LATERAL MOVEMENT RUNBOOK

Over View

Rapid detection, investigation, containment and remediation of lateral movement across identity, endpoint and network layers.

- Collect
- Contain
- Forensics
- Remediate
- Report & Lessons Learned

Kumar Bineet Ranjan

Contents

SOC Lateral Movement Investigation Playbook — Complete (Detailed Manual + Runbook)

This document contains a full, detailed SOC playbook for investigating lateral movement. Each section is structured with: Objective, Detailed Explanation (attacker technique + defender view), Logs/Events/Artifacts, Telemetry (what to check), Example Queries / Commands, Analyst Actions (step-by-step), and Red Flags vs False Positives. Two full worked examples and a final checkpoint are included at the end.

Section 1 — Detection & Alerting

Objective

Detect lateral movement as early as possible across identity, endpoint, and network telemetry to prevent escalation and data exfiltration.

Detailed Explanation (attacker technique + defender view)

Attackers often pivot using legitimate administrative tools and stolen credentials. They may use PsExec, WMIC, PowerShell Remoting, scheduled tasks, services, or RDP. Because these tools are dual-use, defenders must look for anomalous patterns — non-admin usage, off-hours activity, rapid multi-host logons, encoded PowerShell, persistence creation, and LSASS memory access. Early detection reduces dwell time and limits scope.

Logs / Events / Artifacts

- Windows Security: 4624 (Success), 4625 (Failure), 4648 (Explicit Creds), 4672 (Privilege Assigned), 4698/4702 (Scheduled Task create/modify), 7045 (Service Install), 4768/4769 (Kerberos), 4776 (NTLM).
- Sysmon: 1 (ProcessCreate), 3 (NetworkConnect), 8 (CreateRemoteThread), 10 (ProcessAccess), 11 (FileCreate), 19–21 (WMI activity).
- PowerShell: 4103/4104 (Module/ScriptBlock logging).
- EDR: process tree snapshots, file quarantine, LSASS access attempts, sandbox verdicts.
- Network: NetFlow/firewall logs showing east-west RDP/SMB/WinRM flows, PCAP evidence.
- Filesystem: files in ADMIN$, C$, Temp, and unexpected executables or scripts.

Telemetry (what to check)

- 4624 LogonType (10=RDP, 3=Network) and IpAddress/WorkstationName.
- Sysmon ProcessCreate CommandLine and ParentImage; ProcessGuid for joins.
- PowerShell ScriptBlockText raw content and any base64-encoded payloads.
- EDR: process hashes, parent-child relationships, open handles to lsass.exe.
- Network: Source/Destination IP and ports for internal connections.

Example Queries (Splunk / KQL / QRadar)

Splunk - Rapid multi-host logons:

```
index=winevent EventID=4624
| eval user=Account_Name, host=Computer
```

| stats earliest(_time) as first latest(_time) as last dc(host) as host_count by user
| where host_count >= 3 AND (last - first) <= 300

KQL (Sentinel) - Failed then success:

SecurityEvent
| where EventID in (4625, 4624)
| sort by TimeGenerated asc
| serialize
| session window=10m by Account
| where countif(EventID==4625) >=3 and countif(EventID==4624) >=1

QRadar - Internal RDP flows (AQL):

select * from events where (eventname = 'Connection' and (protocol = 'RDP' or destination_port = 3389)) and source_ip in (select ipaddr from asset where asset_type='workstation') and destination_ip in (select ipaddr from asset where asset_type='server') within last 5 minutes

Analyst Actions (step-by-step)

1. Record alert metadata: alert ID, rule name, timestamp (UTC+IST), source host, target host, account, process, command line.
2. Export raw Security, Sysmon and PowerShell logs for ±2 hours around the alert.
3. Request EDR process tree snapshots for source and target endpoints; save JSON export.
4. Check network flows (NetFlow/firewall) for east-west sessions between the implicated hosts.
5. If LSASS access or credential theft is suspected, prioritize memory capture before terminating processes.

Red Flags vs False Positives

- Red Flags: non-admin account running PsExec/WMIC; repeated 4625 followed by 4624; PowerShell -EncodedCommand with downloads/IEX; creation of scheduled tasks/services by low-priv users; LSASS memory access by non-forensic tools.
- False Positives: SCCM/Intune deployment, documented admin maintenance, known backup/AV agents performing expected actions.

Section 2 — Initial Triage

Objective

Rapidly determine whether the alert represents a True Positive (TP) requiring containment or a False Positive (FP) that can be closed or monitored.

Detailed Explanation

Triage requires context: account role, host role, time-of-day, source network (LAN/VPN/external), process legitimacy, and change tickets. Because many tools are dual-use, triage must correlate multiple indicators and check with ITSM or operations when in doubt.

Logs / Events / Artifacts

- AD user attributes (memberOf, admin privileges), ticketing system entries.
- Security events for the user/host (4624/4625/4648).
- EDR process hashes and recent activity.

- Baseline login/process behavior over last 30–90 days.

Telemetry (what to check)

- User type: standard, service, admin; MFA status if available.
- Host type: workstation/server/DC/jump box; asset criticality.
- Time: business hours vs off-hours; local time zone consistency (IST).
- IP/Geo: corporate subnet, VPN gateway, external IP reputation.
- Process legitimacy: command line, parent process, signatures.

Analyst Actions (step-by-step)

6. Look up the user in AD (Get-ADUser -Identity <user> -Properties MemberOf) and note admin membership.
7. Determine the host role and business owner (CMDB lookup).
8. Search ticketing (ITSM) for change tickets referencing the host/user/process/time.
9. Pull 30–90 day baseline of logons and process executions for the user and host to identify anomalies.
10. Apply scoring model: PsExec +5; LSASS/mimikatz +7; scheduled task/service +4; multi-host +3; off-hours +1; change ticket -6; known automation -5. Decide TP/FP per thresholds.

Red Flags vs False Positives

- Red Flags: standard user executing admin tools off-hours without tickets; external IP logins followed by internal lateral moves; multiple distinct hosts accessed in short window.
- False Positives: documented maintenance, known automation with correct hashes and owners.

Section 3 — Evidence Collection

Objective

Collect volatile and persistent evidence in a forensically sound manner, prioritizing volatile data and maintaining chain-of-custody.

Detailed Explanation

Capture memory, EDR snapshots, raw logs, scheduled task/service configs, and network evidence. Volatile items like memory or active network sessions should be captured before containment actions that remove them. All artifacts must be hashed and recorded with chain-of-custody metadata.

Logs / Events / Artifacts

- Memory: lsass.dmp if credential theft suspected; other process dumps as needed.
- Raw EVTX: Security, Sysmon, PowerShell logs for ±2 hours (or more) around detection.
- EDR: process tree JSON, file artifacts, quarantine copies.
- Network: PCAP slices, NetFlow/Firewall logs for implicated subnets.
- System artifacts: Task XML files, Service registry entries, Run keys, Prefetch, MFT.

Telemetry (what to check)

- ProcessGuid and ParentProcessGuid for Sysmon; full CommandLine for process events.
- PowerShell ScriptBlockText and module logging entries.
- File paths and their timestamps; hashes (SHA256).
- Network sessions (src/dst IP, ports, bytes, durations).

Example Commands (collection)

Export EVTX files:

wevtutil epl Security C:\Forensics\Security.evtx

wevtutil epl Microsoft-Windows-Sysmon/Operational C:\Forensics\Sysmon.evtx

Capture LSASS memory (EDR preferred; alternative procdump):

procdump -ma lsass.exe C:\Forensics\lsass.dmp

Compute hash:

certutil -hashfile C:\Forensics\lsass.dmp SHA256

Analyst Actions (step-by-step)

11. If LSASS suspected, capture memory via EDR or procdump and compute hash; record collector and timestamp.
12. Export Security/Sysmon/PowerShell logs for ±2 hours and save raw files to evidence store.
13. Export/screen capture EDR process tree and quarantine suspicious files; record SHA256.
14. Export scheduled task XML and service configs from target hosts.
15. Collect PCAP slices for implicated host IPs and time window from network capture systems.
16. Populate chain-of-custody CSV with fields: Item, CollectedBy, DateTimeUTC, Tool, Hash, StoragePath, Notes.

Red Flags vs False Positives

- Red Flags: procdump/mimikatz executed by non-IR personnel; task/service created pointing to temp path; suspicious LSASS access by unknown process.
- False Positives: authorized AV/backups/IR captures with documented approvals.

Section 4 — Tool & Process Analysis (Deep-Dive)

Objective

For each common lateral tool, understand attacker use-cases, defender detection points, and forensic traces.

Detailed Tool Breakdowns

PsExec: copies psexesvc to ADMIN$ and creates a service; detectable via 7045, Sysmon process create on source and service install on target, SMB connect to 445. Watch for non-admin invocation, service binary in temp, or unexpected cleanup behavior.

WMIC/WMI: stealthy remote execution using WMI consumers/providers; logs in Sysmon 19–21 and often leaves remote process events without explicit file copies. Check for WMI Consumers, Callers, and command strings.

PowerShell Remoting/EncodedCommand: attackers use -EncodedCommand and scriptblocks to obfuscate payloads; PowerShell 4104 captures blocks if enabled. Decode base64 in safe lab and inspect for IEX, Invoke-Mimikatz, credential theft routines.

Scheduled Tasks/Services: persistence often implemented as tasks (4698/4702) or services (7045). Record Task XML and service BinaryPathName. Suspicious if run-as is SYSTEM or points to temp/user locations.

RDP: interactive access produces 4624 logon type 10 and TerminalServices logs (1149). Look for workstation→server interactive logons by non-admin users or off-hours.

SMB: file staging detected via 5140 share access and Sysmon 11 file creates in ADMIN$/Temp. Combined with process creates on target, indicates staging and execution.

Credential dumpers: mimikatz/procdump/lsass scrapers often show as EDR indicators (ReadProcessMemory), process creating a dump file, and subsequent authentication anomalies (NTLM reuse, 4648 events).

Telemetry (what to check)

- CommandLine, ParentImage, ProcessGuid, Sysmon 3 destination IP and port.
- Event 7045/4698 details (author, binary path, task name).
- PowerShell ScriptBlockText content and command lengths (very long = encoded).
- EDR handles to LSASS and file write events in temp/ADMIN$.

Analyst Actions (step-by-step)

17. Search Sysmon/EDR for PsExec/WMIC/PowerShell invocations in timeframe.
18. On each target, export task XML and run `sc qc <service>` to identify binary path.
19. If encoded PowerShell present, decode in lab and search for persistence/credential theft code.
20. Collect EDR process trees and link network connects to subsequent logons on targets.

Red Flags vs False Positives

- Red Flags: PsExec by standard user; PSEXESVC in temp; task that runs mimikatz; LSASS dumps created by unknown tools.
- False Positives: documented admin operations, SCCM, or monitoring agents.

Section 5 — Event ID & Field Analysis (How to interpret core events)

This section provides field-level interpretation for critical Windows and Sysmon events to enable precise correlation and timeline building.

Key Fields

- 4624 - TargetUserName, LogonType, IpAddress, WorkstationName, AuthenticationPackage.
- 4625 - FailureReason, Status, IpAddress.
- 4648 - ProcessName (explicit credentials used), TargetUserName.
- 4698/4702 - TaskName, Action, Author.
- 7045 - ServiceName, BinaryPathName, CreatorName.
- Sysmon 1 - ProcessGuid, Image, CommandLine, ParentProcessGuid.
- Sysmon 3 - SourceIp, DestinationIp, DestinationPort, Image.
- PowerShell 4104 - ScriptBlockText, UserContext.

Analyst Actions

21. Always export raw EVTX or JSON event payloads (do not rely on parsed fields only).
22. Use ProcessGuid and ParentProcessGuid to join Sysmon process events reliably.
23. Retain full CommandLine and decoded PowerShell scriptblocks in evidence repository.

Red Flags vs False Positives

- Red Flags: 4648 showing psexec/wmic as ProcessName for low-priv user; 7045 creating service with binary in temp.

- False Positives: automation tools with proper signatures and tickets.

Section 6 — Timeline Construction & Correlation

Construct a master timeline from all sources and use robust join keys (ProcessGuid or PID+host+timestamp). Normalize to UTC and also show IST for local teams.

Analyst Actions

24. Define timeframe (T0 ±2h, expand as needed).
25. Export logs (EVTX/JSON), EDR snapshots, PCAP slices for window.
26. Normalize timestamps to UTC; create master CSV with fields: TimestampUTC, TimestampLocal(IST), EventSource, EventID, Host, User, Process, ParentProcess, CommandLine, SrcIP, DstIP, SHA256, Notes.
27. Join using ProcessGuid/ParentProcessGuid; where absent, use PID+Host+Time with caution.
28. Annotate links with confidence and produce attack-chain diagram.

Red Flags vs False Positives

- Red Flags: chain of ProcessCreate → NetworkConnect → 4624 on target within seconds; multiple hops by same account.
- False Positives: scheduled automated tasks with matching tickets and periodicity.

Section 7 — FP vs TP Verification (Decision Framework)

Use a documented scoring model and cross-checks (ticketing, baseline, asset criticality) to decide TP/FP and escalation.

Analyst Actions

29. Apply scoring model and thresholds; document each input.
30. Check ITSM for matching ticket and confirm via ops if ambiguous.
31. If TP, escalate to IR and containment. If FP, document and close with rationale.
32. Always log decision and supporting evidence (screenshots, raw events, ticket IDs).

Red Flags vs False Positives

- Red Flags: evidence of credential theft/persistence + multi-host lateral movement.
- False Positives: approved administrative automation with proper documentation.

Section 8 — Containment & Remediation (Tactical Playbook)

Contain attackers while preserving evidence; prioritize memory capture for credential theft cases; use EDR isolation where possible while keeping the agent alive.

Immediate Actions (minutes)

33. Notify Incident Lead, IT Ops, Legal as per incident comms plan.
34. Export EDR process tree and capture memory if LSASS suspected (EDR preferred).
35. Isolate host network via EDR (block lateral comms but keep agent connectivity if possible).
36. Disable compromised account(s) in AD and cloud identities; note timestamp and operator.
37. Block C2 IPs/domains at perimeter and in proxies/DNS.

Short-term Actions (hours)

38. Terminate malicious processes via EDR and record actions.

39. Remove persistence: delete scheduled tasks (export XML first), delete services (sc delete), remove Run keys (export before edit).
40. Quarantine suspicious files and compute hashes; submit to sandbox if safe.
41. Increase monitoring on IOCs across estate (blocklist, detections).

Long-term Actions (days)

42. Rotate credentials for compromised and related accounts; enforce password resets and revoke sessions.
43. Reimage hosts if integrity cannot be guaranteed; validate images are patched and hardened.
44. Perform root cause analysis and apply patching/hardening (disable SMBv1, restrict WinRM/RDP, enforce jump servers).
45. Update SIEM/EDR detection rules and whitelists; train ops on changes.

Red Flags vs False Positives

- Red Flags: C2 beacons persist after isolation; ability of attacker to re-establish persistence on reimaged hosts.
- False Positives: containment executed during approved maintenance — validate via ticket.

Section 9 — Forensic Analysis (Deep Technical)

Perform memory, file, and network analysis in an isolated lab; produce reproducible results and IOCs.

Analyst Actions

46. Set up isolated analysis VM for static/dynamic analysis.
47. Compute SHA256 for all artifacts and log in evidence repository.
48. Run Volatility `sekurlsa::logonpasswords` on lsass.dmp to extract credentials if memory captured.
49. Analyze binaries statically (PE headers, imports, strings) and dynamically in sandbox (Cuckoo).
50. Analyze PCAP for SMB/RDP/C2 sessions; extract files and reconstruct sessions.
51. Document all commands, outputs, and inference steps in a forensic report.

Red Flags vs False Positives

- Red Flags: plaintext credentials extracted from memory; unknown binaries contacting known-malicious IPs/domains.
- False Positives: documented IR forensic captures by authorized staff.

Section 10 — Post-Incident Reporting & Handover

Produce executive and technical reports, deliver IOCs, remediation evidence, and update stakeholders and playbooks.

Analyst Actions

52. Draft a 1–2 page executive summary: initial detection, impact, remediation status, business risk.
53. Assemble technical annex: full timeline CSV, raw logs (EVTX), decoded scriptblocks, artifact hashes, PCAP slices.
54. Prepare IOC list (accounts, hosts, file hashes, services/tasks, IPs/domains) and TLP marking.
55. Attach remediation proof: screenshots/logs showing disabled accounts, firewall rules, deleted tasks/services, reimage confirmations.
56. Distribute to IR, IT Ops, Legal, Exec Sponsor; schedule lessons learned and action-tracking.

Red Flags vs False Positives

- Red Flags: report lacks raw evidence or remediation proof; inconsistent timestamps across logs.
- False Positives: closing incident without attachments — unacceptable.

Section 11 — Continuous Log Monitoring & Detection Hygiene

Maintain detection posture, keep Sysmon/PowerShell logging enabled, centralize logs, and run periodic hunts and FP reviews.

Analyst Actions

57. Deploy Sysmon with recommended config capturing command-line, process GUIDs, network, and image load events.
58. Enable PowerShell Module and ScriptBlock logging; centralize to SIEM.
59. Create correlation rules combining auth+process+network signals.
60. Whitelist known automation (SCCM/Intune/backup) and maintain allowlist hashes.
61. Run quarterly FP clinic to tune noisy rules and add context enrichment.

Red Flags vs False Positives

- Red Flags: sudden spike in encoded PowerShell by standard accounts; many unexpected services/tasks created across servers.
- False Positives: scheduled, documented maintenance activities.

Section 12 — Recommendations (Prevention & Detective Controls)

Implement controls to reduce lateral movement risk and improve detection capabilities.

- Enforce least privilege; remove local admin rights on endpoints.
- Implement MFA for all remote access and admin accounts (including VPN).
- Use LAPS for local admin password rotation and JIT for privileged access.
- Network segmentation: restrict workstation→server RDP/SMB except via jump servers.
- Disable SMBv1 and minimize NTLM usage; use Kerberos where possible.
- Enable Sysmon and centralized PS logging; deploy EDR with auto-isolate capability for high-fidelity signals.

Red Flags vs False Positives

- Red Flags: privileged accounts without MFA; inability to enforce segmentation.
- False Positives: operational exceptions with compensating controls documented.

Section 13 — SIEM & EDR Rule Logic (Examples)

Example correlation recipes and pseudo-logic for SIEM/EDR rules. Use AD/asset context and ticket enrichment to reduce FP.

Example Rules

- High-Severity: Non-admin invokes PsExec AND Windows 7045 (service install) on the target host within 10 minutes -> Alert: HIGH.
- Critical: Any non-allowed process opens handles to lsass.exe (ProcessAccess) -> Auto-isolate host + capture memory.
- Medium: Powershell -EncodedCommand AND file copied to ADMIN$ within 5 minutes -> Alert: Medium.

Pseudo EDR Rule Logic

IF process.name == 'psexec.exe' AND user NOT IN admin_whitelist AND event EXISTS 7045 on target within 10m THEN raise HIGH

Implementation Notes

- Enrich alerts with AD group membership, asset criticality, ITSM ticket ID.
- Suppression during approved maintenance windows using ticket IDs reduces FP.
- Test rule logic against historical data and red-team events before production rollout.

Red Flags vs False Positives

- Red Flags: rules firing with corroborating EDR LSASS access and persistence artifacts.
- False Positives: maintenance ticket present and validated with ops.

Section 14 — Runbook: End-to-End Analyst Flow

A deterministic sequence for triage, collection, containment, analysis, remediation and reporting.

Runbook Steps (concise)

62. Triage (0–15m): validate alert, check user/host/ticket, compute score, assign owner.
63. Evidence Collection (0–60m): export logs, EDR snapshots, capture memory if LSASS suspected.
64. Containment (as needed): isolate hosts, disable accounts, block IOCs.
65. Forensic Analysis (hours-days): memory & binary analysis, PCAP review, build timeline.
66. Remediation & Report (days): rotate creds, reimage as needed, deliver exec+tech reports, update detections.

Red Flags vs False Positives

- Red Flags: inability to capture volatile evidence before containment; agent offline during isolation.
- False Positives: actions taken for documented admin tasks (verify before action).

Section 15 — Detection Tuning & Best Practices

Ongoing tuning and FP reviews keep detection relevant and accurate.

67. Weekly FP review for noisy rules; update whitelist and enrichments.
68. Require multi-signal correlation for high-severity alerts.
69. Maintain rule versioning and test changes on historical/backfilled data.
70. Add asset and AD enrichment to every alert to improve context.

Red Flags vs False Positives

- Red Flags: over-tuned rules that stop detecting real attacks.
- False Positives: initial ramp-up noise before enrichment.

Section 16 — KPIs & SOC Metrics

Measure performance and improvement areas.

- MTTD (Mean Time to Detect): goal target (example 15–30 minutes).
- MTTC (Mean Time to Contain): track trends and targets.
- %TP per rule: improve signal quality.
- Memory capture success rate: % of incidents where memory was captured when needed.

Section 17 — Lessons Learned & Tabletop Exercises

Convert incident outcomes into actionable improvements and validate via tabletop exercises.

71. Hold post-mortem within 7–14 days; capture action items with owners and due dates.
72. Update playbooks and SIEM/EDR rules based on findings.
73. Run a tabletop focusing on the same vector and measure MTTD/MTTC improvements.

Red Flags vs False Positives

- Red Flags: same vector recurs; action items unimplemented.
- False Positives: N/A.

Section 18 — Appendix: Useful Commands & Forensic Snippets

- Export Security log: wevtutil epl Security C:\Forensics\Security.evtx /ow:true
- Export Sysmon log: wevtutil epl Microsoft-Windows-Sysmon/Operational C:\Forensics\Sysmon.evtx /ow:true
- Capture LSASS memory (EDR preferred): procdump -ma lsass.exe C:\Forensics\lsass.dmp
- Hash file: certutil -hashfile <path> SHA256
- Export scheduled task XML: schtasks /Query /TN "UpdateSvc" /XML > C:\Forensics\UpdateSvc.xml
- Query service config: sc qc PSEXESVC
- Decode PowerShell base64 (lab): [Text.Encoding]::Unicode.GetString([Convert]::FromBase64String("<BASE64>"))

Section 19 — Artifact Folder & Report Packaging

Standardized evidence packaging ensures reproducibility and auditability. Use consistent folder structure, compute hashes, and maintain chain-of-custody.

- /evidence/INC-YYYYMMDD-XXX/
- /logs/ (Security/Sysmon/PowerShell EVTX)
- /memory/ (lsass.dmp, other dumps)
- /files/ (suspicious binaries and scripts)
- /network/ (pcap slices, NetFlow)
- /reports/ (executive_summary.pdf, forensic_report.pdf)
- /hashes/ (hashes.txt)
- /chain_of_custody/ (chain_of_custody.csv)
- /screenshots/ (containment evidence)

Example 1 — PsExec → Scheduled Task → Mimikatz → RDP

Objective

End-to-end example showing how an initial workstation compromise leads to lateral movement, credential theft, and interactive access to critical servers.

Scenario Narrative (Detailed)

1) Initial foothold: user 'alice' on HOST-A falls for phishing; attacker obtains her credentials and drops a small toolkit. 2) Lateral execution: attacker uses PsExec from HOST-A to execute on HOST-B, creating a PSEXESVC service. 3) Persistence/credential theft: attacker installs scheduled task 'UpdateSvc' on HOST-B that runs mimikatz to dump LSASS memory and extract privileged credentials. 4) Pivot: attacker uses harvested admin credentials via WMIC to execute on HOST-C. 5) Interactive access: attacker RDPs into HOST-D as admin and performs data access or further compromise.

Logs / Events / Artifacts

- HOST-A: Sysmon 1 process create psexec.exe with CommandLine '\\HOST-B -u ... -p ...'.
- HOST-B: Security 7045 service install PSEXESVC; Sysmon 11 file created in C:\Windows\Temp; 4698 task 'UpdateSvc' created.
- HOST-B: EDR indicates process reading lsass.exe and creating lsass.dmp; Volatility later recovers credentials.
- HOST-C: Sysmon 3 network connect from HOST-B and 4624 logon for same admin account.
- HOST-D: 4624 LogonType=10 recorded for admin account; TerminalServices 1149 connection entry.

Telemetry (what to check)

- ProcessCreate events on HOST-A and HOST-B with ProcessGuid and CommandLine.
- Event 7045 and 4698 entries on HOST-B with Task/Service path and creator.
- PowerShell 4104 if PowerShell used to execute mimikatz or stage tools.
- EDR read handles on lsass.exe and file hashes of mimikatz/procdump.
- Network flows (SMB/RPC/WinRM) between HOST-A/B/C and RDP to HOST-D.

Analyst Actions (step-by-step, commands)

74. Triage: identify 'alice' as standard user and confirm no ticket. Record alert details.
75. Evidence: export Security/Sysmon EVTX from HOST-A/B/C/D and EDR process trees.
76. Memory: capture lsass.dmp from HOST-B (EDR or procdump -ma lsass.exe) before killing processes.
77. Contain: isolate HOST-A and HOST-B via EDR (network containment), disable 'alice', block C2 IPs.
78. Remediate: delete scheduled task `schtasks /Delete /TN "UpdateSvc" /F` and delete PSEXESVC `sc stop PSEXESVC && sc delete PSEXESVC` after export, rotate credentials for compromised admin accounts, reimage hosts if necessary.
79. Forensics: run Volatility `sekurlsa::logonpasswords` against lsass.dmp and document recovered credentials and where they were used.

Illustrative Timeline (example rows)

Time (IST)	User	Source Host	Target Host	Process	Event IDs	Command/Notes
10:05	alice	HOST-A	HOST-B	psexec.exe	Sysmon1,7045	\\HOST-B -u alice -p *****

10:09	SYSTEM	HOST-B	HOST-B	schtasks.exe	4698	Create 'UpdateSvc' -> mimikatz.exe
10:12	SYSTEM	HOST-B	HOST-B	mimikatz.exe	EDR/Volatility	LSASS read; creds dumped
10:18	admin	HOST-B	HOST-C	wmic.exe	Sysmon19-21,4624	Remote exec to HOST-C
10:25	admin	HOST-C	HOST-D	mstsc.exe	4624 LT=10	RDP interactive session

Red Flags vs False Positives

- Red Flags: scheduled task executing mimikatz by a non-admin; lsass memory dumped by procdump/mimikatz; multi-hop lateral movement in short period.
- False Positives: admin-run maintenance with documented ticket and signed binaries.

Example 2 — PowerShell Remoting + Pass-the-Hash → SMB Ransomware

Objective

Demonstrate credential theft (PtH) combined with PowerShell remoting to stage ransomware via SMB.

Scenario Narrative (Detailed)

1) Initial compromise: helpdesk VPN credentials are stolen (no MFA). 2) Attacker uses Pass-the-Hash (NTLM) to authenticate to DC and request service tickets or explicit logons (4776/4648). 3) Using PowerShell remoting (WinRM), attacker copies and registers finance_ransom.dll via ADMIN$ on HOST-G and creates scheduled task to run it. 4) DLL executed and spreads via SMB to other hosts, encrypting files.

Logs / Events / Artifacts

- VPN logs: external IP and username, lack of MFA challenge logs.
- DC logs: 4776 (NTLM auth), 4648 (explicit credentials).
- PowerShell 4104: EncodedCommand invoking remote copy and execution.
- Sysmon 11 on HOST-G: file create finance_ransom.dll; 4698 task create 'FinanceUpdate'.
- EDR: suspicious DLL behavior, file-modifying activity, and potential process injection.

Telemetry (what to check)

- VPN gateway logs and device fingerprinting for the external IP.
- Sequence of DC auth events (time, source ip, account).
- PowerShell scriptblock decoded content and target host commands.
- Sysmon file create and task creation events on HOST-G; file hashes across estate.

Analyst Actions (step-by-step, commands)

80. Immediate: disable 'helpdesk' account and revoke VPN session tokens; block source IP at edge.
81. Export relevant DC logs, PowerShell 4104 entries, Sysmon/Security on HOST-G and suspect hosts.
82. Isolate HOST-G and collect EDR process tree and PCAP slice.
83. Delete scheduled task after export: schtasks /Delete /TN "FinanceUpdate" /F; quarantine finance_ransom.dll and compute SHA256.
84. Hunt for DLL hash across environment and quarantine infected hosts; restore from backups where needed; rotate affected credentials.

Illustrative Timeline (example rows)

Time (IST)	User	Source Host	Target Host	Process	Event IDs	Command/Notes
21:02	helpdesk	VPN-GW	DC01	auth	4776/4648	NTLM auth from external IP 198.51.100.45
21:07	helpdesk	DC01	HOST-G	powershell.exe	4104/Sysmon1	Invoke-Command push DLL to ADMIN$

21:09	SYSTEM	HOST-G	HOST-G	schtasks.exe	4698	Create 'FinanceUpdate' scheduled task
21:12	SYSTEM	HOST-G	HOST-G	rundll32.exe	Sysmon1	Execute finance_ransom.dll
21:20	SYSTEM	HOST-G	HOST-H	smb.exe	NetFlow/Firewall	SMB lateral encryption activity

Red Flags vs False Positives

- Red Flags: PtH usage (NTLM/4648) without MFA, encoded PS remoting, DLL staged in system directories, rapid SMB propagation.
- False Positives: rare; authorized PowerShell remoting only in managed/approved automation.

Final Checkpoint — Analyst Verification & Handover

✓ Detection & Triage

- ☐ Alert metadata (rule, source host, target host, user, timestamp, process) recorded.
- ☐ User type (standard, admin, service) and host type (workstation, server, DC, jump box) identified.
- ☐ Business hours vs. off-hours and IP/Geo context reviewed.
- ☐ False positive vs. true positive decision documented with evidence.

✓ Evidence Collection

- ☐ Security, Sysmon, PowerShell, and relevant EVTX logs exported for ±2 hours around detection.
- ☐ EDR process tree snapshots exported (JSON / screenshots).
- ☐ Memory captured (if LSASS access suspected) and SHA256 hash computed.
- ☐ Suspicious files collected and hashes calculated.
- ☐ Scheduled task XMLs and service configs exported before removal.
- ☐ Network evidence (PCAP slices / NetFlow) collected for implicated hosts.
- ☐ Chain-of-custody form updated with all items.

✓ Analysis

- ☐ Timeline constructed (UTC + IST) with correlated Event IDs, Sysmon ProcessGuids, network flows.
- ☐ Parent-child process chains verified against known attack tools (PsExec, WMIC, PowerShell).
- ☐ Encoded PowerShell decoded and reviewed in lab.
- ☐ Credential theft validated (Sysmon 10 / EDR LSASS access) and accounts flagged.
- ☐ Persistence (4698, 4702, 7045, WMI 19–21) reviewed and removed after export.

✅ Containment

- [ ] Hosts isolated (via EDR/VLAN) while maintaining agent connectivity.
- [ ] Compromised accounts disabled; passwords rotated; active sessions revoked.
- [ ] Malicious processes terminated and persistence removed.
- [ ] C2 domains/IPs blocked at proxy/firewall/DNS.

✅ Remediation

- [ ] Affected hosts reimaged if integrity uncertain.
- [ ] Security patches applied (RDP, SMB, WinRM, etc.).
- [ ] Least-privilege and segmentation reviewed (restrict workstation→server RDP/SMB).
- [ ] MFA enforced for admin and remote accounts.

✅ Reporting & Lessons Learned

- [ ] IOC list (hosts, accounts, hashes, IPs/domains) compiled and distributed.
- [ ] Executive summary and technical forensic annex completed with evidence.
- [ ] SOC rules tuned based on incident (SIEM/EDR correlation).
- [ ] Lessons learned meeting scheduled; action items tracked.

Key Analyst Reminder:
If any box is left unchecked, the investigation is not complete.
A senior analyst or IR lead should validate this checkpoint before closing the case.

Lateral Movement Telemetry + LogonType Reference

Event ID	Key Telemetry Fields	Why Used	Example in Lateral Movement
4624 (Successful Logon)	`TargetUserName`, `IpAddress`, `WorkstationName`, `AuthenticationPackage`, **LogonType** • 2 = Interactive (console) • 3 = Network (SMB, PsExec) • 4 = Batch (scheduled task) • 5 = Service (service account	Identifies how logon occurred. **Types 3, 9, 10 are critical for lateral movement detection.**	Standard user logs into a server via **LogonType=10 (RDP)** at 2 AM from a workstation.

Event ID	Key Telemetry Fields	Why Used	Example in Lateral Movement
	logon) • 7 = Unlock • 8 = NetworkClearText (rare) • 9 = NewCredentials (RunAs / PtH) • 10 = RemoteInteractive (RDP) • 11 = CachedInteractive (offline) • 12 = RemoteCredentialGuard (secure RDP) • 13 = S4U (delegation) • 14 = Proxy • 15 = RemoteInteractiveUnlock		
4625 (Failed Logon)	`FailureReason`, `Status`, `IpAddress`	Detect brute force / password spray before success.	Multiple failures from one IP → followed by a 4624 success.
4648 (Logon with Explicit Creds)	`ProcessName`, `TargetUserName`, `LogonProcessName`	Detects Pass-the-Hash/Ticket and credential misuse.	`psexec.exe` invoked with explicit credentials.
4672 (Special Privileges Assigned)	`PrivilegeList`	Detects admin rights assignment (SeDebugPrivilege).	Account gains `SeDebugPrivilege` → dumps LSASS.
4698 / 4702 (Scheduled Task)	`TaskName`, `Author`, `Command`	Persistence or remote execution.	Attacker creates scheduled task `UpdateSvc` to run mimikatz.
7045 (Service Installed)	`ServiceName`, `BinaryPathName`, `ServiceType`, `AccountName`	Detects PsExec service or persistence.	PsExec installs `PSEXESVC` on HOST-B.
4768 / 4769 (Kerberos Tickets)	`ServiceName`, `TicketOptions`, `EncryptionType`	Detects Kerberoasting or unusual TGS requests.	Non-admin requests `MSSQLSvc` service ticket.
4776 (NTLM Auth)	`WorkstationName`, `IpAddress`, `AuthenticationPackage`	Detects NTLM usage (PtH).	Workstation authenticates to DC with NTLM.

Event ID	Key Telemetry Fields	Why Used	Example in Lateral Movement
5140 (SMB Share Accessed)	`ShareName`, `RelativeTargetName`, `IpAddress`	Detects SMB staging of tools.	`mimikatz.exe` copied via ADMIN$ share.
Sysmon 1 (Process Create)	`Image`, `ParentImage`, `CommandLine`, `User`, `ProcessGuid`	Detect process chains and suspicious execution.	`winword.exe → cmd.exe → powershell.exe -enc ….`
Sysmon 3 (Network Connect)	`SourceIp`, `DestinationIp`, `DestinationPort`, `Image`	Detect east–west lateral traffic.	PowerShell connects to port 5985 (WinRM).
Sysmon 8 (CreateRemoteThread)	`SourceProcessGuid`, `TargetProcessGuid`, `TargetImage`	Detects injection into other processes.	Injection into `lsass.exe`.
Sysmon 10 (Process Access)	`SourceProcess`, `TargetProcess`, `GrantedAccess`	Detects LSASS memory access for credential theft.	`procdump.exe` accessing `lsass.exe`.
Sysmon 11 (File Create)	`TargetFilename`, `Image`, `Hashes`	Detect attacker payloads being staged.	`finance_ransom.dll` dropped in `C:\Windows\Temp`.
Sysmon 19–21 (WMI Events)	`Consumer`, `Filter`, `Binding`, `Operation`	Detect WMI persistence and remote execution.	WMI subscription triggers PowerShell.
4103 (PS Module Logging)	`ModuleName`, `CommandLine`	Detect malicious PowerShell modules loaded.	`Invoke-Mimikatz` imported.
4104 (PS ScriptBlock Logging)	`ScriptBlockText`, `UserContext`, `CommandLength`	Captures obfuscated PowerShell commands.	Attacker runs `-EncodedCommand` payload.
1149 (RDP Logon Success)	`User`, `SourceIP`, `SessionID`	Confirms RDP session after 4624 authentication.	Admin RDPs into server from unusual workstation.

End-to- End lateral movement playbook **for some scenarios**

Scenario 1: Pass-the-Hash (PtH)

Context & Risk

- Attacker dumps **NTLM hashes** from a host and reuses them to log in to other systems without knowing passwords.
- Often linked with **Mimikatz** or LSASS dumps.
- High risk because hashes can allow **admin access** domain-wide.

Step-by-Step Investigation

1. **Isolate host** suspected of dumping credentials (to stop lateral spread).
2. **Memory dump** → check if LSASS process accessed by non-legit processes.
3. **Event Logs (Security.evtx)**:
    - Event ID `4624` (Logon Type 3 = Network). Look for same account accessing multiple hosts.
    - Event ID `4625` (failed logons). Burst of failures may show password spray/hash mismatch.
4. **Netstat**: identify if host has open SMB (445) connections to multiple internal systems.
5. **Hunt for tools** (`mimikatz.exe`, renamed binaries, suspicious procdump.exe).
6. **Check Prefetch** → reveals execution history of mimikatz/procdump.

Indicators & Validation

- Same **NTLM hash** used across systems.
- Processes attaching to **lsass.exe**.
- Lateral SMB sessions without password resets.

MITRE Mapping

- **T1003 – Credential Dumping**
- **T1078 – Valid Accounts**
- **T1021.002 – SMB/Windows Admin Shares**

Remediation & Hardening

- Force password resets for all compromised accounts.
- Deploy **LSA protection** to prevent lsass dumping.
- Disable NTLM where possible (enforce Kerberos + MFA).

Scenario 2: Remote Service Creation (PsExec / sc.exe)

Context & Risk

- Attackers create malicious services remotely to execute payloads.

- PsExec installs `psexesvc.exe`.
- Dangerous because it allows stealthy persistence.

Step-by-Step Investigation

1. **Check Event ID `7045`** (new service installation).
2. **Sysmon (if available)** → Event ID `1` for PsExec or `sc.exe`.
3. **Services Query** (`sc queryex`): look for unusual ImagePaths (e.g., temp folders).
4. **Registry**: HKLM\SYSTEM\CurrentControlSet\Services\ (new suspicious entries).
5. **Collect service binary** → hash + TI lookup.

Indicators & Validation

- Suspicious service names (`XWinSvc`, `Updater`, etc.).
- `psexesvc.exe` present.
- Execution from accounts not in IT group.

MITRE Mapping

- **T1021.002 – SMB/Windows Admin Shares**
- **T1543.003 – Windows Service Creation**
- **T1059 – Command Execution**

Remediation & Hardening

- Block PsExec/sc.exe in non-admin environments.
- Implement application whitelisting.
- Restrict admin service installation to IT group only.

Scenario 3: Admin Share Exploitation (C$ / ADMIN$)

Context & Risk

- Attackers copy tools via `\\HOST\C$` or `ADMIN$` shares, then execute remotely (via WMI, tasks).

Step-by-Step Investigation

1. **Event ID `5140`**: identify access to `ADMIN$` or `C$`.
2. **File System**: check for executables in `C:\Windows\System32` or `C:\Users\Public`.
3. **Event ID `4698`**: suspicious scheduled tasks launching dropped binaries.
4. **Sysmon Event ID 11** (File Create) → check new file drop in admin shares.

Indicators & Validation

- File creation in admin shares.
- Scheduled task execution of remote files.
- Odd binaries named `update.exe`, `svc.exe`.

MITRE Mapping

- **T1021.002 – SMB Admin Shares**
- **T1105 – Ingress Tool Transfer**
- **T1053 – Scheduled Task/Job**

Remediation & Hardening

- Disable default ADMIN$ where not needed.
- Use firewall rules to block SMB lateral traffic.
- Restrict local admin accounts with LAPS.

Scenario 4: Credential Dumping + RDP Hopping

Context & Risk

- Attacker dumps credentials then pivots using RDP across systems.
- Dangerous for **domain-wide escalation**.

Step-by-Step Investigation

1. **Memory dump** → check for LSASS access.
2. **Event ID `4624`**: Logon Type 10 (RDP). Multiple across servers = suspicious.
3. **`qwinsta` or `query user`**: list active sessions.
4. **Prefetch**: check `mstsc.exe` usage.
5. **Event ID `4778/4779`**: RDP reconnect/disconnect logs.

Indicators & Validation

- Burst of RDP logons from one workstation.
- Logons outside work hours.
- New accounts suddenly using RDP.

MITRE Mapping

- **T1003 – Credential Dumping**
- **T1021.001 – RDP**
- **T1078 – Valid Accounts**

Remediation & Hardening

- Enable MFA for RDP.
- Restrict RDP to jump servers.
- Monitor for anomalous RDP logons in SIEM.

Scenario 5: Kerberos Ticket Abuse (Golden Ticket)

Context & Risk

- Attacker forges Kerberos TGTs/TGSs to impersonate users.
- Golden Ticket = **domain dominance**.

Step-by-Step Investigation

1. **Event ID `4769`**: check encryption types (RC4 vs AES).
2. **Event ID `4624`**: logon with impossible lifetime tickets.
3. **Memory dump**: check for Mimikatz Kerberos module.
4. **AD**: check if tickets issued for non-existent accounts.

Indicators & Validation

- Abnormal TGT lifetime.
- Kerberos tickets for fake accounts.
- Mimikatz artifacts.

MITRE Mapping

- **T1558 – Kerberos Ticket Forgery**
- **T1078 – Valid Accounts**
- **T1098 – Account Manipulation**

Remediation & Hardening

- **Reset `krbtgt` twice** (invalidate forged tickets).
- Disable RC4; enforce AES.
- Monitor DCs for abnormal Kerberos activity.

Scenario 6: Standalone Host (No SIEM/EDR)

Context & Risk

- No central monitoring. Analyst relies on manual inspection.

Step-by-Step Investigation

1. Isolate system (disconnect network).
2. Collect volatile data: memory dump, `tasklist`, `netstat`.
3. Export logs → Security (4624, 4625), System, Application.
4. Check persistence: `schtasks`, `sc queryex`, `autoruns`.
5. Collect suspicious binaries → hash and check TI.
6. If confirmed compromise → full disk image + reimage host.

Indicators & Validation

- Event ID `7045` (new service).
- Event ID `4698` (task creation).
- Abnormal logons (`4624/4625`).

MITRE Mapping

- **T1078 – Valid Accounts**
- **T1543 – Create/Modify System Process**
- **T1053 – Scheduled Task**
- **T1105 – Ingress Tool Transfer**

Remediation & Hardening

- Deploy EDR/SIEM immediately post-incident.
- Harden standalone system baseline.
- Limit use of local admin accounts.

Analyst Quick Reference Matrix

Scenario	Logs to Check	Suspicious Signs	MITRE TTPs
Pass-the-Hash	4624 (Type 3), LSASS memory	Same hash reused, Mimikatz	T1003, T1078, T1021.002
Remote Service Creation	7045, Sysmon 1	New services, PsExec	T1021.002, T1543.003
Admin Shares	5140, 4698, Sysmon 11	File drop in ADMIN$, tasks	T1021.002, T1105, T1053
RDP Hopping	4624 (Type 10), 4778/4779	Unusual RDP logons	T1003, T1021.001, T1078
Kerberos Abuse	4769, 4624	Odd ticket lifetimes, RC4	T1558, T1078, T1098
Standalone Host	4624, 4625, 7045, 4698	Suspicious services/tasks	T1078, T1543, T1053, T1105