102 – Windows Lateral Movement

Team Information

Team Name : LuckyVicky

Team Member : Eungchang Lee, Hyun Yi, Juho Heo, Dongkyu Lee

Email Address : dfc_luckyvicky@googlegroups.com

Instructions

Description Monitoring or security solution accounts used by companies are typically part of the administrator group, and for the convenience of management, these accounts are registered with the same password across all systems. Attackers may steal the credentials of these accounts and use lateral movement to access critical systems. To prevent such situations, it is important to manage credentials and prevent password reuse, but it is also crucial to preemptively block various Lateral Movement methods.

Target	Hash (MD5)
102_target	0cb76ba40ee240fcd236c752aaad9e07

Questions

- The attacker used various lateral movement methods to access the system. Analyze the given artifacts to identify the times (yyyy-mm-dd hh:mm:ss UTC+0) when the attacker accessed the system and the lateral movement methods (tools, protocols, etc.: exact matches are not necessary as long as the keywords are included) they used during the access. (100 points)

Teams must:

- Develop and document the step-by-step approach used to solve this problem to allow another examiner to replicate team actions and

results.

- Specify all tools used in deriving the conclusion(s).

Tools used:

Name:	KAPE	**Publisher:**	Kroll
Version:	1.3.0.2		
URL:	https://www.kroll.com/en/services/cyber-risk/incident-response-litigation-support/kroll-artifact-parser-extractor-kape		

Name:	Registry Explorer	**Publisher:**	Eric Zimmerman
Version:	2.0.0.0		
URL:	https://ericzimmerman.github.io/		

Name:	EvtxECmd	**Publisher:**	Eric Zimmerman
Version:	1.5.0.0		
URL:	https://ericzimmerman.github.io/		

Name:	PECmd	**Publisher:**	Eric Zimmerman
Version:	1.5.0.0		
URL:	https://ericzimmerman.github.io/		

Name:	NTFS Log Tracker	**Publisher:**	blueangel
Version:	1.71		
URL:	https://sites.google.com/site/forensicnote/ntfs-log-tracker		

Name:	REGA	**Publisher:**	DFRC
Version:	1.5.3		
URL:	https://dfrc.korea.ac.kr/infra_dfrc_tools		

Name:	impacket	Publisher:	Fortra
Version:	0.11.0		
URL:	https://github.com/fortra/impacket/tree/master		

Name:	Evil-winRM	Publisher:	Hackplayers
Version:	3.5		
URL:	https://github.com/Hackplayers/evil-winrm		

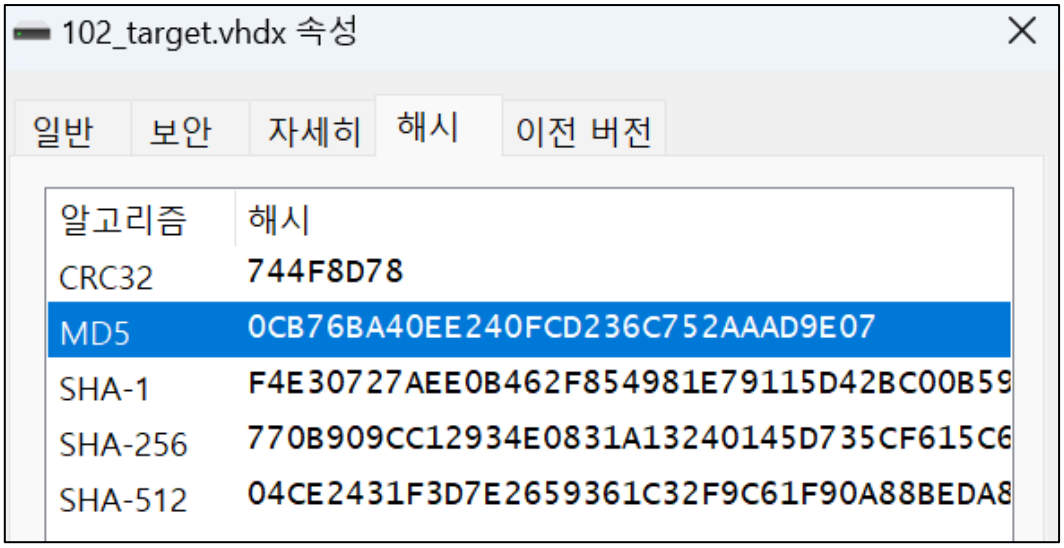
Name:	xfreerdp	Publisher:	Freerdp
Version:	3.8.0		
URL:	https://github.com/FreeRDP/FreeRDP		

VM used:

Name:	Ubuntu	ip	192.168.140.19
Version:	22.04		
URL:	https://releases.ubuntu.com/jammy/		

Name:	Kali-linux	ip	192.168.140.28
Version:	2024.2		
URL:	https://www.kali.org/get-kali/		

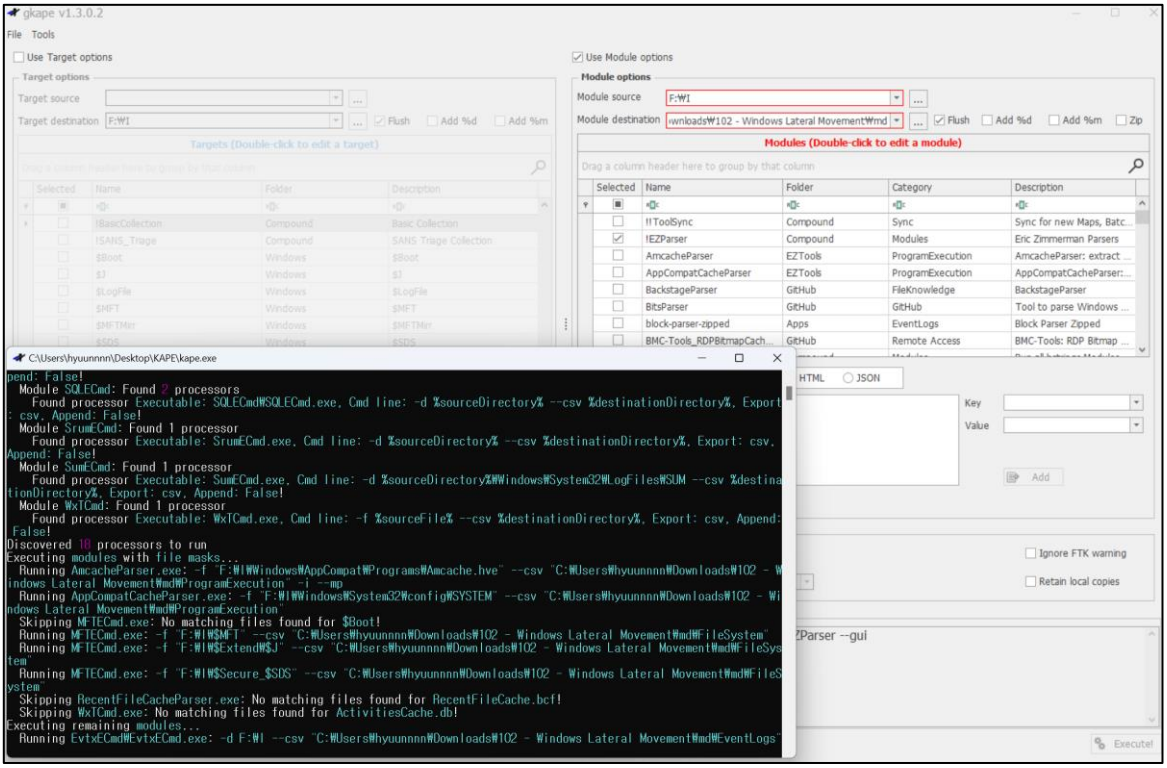
Step-by-step methodology:



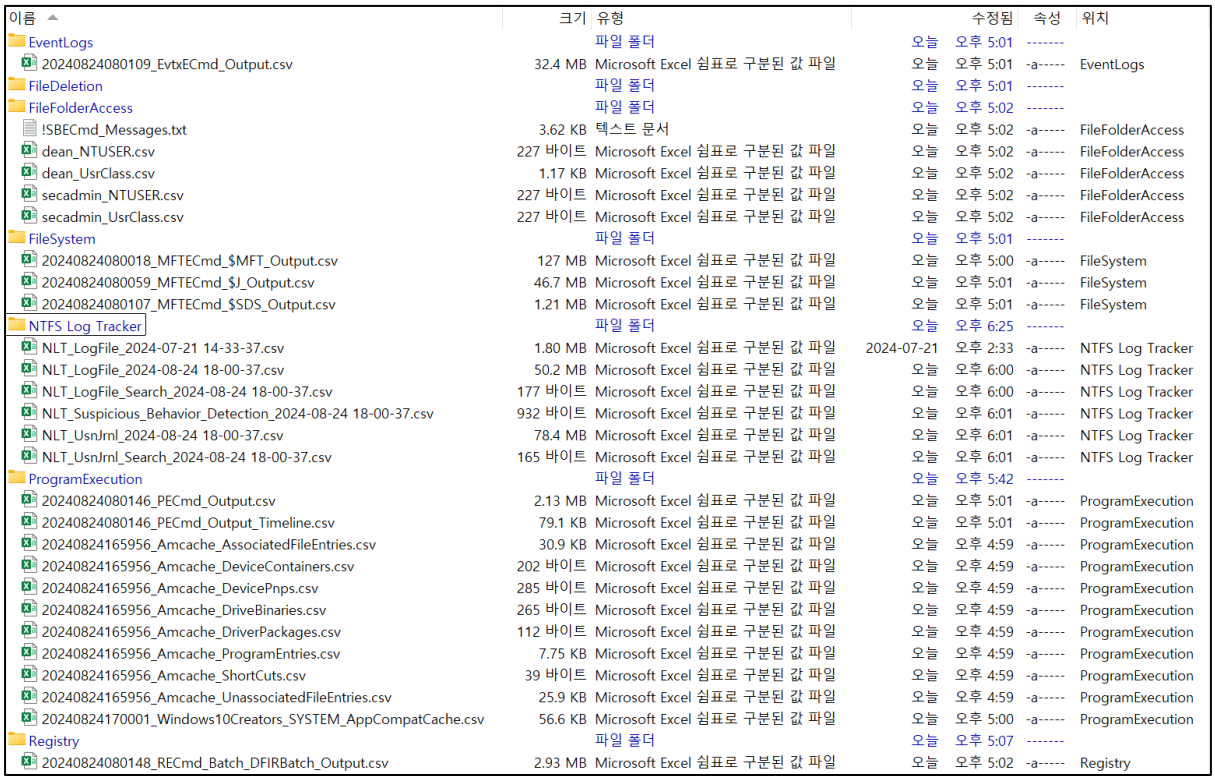
[그림 1] 102_target.vhdx 파일의 md5 해시 값 확인

문제로 주어진 102_target.vhdx 파일의 MD5 해시 값이 일치함을 확인하였습니다.

- The attacker used various lateral movement methods to access the system. Analyze the given artifacts to identify the times (yyyy-mm-dd hh:mm:ss UTC+0) when the attacker accessed the system and the lateral movement methods (tools, protocols, etc.: exact matches are not necessary as long as the keywords are included) they used during the access. (100 points)



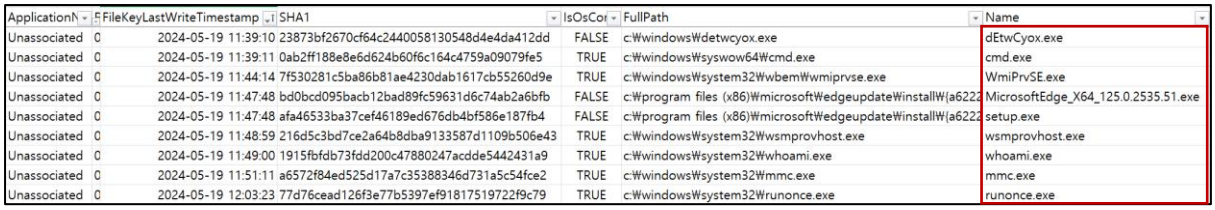
[그림 2] KAPE 도구를 사용하여 분석하는 화면



이름	크기	유형	수정됨	속성	위치
EventLogs		파일 폴더	오늘 오후 5:01	-------	
20240824080109_EvtxECmd_Output.csv	32.4 MB	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 5:01	-a-----	EventLogs
FileDeletion		파일 폴더	오늘 오후 5:01	-------	
FileFolderAccess		파일 폴더	오늘 오후 5:02	-------	
!SBECmd_Messages.txt	3.62 KB	텍스트 문서	오늘 오후 5:02	-a-----	FileFolderAccess
dean_NTUSER.csv	227 바이트	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 5:02	-a-----	FileFolderAccess
dean_UsrClass.csv	1.17 KB	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 5:02	-a-----	FileFolderAccess
secadmin_NTUSER.csv	227 바이트	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 5:02	-a-----	FileFolderAccess
secadmin_UsrClass.csv	227 바이트	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 5:02	-a-----	FileFolderAccess
FileSystem		파일 폴더	오늘 오후 5:01	-------	
20240824080018_MFTECmd_$MFT_Output.csv	127 MB	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 5:00	-a-----	FileSystem
20240824080059_MFTECmd_$J_Output.csv	46.7 MB	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 5:01	-a-----	FileSystem
20240824080107_MFTECmd_$SDS_Output.csv	1.21 MB	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 5:01	-a-----	FileSystem
NTFS Log Tracker		파일 폴더	오늘 오후 6:25	-------	
NLT_LogFile_2024-07-21 14-33-37.csv	1.80 MB	Microsoft Excel 쉼표로 구분된 값 파일	2024-07-21 오후 2:33	-a-----	NTFS Log Tracker
NLT_LogFile_2024-08-24 18-00-37.csv	50.2 MB	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 6:00	-a-----	NTFS Log Tracker
NLT_LogFile_Search_2024-08-24 18-00-37.csv	177 바이트	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 6:00	-a-----	NTFS Log Tracker
NLT_Suspicious_Behavior_Detection_2024-08-24 18-00-37.csv	932 바이트	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 6:01	-a-----	NTFS Log Tracker
NLT_UsnJrnl_2024-08-24 18-00-37.csv	78.4 MB	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 6:01	-a-----	NTFS Log Tracker
NLT_UsnJrnl_Search_2024-08-24 18-00-37.csv	165 바이트	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 6:01	-a-----	NTFS Log Tracker
ProgramExecution		파일 폴더	오늘 오후 5:42	-------	
20240824080146_PECmd_Output.csv	2.13 MB	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 5:01	-a-----	ProgramExecution
20240824080146_PECmd_Output_Timeline.csv	79.1 KB	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 5:01	-a-----	ProgramExecution
20240824165956_Amcache_AssociatedFileEntries.csv	30.9 KB	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 4:59	-a-----	ProgramExecution
20240824165956_Amcache_DeviceContainers.csv	202 바이트	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 4:59	-a-----	ProgramExecution
20240824165956_Amcache_DevicePnps.csv	285 바이트	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 4:59	-a-----	ProgramExecution
20240824165956_Amcache_DriveBinaries.csv	265 바이트	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 4:59	-a-----	ProgramExecution
20240824165956_Amcache_DriverPackages.csv	112 바이트	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 4:59	-a-----	ProgramExecution
20240824165956_Amcache_ProgramEntries.csv	7.75 KB	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 4:59	-a-----	ProgramExecution
20240824165956_Amcache_ShortCuts.csv	39 바이트	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 4:59	-a-----	ProgramExecution
20240824165956_Amcache_UnassociatedFileEntries.csv	25.9 KB	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 4:59	-a-----	ProgramExecution
20240824170001_Windows10Creators_SYSTEM_AppCompatCache.csv	56.6 KB	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 5:00	-a-----	ProgramExecution
Registry		파일 폴더	오늘 오후 5:07	-------	
20240824080148_RECmd_Batch_DFIRBatch_Output.csv	2.93 MB	Microsoft Excel 쉼표로 구분된 값 파일	오늘 오후 5:02	-a-----	Registry

[그림 3] KAPE 도구를 사용하여 분석한 결과 CSV 파일

KAPE의 module에 정의된 각 아티팩트에 맞는 분석 도구가 자동으로 실행되며, [그림 3]과 같이 분석이 완료된 CSV 파일들이 생성됩니다. NTFS Log Tracker 도구는 별도로 실행하여 CSV 파일들을 추출하였습니다



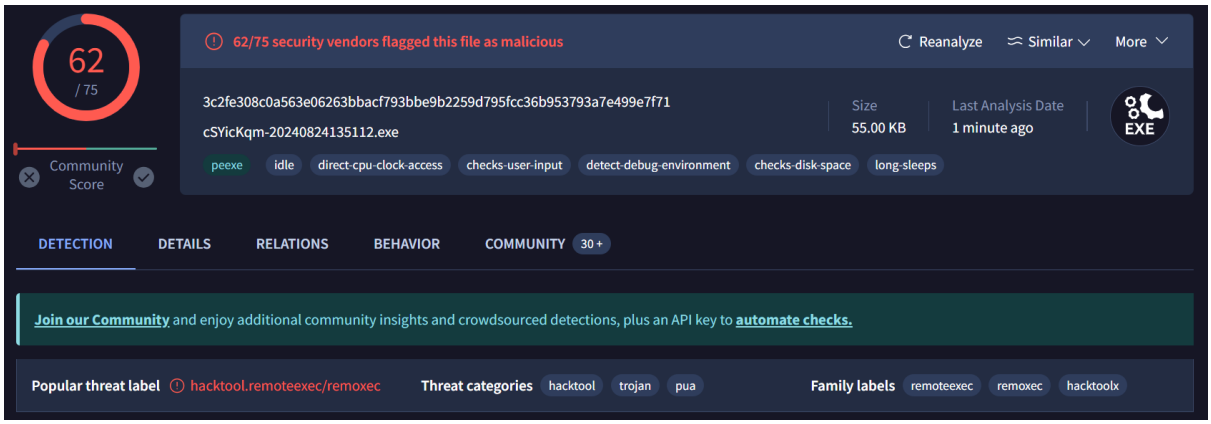
ApplicationN	F	FileKeyLastWriteTimestamp	SHA1	IsOsCor	FullPath	Name
Unassociated	0	2024-05-19 11:39:10	23873bf2670cf64c2440058130548d4e4da412dd	FALSE	c:₩windows₩detwcyox.exe	dEtwCyox.exe
Unassociated	0	2024-05-19 11:39:11	0ab2ff188e8e6d624b60f6c164c4759a09079fe5	TRUE	c:₩windows₩syswow64₩cmd.exe	cmd.exe
Unassociated	0	2024-05-19 11:44:14	7f530281c5ba86b81ae4230dab1617cb55260d9e	TRUE	c:₩windows₩system32₩wbem₩wmiprvse.exe	WmiPrvSE.exe
Unassociated	0	2024-05-19 11:47:48	bd0bcd095bacb12bad89fc59631d6c74ab2a6bfb	FALSE	c:₩program files (x86)₩microsoft₩edgeupdate₩install₩{a6222	MicrosoftEdge_X64_125.0.2535.51.exe
Unassociated	0	2024-05-19 11:47:48	afa46533ba37cef46189ed676db4bf586e187fb4	FALSE	c:₩program files (x86)₩microsoft₩edgeupdate₩install₩{a6222	setup.exe
Unassociated	0	2024-05-19 11:48:59	216d5c3bd7ce2a64b8dba9133587d1109b506e43	TRUE	c:₩windows₩system32₩wsmprovhost.exe	wsmprovhost.exe
Unassociated	0	2024-05-19 11:49:00	1915fbfdb73fdd200c47880247acdde5442431a9	TRUE	c:₩windows₩system32₩whoami.exe	whoami.exe
Unassociated	0	2024-05-19 11:51:11	a6572f84ed525d17a7c35388346d731a5c54fce2	TRUE	c:₩windows₩system32₩mmc.exe	mmc.exe
Unassociated	0	2024-05-19 12:03:23	77d76cead126f3e77b5397ef91817519722f9c79	TRUE	c:₩windows₩system32₩runonce.exe	runonce.exe

[그림 4] Amcache_UnassociatedFileEntries.csv 분석 결과(UTC+0)

Amcache의 Unassociated에는 설치하지 않는 독립적인 파일의 실행 기록들이 남아있기 때문에 악성 exe 파일이 실행되었다면 이곳에서 전반적으로 유의미한 데이터를 얻을 수 있습니다. 이를 통해, 시간 순서대로 시스템에 접근하기 위한 공격자의 Lateral Movement 행동을 시간 순서대로 살펴보았습니다.

#1 Psexec.py

먼저, [그림 4]를 보면 악성으로 의심되는 dEtwCyox.exe 파일이 존재하며, SHA1값을 통해 악성 파일임을 확인할 수 있었습니다. (SHA1: 23873bf2670cf64c2440058130548d4e4da412dd)



[그림 5] Virustotal 검사 결과

TimeStamp(UTC 0)	USN	File/Directory Name	FullPath	EventInfo
2024-05-19 11:39:09	273923208	dEtwCyox.exe	₩Windows₩dEtwCyox.exe	File_Created
2024-05-19 11:39:09	273923296	dEtwCyox.exe	₩Windows₩dEtwCyox.exe	File_Created / Data_Added
2024-05-19 11:39:09	273923384	dEtwCyox.exe	₩Windows₩dEtwCyox.exe	File_Created / Data_Added / File_Closed
2024-05-19 11:39:20	273923472	DETWCYOX.EXE-D166D79A.pf	₩Windows₩Prefetch₩DETWCYOX.EXE-D166D79A.pf	File_Created
2024-05-19 11:39:20	273923584	DETWCYOX.EXE-D166D79A.pf	₩Windows₩Prefetch₩DETWCYOX.EXE-D166D79A.pf	File_Created / Data_Added
2024-05-19 11:39:20	273923696	DETWCYOX.EXE-D166D79A.pf	₩Windows₩Prefetch₩DETWCYOX.EXE-D166D79A.pf	File_Created / Data_Added / File_Closed
2024-05-19 11:39:20	273923808	CMD.EXE-6D6290C5.pf	₩Windows₩Prefetch₩CMD.EXE-6D6290C5.pf	File_Created
2024-05-19 11:39:20	273923912	CMD.EXE-6D6290C5.pf	₩Windows₩Prefetch₩CMD.EXE-6D6290C5.pf	File_Created / Data_Added
2024-05-19 11:39:20	273924096	CMD.EXE-6D6290C5.pf	₩Windows₩Prefetch₩CMD.EXE-6D6290C5.pf	File_Created / Data_Added / File_Closed

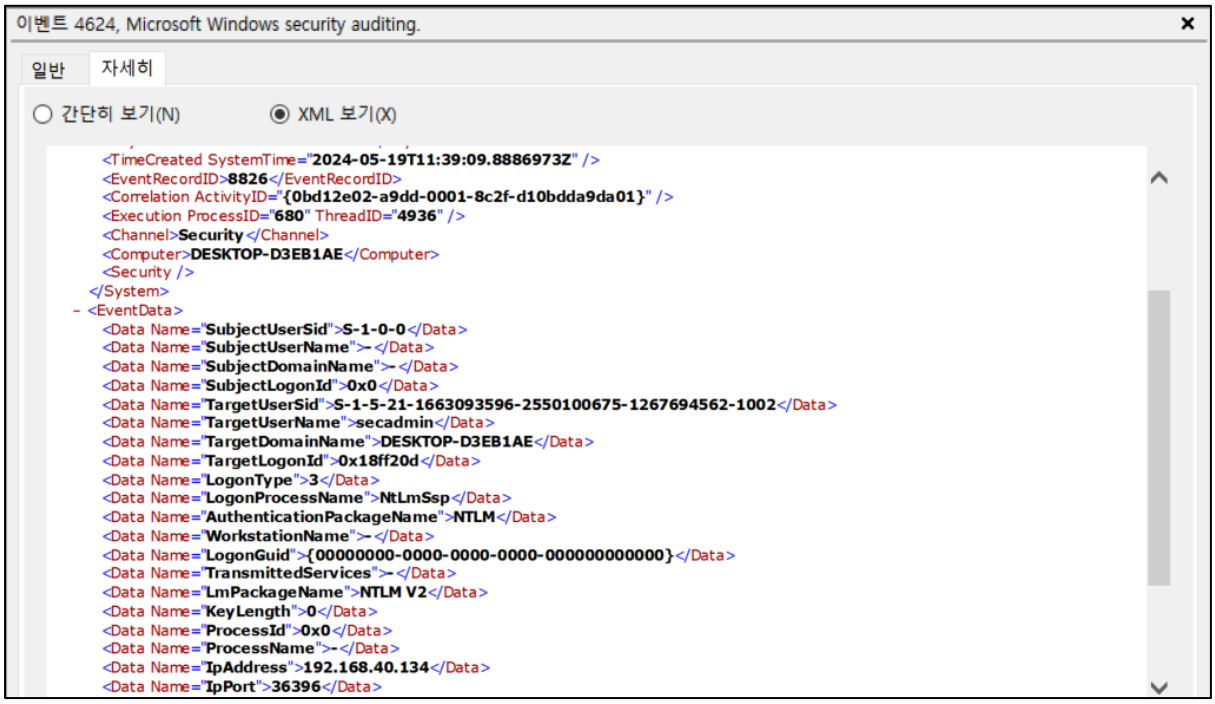
[그림 6] NTFS 로그에 기록된 dEtwCyox.exe 파일 생성 로그(UTC+0)

TimeStamp(UTC 0)	USN	File/Directory Name	FullPath	EventInfo
2024-05-19 11:40:53	273970824	dEtwCyox.exe	₩Windows₩dEtwCyox.exe	File_Closed / File_Deleted

[그림 7] NTFS 로그에 기록된 dEtwCyox.exe 파일 삭제 로그(UTC+0)

[그림 8] 이벤트 로그에 기록된 서비스 등록 로그 – System Event ID 7045

또한, NTFS 로그와 이벤트 로그를 활용하여 교차 검증을 진행한 결과 dEtwCyox.exe 파일이 생성됨과 동시에 서비스(이벤트 ID: 7045)에 등록된 것을 확인할 수 있습니다.



[그림 9] 이벤트 로그에 기록된 로그인 로그 – Security Event ID 4624

그리고, 같은 시각에 192.168.40.134 ip와 36396 포트로 secadmin에 로그인한 흔적이 발견되었습니다.

Timestamp (UTC+9)	TargetUserName	RemoteHost	RemotePort	Logon ProcessName	추정되는 impacket
2024-05-19 20:39:09	secadmin	192.168.40.134	36396	NtLmSsp	psexec.py
2024-05-19 20:39:09	secadmin	192.168.40.134	36402	NtLmSsp	psexec.py
2024-05-19 20:39:09	secadmin	192.168.40.134	36404	NtLmSsp	psexec.py
2024-05-19 20:39:09	secadmin	192.168.40.134	36410	NtLmSsp	psexec.py

[표 1] psexec.py를 통해 접근한 기록으로 추정되는 id 4624 이벤트 로그

또한, Security 이벤트 로그에서 psexec.py를 통해 같은 사설 ip대역에서 remote ip와 port로 접속을 수행한 것으로 보이는 로그를 정리하면 위와 같습니다.

DHCP Subnet Mask	DHCP Server	DHCP Name Server	DHCPIP Address	DHCP Default Gateway	Enabled DHCP
255.255.255.0	192.168.40.254	192.168.40.2	192.168.40.142	192.168.40.2	☑

[그림 10] 사용자 PC가 사용하고 있는 IP

SYSTEM hive의 ROOT₩ControlSet001₩Services₩Tcpip₩Parameters₩Interfaces₩{18f58deb-f0f6-48a0-b68a-a8be9e448fb0} 경로에서 위 그림과 같이 피해자 PC에서 사용 중인 IP를 조회해보면, DHCP로 설정된 192.168.40.142인 것을 알 수 있습니다. 따라서, [그림 9]에서 확인된 192.168.40.134는 공격자 IP로 추정할 수 있습니다.

```
28  class ServiceInstall:
29      def __init__(self, SMBObject, exeFile, serviceName='', binary_service_name=None):
30          self._rpctransport = 0
31          self.__service_name = serviceName if len(serviceName) > 0  else ''.join([random.choice(string.ascii_letters) for i in range(4)])
32
33          if binary_service_name is None:
34              self.__binary_service_name = ''.join([random.choice(string.ascii_letters) for i in range(8)]) + '.exe'
35          else:
36              self.__binary_service_name = binary_service_name
37
38          self.__exeFile = exeFile
```

[그림 11] serviceinstall.py에 존재하는 이름을 설정하는 코드

그리고, 이러한 패턴은 impacket 도구 중 하나인 **psexec.py** 와 매우 유사하며, psexec.py 에서 import 하여 사용하는 **serviceinstall.py** 를 보면 위 사진과 동일한 형태의 랜덤 이름을 설정하는 코드가 존재합니다. 등록된 uulG와 같은 4자리 서비스 이름과 %systemroot%dEtwCyox.exe 와 같은 서비스 파일에 대한 이름 생성은 다음의 두 url에서 확인이 가능합니다.

https://github.com/fortra/impacket/blob/master/examples/psexec.py
https://github.com/fortra/impacket/blob/master/impacket/examples/serviceinstall.py

따라서, 192.168.40.134라는 ip를 가진 공격자는 사설 ip대역임을 고려할 때 내부 PC 장악 후, psexec.py를 실행함으로써 smb 프로토콜을 통해 Lateral Movement를 2024-05-19 20:39:09(UTC+9) 시각에 시도한 것으로 보입니다.

#2 Wmiexec.py

공격자의 다음 lateral movement 행위를 살펴보겠습니다.

TimeStamp(UTC 0)	USN	File/Directory Name	FullPath	EventInfo
2024-05-19 11:44:14	274095024	__1716119051.3501575	₩Windows₩__1716119051.3501575	File_Created
2024-05-19 11:44:14	274095128	__1716119051.3501575	₩Windows₩__1716119051.3501575	File_Created / File_Closed
2024-05-19 11:44:14	274095232	CMD.EXE-0BD30981.pf	₩Windows₩Prefetch₩CMD.EXE-0BD30981.pf	Data_Truncated
2024-05-19 11:44:14	274095336	CMD.EXE-0BD30981.pf	₩Windows₩Prefetch₩CMD.EXE-0BD30981.pf	Data_Added / Data_Truncated
2024-05-19 11:44:14	274095440	CMD.EXE-0BD30981.pf	₩Windows₩Prefetch₩CMD.EXE-0BD30981.pf	Data_Added / Data_Truncated / File_Closed
2024-05-19 11:44:14	274095544	CONHOST.EXE-0C6456FB.pf	₩Windows₩Prefetch₩CONHOST.EXE-0C6456FB.pf	Data_Truncated
2024-05-19 11:44:14	274095656	CONHOST.EXE-0C6456FB.pf	₩Windows₩Prefetch₩CONHOST.EXE-0C6456FB.pf	Data_Added / Data_Truncated
2024-05-19 11:44:14	274095768	CONHOST.EXE-0C6456FB.pf	₩Windows₩Prefetch₩CONHOST.EXE-0C6456FB.pf	Data_Added / Data_Truncated / File_Closed
2024-05-19 11:44:14	274095880	__1716119051.3501575	₩Windows₩__1716119051.3501575	File_Closed / File_Deleted
2024-05-19 11:44:14	274095984	__1716119051.3501575	₩Windows₩__1716119051.3501575	File_Created
2024-05-19 11:44:14	274096128	__1716119051.3501575	₩Windows₩__1716119051.3501575	File_Created / Data_Added
2024-05-19 11:44:14	274096232	__1716119051.3501575	₩Windows₩__1716119051.3501575	File_Created / Data_Added / File_Closed
2024-05-19 11:44:14	274096336	CMD.EXE-0BD30981.pf	₩Windows₩Prefetch₩CMD.EXE-0BD30981.pf	Data_Truncated
2024-05-19 11:44:14	274096440	CMD.EXE-0BD30981.pf	₩Windows₩Prefetch₩CMD.EXE-0BD30981.pf	Data_Added / Data_Truncated
2024-05-19 11:44:14	274096544	CMD.EXE-0BD30981.pf	₩Windows₩Prefetch₩CMD.EXE-0BD30981.pf	Data_Added / Data_Truncated / File_Closed
2024-05-19 11:44:14	274096648	__1716119051.3501575	₩Windows₩__1716119051.3501575	File_Closed / File_Deleted

[그림 12] NTFS 로그에 기록된 __1716119051.3501575 생성 및 삭제 로그 – 1(UTC+0)

TimeStamp(UTC 0)	USN	File/Directory Name	FullPath	EventInfo
2024-05-19 11:44:16	274097080	__1716119051.3501575	₩Windows₩__1716119051.3501575	File_Created
2024-05-19 11:44:16	274097184	__1716119051.3501575	₩Windows₩__1716119051.3501575	File_Created / Data_Added
2024-05-19 11:44:16	274097288	__1716119051.3501575	₩Windows₩__1716119051.3501575	File_Created / Data_Added / File_Closed
2024-05-19 11:44:16	274097392	NTUSER.DAT	₩Users₩dean₩NTUSER.DAT	Data_Overwritten
2024-05-19 11:44:16	274097472	WHOAMI.EXE-9D378AFE.pf	₩Windows₩Prefetch₩WHOAMI.EXE-9D378AFE.pf	File_Created
2024-05-19 11:44:16	274097576	WHOAMI.EXE-9D378AFE.pf	₩Windows₩Prefetch₩WHOAMI.EXE-9D378AFE.pf	File_Created / Data_Added
2024-05-19 11:44:16	274097680	WHOAMI.EXE-9D378AFE.pf	₩Windows₩Prefetch₩WHOAMI.EXE-9D378AFE.pf	File_Created / Data_Added / File_Closed
2024-05-19 11:44:16	274097784	UsrClass.dat	₩Users₩dean₩AppData₩Local₩Microsoft₩Windows₩UsrClass.dat	Data_Overwritten
2024-05-19 11:44:16	274097872	Amcache.hve	₩Windows₩appcompat₩Programs₩Amcache.hve	Data_Overwritten
2024-05-19 11:44:16	274097960	settings.dat	₩Users₩dean₩AppData₩Local₩Packages₩Microsoft.Windows.StartMenuExperienceHost_cw	Data_Overwritten
2024-05-19 11:44:16	274098048	settings.dat	₩Users₩dean₩AppData₩Local₩Packages₩Microsoft.Windows.Search_cw5n1h2txyewy₩Sett	Data_Overwritten
2024-05-19 11:44:16	274098136	NTUSER.DAT	₩Users₩secadmin₩NTUSER.DAT	Data_Overwritten
2024-05-19 11:44:16	274098216	settings.dat	₩Users₩dean₩AppData₩Local₩Packages₩Microsoft.Windows.ShellExperienceHost_cw5n1h	Data_Overwritten
2024-05-19 11:44:17	274098304	Microsoft-Windows-ShellCommon-StartLayoutPopulation%4(	₩Windows₩System32₩winevt₩Logs₩Microsoft-Windows-ShellCommon-StartLayoutPopula	Data_Overwritten
2024-05-19 11:44:17	274098504	__1716119051.3501575	₩Windows₩__1716119051.3501575	File_Closed / File_Deleted

[그림 13] NTFS 로그에 기록된 __1716119051.3501575 파일 생성 및 삭제 로그 – 2(UTC+0)

NTFS Log를 살펴보면, 2024-05-19 11:44:14, 2024-05-19 11:44:16(UTC+0) 시간에 __1716119051.3501575 파일의 생성과 삭제된 로그가 존재합니다. 그리고, [그림 4]의 Amcache에서 확인된 WmiPrvSE.exe 파일의 타임스탬프를 보면 2024-05-19 11:44:14로 동일한 시간대임을 확인할 수 있습니다. 이러한 패턴으로 생성 및 삭제하는 도구는 impacket 도구 중 하나인 **wmiexec.py**입니다.

```
46      OUTPUT_FILENAME = '__' + str(time.time())
47      CODEC = sys.stdout.encoding
```

[그림 14] wmiexec.py에 사용되는 파일 이름 생성 코드

```
def execute_remote(self, data, shell_type='cmd'):
    if shell_type == 'powershell':
        data = '$ProgressPreference="SilentlyContinue";' + data
        data = self.__pwsh + b64encode(data.encode('utf-16le')).decode()

    command = self.__shell + data

    if self.__noOutput is False:
        command += ' 1> ' + '\\\\127.0.0.1\\%s' % self.__share + self.__output + ' 2>&1'
    if PY2:
        self.__win32Process.Create(command.decode(sys.stdin.encoding), self.__pwd, None)
    else:
        self.__win32Process.Create(command, self.__pwd, None)
    self.get_output()
```

In this file

126 __output = '\\' + OUTPUT_FILENAME

3 References Search

In this file

274 self.__output, output_callback)

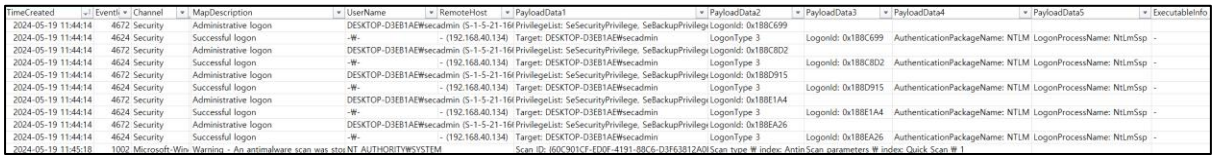
286 self.__output)

296 + self.__output + ' 2>&1'

Search for this symbol

[그림 15] wmiexec.py에 사용되는 명령 코드

위 파일의 코드는 https://github.com/fortra/impacket/blob/master/examples/wmiexec.py에서 확인할 수 있습니다.



TimeCreated	EventI	Channel	MapDescription	UserName	RemoteHost	PayloadData1	PayloadData2	PayloadData3	PayloadData4	PayloadData5	ExecutableInfo
2024-05-19 11:44:14	4672	Security	Administrative logon	DESKTOP-D3EB1AE₩secadmin	(S-1-5-21-16(	PrivilegeList: SeSecurityPrivilege, SeBackupPrivilege	LogonId: 0x1B8C699				
2024-05-19 11:44:14	4624	Security	Successful logon	-₩-	- (192.168.40.134)	Target: DESKTOP-D3EB1AE₩secadmin	LogonType 3	LogonId: 0x188C699	AuthenticationPackageName: NTLM	LogonProcessName: NtLmSsp	-
2024-05-19 11:44:14	4672	Security	Administrative logon	DESKTOP-D3EB1AE₩secadmin	(S-1-5-21-16(	PrivilegeList: SeSecurityPrivilege, SeBackupPrivilege	LogonId: 0x1B8CBD2				
2024-05-19 11:44:14	4624	Security	Successful logon	-₩-	- (192.168.40.134)	Target: DESKTOP-D3EB1AE₩secadmin	LogonType 3	LogonId: 0x188CBD2	AuthenticationPackageName: NTLM	LogonProcessName: NtLmSsp	-
2024-05-19 11:44:14	4672	Security	Administrative logon	DESKTOP-D3EB1AE₩secadmin	(S-1-5-21-16(	PrivilegeList: SeSecurityPrivilege, SeBackupPrivilege	LogonId: 0x188D915				
2024-05-19 11:44:14	4624	Security	Successful logon	-₩-	- (192.168.40.134)	Target: DESKTOP-D3EB1AE₩secadmin	LogonType 3	LogonId: 0x188D915	AuthenticationPackageName: NTLM	LogonProcessName: NtLmSsp	-
2024-05-19 11:44:14	4672	Security	Administrative logon	DESKTOP-D3EB1AE₩secadmin	(S-1-5-21-16(	PrivilegeList: SeSecurityPrivilege, SeBackupPrivilege	LogonId: 0x188E1A4				
2024-05-19 11:44:14	4624	Security	Successful logon	-₩-	- (192.168.40.134)	Target: DESKTOP-D3EB1AE₩secadmin	LogonType 3	LogonId: 0x188E1A4	AuthenticationPackageName: NTLM	LogonProcessName: NtLmSsp	-
2024-05-19 11:44:14	4672	Security	Administrative logon	DESKTOP-D3EB1AE₩secadmin	(S-1-5-21-16(	PrivilegeList: SeSecurityPrivilege, SeBackupPrivilege	LogonId: 0x188EA26				
2024-05-19 11:44:14	4624	Security	Successful logon	-₩-	- (192.168.40.134)	Target: DESKTOP-D3EB1AE₩secadmin	LogonType 3	LogonId: 0x188EA26	AuthenticationPackageName: NTLM	LogonProcessName: NtLmSsp	-
2024-05-19 11:45:18	1002	Microsoft-Win	Warning - An antimalware scan was stop	NT AUTHORITY₩SYSTEM		Scan ID: {60C901CF-ED0F-4191-88C6-D3F63812A0E	Scan type ₩ index: Antin	Scan parameters ₩ index: Quick Scan ₩ 1			

[그림 16] __1716119051.3501575 파일이 생성될 때 기록된 이벤트 로그

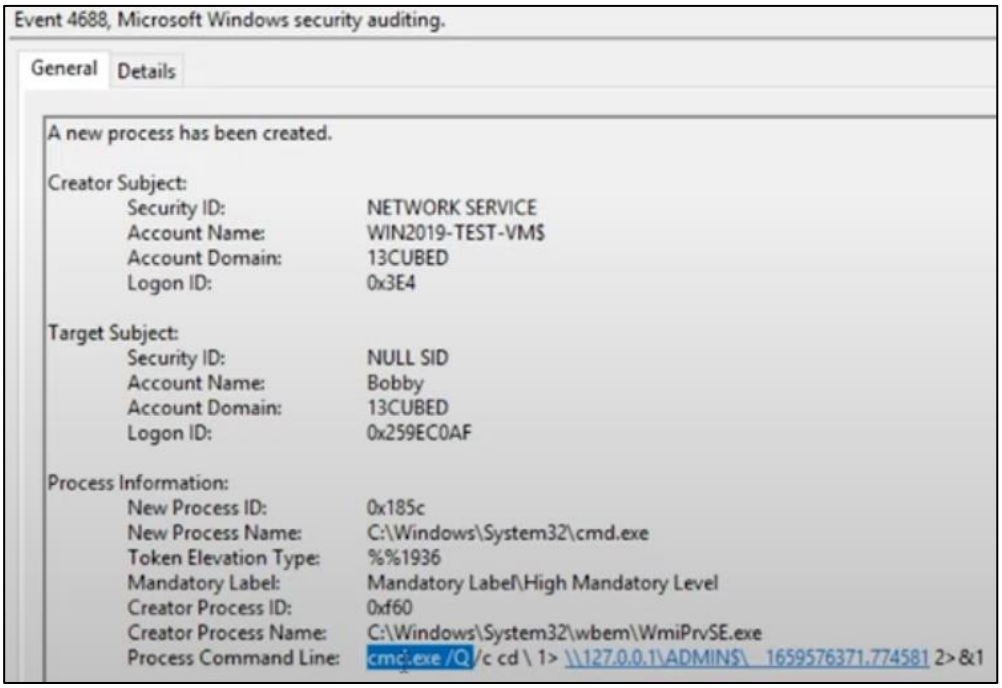
또한 해당 파일이 생성되기 전에 lateral movement에서 자주 사용되는 접근 패턴(이벤트 ID 4672, 4624)을 확인할 수 있습니다.

추가로, Security에 기록된 event id 4624인 로그를 상세히 살펴보면 다음과 같습니다.

Timestamp (UTC+9)	TargetUserName	RemoteHost	RemotePort	Logon ProcessName	추정되는 impacket
2024-05-19 20:44:14	secadmin	192.168.40.134	56226	NtLmSsp	wmiexec.py
2024-05-19 20:44:14	secadmin	192.168.40.134	43572	NtLmSsp	wmiexec.py
2024-05-19 20:44:14	secadmin	192.168.40.134	47348	NtLmSsp	wmiexec.py

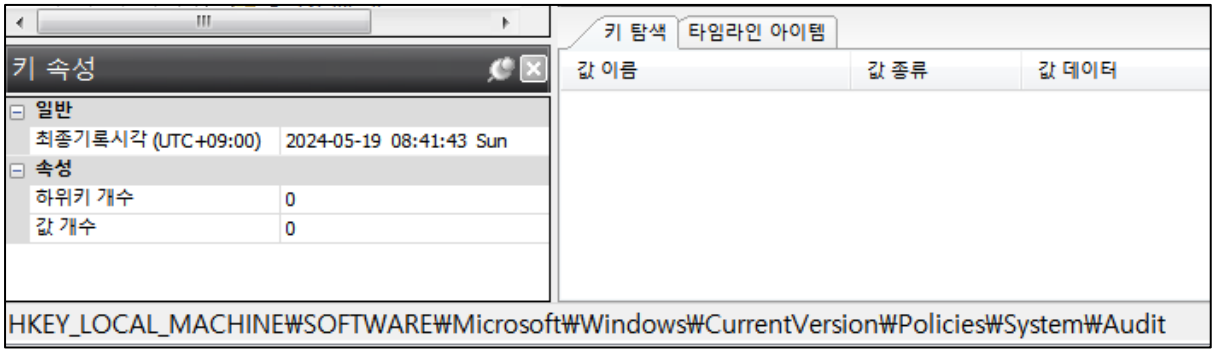
[표 2] wmiexec.py를 통해 접근한 기록으로 추정되는 id 4624 이벤트 로그

__1716119051.3501575 파일 생성 시각에 192.168.40.134 ip로부터 다양한 port를 통해 시도 요청을 했음을 알 수 있습니다.



[그림 17] Impacket Impediments - Finding Evil in Event Logs (Youtube: 13Cubed, 34:54)[1]

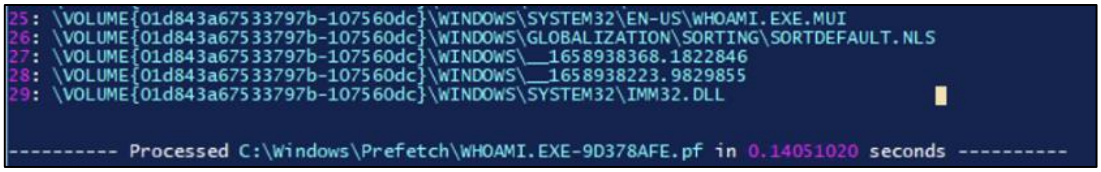
그러나, 일반적으로 wmiexec.py impacket을 통해 lateral movement를 시도할 경우, Security에 이벤트 ID가 4688이면서 프로세스 명령 실행 줄에 명령어가 기록된 이벤트가 위 그림처럼 발견되어야 합니다. 하지만, 주어진 target 파일 내 wmiexec.py를 실행한 것으로 추정되는 시각(2024-05-19 20:44:14, UTC+9)에 생성된 id가 4688인 이벤트 로그는 존재하지 않았습니다.



[그림 18] 프로세스 명령줄이 기록되는 지 활성화 여부를 알 수 있는 레지스트리 경로

이벤트 로그도 존재하지 않았지만, 위 경로에서 ProcessCreationIncludeCmdLine_Enabled 라는 레지스트리 키가 존재하고 값이 1이어야 4688 이벤트 로그에서 프로세스 명령줄이 일반적으로 기록되지만, 그러한 흔적을 찾을 수 없었습니다.

[1] https://www.youtube.com/watch?v=UMogme3rDRA



[그림 19] wmiexec.py 실행 시 whoami 프리패치에서 확인가능한 outputfile[2]

또한, PECmd를 통해 whoami 프리패치를 살펴본 결과, 위 그림처럼 wmiexec.py 실행 시 생성되는 __1716119051.3501575를 확인할 수는 없었습니다.

따라서, 로그온 성공 기록은 존재하지만 그 뒤의 행위는 선별 압수된 증거로는 정확히 파악할 수 없었으나, **__1716119051.3501575 라는 포맷의 파일 생성을 수행하는 impacket은 wmiexec.py**뿐이고, amcache에 기록된 WmiPrvSe.exe 시각을 통해 192.168.40.134라는 ip를 가진 공격자가 이번에는 wmiexec.py사용하여 WMI 프로토콜을 통해 Lateral Movement를 2024-05-19 20:44:14(UTC+9) 시각에 시도하였다는 것을 알 수 있습니다.

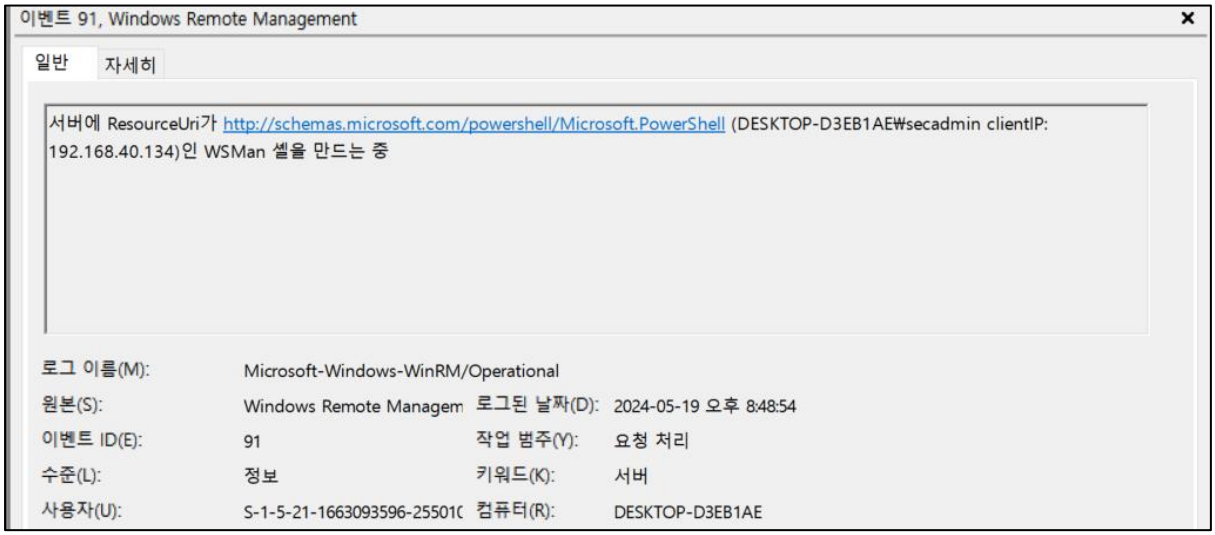
[2] https://www.crowdstrike.com/blog/how-to-detect-and-prevent-impackets-wmiexec/

#3 Evil-winRM

공격자의 다음 주요 lateral movement 행위를 살펴보겠습니다.

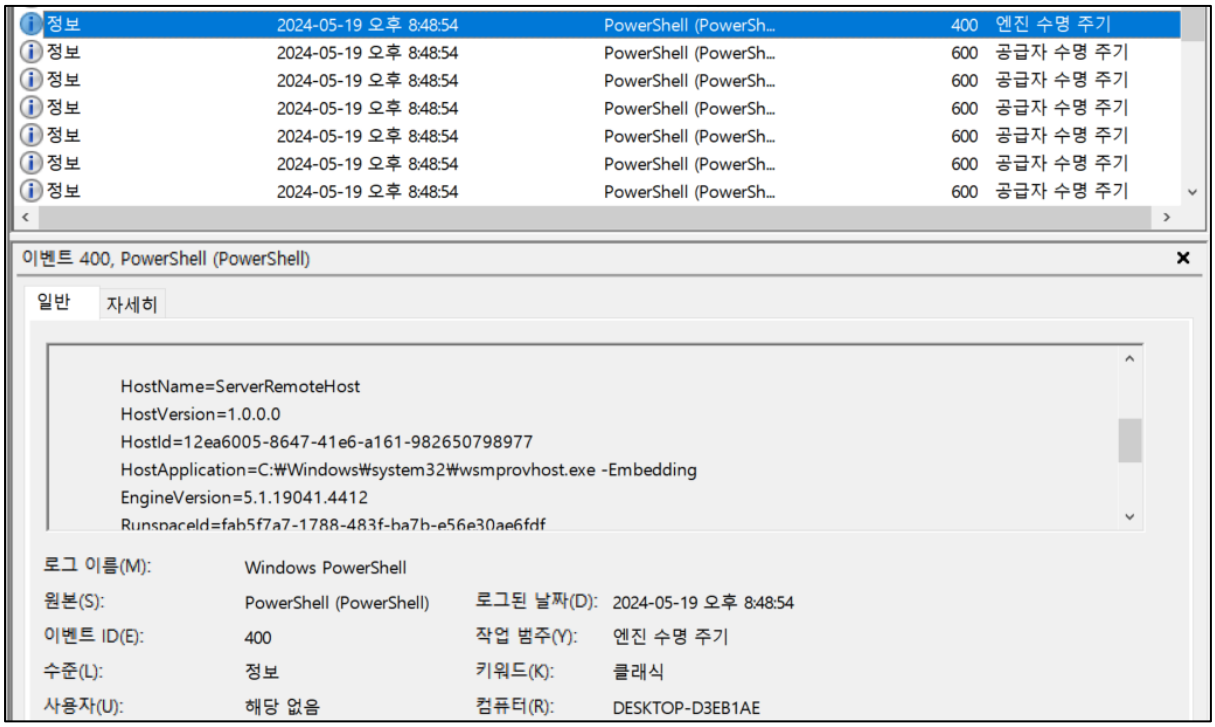
[그림 4]에 존재하는 timestamp가 11:48:59인 wsmprovhost.exe에 대한 기록을 살펴보았습니다.

[그림 20] Security 이벤트로그에 기록된 로그온 이벤트



[그림 21] WinRM 이벤트로그에 기록된 event id가 91인 로그

먼저, 2024-05-19 20:48:53에 공격자 ip로 추정되는 192.168.40.134가 32876 포트로 로그온을 성공했고, 20:48:54에 event id가 91인 winRM 이벤트로그가 기록된 것을 알 수 있습니다. 그림에는 없지만 로그온 이벤트는 20:50:01에도 기록이 되었습니다.



[그림 22] Windows Powershell 이벤트 로그

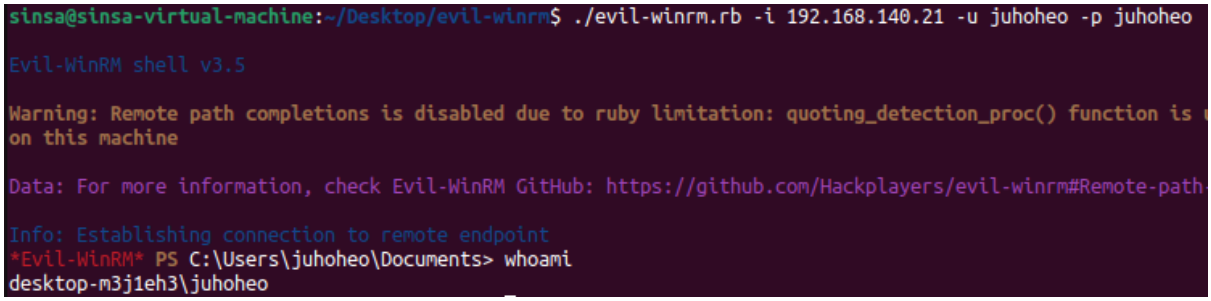
그리고, 동일 시각에 Windows Powershell 이벤트 로그에도 wsmprovhost.exe -Embedding 이라는 HostApplication 기록이 여러 개 남아있는 것을 확인할 수 있습니다.

Provider	EventID	Description	Comments
PowerShell	400	Engine state is changed from None to Available.	HostName is set to ServerRemoteHost
WinRM/Operational	91	Creating WSMan shell on server with ResourceUri: %1	UTF16LE data correspond to `http://schemas.microsoft.com/powershell/Microsoft.PowerShell`

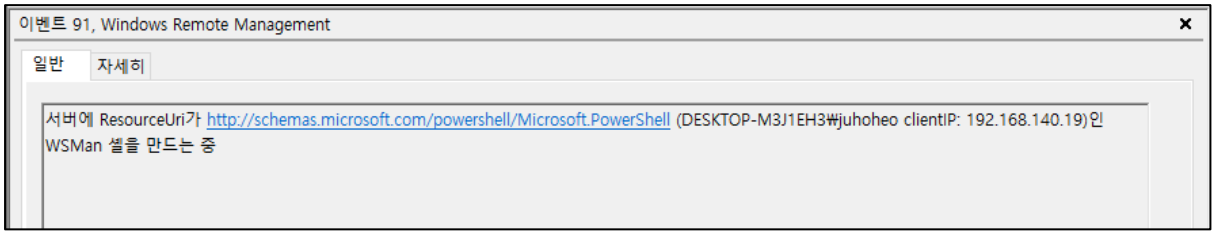
[그림 23] evil-winRM 실행 시 이벤트로그에 기록되는 아티팩트

해당 windows 헌팅 참고자료에 따르면, [3] **evil-winRM** 도구 실행 시 winRM에서 event id 91인 이벤트로그와 windows powershell에서 event id 400인 이벤트로그 흔적이 기록된다는 것을 알 수 있습니다.

[3] https://www.synacktiv.com/publications/traces-of-windows-remote-command-execution.html



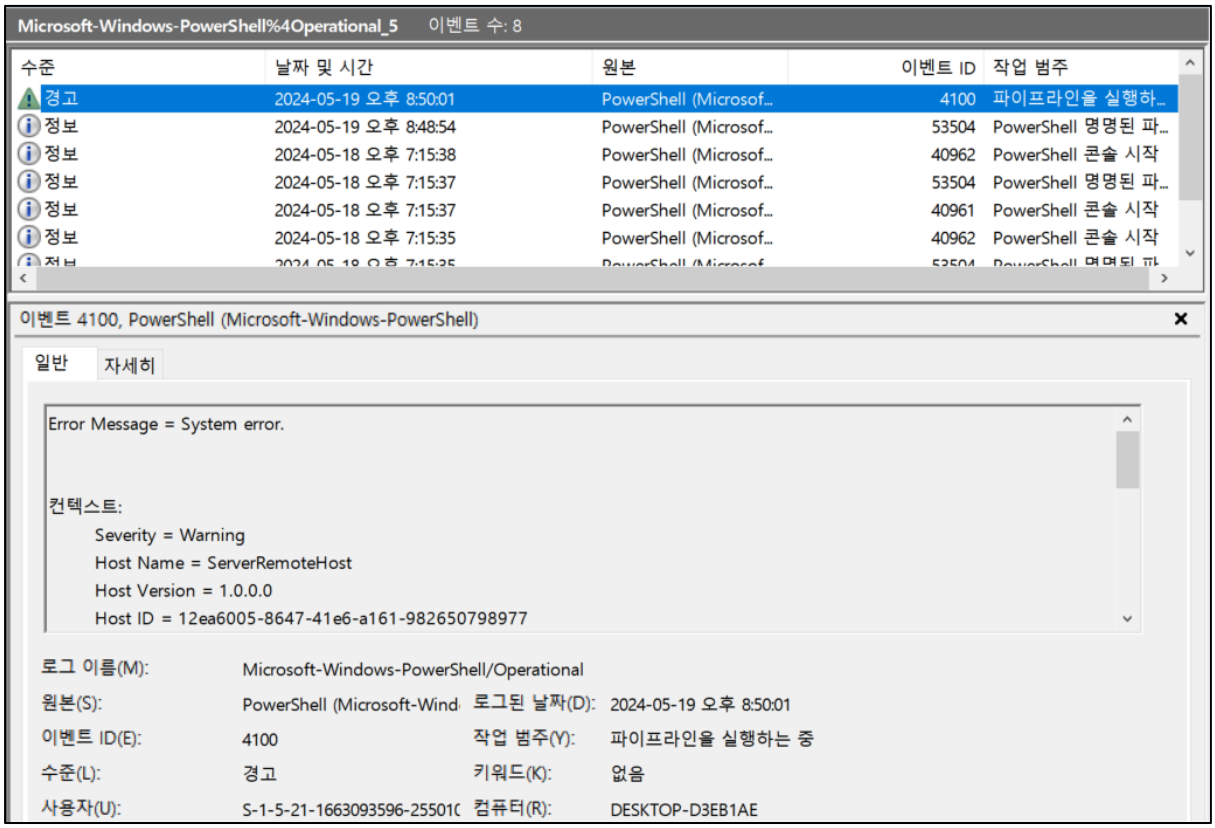
[그림 24] evil-winRM 아티팩트 검증 테스트 - 1



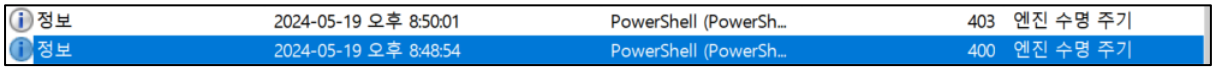
[그림 25] evil-winRM 아티팩트 검증 테스트 - 2

[그림 26] evil-winRM 아티팩트 검증 테스트 - 3

또한, 위 검증 테스트를 통해서 target파일에서 남은 이벤트 로그와 일치한다는 점을 알 수 있습니다.



[그림 27] 피해자 PC에 기록된 winRM 접속 흔적 (Microsoft windows powershell/Operational)



[그림 28] 피해자 PC에 기록된 winRM 접속 흔적 (windows powershell)

그리고, Security Eventlog 로그온 기록에는 20:48:53, 20:50:01경에 접속한 기록이 남아 있었고, 해당 기록은 위 그림들과 같이 Microsoft-Windows-Powershell/Operational 이벤트로그와 powershell 이벤트 로그에서 확인할 수 있습니다.

따라서, 192.168.40.134 ip로 추정되는 공격자는 **evil-winRM** 도구를 통해 2024-05-19 20:48:53(UTC+9)에 접속 시도를 하였습니다. 그리고, Lateral Movement 행위는 2024-05-19 20:50:01(UTC+9)에 다시 로그온 되었다가 event id 4100 System error 이벤트로그와 event id 403 engine stop 이벤트로그와 함께 종료된 것으로 파악됩니다.

#4 Dcomexec.py

공격자의 다음 Lateral Movement 행위를 살펴보겠습니다.

TimeStamp(UTC 0)	USN	File/Directory Name	FullPath	EventInfo
2024-05-19 11:49:54	274354440	Windows PowerShell.evtx	₩Windows₩System32₩winevt₩Logs₩Windows PowerShell.evtx	Data_Overwritten
2024-05-19 11:49:54	274354552	Microsoft-Windows-PowerShell%4Operational.evtx	₩Windows₩System32₩winevt₩Logs₩Microsoft-Windows-PowerShell%4Operational.evtx	Data_Overwritten
2024-05-19 11:51:11	274354704	__17161	₩Windows₩__17161	File_Created
2024-05-19 11:51:11	274354784	__17161	₩Windows₩__17161	File_Created / File_Closed
2024-05-19 11:51:11	274354864	CMD.EXE-0BD30981.pf	₩Windows₩Prefetch₩CMD.EXE-0BD30981.pf	Data_Truncated
2024-05-19 11:51:11	274354968	CMD.EXE-0BD30981.pf	₩Windows₩Prefetch₩CMD.EXE-0BD30981.pf	Data_Added / Data_Truncated
2024-05-19 11:51:11	274355072	CMD.EXE-0BD30981.pf	₩Windows₩Prefetch₩CMD.EXE-0BD30981.pf	Data_Added / Data_Truncated / File_Closed
2024-05-19 11:51:11	274355176	CONHOST.EXE-0C6456FB.pf	₩Windows₩Prefetch₩CONHOST.EXE-0C6456FB.pf	Data_Truncated
2024-05-19 11:51:11	274355288	CONHOST.EXE-0C6456FB.pf	₩Windows₩Prefetch₩CONHOST.EXE-0C6456FB.pf	Data_Added / Data_Truncated
2024-05-19 11:51:11	274355400	CONHOST.EXE-0C6456FB.pf	₩Windows₩Prefetch₩CONHOST.EXE-0C6456FB.pf	Data_Added / Data_Truncated / File_Closed
2024-05-19 11:51:12	274355512	__17161	₩Windows₩__17161	File_Closed / File_Deleted
2024-05-19 11:51:12	274355592	__17161	₩Windows₩__17161	File_Created
2024-05-19 11:51:12	274355672	__17161	₩Windows₩__17161	File_Created / Data_Added
2024-05-19 11:51:12	274355752	__17161	₩Windows₩__17161	File_Created / Data_Added / File_Closed
2024-05-19 11:51:13	274355832	__17161	₩Windows₩__17161	File_Closed / File_Deleted
2024-05-19 11:51:20	274355912	MMC.EXE-0FCCACD8.pf	₩Windows₩Prefetch₩MMC.EXE-0FCCACD8.pf	File_Created
2024-05-19 11:51:20	274356016	MMC.EXE-0FCCACD8.pf	₩Windows₩Prefetch₩MMC.EXE-0FCCACD8.pf	File_Created / Data_Added
2024-05-19 11:51:20	274356120	MMC.EXE-0FCCACD8.pf	₩Windows₩Prefetch₩MMC.EXE-0FCCACD8.pf	File_Created / Data_Added / File_Closed
2024-05-19 11:51:21	274356224	__17161	₩Windows₩__17161	File_Created
2024-05-19 11:51:21	274356304	__17161	₩Windows₩__17161	File_Created / Data_Added
2024-05-19 11:51:21	274356384	WHOAMI.EXE-9D378AFE.pf	₩Windows₩Prefetch₩WHOAMI.EXE-9D378AFE.pf	Data_Truncated
2024-05-19 11:51:21	274356488	WHOAMI.EXE-9D378AFE.pf	₩Windows₩Prefetch₩WHOAMI.EXE-9D378AFE.pf	Data_Added / Data_Truncated
2024-05-19 11:51:21	274356592	WHOAMI.EXE-9D378AFE.pf	₩Windows₩Prefetch₩WHOAMI.EXE-9D378AFE.pf	Data_Added / Data_Truncated / File_Closed
2024-05-19 11:51:21	274356696	__17161	₩Windows₩__17161	File_Created / Data_Added / File_Closed
2024-05-19 11:51:22	274356776	__17161	₩Windows₩__17161	File_Closed / File_Deleted

[그림 29] NTFS 로그에 기록된 __17161 파일 생성 및 삭제 로그

2024-05-19 11:51:11, 2024-05-19 11:51:21 시간에 생성된 __17161 파일의 생성과 삭제된 로그가 존재합니다. 또한 mmc.exe 이후에 whoami.exe 파일의 프리패치 로그가 존재하는데 프리패치는 실행 시간을 기록하면서 파일이 수정되기 때문에 해당 시간에 실행되었다고 볼 수 있습니다. 이러한 패턴으로 생성 및 삭제하는 도구는 impacket 도구 중 하나인 **dcomexec.py**입니다.

```
62      OUTPUT_FILENAME = '__' + str(time.time())[:5]
63      CODEC = sys.stdout.encoding
```

[그림 30] dcomexec.py에 사용되는 파일 이름 생성 코드

```
def execute_remote(self, data, shell_type='cmd'):
    if self._silentCommand is True:
        self._shell = data.split()[0]
        command = ' '.join(data.split()[1:])
    else:
        if shell_type == 'powershell':
            data = '$ProgressPreference="SilentlyContinue";' + data
            data = self.__pwsh + b64encode(data.encode('utf-16le')).decode()
        command = '/Q /c ' + data

    if self._noOutput is False:
        command += ' 1> ' + '\\\\127.0.0.1\\%s' % self._share + self._output + ' 2>&1'
```

In this file

210 _output = '\\' + OUTPUT_FILENAME

4 References Search

In this file

362 self._output, output_callback)

374 self._output)

387 + self._output + ' 2>&1'

456 + self._output + ' 2>&1'

[그림 31] dcomexec.py에 사용되는 명령 코드

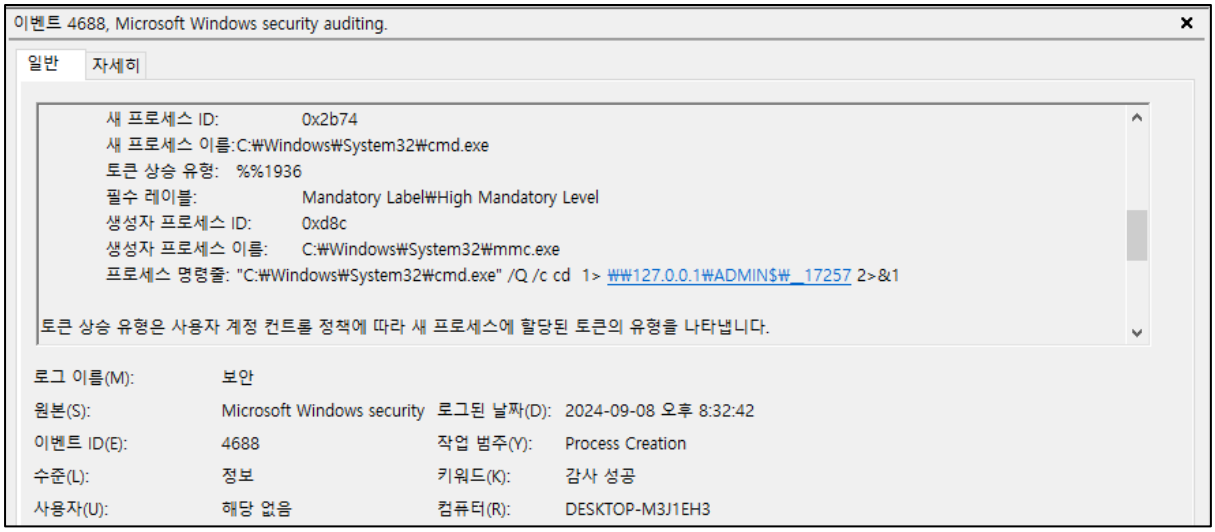
위 파일의 코드는 https://github.com/fortra/impacket/blob/master/examples/dcomexec.py에서 확인할 수 있습니다.

다만, dcomexec.py 실행 흔적도 wmiexec.py와 마찬가지로 cmd.exe를 통해 프로세스 명령줄 실행 기록이 적힌 4688 event id 로그가 존재하지 않습니다.

```
sinsa@sinsa-virtual-machine:~/Desktop/impacket/examples$ python3 dcomexec.py -object MMC20 -debug juhoheo:juhoheo@192.16
8.140.21
Impacket v0.10.0 - Copyright 2022 SecureAuth Corporation

[+] Impacket Library Installation Path: /home/sinsa/.local/lib/python3.10/site-packages/impacket
[*] SMBv3.0 dialect used
[+] Target system is 192.168.140.21 and isFQDN is False
[+] StringBinding: DESKTOP-M3J1EH3[61378]
[+] StringBinding: 192.168.140.21[61378]
[+] StringBinding chosen: ncacn_ip_tcp:192.168.140.21[61378]
[!] Launching semi-interactive shell - Careful what you execute
[!] Press help for extra shell commands
C:\>
```

[그림 32] dcomexec.py 테스트 – 공격자



[그림 33] dcomexec.py 테스트 결과 – 피해자 로그

실제로 교차 검증을 위해 테스트를 해본 결과, 피해자 로그엔 4688 event id로 mmc.exe 생성자와 함께 프로세스 명령줄로 cmd.exe 명령어가 위 그림과 같이 존재해야 합니다. 하지만, target 파일에는 4688 이벤트 로그 자체는 로깅을 하지만 wmiexec.py나 dcomexec.py 실행 시 기록되는 로그가 각 시각에 정확하게 존재하지 않았습니다.

따라서, dcomexec.py역시 성공과 실패 여부는 선별 압수된 증거로는 정확히 파악할 수 없었으나, **__17161 라는 포맷의 파일 생성을 수행하는 impacket은 dcomexec.py**뿐이고, mmc.exe와 whoami.exe 실행 시각을 통해 192.168.40.134라는 ip로 추정되는 공격자가 이번에는 dcomexec.py사용하여 DCOM 프로토콜을 통해 Lateral Movement를 2024-05-19 20:51:11(UTC+9) 시각에 시도하였다는 것을 알 수 있습니다.

#5 Smbexec.py

공격자의 다음 Lateral Movement 행위를 살펴보겠습니다.

TimeStamp(UTC 0)	USN	File/Directory Name	FullPath	EventInfo
2024-05-19 11:56:57	274372600	bREaoDkY.bat	₩Windows₩bREaoDkY.bat	File_Created
2024-05-19 11:56:57	274372688	bREaoDkY.bat	₩Windows₩bREaoDkY.bat	File_Created / Data_Added
2024-05-19 11:56:57	274372776	bREaoDkY.bat	₩Windows₩bREaoDkY.bat	File_Created / Data_Added / File_Closed
2024-05-19 11:56:57	274372864	__output	₩__output	File_Created
2024-05-19 11:56:57	274372944	__output	₩__output	File_Created / Data_Added
2024-05-19 11:56:57	274373024	__output	₩__output	File_Created / Data_Added / File_Closed
2024-05-19 11:56:57	274373104	bREaoDkY.bat	₩Windows₩bREaoDkY.bat	File_Closed / File_Deleted
2024-05-19 11:56:57	274373192	CMD.EXE-0BD30981.pf	₩Windows₩Prefetch₩CMD.EXE-0BD30981.pf	Data_Truncated
2024-05-19 11:56:57	274373296	CMD.EXE-0BD30981.pf	₩Windows₩Prefetch₩CMD.EXE-0BD30981.pf	Data_Added / Data_Truncated
2024-05-19 11:56:57	274373400	CMD.EXE-0BD30981.pf	₩Windows₩Prefetch₩CMD.EXE-0BD30981.pf	Data_Added / Data_Truncated / File_Closed
2024-05-19 11:56:57	274373504	CONHOST.EXE-0C6456FB.pf	₩Windows₩Prefetch₩CONHOST.EXE-0C6456FB.pf	Data_Truncated
2024-05-19 11:56:57	274373616	CONHOST.EXE-0C6456FB.pf	₩Windows₩Prefetch₩CONHOST.EXE-0C6456FB.pf	Data_Added / Data_Truncated
2024-05-19 11:56:57	274373728	CONHOST.EXE-0C6456FB.pf	₩Windows₩Prefetch₩CONHOST.EXE-0C6456FB.pf	Data_Added / Data_Truncated / File_Closed
2024-05-19 11:56:57	274373840	__output	₩__output	File_Closed / File_Deleted

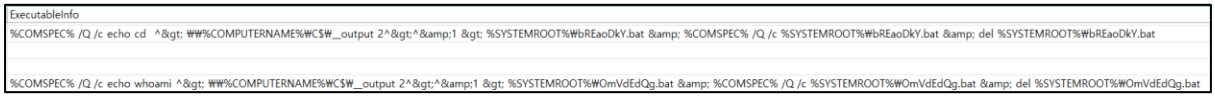
[그림 34] NTFS 로그에 기록된 bREaoDkY.bat, __output 파일

TimeStamp(UTC 0)	USN	File/Directory Name	FullPath	EventInfo
2024-05-19 11:57:16	274375824	OmVdEdQg.bat	₩Windows₩OmVdEdQg.bat	File_Created
2024-05-19 11:57:16	274375912	OmVdEdQg.bat	₩Windows₩OmVdEdQg.bat	File_Created / Data_Added
2024-05-19 11:57:16	274376000	OmVdEdQg.bat	₩Windows₩OmVdEdQg.bat	File_Created / Data_Added / File_Closed
2024-05-19 11:57:16	274376088	__output	₩__output	File_Created
2024-05-19 11:57:16	274376168	__output	₩__output	File_Created / Data_Added
2024-05-19 11:57:16	274376248	WHOAMI.EXE-9D378AFE.pf	₩Windows₩Prefetch₩WHOAMI.EXE-9D378AFE.pf	Data_Truncated
2024-05-19 11:57:16	274376352	WHOAMI.EXE-9D378AFE.pf	₩Windows₩Prefetch₩WHOAMI.EXE-9D378AFE.pf	Data_Added / Data_Truncated
2024-05-19 11:57:16	274376456	WHOAMI.EXE-9D378AFE.pf	₩Windows₩Prefetch₩WHOAMI.EXE-9D378AFE.pf	Data_Added / Data_Truncated / File_Closed
2024-05-19 11:57:16	274376560	__output	₩__output	File_Created / Data_Added / File_Closed
2024-05-19 11:57:16	274376640	OmVdEdQg.bat	₩Windows₩OmVdEdQg.bat	File_Closed / File_Deleted
2024-05-19 11:57:16	274376728	__output	₩__output	File_Closed / File_Deleted

[그림 35] NTFS 로그에 기록된 OmVdEdQg.bat, __output 파일

2024-05-19 11:56:57, 2024-05-19 11:57:16 시간에 생성된 bat 파일과 __output 파일이 존재하며, 이후에 삭제된 것까지 확인할 수 있습니다.

[그림 36] ssdpxvtN으로 등록된 이벤트 로그 기록



[그림 37] 서비스에 등록된 페이로드

이번에도, 동일한 패턴(이벤트 ID 4624, 4672)의 로그를 확인할 수 있으며, 이전과 다르게 페이로드까지 이벤트 로그에 남아있는 것을 확인할 수 있습니다.

```
57      OUTPUT_FILENAME = '__output'
58      SMBSERVER_DIR   = '__tmp'
59      DUMMY_SHARE     = 'TMP'
60      CODEC = sys.stdout.encoding
```

[그림 38] smbexec.py에 사용되는 파일 이름

```
def execute_remote(self, data, shell_type='cmd'):
    if shell_type == 'powershell':
        data = '$ProgressPreference="SilentlyContinue";' + data
        data = self.__pwsh + b64encode(data.encode('utf-16le')).decode()

    batchFile = '%SYSTEMROOT%\\' + ''.join([random.choice(string.ascii_letters) for _ in range(8)]) + '.bat'

    command = self.__shell + 'echo ' + data + ' ^> ' + self.__output + ' 2^>^&1 > ' + batchFile + ' & ' + \
              self.__shell + batchFile

    if self.__mode == 'SERVER':
        command += ' & ' + self.__copyBack
    command += ' & ' + 'del ' + batchFile
```

[그림 39] smbexec.py에 사용되는 명령 코드

```
if serviceName is None:
    self.__serviceName = ''.join([random.choice(string.ascii_letters) for i in range(8)])
else:
    self.__serviceName = serviceName
```

[그림 40] ssdpxvtN과 같은 서비스 이름 생성 코드

해당 페이로드 형태의 공격으로 미루어 볼 때, **smbexec.py**를 사용한 것으로 보이며 위 파일의 코드는 https://github.com/fortra/impacket/blob/master/examples/smbexec.py 에서 확인할 수 있습니다.

따라서, .bat파일과 __output파일의 생성, ssdpxvtN 서비스 이름 생성 및 7045 이벤트로그에 기록된 명령어 정보 등을 토대로, 192.168.40.134라는 ip로 추정되는 공격자가 이번에는 **smbexec.py**를 사용하여 SMB 프로토콜을 통해 Lateral Movement를 2024-05-19 20:56:57(UTC+9) 시각부터 시도하였다는 것을 알 수 있습니다.

이후에, [그림 4]에 21:03:23(UTC+9) 시각에 찍힌 runonce.exe는 별도의 event id가 4624인 logon 이벤트에서 외부 ip 흔적을 찾을 수 없었습니다.

#6 xfreerdp

다음으로 공격자의 lateral movement를 살펴보겠습니다.

[그림 41] event id 131 RDP 이벤트 로그

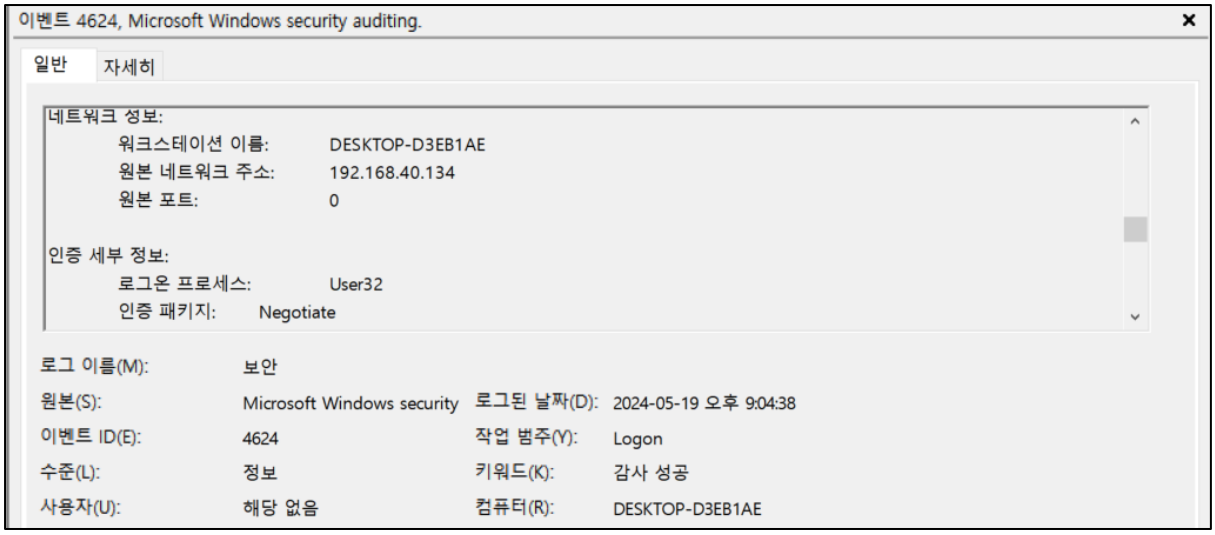
2024-05-19 21:04:36(UTC+9)에 위 그림과 같이 RDP 로그가 기록되어 있습니다.

[그림 42] event id 1149인 RemoteConnectionManager 이벤트로그

그리고, 비슷한 시각에는 위 그림과 같이 1149 event id를 가진 이벤트로그에서 원격 데스크톱 서비스: 사용자 인증 성공이라는 문구와 함께 원본 네트워크 주소가 공격자의 ip로 추정되는 192.168.40.134임을 알 수 있습니다.

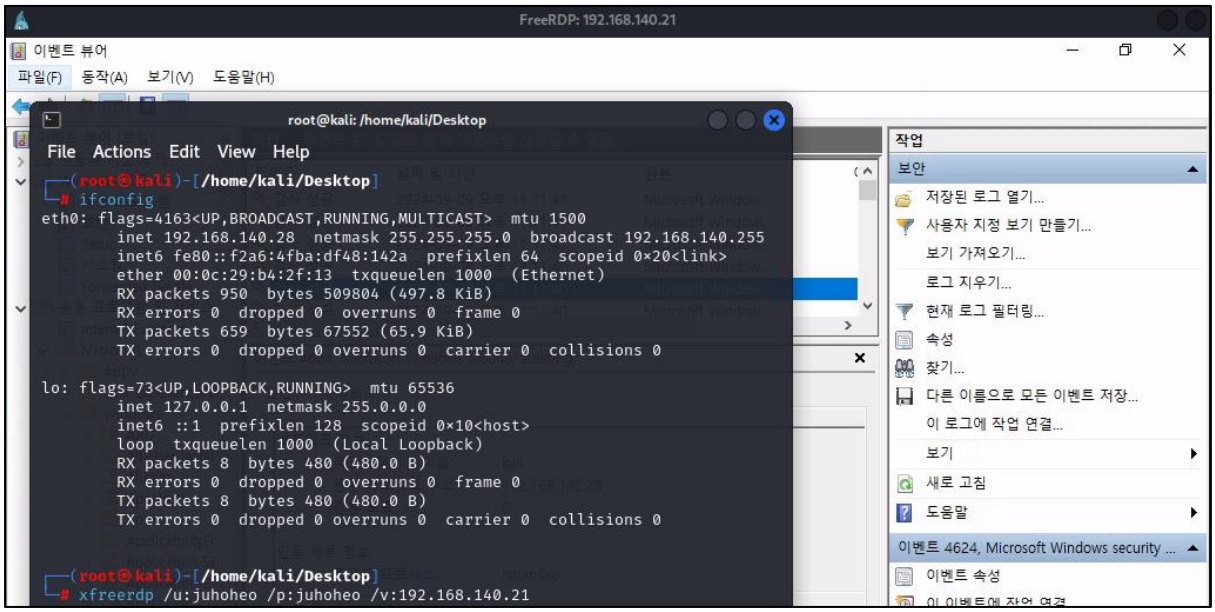
[그림 43] event id 4624 logon 이벤트 로그 - 1

그리고 [그림 41]과 시각에 Logon type이 3이고 secadmin으로 로그인한 기록이 남아있는데, 네트워크 정보에 워크스테이션 이름이 kali고 원본 네트워크 주소는 공격자로 추정되는 192.168.40.134 ip였습니다.



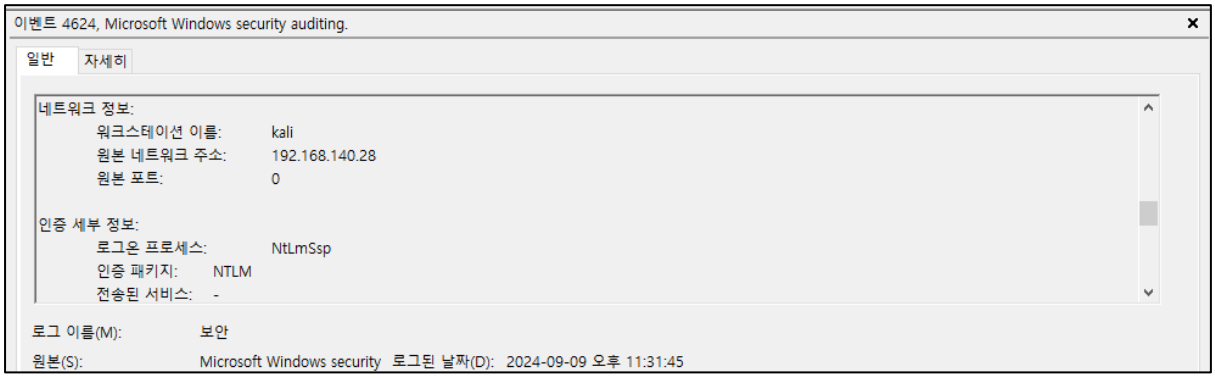
[그림 44] event id 4624 logon 이벤트 로그 – 2

2초 간격 뒤에는 동일한 logon type 10으로 워크스테이션 이름이 DESKTOP-D3EB1AE인 자기자신을 가리키지만 원본 네트워크 주소는 공격자의 ip로 추정되는 192.168.40.134로 기록되어 있습니다. 로그온 유형 10은 원격 대화형 로그온이나 RDP, 네트워크 연결에 사용되는 유형인데, kali 이후 자기자신 컴퓨터 이름으로 로그온 된 기록의 패턴을 찾기 위해 kali VM에서 테스트를 시도해보았습니다.



[그림 45] Kali VM(192.168.140.28)에서 target PC인 192.168.140.21로 테스트

Kali VM(192.168.140.28)에서 target PC인 192.168.140.21로 **xfreerdp**라는 도구를 활용해서 RDP 연결을 시도하였습니다.



[그림 46] kali에서 xfreerdp를 통해 접속한 후 로그온 기록 - 1

[그림 47] kali에서 xfreerdp를 통해 접속한 후 로그온 기록 - 2

[그림 48] kali에서 xfreerdp를 통해 접속한 후 1149 이벤트 로그 기록

위 [그림 46], [그림 47], [그림 48]을 통해서 kali 워크스테이션 이후 피해자 PC의 컴퓨터 이름이 남고, 원격 데스크톱 서비스 기록도 남은 것으로 보아 target에 남겨진 흔적과 동일함을 알 수 있습니다.

따라서, 공격자는 192.168.40.134 ip를 통해 2024-05-19 21:04:36(UTC+9) 시각에 **xfreerdp**라는 도구를 kali환경에서 RDP 프로토콜을 통해 Lateral Movement 행위를 시도했음을 알 수 있습니다.

앞서 살펴본 공격자의 lateral movement 행위와 연관된 이벤트 로그에서 얻을 수 있는 정보와 사용된 도구에 대해 표로 최종 정리하였습니다.

Timestamp (UTC+9)	TargetUserName	RemoteHost	RemotePort or workstation name	Logon ProcessName	사용된 도구
2024-05-19 20:39:09	secadmin	192.168.40.134	36396	NtLmSsp	psexec.py
2024-05-19 20:39:09	secadmin	192.168.40.134	36402	NtLmSsp	psexec.py
2024-05-19 20:39:09	secadmin	192.168.40.134	36404	NtLmSsp	psexec.py
2024-05-19 20:39:09	secadmin	192.168.40.134	36410	NtLmSsp	psexec.py
2024-05-19 20:44:13	secadmin	192.168.40.134	56226	NtLmSsp	wmiexec.py
2024-05-19 20:44:13	secadmin	192.168.40.134	43572	NtLmSsp	wmiexec.py
2024-05-19 20:44:13	secadmin	192.168.40.134	47348	NtLmSsp	wmiexec.py
2024-05-19 20:48:53	secadmin	192.168.40.134	32876	NtLmSsp	evil-winRM
2024-05-19 20:50:01	secadmin	192.168.40.134	48558	NtLmSsp	evil-winRM
2024-05-19 20:51:10	secadmin	192.168.40.134	56660	NtLmSsp	dcomexec.py
2024-05-19 20:51:10	secadmin	192.168.40.134	50938	NtLmSsp	dcomexec.py
2024-05-19 20:51:10	secadmin	192.168.40.134	59700	NtLmSsp	dcomexec.py
2024-05-19 20:56:57	secadmin	192.168.40.134	59980	NtLmSsp	smbexec.py
2024-05-19 21:04:36	secadmin	192.168.40.134	53552/kali	NtLmSsp	xfreerdp
2024-05-19 21:04:38	secadmin	192.168.40.134	53552/DESKTOP-D3EB1AE	NtLmSsp	xfreerdp

[표 3] 공격자의 impacket lateral movement 행위에 대한 정보