There's an Actor In My Pocket!

Daniel Garcia and Jennifer Chavarria Reindl
Threat Hunters
Safety First!

PLANNED FIRE ALARMS

NEAREST FIRE EXIT

MUSTER POINT
“Did you ever have the feeling there’s a WASKET in your BASKET?...”

...or an actor in your logs?
C:\>net localgroup presenters
Members

Daniel Garcia
Jennifer Chavarria Reindl
C: \Windows\System32\cmd.exe

C:\>quser "Daniel Garcia"

****Loading Custom Output ****

> Sr. Threat Hunter for Shell Oil
> 11+ years of experience in IT and Cybersecurity
> Majority of those years in the Oil & Gas sector
> Certs: Mostly SANS and Microsoft
> Previous roles: IR, Malware Analysis, Threat Intelligence
> Twitter: @43nsicbot
C:\\>quser "Jennifer Chavarria Reindl"
****Loading Custom Output****

> Threat Hunter for Shell Oil
> 5+ years of experience in IT and Cybersecurity
> Certs: GCIH, GRID
> Previous roles: IR, User Awareness, Application Support
> Twitter: @jennxxdcs
C:\Windows\System32\cmd.exe

C:\Agenda>tree
Folder PATH listing for volume Local Disk
Volume serial number is 1337 D34D:833F

C:.

1. What is Threat Hunting to Shell?
   └── Hunting Process
   └── Hunting Outcomes

2. Hunting Examples

3. Bring It All Together

4. Conclusion

Q&A
What is Threat Hunting to Shell?
Shell Business Overview

01 Exploration
- Exploring for oil and gas: offshore
- Exploring for oil and gas: onshore

02 Development and extraction
- Developing fields
- Producing oil and gas
- Extracting bitumen

03 Manufacturing and energy production
- Upgrading bitumen
- Refining oil into fuels and lubricants
- Converting gas into liquid products (GTL)
- Producing petrochemicals
- Producing biofuels
- Generating power

04 Transport and trading
- Liquefying gas by cooling (LNG)
- Shipping and trading
- Regasifying (LNG)
- Supply and distribution

05 Sales and marketing
- Retail
- Lubricants
- Aviation

Customers
What Is Threat Hunting to Shell?

Proactively searching for adversary behaviours that evade existing security controls in our environment by utilizing a hypothesis-driven approach.
Hunting Process

**Threat Intel**
Tactics, techniques, procedures observed, peer interactions publicly and privately (subscription).

**Internal Incidents**
What techniques or procedures were observed?

**Hypothesis/Hunches**
What new techniques or procedures can an adversary deploy to achieve operational goals?

Adversaries can execute code or maintain persistence by utilizing native windows scheduled tasks on compromised Windows hosts in our environment.
Hunting Process – Adversary Based Hypothesis


Adversary X can deliver payloads via phishing emails in the next two months on E [location] whereby F [victim] is affected, because G [motive]

to U.S. refineries whereby operators are affected, because data can be destroyed.
Hunting Process – Tools and Data Sources

Security Products
- Anti-Virus/Anti-Malware
- Firewall
- Endpoint Detection & Response
- Web Proxy
- Threat Intelligence Platform
- Application Security
- Data Loss Prevention
- Email Gateway
- Intrusion Prevention System
- Behavioural Analytics
- Mobile Device Security
- Identity and Access Management
- VPN

Target a set of endpoints

MITRE ATT&CK™
Hunting Outcomes

Findings
- Security control gaps
- Compliance violations

New Intel
- Find new techniques, procedures, adversaries

Incidents
- Find adversaries!

Use Cases
- Detection logic and alerts

Nothing
- Document and revisit in the future
Hunting Examples

The following slides are real examples from our environment but have been anonymized to protect the innocent.

Keep an eye out for a particular user - “Alice Tellar!”
Hunt #1 – Accessibility Utilities Abuse “Sticky Keys”

Input: Threat Intel

Testing of Hypothesis

Outcome

Frequently, we also see the use of the “sticky keys” trick for maintaining malware-free persistence on a victim network. With such trick, the adversary will modify the registry on a remote machine (typically using WMI) to set “cmd.exe” as a Debugger for tools like sethc.exe (Sticky Keys) and osk.exe (On-screen keyboard). Once that's done, an attacker can RDP into that machine and press the Sticky Keys or On-Screen Keyboard hotkeys and instantly get a command prompt running with System-level privileges without even requiring a login into the remote server. Thus, even if passwords are reset across the victim environment, the adversary may still maintain persistent access unless all the registry entries are cleaned up.

https://www.crowdstrike.com/blog/adversary-tricks-crowdstrike-treats/
Hunt #1 – Accessibility Utilities Abuse “Sticky Keys”

**Hypothesis**

Adversaries have replaced or changed the registry debugger value for accessibility utilities to gain persistence and elevated access to our endpoints.

**ATT&CK**

- **Tactics:** TA0003: Persistence, TA0004: Privilege Escalation
- **Technique:** T1015 – Accessibility Features
# Hunt #1 – Accessibility Utilities Abuse “Sticky Keys”

<table>
<thead>
<tr>
<th>Input: Threat Intel</th>
<th>Testing of Hypothesis</th>
<th>Outcome</th>
</tr>
</thead>
</table>

## Hunting Logic

**title:** Accessibility Utilities Abuse

**description:** detects accessibility utilities executing as a parent process

**tags:**
- attack.privilege_escalation
- attack.persistence

**level:** high

**logsource:**
- category: process_execution

**detection:**

<table>
<thead>
<tr>
<th>selection_1:</th>
</tr>
</thead>
<tbody>
<tr>
<td>ParentImage:</td>
</tr>
<tr>
<td>- *\sethc.exe</td>
</tr>
<tr>
<td>- *\utilman.exe</td>
</tr>
<tr>
<td>- *\osk.exe</td>
</tr>
<tr>
<td>- *\narrator.exe</td>
</tr>
<tr>
<td>- *\magnify.exe</td>
</tr>
<tr>
<td>- *\displayswitch.exe</td>
</tr>
<tr>
<td>- *\atbroker.exe</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Image:</th>
</tr>
</thead>
<tbody>
<tr>
<td>- **.exe</td>
</tr>
</tbody>
</table>

**condition:** selection_1
Hunt #1 – Accessibility Utilities Abuse “Sticky Keys”

<table>
<thead>
<tr>
<th>Input: Threat Intel</th>
<th>Testing of Hypothesis</th>
<th>Outcome</th>
</tr>
</thead>
</table>

![Registry Editor](Image)

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Debugger</td>
<td>REG_SZ</td>
<td>C:\windows\system32\cmd.exe</td>
</tr>
</tbody>
</table>
Hunt #1 – Accessibility Utilities Abuse “Sticky Keys”

Input: Threat Intel  
Testing of Hypothesis  
Outcome

- **sethc.exe Properties**
  - **Description:** Windows Command Processor
  - **Location:** C:\Windows\System32
  - **Size:** 227 KB (232,448 bytes)
  - **Size on disk:** 228 KB (233,472 bytes)

- **sethc.exe Properties**
  - **Description:** Accessibility shortcut keys
  - **Location:** C:\Windows\System32
  - **Size:** 265 KB (271,360 bytes)
  - **Size on disk:** 268 KB (274,432 bytes)
## Hunt #1 – Accessibility Utilities Abuse “Sticky Keys”

**Input:** Threat Intel  
**Testing of Hypothesis**  
**Outcome**

### Outcome: Possible Use Case, Lead for IR

<table>
<thead>
<tr>
<th>hostname</th>
<th>parent_process</th>
<th>child_process</th>
<th>user</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laptop-98915</td>
<td>C:\Windows\System32\sethc.exe</td>
<td>c:\windows\system32\cmd.exe</td>
<td>SYSTEM</td>
</tr>
<tr>
<td>Laptop-87459</td>
<td>C:\Windows\System32\osk.exe</td>
<td>c:\windows\system32\cmd.exe</td>
<td>SYSTEM</td>
</tr>
</tbody>
</table>

**Accessibility Utilities launching cmd.exe??**
Hunt #2 – Domain Fronting

Input: Threat Intel

Testing of Hypothesis

Outcome

CONNECT reputable.cloudprvdr.com HTTP/1.1
GET / HTTP/1.1
Host: attacker.cloudprvdr.com

https://blog.cobaltstrike.com/2017/02/06/high-reputation-redirectors-and-domain-fronting/
Hunt #2 – Domain Fronting

Input: Threat Intel  
Testing of Hypothesis  
Outcome

Hypothesis

• Adversaries leverage Domain Fronting to obfuscate C2 traffic and are potentially going undetected in our environment.

ATT&CK

• Tactic: TA0011 - Command and Control
• Technique: T1172 - Domain Fronting
Hunt #2 – Domain Fronting

Input: Threat Intel

Testing of Hypothesis

Outcome

Hunting Logic

**title**: Domain Fronting
**description**: detects suspicious web requests that are potential domain fronts
**tags**:
  - attack.c2
**level**: medium
**logsource**:
  - **category**: proxy
**detection**:
  - **selection_1**:
    - **http_method**: ‘-GET’
  - **condition**: selection_1 and not (host_field = original_domain)
Hunt #2 – Domain Fronting

Input: Threat Intel  
Testing of Hypothesis  
Outcome

wget -d --config=C:\Users\test.user\test\waetrc.txt --secure-protocol=TLSv1 --no-check-certificate --save-headers --header="Host: redirect-example.com" https://example.com/file-fetched

<table>
<thead>
<tr>
<th>http_method</th>
<th>dest_host</th>
<th>http_status</th>
<th>useragent</th>
<th>url</th>
</tr>
</thead>
<tbody>
<tr>
<td>CONNECT</td>
<td>EXAMPLE.COM</td>
<td>200</td>
<td>Wget/1.19.1 (cygwin)</td>
<td>EXAMPLE.COM:443</td>
</tr>
<tr>
<td>GET</td>
<td>REDIRECT-EXAMPLE.COM</td>
<td>403</td>
<td>Wget/1.19.1 (cygwin)</td>
<td>REDIRECT-EXAMPLE.COM/FILE-FETCHED</td>
</tr>
</tbody>
</table>

This redirection should not happen!
Hunt #2 – Domain Fronting

Input: Threat Intel  Testing of Hypothesis  Outcome

Outcome: Nothing (yet), Findings Reported

- Can we detect or protect against it? No, but requests have been made to the vendor to investigate:
  - Why does the redirection happen?
  - Can the Host Header field be logged for detection?
Hunt #3 – Malware Delivered Through UDF Image

Input: Threat Intel  Testing of Hypothesis  Outcome

Malware Sample Delivered Through UDF Image

I found an interesting phishing email which was delivered with a malicious attachment: an UDF image (.img). UDF means “Universal Disk Format” and, as said by Wikipedia[1], is an open vendor-neutral file system for computer data storage. It has supplanted the well-known ISO 9660 format (used for burning CD & DVD) that was also used in previous campaign to deliver malicious files[2].

https://isc.sans.edu/forums/diary/Malware+Sample+Delivered+Through+UDF+Image/24854/
Hunt #3 – Malware Delivered Through UDF Image

Input: Threat Intel
Testing of Hypothesis
Outcome

Hypothesis

• Adversaries utilize UDF attachments to enclose payloads in phishing lures to bypass perimeter scan controls (Email AV+Sandbox) in our environment.

ATT&CK

• Tactics: TA0001 - Initial Access, TA0002 - Execution
• Techniques: T1193 – Spearphishing Attachment, T1204 – User Execution
Hunt #3 – Malware Delivered Through UDF Image

**Hunting Logic**

**title:** UDF Attachments

**description:** detects UDF attachments that may enclose adversary payloads

**tags:**
- attack.initial_access
- attack.execution

**level:** high

**logsource:**
- category: endpoint
- description: filewrites

**detection:**

**selection_1:**
- FileWritePath: 
  - *\users\*
- FileExtension: 
  - ^(iso | img)$
- Image: 
  - *\outlook.exe

**condition:** selection_1 and FileSize > 5 MB
# Hunt #3 – Malware Delivered Through UDF Image

**Input:** Threat Intel  
**Testing of Hypothesis**  

**Outcome:** Possible Use Case, Lead for IR, Candidate for blocking

<table>
<thead>
<tr>
<th>hostname</th>
<th>fwrite_path</th>
<th>file_name</th>
<th>process</th>
<th>md5</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laptop-87475</td>
<td>C:\Users\r.white\AppData\Local\Microsoft\Windows\INetCache\Content.Outlook\G4DQB8FH\Shipment For Pickup.img</td>
<td>Shipment For Pickup.img</td>
<td>OUTLOOK.EXE</td>
<td>c31d8cd98f00b204e9800998ecf8427</td>
</tr>
<tr>
<td>Laptop-87459</td>
<td>C:\Users\a.tellar\AppData\Local\Microsoft\Windows\INetCache\Content.Outlook\O3LXG7HW\REF#05012019.pdf.img</td>
<td>REF#05012019.pdf.img</td>
<td>OUTLOOK.EXE</td>
<td>g21d8cd98f00b203e9800998ecf8489</td>
</tr>
</tbody>
</table>
Eventing Requirements

1. Event filter
   - The event of interest; user logon, removable media mounted, etc.

2. Event consumer
   - An action to perform upon triggering an event; ActiveScript or CommandLineEvent

3. Filter to consumer binding
   - The registration mechanism that binds a filter to a consumer
Hunt #4 – WMI Subscriptions

Input: Threat Intel

Testing of Hypothesis

Outcome

Hypothesis

• Adversaries abuse WMI subscriptions as an obscure method to persist and execute a payload on our endpoints.

ATT&CK

• Tactics: TA0003 - Persistence, TA0002 - Execution

• Technique: T1084 - Windows Management Instrumentation Event Subscription
Hunt #4 – WMI Subscriptions

**Hunting Logic**

**title:** WMI Subscriptions  
**description:** detect wmi subscription creation  
**tags:**  
- attack.persistence  
- attack.execution  
**level:** high  
**logsource:**  
**category:** endpoint  
**description:** filewrites, cmdline  
**detection:**  
**selection_1:**  
**FileWritePath:**  
- *\appdata\roaming\Microsoft\windows\powershell\psreadline  
**selection_2:**  
**commandline:**  
- subscription.* eventfilter | activescripteventconsumer | commandlineeventconsumer | filtertoconsumerbinding \s+create \s+name  
**selection_3:**  
**Image:** wmic.exe  
**commandline:**  
- root.*subscription[^s]  
**condition:** selection_1 or selection_2 or selection_3
### Hunt #4 – WMI Subscriptions

**Input:** Threat Intel  ➔ **Testing of Hypothesis**  ➔ **Outcome**

**Outcome:** Possible Use Case, Lead for IR

<table>
<thead>
<tr>
<th>host</th>
<th>process</th>
<th>parent</th>
<th>user</th>
<th>cmdline</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laptop-92124</td>
<td>WMIC.exe</td>
<td>cmd.exe</td>
<td>j.wick</td>
<td>WMIC /NAMESPACE:\root\subscription PATH __EventFilter WHERE __CLASS=&quot;__EventFilter&quot; AND __EventFilter.Name=&quot;VolumeArrival&quot;</td>
</tr>
<tr>
<td>Laptop-92125</td>
<td>WMIC.exe</td>
<td>cmd.exe</td>
<td>a.seltzer</td>
<td>WMIC /NAMESPACE:\root\subscription PATH __EventFilter WHERE __CLASS=&quot;__EventFilter&quot; AND __EventFilter.Name=&quot;VolumeArrival&quot;</td>
</tr>
<tr>
<td>Laptop-87459</td>
<td>powershell.exe</td>
<td>cmd.exe</td>
<td>a.tellar</td>
<td>powershell $iConsumer = ([wmiclass]&quot;.</td>
</tr><tr>
<td>oot\DEFAULT:CommandLineEventConsumer&quot;).CreateInstance()</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Hunt #5 – Anomalous RDP Traffic (CVE-2019-0708 – “BlueKeep”)  

Input: Threat Intel  
Testing of Hypothesis  
Outcome

Prevent a worm by updating Remote Desktop Services (CVE-2019-0708)

Today Microsoft released fixes for a critical Remote Code Execution vulnerability, CVE-2019-0708, in Remote Desktop Services – formerly known as Terminal Services – that affects supported versions of Windows. The Remote Desktop Protocol (RDP) itself is not vulnerable. The vulnerability is pre-authentication and requires no user interaction. In other words, the vulnerability is "wormable", meaning that any attacker malware that exploits this vulnerability could propagate from vulnerable computers to vulnerable computer in a similar way as the WannaCry malware spread across the globe in 2017. While we have observed no exploitation of this vulnerability, it is highly likely that malicious actors will write an exploit for this vulnerability and incorporate it into their malware.

Wormable, and can spread like WannaCry!
Hunt #5 – Anomalous RDP Traffic (CVE-2019-0708 – “BlueKeep”)

**Input: Threat Intel**

**Testing of Hypothesis**

**Outcome**

**Hypothesis**

- Adversaries exploit CVE-2019-0708 to distribute arbitrary code to multiple systems in our environment in a wormable fashion.

**ATT&CK**

- **Tactic:** TA0002 – Execution, TA0008 - Lateral Movement
- **Technique:** T1210 - Exploitation of Remote Services
Hunt #5 – Anomalous RDP Traffic (CVE-2019-0708 – “BlueKeep”)
### Outcome: Possible Use Case, Lead for IR, Possible Whitelisting

<table>
<thead>
<tr>
<th>src</th>
<th>src_user</th>
<th>traffic</th>
<th>file_path</th>
<th>user_dept</th>
<th>unique_dest</th>
</tr>
</thead>
<tbody>
<tr>
<td>Laptop-92018</td>
<td>k.pepperpot</td>
<td>Outgoing</td>
<td>C:/tools/wfreerdp.exe</td>
<td>Developer</td>
<td>4</td>
</tr>
<tr>
<td>Laptop-92019</td>
<td>j.doe</td>
<td>Outgoing</td>
<td>C:/Program Files/WindowsApps/Microsoft.RemoteDesktop 10.1.1098.0 x86_8wekyb3d8bbwe/RdClient.Windows.exe</td>
<td>IT</td>
<td>3</td>
</tr>
<tr>
<td>Laptop-87459</td>
<td>a.tellar</td>
<td>Outgoing</td>
<td>C:/users/a.tellar/appdata/local/temp/Microsoft/zsca.exe</td>
<td>Research</td>
<td>37</td>
</tr>
</tbody>
</table>
Hunting Outcomes

- **Accessibility Utilities Abuse**
  - Lead for IR
  - Use Case

- **Domain Fronting**
  - Security Vendor Gap

- **Malware Delivered via UDF Image**
  - Lead for IR
  - Use Case
  - Blocking

- **WMI Subscription Abuse**
  - Lead for IR
  - Use Case

- **Anomalous RDP Traffic**
  - Lead for IR
  - Use Case
  - Whitelisting

A.Tellar or Alice Tellar
Bring It All Together
Bring It All Together – Team Collaboration

CyberDefence Program

Threat Intelligence

Hunting

Incident Response

Create Incident to investigate Alice Tellar’s system
Create Incident to investigate Alice Tellar’s system

Uncovered a Phishing email, RDP lateral movement, and exfiltration of sensitive documents
Bring It All Together – Team Collaboration

Quimney threat group
Financially motivated

Create Incident to investigate Alice Tellar’s system

Uncovered a Phishing email, RDP lateral movement, and exfiltration of sensitive documents

CyberDefence Program

Threat Intelligence

Hunting

Incident Response

Copyright of Shell International B.V.

September 2019

46
Bring It All Together – Team Collaboration

CyberDefence Program

Threat Intelligence

Hunting

Incident Response

Quimney threat group
Financially motivated

Create Incident to investigate Alice Tellar’s system

Uncovered a Phishing email, RDP lateral movement, and exfiltration of sensitive documents

Copyright of Shell International B.V.

September 2019

47
Bring It All Together – Threat Group Correlation

- Use a template to share indicators between teams, for example:
  1. Incident Response collects and shares indicators with the Threat Intelligence team
  2. Threat Intelligence team uses these to correlate the Incident to a Threat Group

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Indicator_Type</th>
<th>ATT&amp;CK</th>
<th>Confidence Rating</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
## Bring It All Together – Threat Group Correlation

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Indicator_Type</th>
<th>ATT&amp;CK</th>
<th>Confidence Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>wonderlanddcorp.com</td>
<td>Network</td>
<td>Establish and Maintain Infrastructure</td>
<td>High</td>
</tr>
</tbody>
</table>

**Remarks**

Domain was used by phishing payload to download additional payloads onto compromised system.
### Bring It All Together – Threat Group Correlation

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Indicator_Type</th>
<th>ATT&amp;CK</th>
<th>Confidence Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>powershe11.exe -mx9 -t7z C:\windows\inf\en-gb\foi_1.7z C:\windows\inf\en-gb*. * - pfFireknight1337!</td>
<td>Procedure</td>
<td>Exfiltration, Data Compressed</td>
<td>High</td>
</tr>
</tbody>
</table>

### Remarks

Adversary staged files under C:\windows\inf under a created a folder “en-gb” to blend in. Recovery of archive was possible and contained sensitive files.
## Bring It All Together – Threat Group Correlation

<table>
<thead>
<tr>
<th>Indicator</th>
<th>Indicator_Type</th>
<th>ATT&amp;CK</th>
<th>Confidence Rating</th>
</tr>
</thead>
<tbody>
<tr>
<td>C:\windows\net1 -v -sP 192.168.1.1/24</td>
<td>Procedure</td>
<td>Discovery</td>
<td>High</td>
</tr>
<tr>
<td>C:\windows\net1 -v -sTV -top-ports 100 192.168.1.103</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

### Remarks

Adversary leveraged a renamed `nmap` executable to perform network scans against a machine.
Conclusion

“That’s the kind of house I live in. And I hope we never leave it.”

- Dr. Seuss
**Takeaways**

Discover new techniques, procedures, adversaries.

Leverage existing tools, data/logs.

Know the many hunting outcomes.

Foster collaboration within your existing CyberDefence functions.

Define what hunting means for your organization.
I was told…

…there would be Q&A
References

• MITRE ATT&CK Framework: https://attack.mitre.org/
• The Intelligence Cycle: https://www.sans.org/reading-room/whitepapers/threatintelligence/cyber-threat-intelligence-support-incident-handling-38150
• Operational Levels of Cyber Intelligence: https://www.insaonline.org/wp-content/uploads/2017/04/INSA_OperCyberIntelligence_WP.pdf
• Domain Fronting: https://blog.cobaltstrike.com/2017/02/06/high-reputation-redirections-and-domain-fronting/
• Malware Sample Delivered Through UDF Image: https://isc.sans.edu/forums/diary/Malware+Sample+Delivered+Through+UDF+Image/24854/
• SIGMA Rules (Florian Roth) — https://github.com/Neo23x0/sigma/wiki/Specification