"Explore a meticulously compiled dossier spotlighting event log entries, registry modifications, and file creations or changes linked to lateral movement. This comprehensive file meticulously examines the nuances of lateral movement occurrences, shedding light on both the origins and destinations of these actions. Immerse yourself in meticulously categorized sections that unveil crucial details surrounding lateral movement scenarios, offering invaluable insights into their dynamics."

Lateral movement Artifacts In case of RDP **Using Event Ids**:

| *Source system Artifacts:* | *Destination system Artifacts:* |
|---|---|
| | |
| *security.evtx* | *security.evtx* |
| 4648 - Logon specifying alternate credentials - if NLA enabled on destination<br>- Current logged-on Username<br>- Alternate Username<br>- Destination Host Name/IP<br>- Process Name | 4624 -Logon Type 10<br>-Source IP/Logon Username.<br><br>4778/4779<br>-IP Address of Source/Source System Name<br>-Logon Username |
| | |
| *Microsoft-Windows-Terminal Services-RDPClient 4Operational.evtx* | *Microsoft-Windows-RemoteDesktopservices-RdpCoreTS%40perational.evtx* |
| 1024<br>-- Destination Host Name<br>1102<br>--Destination IP Address | 131<br>-Connection Attempts -Source IP<br>98<br>-Successful Connections |
| | |
| | *Microsoft-Windows-Services RemoteConnection Manager%40perational.evtx* |
| | 1149<br>-Source IP/Logon User Name Blank user name may indicate use of Sticky |
| | |
| | *Microsoft-Windows-Terminal Services-LocalSessionManager%40perational.evtx* |
| | 21, 22, 25<br>-Source IP/Logon Username<br>41<br>-Logon Username |

Lateral movement Artifacts In case of Windows Admin share **Using Event Ids**:

| *Source system Artifacts:* | *Destination system Artifacts:* |
|---|---|
| | |
| *security.evtx* | *security.evtx* |

Akash's sheet

| | |
|---|---|
| 4648 - Logon specifying alternate credentials<br>- Current logged-on Username<br>- Alternate Username<br>- Destination Host Name/IP<br>- Process Name | 4624 - Logon Type 3<br>- Source IP/Logon Username<br>4672<br>- Logon User Name<br>- Logon by user with administrative rights<br>- Requirement for accessing default shares such as c$ and ADMIN$<br>4776 - NTLM if authenticating to Local System<br>-Source Host Name/Logon User Name<br>4768 - TGT Granted<br>-Source Host Name/Logon User Name<br>- Available only on domain controller<br>4769 - Service Ticket Granted if authenticating to Domain Controller<br>- Destination Host Name/Logon Username<br>-Source IP<br>- Available only on domain controller<br>5140<br>- Share Access<br>5145<br>- Auditing of shared files - NOISY! |
| | |
| *Microsoft-Windows-SmbClient 4Security.evtx* | |
| 31001 Failed logons to destination<br>- Destination Host Name<br>- User Name for failed logon<br>- Reason code for failed destination logon (e.g. bad password) | |

Akash's sheet

Lateral movement Artifacts In case of PsExec **Using Event Ids**:

It can push and execute code non-interactively, make built-in system commands "remote-capable" by sending data back to the originating system, and even be used for interactive console sessions.

| *Source system Artifacts:* | *Destination system Artifacts:* |
|---|---|
| | |
| *security.evtx* | *security.evtx* |
| 4648 - Logon specifying alternate credentials<br>- Current logged-on Username<br>- Alternate Username<br>- Destination Host Name/IP<br>- Process Name | 4624 Logon Type 3 (and Type 2 if "-u" Alternate Credentials are used)<br>-Source IP/Logon Username<br>4672<br>- Logon Username<br>-Logon by a user with administrative rights |

| | |
|---|---|
| | - Requirement for access default shares such as c$ and ADMIN$<br>5140 - Share Access<br>- ADMIN$ share used by PsExec<br>system.evtx<br>7045<br>-Service Install |
| | |
| registry key is created, NTUSER\Software\SysIntemals\PsExec\EulaAccepted | If a binary is executed that does not currently exist on the target, the-c argument tells PsExec to copy it to the system.<br>Keep in mind that PsExec-c can copy a binary anywhere in the file system, and unless the command line was captured, it may take additional artifacts to determine what was executed.<br>***** **Newer versions of PsExec include the "-r" option, allowing an attacker to change this name to anything they like.** *** |

Akash's sheet

Lateral movement Artifacts In case of Remote management tool (Remote service) **Using Event Ids**:

| *Source system Artifacts:* | *Destination system Artifacts:* |
|---|---|
| | |
| *security.evtx* | *security.evtx* |
| | 4624 Logon Type 3<br>-Source IP/Logon Username<br>4697<br>- Security records service install, if enabled<br>- Enabling non-default Security events such as ID 4697 are particularly useful if only the Security logs are forwarded to a centralized log server |
| | |
| | *system.evtx* |
| | 7034<br>- Service crashed unexpectedly<br>7035<br>-Service sent a Start/Stop control<br>7036<br>- Service started or stopped<br>7040<br>- Start type changed (Boot On Request \| Disabled)<br>7045 |

| | |
|---|---|
| | - A service was installed on the system |
| | |
| *Scheduled task Artifact in case Remote management tool used* | |
| | |
| *security.evtx* | *security.evtx* |
| | |
| 4648 -Logon specifying alternate credentials<br>- Current logged-on Username - Alternate Username<br>- Destination Host Name/IP<br>- Process Name | 4624 - Logon Type 3<br>-Source IP/Logon Username<br>4672<br>-Logon Username<br>-Logon by a user with administrative rights<br>- Requirement for accessing default shares such as cs and ADMINS<br>4698<br>- Scheduled task created<br>4702<br>- Scheduled task updated<br>4699<br>- Scheduled task deleted<br>4700/4701<br>- Scheduled task enabled/disabled |
| | |
| | *Microsoft-Windows-Task Scheduler 40perational.evtx* |
| | 106<br>-Scheduled task created<br>140<br>- Scheduled task updated<br>141<br>- Scheduled task deleted<br>200/201<br>-Scheduled task executed/completed |

Akash's sheet

Lateral movement Artifacts In case of WMI **Using Event Ids**:

| *Source system Artifacts:* | *Destination system Artifacts:* |
|---|---|
| | |
| *security.evtx* | *security.evtx* |
| 4648 - Logon specifying alternate credentials<br>- Current logged-on Username<br>- Alternate Username<br>- Destination Host Name/IP<br>- Process Name | 4624 Logon Type 3<br>-Source IP/Logon Username<br>4672<br>-Logon Username<br>-Logon by a user with administrative rights |
| | |
| | *Microsoft-Windows-WMI-Activity 40perational.evtx* |

| | |
|---|---|
| | 5857<br>- Indicates time of wmipryse execution and path to provider DLL<br>- attackers sometimes install malicious WMI provider DLLS<br>5860, 5861<br>-Registration of Temporary (5860) and Permanent (5861) Event Consumers. Typically used for persistence but can be used for remote execution. |
| ******The most common WMI command used for lateral movement is "process call create"****** | ***** The destination file system can help us identify any executables copied to the remote system (**especially if "process call create" was in use)**. **Evidence of the creation of .mof files or the execution of mofcomp.exe can provide early indications of WMI event consumers, as .mof files are one of the easiest ways to implement them.**<br>Once the activity has been identified, review of the WMI Repository can identify the type of persistence and what was scheduled to be executed (PowerShell can be helpful for auditing this). ***** |

Akash's sheet

Lateral movement Artifacts In case of PowerShell **Remoting Using Event Ids:**

| *Source system Artifacts:* | *Destination system Artifacts:* |
|---|---|
| | |
| *security.evtx* | *security.evtx* |
| 4648 - Logon specifying alternate credentials<br>- Current logged-on Username<br>- Alternate Username<br>- Destination Host Name/IP<br>- Process Name | 4624 Logon Type 3<br>-Source IP/Logon Username<br>4672<br>- Logon Username<br>-Logon by an a user with administrative rights |
| | |
| *Microsoft-Windows-WinRM 40perational.evtx* | *Microsoft-Windows-PowerShell\40perational.evtx* |
| 6 - WSMan Session initialize<br>- Session created<br>- Destination Host Name or IP<br>- Current logged-on Username<br>8, 15, 16, 33 - WSMan Session deinitialization<br>- Closing of WSMan session<br>- Current logged-on Username | 4103, 4104 - Script Block logging<br>-Logs suspicious scripts by default in PS v5<br>-Logs all scripts if configured<br>53504<br>-Records the authenticating user |
| | |

| *Microsoft-Windows-PowerShell\40perational.evtx* | *Windows PowerShell.evtx* |
|---|---|
| 40691, 40692<br>- Records the local initiation of powershell.exe and associated user account<br>8193 & 8194<br>- Session created<br>8197 - Connect<br>- Session closed | 400/403<br>-"ServerRemoteHost" indicates start/end of Remoting session<br>800<br>-Includes partial script code |
| | |
| | *Microsoft-Windows-WinRM 40perational.evtx* |
| | 91<br>-Session creation<br>168<br>-Records the authenticating user |

# Registry/File system:

Lateral movement Artifacts In case of RDP **using registry/File system:**

| *Source system Artifacts:* | *Destination system Artifacts:* |
|---|---|
| | |
| ***Registry*** | ***Registry*** |
| Remote desktop destinations:<br>- NTUSER\Software\Microsoft\Terminal Server Client\Server\<br><br>Shimcache: (**System**)<br>- **mstsc.exe** (Remote desktop client)<br><br>BAM/DAM: (**system**)- Last time executed<br>- **mstsc.exe** (Remote desktop client)<br><br>Amcache.hve-First time executed<br>- **mstsc.exe** (Remote desktop client)<br><br>UserAssist- **NTUSER.DAT**<br>- **mstsc.exe** (Remote desktop client)<br>- Last time executed<br>- Number of times exeuted<br><br>RecentApps -**NTUSER.DAT**<br>- **mstsc.exe** (Remote desktop client)<br>- Last time executed | Shimcache -**SYSTEM**<br>- rdpclient.exe<br>- tstheme.exe<br><br>Amcache.hve -First Time Executed<br>- rdpclient.exe<br>- tstheme.exe |

Akash's sheet

| | |
|---|---|
| - Number of times executed<br>- Recent Items subkey tracks connection destination and times | |
| | |
| ***File System*** | ***File System*** |
| JumpLists: -<br>-C:\Users\<Username>\Appdata\Roaming\Microsoft\Windows\Recent\AutomaticDestinations\<br>• {MSTSC-APPID} automaticDestinations-MS<br>• Tracks remote desktop connection destination and times<br>Prefetch: -<br>-C:\Windows\Prefetch\<br>• Mstsc.exe- {Hash).pf<br><br>Bitmap cache: -<br>-C:\Users\<Username>\Appdata\Local\Microsoft\Terminal Server Client\Cache<br>• Backache##.bmc<br>• Cache####.bin | Prefetch: -<br>-C:\Windows\Prefetch\<br>• rdpclient.exe - {Hash}.pf<br>• tstheme.exe - {Hash).pf |
| | |

Akash's sheet

Lateral movement Artifacts In case of Windows Admin share **using registry/File system:**

Net user z: \\host\c$ /user:domain\username <password>

| *Source system Artifacts:* | *Destination system Artifacts:* |
|---|---|
| | |
| ***Registry*** | ***Registry*** |
| MountPoints2- Remotely Mapped Shares<br>- NTUSER\Software\Microsoft\Windows\CurrentVersion\Explorer\MountPoint2<br><br>Shell Bags -USRCLASS.DAT<br>• Remote folder accessed inside an interaction session via explorer by attackers.<br><br>Shimcache -SYSTEM<br>• Net.exe<br>• Net1.exe<br><br>BAM/DAM: (**system**)- Last time executed.<br>• Net.exe<br>• Net1.exe<br><br>Amcache.hve-First time executed<br>• Net.exe<br>• Net1.exe | |

| | |
|---|---|
| | |
| ***File System*** | ***File System*** |
| Prefetch: -<br>-C:\Windows\Prefetch\<br>• Net.exe - {Hash}.pf<br>• Net1.exe - {Hash).pf<br><br>User Profile artifacts: -<br>• Review shortcut files and jump lists for remote files accesses by attackers, if they had interactive access (RDP) | File Creation<br>• Attacker's files (malware) copied to destination system.<br>• Looked for modified time before creation time.<br>• Creation time is time of file copy. |

Akash's sheet

Lateral movement Artifacts In case of PsExec **using registry/File system:**

Psexec.exe \\host -acceptuela -d -c c:\temp\evil.exe

| *Source system Artifacts:* | *Destination system Artifacts:* |
|---|---|
| | |
| ***Registry*** | ***Registry*** |
| NTUSER.Dat:<br>- Software\Sysinternals\PsExec\EulaAccepted<br><br>Shimcache: (**System**)<br>- psexec.exe<br><br>BAM/DAM: (**system**)- Last time executed<br>- psexec.exe<br><br>Amcache.hve-First time executed<br>- psexec.exe | Newer versions of PsExec include the "-r" option, allowing an attacker to change this name to anything they like.<br><br>New Service creation configured in:<br>SYSTEM\CurrentControlSet\Services\PSEXESVC<br>• "-r" option can allow attacker to rename service<br><br>Shimcache: (**System**)<br>- psexec.exe<br><br>Amcache.hve-First time executed<br>- psexec.exe |
| ***File System*** | ***File System*** |
| Prefetch: -<br>-C:\Windows\Prefetch\<br>• psexec.exe - {Hash).pf | Prefetch: -<br>-C:\Windows\Prefetch\<br>• psexesvc.exe - {Hash).pf<br>• evil.exe - {Hash).pf |

| | |
|---|---|
| • possible reference to other files accesses by psexec.exe, such as executables copied to target system with the -c option.<br><br>Files creation: -<br>• psexec.exe file downloaded and created on local host as the file is not native to windows | Files creation: -<br>• User profile directory structure created unless "-e" option used<br>• Psexesvc.exe will be places in ADMIN$(\Windows) by default, as well as other executable (Evil.exe) pushed by PsExec |

Lateral movement Artifacts In case of Remote management tool (Remote service) **using registry/File system:**

| *Source system Artifacts:* | *Destination system Artifacts:* |
|---|---|
| | |
| ***Registry*** | ***Registry*** |
| Shimcache: (**System**)<br>- sc.exe<br><br>BAM/DAM: (**system**)- Last time executed<br>- sc.exe<br><br>Amcache.hve-First time executed<br>- sc.exe | SYSTEM\CurrentControlSet\Services\<br>(New service creation)<br><br>Shimcache: (**System**)<br>- evil.exe<br>- Shimcache records existence of malicious service executable unless implemented as service DLL<br><br>Amcache.hve-First time executed<br>- evil.exe |
| | |
| ***File System*** | ***File System*** |
| Prefetch: -<br>-C:\Windows\Prefetch\<br>• sc.exe - {Hash).pf | Prefetch: -<br>-C:\Windows\Prefetch\<br>• evil.exe- {Hash).pf<br><br>File creation.<br>• evil.exe pr evil.dll malicious service executable or service DLL |
| | |
| *Scheduled task Artifact in case Remote management tool used* | |
| | |
| ***Registry*** | ***Registry*** |
| | |
| Shimcache: (**System**)<br>- at.exe<br>- Schtasks.exe | Software:<br>- Microsoft\Windows NT\CurrentVersion\Schedule\TaskCache\Tasks\<br>- Microsoft\Windows NT\CurrentVersion\ |

Akash's sheet

| | |
|---|---|
| BAM/DAM: (**system**)- Last time executed<br>- at.exe<br>- Schtasks.exe<br><br>Amcache.hve-First time executed<br>- at.exe<br>- Schtasks.exe | Schedule\TaskCache\Tree\<br><br><u>Shimcache:</u> (**System**)<br>- evil.exe<br><br>Amcache.hve-First time executed<br>- evil.exe |
| | |
| ***File System*** | ***File System*** |
| Prefetch: -<br>-C:\Windows\Prefetch\<br>• at.exe - {Hash).pf<br>• Schtasks.exe – {Hash}.pf | File creation<br>• Evil.exe<br>• Job files created in C:\Windows\Tasks<br>• XML Tak file created in C:\Windows\System32\Tasks<br>- Author tag under "registrationinfo" can identify (Source system name, creator name)<br><br>Prefetch: -<br>-C:\Windows\Prefetch\<br>• evil.exe- {Hash).pf |

<u>Lateral movement Artifacts In case of WMI **using registry/File system:**</u>

| *Source system Artifacts:* | *Destination system Artifacts:* |
|---|---|
| | |
| ***Registry*** | ***Registry*** |
| <u>Shimcache:</u> (**System**)<br>-wmic.exe<br><br>BAM/DAM: (**system**)- Last time executed<br>-wmic.exe<br><br>Amcache.hve-First time executed<br>-wmic.exe | <u>Shimcache:</u> (**System**)<br>-scrcons.exe<br>-mofcomp.exe<br>-wmiprvse.exe<br>-evil.exe<br><br>Amcache.hve-First time executed<br>-scrcons.exe<br>-mofcomp.exe<br>-wmiprvse.exe<br>-evil.exe |
| | |
| ***File System*** | ***File System*** |
| Prefetch: -<br>-C:\Windows\Prefetch\<br>• wmic.exe - {Hash).pf | File creation<br>• Evil.exe |

Akash's sheet

| | |
|---|---|
| | • Evil.mof -mof files can be used to manage the wmi repository<br><br>Prefetch: -<br>-C:\Windows\Prefetch\<br>• scrcons.exe - {Hash).pf<br>• mofcomp.exe - {Hash).pf<br>• wmiprvse.exe - {Hash).pf<br>• evil.exe - {Hash).pf<br><br>Unauthorized changes to the wmi repositories in C:\Windows\System32\wbem\repository |
| ******The most common WMI command used for lateral movement is "process call create"****** | ***** The destination file system can help us identify any executables copied to the remote system (**especially if "process call create" was in use)**. **Evidence of the creation of .mof files or the execution of mofcomp.exe can provide early indications of WMI event consumers, as .mof files are one of the easiest ways to implement them.**<br>Once the activity has been identified, review of the WMI Repository can identify the type of persistence and what was scheduled to be executed (PowerShell can be helpful for auditing this). ***** |

Akash's sheet

Lateral movement Artifacts in case of PowerShell Remoting **using registry/File system**:

| *Source system Artifacts:* | *Destination system Artifacts:* |
|---|---|
| | |
| ***Registry*** | ***Registry*** |
| Shimcache: (**System**)<br>-powershell.exe<br><br>BAM/DAM: (**system**)- Last time executed<br>- powershell.exe<br><br>Amcache.hve-First time executed<br>- powershell.exe | Software:<br>- Microsoft\PowerShell\1\ShellIDs\Microsoft.PowerShell\ExecutionPolicy<br>• Attacker may change execution policy to less restrictive setting such as "bypass"<br><br>Shimcache: (**System**)<br>- wsmprovhost.exe<br>- evil.exe<br><br>Amcache.hve-First time executed<br>- wsmprovhost.exe<br>- evil.exe |
| | |
| ***File System*** | ***File System*** |

| | |
|---|---|
| Prefetch: -<br>-C:\Windows\Prefetch\<br>• powershell.exe - {Hash).pf<br>• Powershell scripts that run within 10 seconds of powershll.exe launching will be tracked in powershell.exe prefetch file<br><br>Command history:<br>C:\Users\<Username>\Appdata\Roaming\Microsoft\Windows\Powershell\PSReadline\ConsoleHost_history.txt<br>• With PS V5+ a history file with previous 4096 command is maintained per user | Prefetch: -<br>-C:\Windows\Prefetch\<br>• Wsmprovhost.exe- {Hash).pf<br>• evil.exe - {Hash).pf<br><br>File creation: -<br>-evil.exe<br>-With Enter-PSSession, a user profile directory may be created |

Akash's sheet