Defense and Government

- 2 x DFIR Summit Speaker (USA)

- Manager Falcon Overwatch
  - International Team of Hunters \ Intrusion Analysts

- Outdoor Enthusiast
THE PROBLEM

- Adversaries can compromise hosts at a rate faster than we can investigate and respond with reasonable accuracy in a timely fashion. Traditional tools and methods don’t scale to meet the demands of today’s intrusions.
  - 15-30 minutes timeline for a smash and grab
  - 10-20 hosts impacted
  - Mixed TTPs
    - Powershell
    - Custom tools
    - Built in operating system commands
YOUR OODA LOOP

1. Understand the detection mechanism
2. Review the detection details
3. What’s normal for this computer/user
4. What’s normal for this segment/BU
5. What’s normal across the environment
6. Peer review

Observe
Act
Decide
Orient

Detections
Start IR
Block Indicators
Add/Change detections
Malicious
Suspicious
Benign
## EXAMPLE INTRUSION TIMELINE

<table>
<thead>
<tr>
<th>Stage</th>
<th>Actor</th>
<th>Date Added</th>
<th>Host Name</th>
<th>Date (UTC)</th>
<th>Activity</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td></td>
<td>2015</td>
<td>140</td>
<td>2015-12-26T16:34:11Z</td>
<td>C:\Windows\System32\cmd.exe /c net use \docs.live.net@outlook.com</td>
</tr>
<tr>
<td>2</td>
<td></td>
<td>2015</td>
<td>140</td>
<td>2015-12-26T16:34:11Z</td>
<td>C:\Windows\System32\cmd.exe /c &quot;reg add HKCU\Software\Microsoft\Windows\CurrentVersion\Run\ /f /v &quot;AdobeARM&quot; /t REG_SZ /d &quot;C:\Users[redacted]\AdobeARM.exe&quot;&quot;</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>2015</td>
<td>140</td>
<td>2015-12-26T16:34:11Z</td>
<td>powershell.exe -windowstyle hidden -enc</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>2015</td>
<td>140</td>
<td>2015-12-26T16:34:11Z</td>
<td>C:\Windows\System32\cmd.exe /c &quot;whoami&quot;</td>
</tr>
<tr>
<td>3</td>
<td></td>
<td>2015</td>
<td>140</td>
<td>2015-12-26T16:34:11Z</td>
<td>&quot;C:\Windows\System32\cmd.exe&quot; /c dir /b \network.local\SYSVOL &gt; C:\Users.\AppData\Local\Temp\waudit\dc\network.local\network.local.txt&quot;</td>
</tr>
</tbody>
</table>
TRADITIONAL ANALYSIS:

- Network Forensics:
  - Intrusion Detection
  - Full PCAP
  - DNS Logs
  - Proxy Logs
- Disk Forensics:
  - Encase
  - Autopsy
  - FTK
  - X-Ways
- Memory Forensics:
  - Volatility
  - RedLine
- Windows Log Analysis
KEYS TO SUCCESS

- Reduce your time to **Orient**
- Increase what you have at your disposal to **Decide**
- Decrease your time to **Action**
REDUCE YOUR TIME TO ORIENT

- Automate repetition
  - Evidence collection
  - Data enrichment
- Bring the data closer to your analyst
  -Unified views
  - Workflow automation
  - Context rich sources of information
INCREASE WHAT YOU HAVE … TO DECIDE

- Our Example Intrusion Timeline:
  - Full PCAPS required to extract artifacts from initial infection, exfiltration determination
  - DNS or Proxy logs for C2 determination
  - Registry Analysis for Persistence
  - Powershell logging to determine intent and actions taken by the script
  - Memory Dumps
  - Traditional forensics
    - Timeline generation
    - Carve or capture reconnaissance files
    - Command line executions …maybe
THE TECHNOLOGIES: PART 1

- Endpoint Polling / Forensics – targeted post incident collection or polling of forensic artifacts.
  - Commercial:
    - Mandiant MIR
    - Encase Endpoint Investigator
  - Opensource / Free
    - Crowd Response
    - Google Rapid Response (GRR)
    - Creative administrative solutions
END POINT POLLING \ FORENSICS

<table>
<thead>
<tr>
<th>Pros:</th>
<th>Cons:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- Easy to implement</td>
<td>- Slower than EDR</td>
</tr>
<tr>
<td>- Low storage requirements</td>
<td>- Requires separate network collection</td>
</tr>
<tr>
<td>- Targeted collection</td>
<td>- Need separate Network infrastructure for network logs etc.</td>
</tr>
<tr>
<td>- Leverages more traditional skills and analysis techniques</td>
<td>- Can’t tie network events to processes</td>
</tr>
<tr>
<td>- Faster than traditional forensics</td>
<td></td>
</tr>
</tbody>
</table>
CROWD RESPONSE

- Scan all running processes, loaded DLLs
- Scan on-disk image binaries
- Scan arbitrary files and folders
- YARA detection engine
- Collect Task Scheduler & AT jobs, Registry, SHIM Cache, Pre\SuperFetch
- Limit scanning by regular expression
- Download rules from a central URL
- Splunk App available for data ingestion
CROWD RESPONSE

- **Pros:**
  - Detection capabilities
  - Enables Hunting for known \ unknown (Frequency analysis)
  - Splunk ingestion App w/prebuilt dashboards

- **Cons**
  - Missing some required traditional artifacts
    - Windows logs
    - MFT
THE BARE MINIMUM TO DECIDE

- Process information OR Command Lines
- Windows event logs OR Login information (type of login: Remote, Interactive, Network, etc)
- Powershell logs (PS 5.0) OR Command Lines
- Registry persistence keys
- Full PCAP OR (DNS, Proxy, and Network Flow information)
THE TECHNOLOGIES: PART 2

- Endpoint Detection and Response (EDR) – real time streaming of operating system events (process creation, files written, registry changes, DNS events, user identity events, and network events, etc.)
  - Commercial:
    - CrowdStrike Falcon Host (Windows, OSX, Linux)
    - Carbon Black (Windows, OSX, Linux)
    - Windows Defender ATP (Windows 10 only)
  - Opensource \ Free:
    - Lima Charlie (Multi-platform, contains detection components)
    - Sysmon (Windows logging only)
    - Windows Event Forwarding (Windows only)
ENDPOINT DETECTION RESPONSE (EDR)

**Pros:**
- Process information - > CMD LINES
- Network information
- DNS information
- File writes
- Registry changes
- Detection components
- Linkage between events
- Logon information
- One team, one tool

**Cons:**
- Events can be voluminous
- Depending on the solution may require significant tuning to start
- Data storage, searching, and analytics requirements vary and impacts on privacy laws based on country
SYSMON

- Process Creations
  - Parent \ Child relationships
  - Hashes of processes, loaded DLLs & drivers

- Network Connections (Ties it to source process)
  - Source \ Destination
  - Ports
  - Hostnames

- File creation times
SYSMON

- **Pros:**
  - Free (Windows 7 +, Windows Server 2012 +)
  - A great data source for hunting
  - Can be stored locally or sent to central collection points
  - COMMAND LINES
  - Splunk Add-ons

- **Cons:**
  - Windows Only
  - Missing Registry artifacts
  - Will need extensive tuning based on environment
  - No built in detection capabilities
  - Should be supplemented with additional windows logs
    - Powershell logs Optional (with the command line arguments you would gain very little)
    - Windows Event logs (logon events, etc.)
EDR = WINNING

- **EDR**
  - Single Data OR minimal data sources - > reduced **ORIENT**
  - Streaming data - > reduced **ORIENT**
    - Real time or near real time situational awareness
  - Increased Contextual awareness - > reduced **DECIDE**
    - Did they open the attachment?
    - What did they get?
    - We recovered the exfil files but they’re password protected and encrypted. Do we have the password?
    - What process made the connection to the C2?
    - No longer need to interview employees for “context”
  - Story time!
QUESTIONS?

Interested in investigating and hunting adversaries, I’m hiring!

Contact info:
- Christopher.Witter@crowdstrike.com