

# Incident Response: Analysis

<https://t.me/learningnets>





# Alexis Ahmed

Offensive Security/Red Team Instructor @INE  
Senior Pentester & Red Team Lead @HackerSploit

---

<https://t.me/learningnets>

# Key Concepts

- + Deep-dive forensic investigations
- + Evidence triage & endpoint analysis
- + Network & log data interpretation
- + Tool-driven analysis workflows
- + Real-world incident response practices

# MAJOR TOPICS

- + First response & deep-dive analysis techniques
- + Endpoint triage: Windows, Linux, live & offline systems
- + Log analysis with Splunk, EvtxECmd & Sigma
- + PCAP analysis & network investigations
- + Evidence timeline creation & correlation



## LEARNING OUTCOMES

- + Confidently perform endpoint & network analysis
- + Identify & interpret critical forensic artifacts
- + Use industry-standard tools in live investigations
- + Differentiate analysis strategies based on scenario

# PREREQUISITES

- + Completion of *Incident Response: Detection* recommended
- + Basic understanding of operating systems & networking



# LET'S GO!

<https://t.me/learningnets>





# Bridging the Gap: Detection & Analysis

<https://t.me/learningnets>



# Bridging the Gap: Detection → Analysis

In the previous course, *Incident Response – Detection*, you learned how logs are collected, parsed, correlated, and triaged so that only credible alerts reach an incident responder.

**Detection answers the questions:** “Something suspicious just happened, do we care?”

The moment a Tier 2 analyst/Incident responder confirms “**Yes, we do,**” the workflow ***pivots to Analysis, turning that alert into a precise understanding of what happened, how far it spread, and how to stop it.***

# Bridging the Gap: Detection → Analysis



Picture a modern museum at night...

From a security perspective, intrusion detection is facilitated via the laser-grid sensors in the exhibit halls.

# Bridging the Gap: Detection → Analysis



The moment a beam is broken/disrupted, a silent alarm tells (alerts) the security personnel something is **moving where it shouldn't be.**

This silent alarm triggers the security team to investigate (analyse) the cause of the alarm.

# Bridging the Gap: Detection → Analysis



Analysis/investigation of the root cause of the alarm involves the security team fanning out with flashlights and reviewing CCTV playback.

This is done to determine whether there actually is a legitimate intrusion (or whether it is a false alarm/positive), if legitimate, determine where the intruder is, and what has been stolen (scope).



They trace the intruder's path, note which display cases were opened, check for missing artifacts, and decide which galleries to seal off before the thief can escape.

A man with glasses is shown in profile, looking at a computer monitor in a dark room. The monitor displays some code or data. The overall scene is dimly lit, with the primary light source being the screen. An orange vertical bar is on the left side of the image.

# The Analysis Phase of IR

<https://t.me/learningnets>



# The Analysis Phase of IR

The analysis phase of incident response is the disciplined investigation that begins the moment an escalated alert is accepted as a credible security event/incident.

**Detection** determines that *something suspicious occurred/occurring*, **analysis determines what, how, where, and how far it went (scope)**.

**Working from a combination of evidence sources** such as SIEM events, endpoint telemetry, memory captures, disk images, network packet captures, and open-source intelligence (OSINT), **responders validate the incident, reconstruct the attacker's actions, and assess business impact**.

# The Analysis Phase of IR

**Analysis exists to remove uncertainty. *False positives* are eliminated, *true positives* are confirmed, and *silent false negatives* are uncovered by pivoting through related logs and artifacts.**

**By accurately scoping affected hosts, users, data sets, and timelines, responders prevent both over-reaction (needless outages) and under-reaction (missed footholds).**

The phase also seeks to ***identify the root cause of the incident*** (phishing e-mail, unpatched web vulnerability etc.) ***so that remediation addresses the real weakness rather than just the symptoms.***

# The Analysis Phase of IR

## Pivoting through logs... what does that mean?

Pivoting through logs is the *investigative practice* of starting with **one piece of evidence** in your **log data** like an IP address, process hash, username, file path, URL, etc, and *using it as a pivot point* to *discover related events that broaden the picture of an incident.*

# Why Analysis Exists

- + **Verification** – Ensure the incident is real (eliminate false positives).
- + **Scoping** – Identify all affected hosts, users, data, and timelines.
- + **Root-Cause Discovery** – Understand the entry vector and exploited weakness.
- + **Impact Assessment** – Quantify business and regulatory ramifications.
- + **Decision Support** – Provide actionable intelligence for containment, eradication, and recovery.

# What Analysis Entails

- + **Evidence Preservation** – Capture volatile memory, secure log exports, and disk images to ensure data integrity.
- + **Timeline Reconstruction** – Merge endpoint, network, and cloud events into a minute-by-minute chronology of attacker activity.
- + **IOC Expansion & Hunting** – Extract new hashes, domains, IPs, registry keys, and beacon patterns; search enterprise-wide to reveal additional compromise.
- + **Root-Cause & Impact Analysis** – Determine the entry vector, exploited vulnerabilities, lateral-movement paths, and any data touched or exfiltrated.
- + **Business Context Mapping** – Cross-reference asset criticality and regulatory scope (PCI, PHI, PII) to assign the correct severity and compliance actions.

# What Analysis Entails

Category	Concrete Activities & Tools
Evidence Preservation	Memory capture (Velociraptor, WinPMem), disk imaging, log export to secure bucket.
Timeline Reconstruction	Merge Sysmon, Windows Event, EDR, NetFlow, Zeek into a unified chronology (Timesketch, Plaso).
IOC Expansion & Hunting	Extract new hashes/IPs/domains; pivot enterprise-wide with SIEM or EDR search.
Host Artefact Analysis	Examine processes, registry, scheduled tasks, persistence keys, \$MFT/USN (Windows) or journald/ext journal (Linux).
Malware & Script Triage	Static strings, sandbox run, identify C2 config, map to ATT&CK techniques.
Network Traffic Verification	Decrypt or examine PCAPs for beaconing, exfil patterns; correlate with DNS queries.
Business Context Mapping	Cross-reference asset criticality (CMDB), data sensitivity (PCI, PHI), user privilege.

# Outputs & Deliverables

Deliverable	Description	Immediate Use
<b>Incident Timeline</b>	Unified chronology of attacker actions and defender observations.	Prioritises containment order and supports legal/audit review.
<b>Scope Matrix</b>	List of all compromised systems, accounts, and data types.	Guides isolation, notification, and recovery procedures.
<b>IOC Package</b>	Validated hashes, IPs, domains, YARA/SIGMA rules.	Blocks C2, enables enterprise-wide hunts, tunes detections.
<b>Root-Cause Report</b>	Detailed explanation of entry vector and exploited weakness.	Drives patching, configuration changes, and security control improvements.
<b>Detection Gap Analysis</b>	List of missed signals or noisy rules.	Feeds detection-engineering backlog to reduce future MTTR/FP rates.

A man with glasses is shown in profile, looking at a computer monitor in a dark room. The monitor displays some code or data. The overall scene is dimly lit, with the primary light source being the screen. An orange vertical bar is on the left side of the image.

# Transition to the Next IR Phases

<https://t.me/learningnets>



# Transition to the Next IR Phases

The analysis phase hands off a *fully scoped, evidence-rich incident* to containment and eradication teams.

Isolation commands, firewall blocks, EDR quarantines, and patch instructions are grounded in the timeline and IOC package.

**Recovery teams** rely on the **same outputs** to validate that systems are clean and to schedule **safe return to production**.

*Without a rigorous analysis phase, downstream actions risk being mis-targeted, incomplete, or out of proportion to the true impact, leaving the organisation vulnerable to reinfection or regulatory penalties.*

# Transition to the Next IR Phases

## 1. Containment

- + Isolation decisions use the scope matrix and timeline to prioritise high-value or actively beaconing hosts.
- + Blocking actions apply the IOC package to firewalls, proxies, and EDRs.

## 1. Eradication

- + Removal scripts (delete persistence keys, scheduled tasks) are crafted from artefacts found in host analysis.
- + Patch guidance leverages the root-cause report to close exploited vulnerabilities.

## 1. Recovery

- + Validation checks rely on updated detection rules to confirm no further malicious activity.
- + Business impact reports use the impact assessment to inform stakeholders and regulators.

# Key Take-Away

Detection rings the alarm; Analysis turns that alarm into a clear, actionable picture.

Without thorough analysis, containment may be incomplete, eradication mis-targeted, and recovery short-lived.

**Mastering this phase ensures every subsequent IR action is decisive, efficient, and fully justified.**

A man with glasses is shown in profile, looking at a computer screen in a dark room. The screen displays some code or data. The overall atmosphere is professional and focused.

# First Response: The First 5 Minutes

<https://t.me/learningnets>



# What is “First Response” / “Hot Triage”

When an alert is escalated, the responder’s very *first actions* are called first response (often “hot triage”).

It is a *flash-assessment* phase that converts “This looks bad” into “Yes or no, and what do we do in the next hour?”

**First Response** is done long before disk images, memory forensics, or board-room briefings begin.

# Purpose of First Response

Goal	Why it matters
<b>Validate the alert</b>	Eliminate false escalations before mobilising people or taking production-impacting actions.
<b>Preserve volatile evidence</b>	Live network connections, RAM artefacts, and log buffers can vanish within minutes.
<b>Estimate scope &amp; risk</b>	Decide whether one host, an entire subnet, or cloud tenants are involved.
<b>Trigger rapid containment if needed</b>	Early host isolation or IP blocking can stop data theft or ransomware detonation.

# Typical Activities (2 - 10 Minutes)


- + Open & orient — Read the ticket, note asset criticality, the SLA, and any auto-containment already applied.
- + Re-run the triggering query with a slightly wider time window to confirm it isn't a parsing glitch or dev-lab traffic.
- + Pivot once on the key IOC (IP, hash, user) to spot obvious lateral spread or repeat hits.
- + Snapshot volatile data if the host is still online (live memory capture, open-file list, current network sockets).
- + Take a containment decision — Immediate isolation/block if the activity is ongoing and risk is high, otherwise flag for coordinated containment later.
- + Document & notify — One or two clear lines in the case record plus a ping to the shift lead/asset owner.

# How it Differs from Full/Deep Analysis

Hot Triage (First Response)	Full/Deep Analysis
2–10 minutes	Minutes to hours (or days).
High-level scoping, stop-the-bleed decisions	Deep host & network analysis & forensics, root-cause, impact quantification.
Limited, fast queries & live-response tools	Full SIEM pivots, memory & disk imaging, malware reverse engineering.
Executed by on-call responder	May involve entire IR team, specialists, management.

# Outputs/Outcomes of First Response

- + **Validation statement** – “Confirmed malicious PowerShell on FIN-WS-01.”
- + **Preliminary scope** – affected host(s), user(s), earliest timestamp.
- + **Immediate actions taken** – isolation, firewall block, password reset.
- + **Ticket update** – evidence snippets and next steps for detailed analysis.

A man with glasses is shown in profile, looking at a computer screen in a dimly lit room. The background is dark, with some faint light from the screen and other monitors. The overall mood is professional and focused.

# The First 5 Minutes: Rapid Incident-Validation Checklist

<https://t.me/learningnets>



# The First 5 Minutes: Rapid Incident-Validation Checklist

The clock starts the instant an escalated alert hits your queue.

**These first five minutes** decide whether *you contain* a breach quickly or **waste time** on a false alarm.

The next slide contains a compact **Rapid Incident-Validation checklist** that can be used every time you take ownership of a new case.

# The First 5 Minutes: Rapid Incident-Validation Checklist

Minute	Action	Purpose
<b>0 – 1 min → Open &amp; Orient</b>	<ul style="list-style-type: none"><li>• Acknowledge the ticket.</li><li>• Read the Tier-1 summary: rule name, severity, host/user, any auto-containment.</li><li>• Note SLA deadline.</li></ul>	<b>Avoid duplicating work and know what triage already handled.</b>
<b>1 – 2 min → Re-run &amp; Broaden Search</b>	<ul style="list-style-type: none"><li>• Re-execute the original SIEM query ±5 min.</li><li>• Inspect raw events for parsing errors or test traffic.</li></ul>	<b>Confirm the alert is real (eliminate false escalation/positives).</b>
<b>2 – 3 min → Scope Snapshot</b>	<ul style="list-style-type: none"><li>• Pivot on IP, hash, user: “Who else? Where else?”</li><li>• Check asset criticality and prior alerts on entity.</li></ul>	<b>Identify the initial blast radius/scope; spot lateral movement early.</b>
<b>3 – 4 min → Containment Decision</b>	<ul style="list-style-type: none"><li>• Isolate host or block IOC if beaconing/impact is active.</li><li>• Otherwise flag “containment pending” for later plan.</li></ul>	<b>Prevent ongoing damage while keeping business impact reasonable.</b>
<b>4 – 5 min → Document &amp; Notify</b>	<ul style="list-style-type: none"><li>• Add concise note: validation result, scope, action taken.</li><li>• Attach raw evidence.</li><li>• Alert stakeholders (on-call lead, asset owner).</li></ul>	<b>Creates audit trail, keeps the team synced and starts MTTR counter.</b>

A man with glasses is shown in profile, working at a computer in a dark room. The background is dark, with the light from the computer screen illuminating his face and the desk. The overall mood is professional and focused.

# Example Scenario

<https://t.me/learningnets>



# The First 5 Minutes: Example Scenario

At **02:17 AM** the SOC's paging system flags a high-severity alert:

“An outbound PowerShell script on FIN-APP-SVR01 (a finance server) contacted 45.145.12.77, an IP tied to the APT 29 infrastructure seen in recent threat-intel feeds.”

A **Tier-1 analyst** has already ***triaged the alert, judged it credible,*** and **escalated the ticket to you,** the ***on-call Incident Responder.***

# The First 5 Minutes: Example Scenario

## 1 - Acknowledge & Orient (Minute 0–1)

- + **Open the ticket** in the IR platform (e.g., TheHive or ServiceNow).
- + Skim the **triage summary**: rule name, severity, affected host, timestamp, and any automatic containment (e.g., EDR network isolation).
- + Confirm SLA clock has started.

**Why:** Prevent duplicate work and ensure you know what Tier 1 already did.

# The First 5 Minutes: Example Scenario

## 2 - Quick Validation (Minute 1-2)

- + Re-run the triggering SIEM query, but widen the window ±5 minutes.

### Splunk SPL Query Example:

```
index=oswin host=FIN-APP-SVR01 process_name=powershell.exe  
| search CommandLine="*45.145.12.77*"  
| table _time, User, CommandLine, ParentProcess
```

- + Confirm the event is not a parsing glitch, test traffic, or sandbox detonation.

**Why:** Catch any false escalation before you burn more hours.

# The First 5 Minutes: Example Scenario

## 3 - Scope Snapshot (Minute 2-3)

- + **Pivot** on hash, user, and IP to see if:
  - + Other hosts contacted the same IP.
  - + The same PowerShell hash ran elsewhere.
- + Check asset criticality: finance server = crown-jewel → potential high impact.
- + Note whether **FIN-APP-SVR01** shows other alerts in the last 24 h.

**Why:** Decide if this is isolated or already lateral. This will guide containment breadth.

# The First 5 Minutes: Example Scenario

## 4 - Initial Containment Decision (Minute 3-4)

- + If the server is still beaconing, trigger EDR isolate-host or firewall block on the IP.
- + If beaconing stopped and business risk of isolation is high, plan deferred containment but start paperwork.

**Why:** Early action can stop data theft or second-stage payloads.

# The First 5 Minutes: Example Scenario

## 5 - Document & Communicate (Minute 4–5)

- + Add a concise note to the ticket:  
“Validated malicious PowerShell download from 45.145.12.77 on FIN-APP-SVR01. No other hosts contacted IP. Host isolated via CrowdStrike. Next step: memory capture and persistence check.”
- + Page the “Finance-IT” contact (if isolation may disrupt services).
- + Attach raw event JSON and your expanded search results.

**Why:** Creates a clear audit trail and keeps stakeholders informed.

# The First 5 Minutes: Example Scenario

## What Happens Next?

- + With the **incident validated** and **initial scope defined**, you ***transition*** into **deeper analysis**: memory dumps, disk triage, persistence hunting, and enterprise-wide IOC sweeps.

But those **first five minutes** that **involve acknowledging, validating, scoping, containing, and documenting**, set the ***foundation*** for every decision you and the wider IR team will make from here on.

A man with glasses is shown in profile, looking at a computer screen in a dark room. The screen displays some blurred text or code. The overall atmosphere is professional and focused.

# Beyond First Response: Deep Analysis

<https://t.me/learningnets>



# What “Deep Analysis” Means

*In incident response*, “Deep Analysis” is the transition from a rapid “yes/no, how-big” validation **to a forensic-level investigation/analysis** that seeks to determine **how the threat operated, what was modified, which data was accessed, and how the threat can be eradicated without collateral damage.**

It is triggered when **first-response** findings show that:

- + The activity is confirmed as malicious and not yet fully scoped.
- + Crown-jewel (critical) systems or regulated data are involved.
- + Or, when containment decisions require surgical accuracy (e.g., root-cause patching, selective rollback, legal evidence preservation).

# What “Deep Analysis” Entails

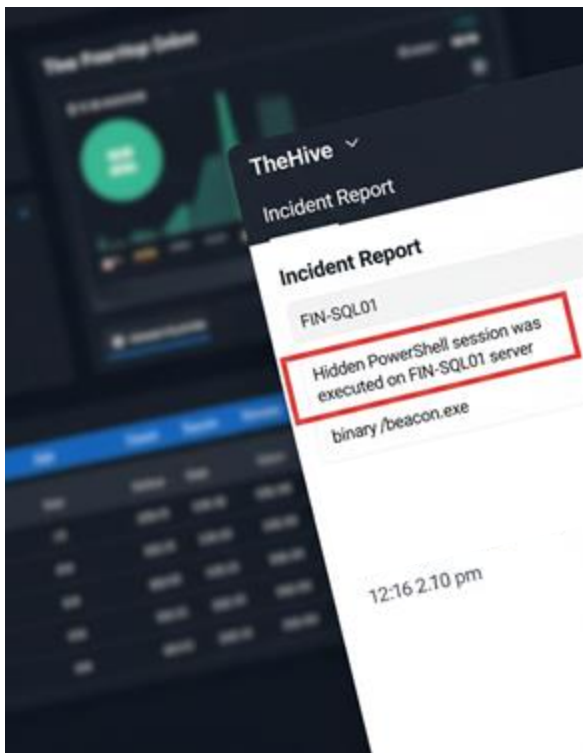
Area	Core Tasks	Typical Tools & Artefacts
<b>Endpoint-centric Analysis</b>	<ul style="list-style-type: none"><li>• Full memory forensics (Volatility, Rekal) to extract injected code, credentials, malware configs.</li><li>• Disk &amp; registry analysis to find persistence, timestamps, deleted files.</li><li>• Binary triage/reverse engineering (Ghidra, CyberChef) performed on dropped executables and scripts.</li><li>• Super-timeline construction (Plaso/Timesketch) combining file-system, log, and registry timestamps.</li><li>• User-artifact review: shellbags, jump-lists, browser history.</li></ul>	RAM dumps, \$MFT/USN, Sysmon & Windows logs, scheduled-task XML, malicious binaries
<b>Network-centric Analysis</b>	<ul style="list-style-type: none"><li>• Full-packet capture replay to decrypt or inspect payloads (Wireshark, Zeek PCAP analyzer).</li><li>• Beacon-pattern analysis (RITA, Arkime) for C2 timing and jitter.</li><li>• Protocol carving to extract exfiltrated files, certificates, HTTP/2 multiplex streams.</li><li>• Threat-intel correlation of ASN, TLS fingerprints, JA3/JA4 hashes.</li><li>• Graphing lateral flows (Malcom, Splunk Traffic App) to map movement between subnets.</li></ul>	PCAPs, NetFlow, Zeek logs, TLS handshake data, DNS query logs

# Example Scenario

<https://t.me/learningnets>



# Initial Alert, Triage & Escalation



→ Alert (00:00) → Splunk raises a critical alert:

The Tier 1 analyst has performed initial triage and has escalated this to you, the on-call incident responder for validation and analysis (if required).

The ticket/case contains the following information:

A hidden `PowerShell` session has been executed on `FIN-SQL01` that downloads a binary/executable called "`beacon.exe`" from a remote server on `198.51.100.77`.

# First 5 Minutes (Hot Triage/First Response)

Minute	Action	Outcome
0–1	You open the ticket and read the information included by the Tier 1 analyst.	Confirm the affected host, user, initial hash.
1–2	Re-run SIEM search ( $\pm 5$ min).	You identify three matching PowerShell events.
2–3	Pivot on hash & IP.	No other hosts show activity, the asset is a production SQL server.
3–4	EDR isolates host and firewall blocks IP.	Active beaconing stopped.
4–5	Update ticket, attach logs and page DBA owner.	Incident validated, scoped as <i>single host—critical asset</i> .

**Why deeper analysis is now required:** A **mission-critical database server ran untrusted code**, as a result, regulators may require **proof** of whether data was read or modified.

***Root-cause and persistence must be located before re-adding the server to production.***

# Deeper Endpoint Analysis Steps

## 1 - Memory Forensics

- + Dump RAM with Velociraptor, run Volatility malfind and find injected shellcode in `sqlservr.exe`.
- + Extract C2 URL and AES key from process strings.

## 2 - Disk & Registry Examination

- + Parse \$MFT and Sysmon Event 11 to confirm `beacon.exe` dropped into “C:\ProgramData\Adobe”.
- + Registry hive shows new run-key “HKCU\Software\Microsoft\Windows\CurrentVersion\Run\AdobeUpdate”.

# Deeper Endpoint Analysis Steps

## 3 - Binary Triage

- + Static strings reveal **7-Zip** self-extract plus **Mimikatz** DLL.
- + Execution in a sandbox (executed offline) confirms credential dump module.

## 4 - Timeline Development

- + Combine MFT, Prefetch, Windows Event 4688, and Sysmon into **Timesketch**.
- + Identify initial execution via an RDP session from HR-LAPTOP02 **18 minutes** earlier.

# Deeper Endpoint Analysis Steps

## 5 - IOC Expansion & Enterprise Hunt

- + Hashes of **beacon.exe**, shellcode, and **7-Zip** payload searched across EDR fleet, two more hits on staging servers.
- + Staging servers queued for separate containment.

## 6 - Root-Cause Identification

- + **RDP** logs show successful login with **domain admin** “**svc\_backup**”.
- + Correlate with AD logs → a password-spray alert an hour prior on **HR-LAPTOP02**.

# Deliverables/Outputs

Output	How It Feeds Containment & Recovery
IOC set (hashes, IP, registry key)	New firewall block rules, EDR quarantine rules and SIEM detection rules.
Root-cause report (compromised RDP creds)	Perform Immediate password reset(s), roll-out 2FA and update the RDP audit policy.
Forensic timeline & data-access list	Database Administrator (DBA) verifies database integrity and legal/compliance team evaluates breach-notification duty.
Detection gap (no PowerShell downgrade rule)	Detection engineering team/personnel write new controls and the SOAR playbook is updated.

# Key Takeaway

**The bottom line:** Deeper analysis turns the quick “we have malware on FIN-SQL01” finding into a complete, and accurate narrative that includes key information like:

- + The entry vector / Initial access vector
- + Lateral reach (scope)
- + Credential exposure
- + Data impact
- + Concrete eradication steps

***This ensures that containment is precise and recovery is both safe and compliant.***

A man with glasses is shown in profile, looking at a computer monitor in a dark room. The monitor displays some code or data. The overall scene is dimly lit, with the primary light source being the screen. An orange vertical bar is on the left side of the image.

# Evidence Triage & Collection

<https://t.me/learningnets>





# Evidence Triage

<https://t.me/learningnets>



# Evidence Triage - Prioritising, Preserving & Staying Focused

After an **incident is validated**, the responder will potentially have more artefacts to investigate as part of their subsequent analysis. In order to analyze these artefacts, you need to collect evidence/data from endpoints.

However, before you begin the process of data/evidence collection (aka acquisition), you need to perform an “**incident-specific**” **evidence triage** process to determine what data/evidence to collect and the order in which it is to be collected.

**Evidence triage** imposes a quick-but-structured decision process so you:

- + **Collect the right artefacts** in the **right order** (highest value, most volatile first).
- + **Maintain chain of custody** so every item is legally and technically defensible.
- + **Prevent analysis paralysis** by using a **repeatable/reusable scoring matrix** instead of ad-hoc choices.

# How Evidence Triage Shapes Collection/Acquisition

**Evidence triage** is the *decision filter* applied before acquisition.

It answers two questions:

1. **Forensic Value:** “Which artefacts provide the highest investigative value?”
2. **Volatility:** “Which artefacts will disappear or change first?”

**Responders** typically *rank each prospective artefact* on two scales; forensic value and volatility using a quick scoring matrix (**1 = low, 3 = high**).

The sum total dictates collection order.

# Evidence Triage: Simple Two-Factor Scoring Matrix

Factor	1 pt	2 pts	3 pts
<b>Forensic Value (how much it tells you)</b>	Low (routine noise)	Medium (context)	High (direct attacker evidence)
<b>Volatility (how fast it disappears)</b>	Low (disk image, server logs)	Medium (Sysmon, event logs w/ rollover)	High (RAM, active sockets, running processes)

## How it works:

1. Score each artefact (RAM dump, Sysmon log, PCAP, registry hive, malware sample, etc.) on both scales/dimensions.
2. Add the two numbers, in this case the potential maximum score is = 6.
3. Collect in descending order of total score.

# Evidence Triage: Simple Two-Factor Scoring Matrix

Factor	1 pt	2 pts	3 pts
<b>Forensic Value (how much it tells you)</b>	Low (routine noise)	Medium (context)	High (direct attacker evidence)
<b>Volatility (how fast it disappears)</b>	Low (disk image, server logs)	Medium (Sysmon, event logs w/ rollover)	High (RAM, active sockets, running processes)

## Example:

- + RAM (*Forensic Value* = 3, *Volatility* = 3): **3 + 3 = 6** → Capture first.
- + Sysmon log: (**3 + 2 = 5**) → Next.
- + Full disk image: (**2 + 1 = 3**) → Later, after volatile items are safe.



# Evidence Collection/Acquisition

<https://t.me/learningnets>



# Evidence Collection/Acquisition

Once an incident has been validated, scoped and evidence triage has been performed. The next step will involve the collection/acquisition of evidence to support deeper analysis.

**Evidence collection/acquisition** is the process of **capturing a full(complete), defensible copy of every artefact (identified during evidence triage)** for the purposes of deeper analysis, so as to answer key questions like:

- + What was the cause of the attack/incident?
- + How the attack worked/unfolded.
- + What systems were accessed and what was changed?
- + What data was accessed?
- + Was any data stolen?

# Evidence Collection/Acquisition Process

Phase	Core Actions	Typical Tools & Artefacts
<b>Preserve volatile data first</b>	<ul style="list-style-type: none"><li>• Capture live memory (RAM), active process lists, open sockets, ARP/cache tables.</li><li>• Stop and hash quarantined binaries before they're deleted by EDR or AV.</li></ul>	WinPMem / LiME, Velociraptor live-response, <a href="#">lsof</a> , <a href="#">netstat</a> , EDR remote-collection APIs
<b>Secure transient logs</b>	<ul style="list-style-type: none"><li>• Export last 24-48h of Sysmon, Security, PowerShell, Linux <code>journalctl/auth.log</code> before rollover.</li><li>• Pull Zeek/Suricata ring buffers or switch PCAPs.</li></ul>	<a href="#">wevtutil epl</a> , Winlogbeat pull, tcpdump, Zeek <a href="#">capture_loss.bz2</a>
<b>Acquire non-volatile artefacts</b>	<ul style="list-style-type: none"><li>• Forensic disk or volume images, registry hives, scheduled-task XML, <a href="#">\$MFT/USN Journal</a>.</li></ul>	FTK Imager, <a href="#">dd</a> /Clonezilla
<b>Hash &amp; chain-of-custody</b>	<ul style="list-style-type: none"><li>• SHA-256 each file at time of capture.</li><li>• Record <i>who, what, when, where, hash</i> in a custody log.</li><li>• Store copies in write-once or versioned storage.</li></ul>	<a href="#">sha256sum</a> , immutable S3 bucket, ServiceNow/TheHive ticket fields
<b>Validate acquisitions</b>	<ul style="list-style-type: none"><li>• Re-hash after transfer, mount images read-only, verify memory dump integrity with Volatility.</li></ul>	<a href="#">sha256sum disk.img</a> , Volatility <a href="#">imageinfo</a>

# Evidence Acquisition: Tools & Evidence You Grab First

Now that we have an understanding of the evidence triage and collection process, we can turn our attention to the various sources of evidence typically used for analysis and the respective tools through which they can be acquired.

The next slide contains an operational checklist you can reference during first response and the deeper analysis that follows.

Each row tells you what to collect, why it's valuable, and a proven utility for pulling/collecting it.

***Always hash, log, and secure every artefact immediately to preserve chain of custody.***

# Evidence Acquisition: Tools & Evidence You Grab First

Evidence Source	Why You Want It	Go-To Utilities & Quick Notes
<b>Endpoint Logs</b> ( <i>Windows, Linux</i> )	<ul style="list-style-type: none"><li>• First layer of truth.</li><li>• Contains info on process launches, auth events, script blocks, EDR telemetry.</li></ul>	<ul style="list-style-type: none"><li>• Windows: <a href="#">wevtutil epl</a> (local EVTX export).</li><li>• Sysmon &amp; PowerShell via Winlogbeat/Splunk UF.</li><li>• Linux: journalctl, auditd logs.</li></ul>
<b>RAM / Memory Dump</b>	<ul style="list-style-type: none"><li>• Stores the most volatile artefacts: in-memory malware, injected DLLs, credentials in LSASS, encryption keys.</li></ul>	<ul style="list-style-type: none"><li>• WinPMem</li><li>• Velociraptor live-response module</li><li>• LiME for Linux</li></ul>
<b>Disk / Volume Image</b>	<ul style="list-style-type: none"><li>• Contains full persistence evidence, deleted files, timestomping clues.</li><li>• Ensures repeatable offline work.</li></ul>	<ul style="list-style-type: none"><li>• FTK Imager GUI / CLI,</li><li>• Clonezilla for bit-for-bit imaging.</li></ul>
<b>Process &amp; Execution Trees</b>	<ul style="list-style-type: none"><li>• Analysis of Parent/child process chains reveal LOLBins, suspicious scripts, privilege-escalation attempts.</li></ul>	<ul style="list-style-type: none"><li>• EDR console (Process Graph),</li><li>• Sysinternals ProcDump / Process Explorer live capture</li><li>• Velociraptor hunt: <a href="#">Windows.System.Pstree</a>.</li></ul>

# Evidence Acquisition: Tools & Evidence You Grab First

Evidence Source	Why You Want It	Go-To Utilities & Quick Notes
<b>Network Traffic: PCAP &amp; NetFlow</b>	<ul style="list-style-type: none"><li>Reveals info like Beacon timing, C2 domains, exfil payloads, lateral RDP/SMB flows.</li></ul>	<ul style="list-style-type: none"><li>tcpdump ring buffer</li><li>Zeek for session logs</li><li>Arkime / Moloch for full-packet indexing</li><li>nfdump/nfcapd for NetFlow.</li></ul>
<b>EDR Artifacts</b>	<ul style="list-style-type: none"><li>Real-time blocking actions, quarantined files, EDR-captured hashes.</li></ul>	<ul style="list-style-type: none"><li>API pulls: CrowdStrike Falcon API</li><li>Defender for Endpoint Advanced Hunting queries</li><li>SentinelOne Deep Visibility.</li></ul>
<b>Browser &amp; User-Activity Artifacts</b>	<ul style="list-style-type: none"><li>Initial phishing click, credential reuse, suspicious file downloads.</li></ul>	<ul style="list-style-type: none"><li>Hindsight (Chrome/Edge)</li><li>jump list parsing with JLECmd</li><li>shellbags via ShellBags Explorer</li></ul>

# Evidence Acquisition: Quick Workflow

## 1. Prioritise by Volatility → Value

- + RAM > live process list > transient logs > disk image.

## 1. Use Trusted, Version-controlled Tools

- + Keep a hashed copy of every acquisition binary in your evidence share.

## 1. Hash Immediately

- + SHA-256 every dump (`sha256sum mem.raw > mem.raw.sha256`).

## 1. Log Every Step

- + Ticket comment or dedicated chain-of-custody form: who, what, where, when, hash.

## 1. Secure Storage

- + Write-once S3 bucket, evidence NAS with immutability, or encrypted external drive.

# Key Takeaways

**Evidence triage** is a rapid-fire prioritization step.

*You decide what needs collecting, how quickly, and in what order, based on each artefact's volatility and forensic value.*

Data acquisition follows: **Once priorities are set**, you use the appropriate tools to capture the high-scoring items first (e.g., RAM dump before disk image), hashing and logging each artefact to preserve chain of custody.

**Think of evidence triage as drafting the shopping list under time pressure; acquisition is the act of grabbing those items from the shelf. Without triage, you risk wasting precious minutes on low-value or low-volatility data while the most critical, short-lived evidence disappears.**



# Introduction to Endpoint Analysis

<https://t.me/learningnets>



# Endpoint Analysis

**Endpoint analysis** is the *systematic examination* of a single host—workstation, laptop, server, VM, container, or even a mobile device—after an alert has indicated it might be involved in malicious activity.

**Using endpoint/host-resident evidence** such as event logs, memory, registry hives, file-system artefacts, running processes, and network sockets, **the responder reconstructs how the attacker interacted** with that specific **endpoint:**

- + Entry vector – phishing link, RDP brute-force, USB drop, etc.
- + Execution & privilege escalation – scripts, LOLBins, service installs.
- + Persistence – scheduled tasks, Run-keys, WMI event subscribers.
- + Credential theft & lateral movement – LSASS dumps, cached tokens.
- + Local impact – files exfiltrated, databases modified, ransomware dropped.

# Endpoint Analysis

While a SIEM may tell you something suspicious happened on “FIN-WS-01”, **endpoint analysis reveals the step-by-step reality on that host:** processes spawned, files dropped, registry keys modified, credentials harvested, and evidence left in volatile memory or on disk.

In short, endpoint analysis transforms a vague “something bad happened on Host X” into a precise, evidence-backed narrative, allowing incident responders to act surgically, eradicate the threat, and prevent it from returning.

# Endpoint Analysis - Objectives

Objective	Why It's Crucial
<b>Validate the alert</b>	Confirms a true positive by showing real malicious artefacts—avoids costly over-reaction to a false alarm.
<b>Scope the compromise</b>	Determines whether the host is <i>patient zero</i> or just one victim among many; clarifies how far the intruder progressed.
<b>Identify root cause</b>	Pinpoints the exact weakness (e.g., unpatched DLL hijack, weak password) so remediation addresses the source, not just symptoms.
<b>Harvest Indicators of Compromise (IOCs)</b>	Extracts hashes, C2 domains, registry keys, persistence paths—feeds hunts across the rest of the environment and updates detection rules.
<b>Preserve evidence</b>	Collects forensically sound artefacts (RAM dumps, disk images) needed for legal, regulatory, or insurance obligations.
<b>Inform containment &amp; eradication</b>	Tells defenders which services to isolate, what persistence keys to delete, what patches or password resets to roll out—preventing re-infection.
<b>Quantify business impact</b>	Clarifies if sensitive data was accessed or altered, guiding disclosure decisions and recovery priorities.

# Endpoint Analysis - Where It Fits in the IR Process

1. **Detection & Analysis** – Endpoint analysis begins after an escalated alert is validated.
1. **Containment & Eradication** – Findings drive host isolation, malware removal, and patching.
1. **Recovery** – Verified evidence assures systems are clean before going back to production.
1. **Lessons Learned** – Artefacts and root-cause insight feed improved detections and security controls.

# Major Categories (Types) of Endpoint Analysis

Category	Core Questions Answered	Key Artefacts & Tools
Log Analysis	<i>What did the OS/EDR record?</i>	Windows Event/Sysmon, Linux <code>journalctl/auth.log</code> ; viewed via SIEM, EVTExtract, ELK, Splunk.
Process & Execution-Tree Analysis	<i>Which processes ran, and in what order?</i>	Parent/child PIDs, command lines, parent hashes; tools: Process Explorer live, EDR console, Kape, ProcDump timelines.
Persistence & Autostart Analysis	<i>How does the malware survive reboots?</i>	Registry Run keys, scheduled tasks, services, WMI subscriptions, cron/systemd units; tools: Autoruns, RegRipper, Velociraptor hunts.
Memory (Volatile) Forensics	<i>What's injected or running only in RAM?</i>	In-memory DLLs, reflective loaders, credentials in LSASS, encryption keys; tools: Volatility, Rekall, Comae.
File-System & Disk Forensics	<i>What files were created, modified, or deleted?</i>	<code>\$MFT</code> , USN Journal, shadow copies, timestamps; tools: FTK Imager, Autopsy, Sleuth Kit.
Registry / Configuration Hive Analysis (Windows)	<i>What config changes reveal user or malware activity?</i>	RecentDocs, Shellbags, UserAssist, SAM modifications; tools: RegRipper, hivex.

# Major Categories (Types) of Endpoint Analysis

Category	Core Questions Answered	Key Artefacts & Tools
User-Activity Artefact Analysis	<i>What did the user open or execute?</i>	Jump lists, LNK files, browser history, RDP cache, bash history.
Credential & Account Analysis	<i>Were credentials dumped, created, or abused?</i>	Password hashes, keytab tickets, new admin accounts; tools: Mimikatz output, SAM diffing, Kerberos logs.
Binary / Malware Triage	<i>What does the payload do?</i>	Hash identification, static strings, sandbox detonation; tools: Detect-It-Easy, PE-Studio, CyberChef, Any.Run.
Driver & Kernel-Module Analysis	<i>Is there a rootkit or malicious driver?</i>	Unsigned drivers, SSDT hooks, kernel callbacks; tools: gmmsup, WinDbg, Kernel Detective.
Timeline Reconstruction (Super-Timeline)	<i>When did every artefact change?</i>	Combined view of log, file, registry timestamps; tools: Plaso, Timesketch.
Live Response vs. Dead-Box	<i>Do we collect artefacts while host runs or offline?</i>	Live: Velociraptor, GRR; Dead-box: image, then offline analysis. Choice depends on volatility and business impact.



# Live Response vs. Dead-Box

<https://t.me/learningnets>



# Live Response vs. Dead-Box

When an endpoint is under investigation, responders must **choose between two evidence-collection strategies**.

**Live response grabs volatile data** *like running processes, RAM-resident malware and active network sockets* **while the system is still powered on**, preserving evidence that is volatile in nature.

**Dead-box** forensics powers the host down and acquires bit-for-bit disk images for pristine, court-defensible analysis, but sacrifices in-memory artefacts and may disrupt operations.

***Knowing when to apply each approach, or how to combine them, ensures critical evidence is captured without needless business impact.***

# Live Response

Aspect	Summary
<b>Definition</b>	Evidence collection while the machine is still powered on and running. Uses an agent or short scripts to pull volatile artefacts without shutting the host down.
<b>When to Use</b>	<ul style="list-style-type: none"><li>• Active threat still beaconing or moving laterally.</li><li>• Host provides critical services and an immediate shutdown would harm the business.</li><li>• Need to capture highly volatile data (process memory, network sockets, RAM-only malware).</li></ul>
<b>What It Entails</b>	<ol style="list-style-type: none"><li>1. Deploy a live-response tool (Velociraptor, GRR, CrowdStrike RTR, remote PowerShell).</li><li>2. Grab RAM dump, running processes, loaded DLLs, active connections, clipboard, in-memory registry, dark-corner logs (\$LogFile, recent EVTX).</li><li>3. Hash each artefact, copy to a secure share, and log every step in the ticket.</li></ol>
<b>Advantages</b>	<ul style="list-style-type: none"><li>• Preserves evidence that vanishes at shutdown.</li><li>• Allows quick containment actions (isolation, process kill) without full power-off.</li><li>• Minimal disruption if done correctly.</li></ul>
<b>Trade-offs</b>	<ul style="list-style-type: none"><li>• Risk of altering evidence (new files, touching timestamps).</li><li>• Malware may detect and react.</li><li>• Requires tight procedural control to remain defensible in court.</li></ul>

# Dead-Box

Aspect	Summary
<b>Definition</b>	Evidence collection after the machine is powered off—typically a bit-for-bit image of disks or volumes, plus post-mortem analysis of that static media.
<b>When to Use</b>	<ul style="list-style-type: none"><li>• Threat is contained or the host can be safely taken offline.</li><li>• Legal or regulatory requirements demand a pristine, unchanged snapshot.</li><li>• Need to examine deep-disk artefacts (deleted files, timestomps, slack space) without risk of live tampering.</li></ul>
<b>What It Entails</b>	<ol style="list-style-type: none"><li>1. Shut down or yank power (only if safe to do so).</li><li>2. Remove drives or boot from trusted forensic media.</li><li>3. Capture full disk/volume images with FTK Imager, <code>dd</code>, or Clonezilla.</li><li>4. Calculate hashes, store on write-once media, mount read-only for analysis (Autopsy, Sleuth Kit, EnCase, X-Ways).</li></ol>
<b>Advantages</b>	<ul style="list-style-type: none"><li>• Forensic soundness—little risk of evidence alteration.</li><li>• Enables deep file-system and slack-space analysis.</li><li>• Malware cannot react or self-destruct.</li></ul>
<b>Trade-offs</b>	<ul style="list-style-type: none"><li>• No volatile artefacts; RAM-only implants and live network state are lost.</li><li>• System downtime may impact operations.</li><li>• Imaging large disks is slow and storage-intensive.</li></ul>

# Choosing Between Them

## 1. Volatility vs. Business Impact

- + If data in memory is critical and the host must stay up → Live Response first.
- + If threats are dormant or host can be pulled offline safely → Dead-Box is safer.

## 1. Containment Urgency

- + Ongoing C2 beacon: isolate network, run live response to grab RAM, then image disk.
- + No active communication: shut down cleanly and proceed to dead-box imaging.

## 1. Legal / Regulatory Requirements

- + Some jurisdictions prefer live capture of RAM plus dead-box imaging for a complete chain of custody.

# Key Takeaway

Live response saves what disappears first; dead-box imaging preserves everything that lasts.

**A skilled responder knows when to employ one, the other, or a carefully sequenced blend of both.**



# Introduction to Log Analysis

<https://t.me/learningnets>



# What is Log Analysis?

On a host/endpoint, **logs are the operating system's diary**: every process start, user login, registry change, kernel error, and network socket can be recorded as a time-stamped entry.

**Endpoint log analysis** is the practice of searching, filtering, correlating, and interpreting those entries to answer five incident-response questions:

Did anything malicious actually happen on this host?

- + When did it start and how far did it go?
- + How did the attacker get in and stay in?
- + Which identities, files, or data were touched?
- + What concrete indicators (hashes, IPs, domains, registry keys) can we use to protect the rest of the environment/other endpoints?

# The Role Logs Play in an Investigation

IR Need	How Host Logs Fulfil It
Alert Validation	A SIEM fires on “suspicious PowerShell” → Security/Sysmon logs either confirm or refute the execution chain.
Scoping & Timeline	Ordered log timestamps let responders rebuild every step from initial exploit to persistence and C2.
Root-Cause Discovery	First evidence of lateral movement or privilege escalation often appears only in local logs (e.g., Event 4674 for S-Privilege use).
IOC Harvesting	Logs expose hashes, file paths, PIDs, IPs, and user accounts that become search pivots across the enterprise.
Regulatory Proof & Lessons Learned	Forensic-sound log exports underpin breach notifications, audits, and post-incident detection tuning.

# Why Log Analysis Is Critical

- + **Earliest, *Cheapest (most-accessible)* Evidence** – Unlike disk images or memory dumps, logs are usually already present and relatively simple to acquire.
- + **High Signal-to-Noise** – Properly parsed Windows/Sysmon and Linux audit logs contain ***ATT&CK-mappable*** signals with relatively few false positives.
- + **Bridging Endpoint & Network** – Process IDs and socket events link host activity with firewall or IDS alerts, giving you a more complete storyline of the attacker activity.
- + **Rapid IOC Expansion** – New hashes or domains pulled from an endpoint's logs drive proactive hunts on thousands of others.

# What an Incident Responder Must Know & Be Able to Do

Competency	Details
<b>Log-Source Literacy</b>	<ul style="list-style-type: none"><li>• Windows: Security, Sysmon, PowerShell/Operational, Scheduled Tasks, WMI-Activity.</li><li>• Linux: <a href="#">journalctl</a>, <a href="#">auth.log/secure</a>, <a href="#">audit.log</a>, sshd &amp; sudo logs.</li><li>• EDR local caches &amp; application logs (browser, database, backup agents).</li></ul>
<b>Core Event IDs &amp; Messages</b>	Memorise or reference the high-value set—e.g., <ul style="list-style-type: none"><li>• Windows: 4624/4625 (logon), 4688 (process start), 7045 (service install), Sysmon 1/3/11</li><li>• Linux: auditd <a href="#">USER_START</a>, <a href="#">EXECVE</a>, <a href="#">CHMOD</a>.</li></ul>
<b>Parsing &amp; Normalisation</b>	Create or tune field extractions so that “UserName” or “event.user” is searchable across sources.
<b>Query &amp; Pivot Skills</b>	Craft SPL/KQL/SQL/grep to chain events: <a href="#">4688 cmd</a> → <a href="#">Sysmon 3 (IP)</a> → <a href="#">Sysmon 11 (file)</a> and quickly switch pivots (hash ↔ PID ↔ IP ↔ Username).
<b>Timeline Construction</b>	Export multi-log slices to CSV/SQLite and load into Timeline Explorer, Plaso/Timesketch, or Kibana TSVB to visualise the attack sequence.

# What an Incident Responder Must Know & Be Able to Do

Competency	Details
<b>IOC Extraction &amp; Enrichment</b>	Parse out hashes/IPs/URLs, enrich with VirusTotal or threat-intel, then feed back into SIEM hunts and block lists.
<b>Noise Reduction &amp; Tuning</b>	Measure FP (false positive) rate, build suppression filters or Sigma rule tweaks, and document the impact on MTTT / escalation rate.
<b>Chain-of-Custody Practices</b>	Hash exported EVTX/JSON, log collector versions, and store evidence on write-once media or an immutable bucket.


# Bottom Line

**Log analysis** transforms an ambiguous alert into a precise, host-level narrative.

**Mastering** the art of **querying, correlating,** and **interpreting endpoint logs** is therefore a ***non-negotiable skill for any incident responder*** who wants to validate breaches quickly, scope them accurately, and ***base containment on hard evidence, not hunches.***

# Common Types/Modes of Log Analysis

Mode	How It Works	When It Shines	Example Stack / Tool
<b>SIEM-Centric Analysis</b>	Logs flow into an indexed platform; <ul style="list-style-type: none"><li>Analysts query with SPL/KQL</li><li>Build correlation searches, dashboards, alerts.</li></ul>	Enterprise-wide scale, cross-source joins, long retention, RBAC & audit built in.	Splunk ES, Elastic Security, Microsoft Sentinel, IBM QRadar.
<b>Local/Offline GUI Review</b>	Drag-and-drop EVTX or log folders into a desktop viewer; filter, regex-search, export.	Single host, air-gapped forensics, quick GUI triage.	Event Log Explorer, Glogg, G4LogViewer.
<b>CLI / Scripted Hunts</b>	Parse logs on disk with command-line tools or scripts; automate searches in CI pipelines.	Speed, repeatability, integration with DevOps or IR scripts.	Chainsaw + Sigma, <a href="#">grep/awk/jq</a> , PowerShell <a href="#">Get-WinEvent</a> , Log Parser Lizard.
<b>Live-Response Agent Queries</b>	Query logs in real time from a running endpoint; pull back rows that match IOC patterns.	Volatile evidence, minimal data transfer, remote edge devices.	Velociraptor VQL, GRR Hunts, EDR “Live Query” (Falcon, Def Endpoint).



# Log Analysis with Splunk: Investigating a Linux Intrusion

<https://t.me/learningnets>



# Investigating a Linux Intrusion

**Case ID:** IR-2025-1024


**Asset:** LINUX01 – production Ubuntu server (application stack)

**Hand-Off:** Tier-1 analyst escalated a High-severity ticket to you.

You have been tasked with analyzing existing logs using Splunk to validate and identify the origin of suspicious activity on a Linux server.

Your objective is to build a story regarding what has occurred on the server, and use that information to plan out required next steps.

1. Perform searches in Splunk to identify & validate potential suspicious or malicious activity.
2. Based on the data found, identify any gaps in logging that may exist.
3. Plan out next steps, which should be determined by the results of the log analysis.

A man with glasses is shown in profile, looking at a computer screen in a dark room. The screen displays some data or code, though it is not clearly legible. The overall atmosphere is focused and technical.

# Lab Demo: Log Analysis with Splunk: Investigating a Linux Intrusion

<https://t.me/learningnets>



# Timeline of Events

Based on our analysis of logs, we've put together an interesting timeline of events:

1. `labadmin` (possibly compromised) created `newadmin`
2. `labadmin` switched users to act as `newadmin`
3. `newadmin` then created `maliciousaccount`
4. `maliciousaccount` logged in from **DC01** using SSH (DC01 possibly compromised)
5. `maliciousaccount` created/uploaded some form of malware on **LINUX01**

# Next Steps

Based on the information obtained from our analysis, here are some possible next steps:

1. Disable **labadmin**, **newadmin**, and **maliciousaccount** accounts
2. Evaluate which of those accounts are legitimate and are needed. Change passwords and secure as necessary.
3. If possible, take DC01 offline for further investigation. This is only possible if there are other, redundant domain controllers. If we can't disconnect it from the network, then IMMEDIATE further investigation is required.
4. Do the same with LINUX01, since it is possibly compromised with malware



# Log Analysis Using Linux

<https://t.me/learningnets>






# Lab Demo: Log Analysis Using Linux

<https://t.me/learningnets>



A man with glasses is shown in profile, looking at a computer screen in a dark room. The background is dimly lit, showing the glow of the screen and some office equipment. The overall mood is professional and focused.

# Windows Logging Primer: Channels, Providers & Key Logs

<https://t.me/learningnets>



# Understanding Windows Logging for IR

This video will provide you with a ***technical primer*** on how ***Windows logging works***, specifically, breaking down the ***key components like event channels, log providers, and collection mechanisms***.

We'll explore the most critical log sources for defenders, including the Security, Sysmon, PowerShell, ScheduledTasks, and WMI-Activity channels.

***Each plays a unique role in capturing system activity, and together they form a comprehensive view of potential malicious behavior within Windows environments.***

# Fundamental Components

## Providers

- + A **provider** is a software component that *generates log events*.
- + **Examples:** Microsoft-Windows-Security-Auditing, Microsoft-Windows-Sysmon, Microsoft-Windows-PowerShell.
- + Each provider defines the structure and content of the events it emits.

# Fundamental Components

## Channels

- + A channel is a log stream where events from providers are published.
- + Think of a channel as a log file or container that aggregates specific kinds of events.

## Types of channels:

- + **Admin:** High-level operational events for troubleshooting.
- + **Operational:** Detailed logs used for auditing and monitoring.
- + **Analytic:** High-volume, fine-grained events, typically off by default.
- + **Debug:** Developer-focused; very low-level details.

# Fundamental Components

## Event Collection

- + Mechanisms like Windows Event Forwarding (WEF) or Syslog agents (like Nxlog, Winlogbeat) gather logs centrally.
- + Useful for real-time monitoring and correlation in SIEM platforms.



# Key Channels & Their Use in DFIR

<https://t.me/learningnets>



# Key Channels

## 1 - Security Channel

**Path:** Event Viewer → Windows Logs → Security

**Provider:** Microsoft-Windows-Security-Auditing

- + Contains logs related to:
  - + Logon attempts (Event ID 4624, 4625)
  - + Privilege use (Event ID 4670+)
  - + Object access (e.g., file, registry, Event ID 4663)
  - + Audit Policy changes, Account management

**Why it's important:** Crucial for detecting brute force, lateral movement, privilege escalation.

# Key Channels

## 2 - Sysmon Channel

**Path:** Applications and Services Logs → Microsoft → Windows → Sysmon → Operational

**Provider:** Microsoft-Windows-Sysmon

- + Contains:
  - + Process creation (ID 1)
  - + Network connections (ID 3)
  - + File creations (ID 11)
  - + Registry and WMI activity

**Why it's important:** Offers rich, correlated data not available in standard logs.

# Key Channels

## 3 - PowerShell Channel

### Paths:

- + **Windows PowerShell:** Traditional PowerShell host events.
- + **Microsoft-Windows-PowerShell/Operational:** Script block logging, transcription, module loads.
- + Key Event IDs:
  - + 4104: Script block logging (reveals full PowerShell code)
  - + 4103: Command invocation

**Why it's important:** Detects fileless malware, Living-off-the-Land (LotL) attacks.

# Key Channels

## 4 - Scheduled Tasks Channel

**Path:** Microsoft-Windows-TaskScheduler/Operational

**Provider:** Microsoft-Windows-TaskScheduler

- + Key Events:
  - + 106: Task registered (creation)
  - + 200/201: Task started
  - + 102: Task completed

**Why it's important:** Attackers often create scheduled tasks for persistence or execution.

# Key Channels

## 5 - WMI Activity Channel

**Path:** Microsoft-Windows-WMI-Activity/Operational

**Provider:** Microsoft-Windows-WMI-Activity

- + Key Event:
  - + 5857–5861: WMI queries, providers, consumer creation

**Why it's important:** WMI is a stealthy method for reconnaissance and persistence. Logging WMI use is critical for identifying advanced threats.

# Why This Matters

- + **Contextual visibility:** Each log source gives a piece of the puzzle. Combined, they reveal attacker behavior.
- + **Detection:** Events from PowerShell and Sysmon channels often provide early indicators of compromise (IOCs).
- + **Response and investigation:** Helps trace execution flow, lateral movement, and establish timelines.
- + **Persistence detection:** WMI, Scheduled Tasks, and Registry modifications are key areas for hidden backdoors.



# **Sysmon Essentials for Incident Responders**

<https://t.me/learningnets>



# What is Sysmon?

**Sysmon (System Monitor)** is a free, lightweight tool from Microsoft's Sysinternals suite.

Sysmon is a system monitoring tool that runs in the background as a Windows service. Once installed and configured, it logs various system events to the Windows Event Log under:

**Applications and Services Logs → Microsoft → Windows → Sysmon → Operational**

# How Does Sysmon Work?

1. Kernel driver hooks key system calls (process creation, network connect, file write, registry set, etc.).
2. User-mode service formats the captured data, calculates SHA-256 hashes, and appends optional geolocation / image metadata.
3. Events are published as numbered records (Event ID 1, 3, 11 ... 26) that can be filtered by an XML configuration (`sysmon.exe -c config.xml`) to include or exclude noise.
4. Because Sysmon writes into the standard Windows Event Log framework, the data can be forwarded by WEF, Winlogbeat, Splunk UF, or any SIEM agent without extra plumbing.

# Why Is Sysmon Used?

Sysmon is widely used for:

- + **Security monitoring:** Provides high-fidelity telemetry that SIEMs and EDRs often rely on.
- + **Threat hunting:** Investigators can trace malware behavior through process trees, network activity, and registry changes.
- + **Incident response:** Helps reconstruct the timeline of an attack, showing what was executed, when, and by whom.
- + **Behavioral analysis:** Assists in identifying anomalous patterns indicative of compromise.
- + **Forensic investigation:** Preserves detailed logs that may reveal indicators of compromise (IOCs) long after the initial event.

# Why Is Sysmon Valuable to Incident Responders?

1. **Visibility:** It gives responders insight into low-level OS behavior without requiring kernel debugging.
1. **Correlated data:** Linking processes, network activity, and registry modifications allows for narrative reconstruction of attacks.
1. **Lightweight & Customizable:** Low resource footprint with customizable configs to minimize noise.
1. **Persistent logs:** Even if an attacker clears standard logs, if Sysmon logs are forwarded to a SIEM, evidence is preserved.
1. **Detection of advanced threats:** Sysmon can catch subtle indicators like PowerShell usage, unusual parent-child relationships, or lateral movement attempts.

# Key Event IDs to Know


ID	Quick Meaning	Typical IOC Value
1	Process Create	Image path, command line, parent hash
3	Network Connection	Src/Dst IP & port, Process GUID
11	File Create	Target filename, hash
13	Registry Value Set	Key path, new data
22	DNS Query	Query name, PID
23– 24	File Delete & Clipboard	Signs of data staging or exfil

A man with glasses is shown in profile, looking at a computer monitor in a dark room. The monitor displays some code or data. The overall scene is dimly lit, with the primary light source being the screen.

# Lab Demo: Deploying Sysmon

<https://t.me/learningnets>



A man with glasses is shown in profile, looking at a computer screen in a dark room. The background is dimly lit, showing the glow of the screen and some office equipment. The overall mood is professional and focused.

# High-Value Windows Event IDs Every Responder Should Know

<https://t.me/learningnets>



# High-Value Windows Event IDs Every Responder Should Know

In the ever-evolving landscape of cyber threats, **Windows Event Logs** serve as one of the most powerful tools in an *incident responder's arsenal*.

These logs provide **granular visibility** into everything from user logins and privilege changes to process execution and network access.

But the vast volume of logs generated by a typical Windows environment can be **overwhelming**—*knowing which Event IDs matter most is key to cutting through the noise*.

# Authentication & Logon Events

Event ID	Description	Why It Matters
4624	Successful logon	Know who logged in and how (type 2 = local, type 3 = network, type 10 = RDP)
4625	Failed logon	Detect brute-force or credential stuffing attempts
4648	Logon using explicit credentials	Often used in pass-the-hash/ticket attacks
4675	SIDs were filtered	Can indicate user logon attempt with added SIDs (privilege abuse)

# User Account and Privilege Management

Event ID	Description	Why It Matters
4720	User account created	Unusual new accounts can indicate persistence
4722	User account enabled	Can show reactivation of dormant accounts
4723, 4724	Password changes/resets	May indicate account compromise or lateral movement
4732, 4756	User added to security group	Critical for spotting privilege escalation
4670	Permissions on an object were changed	Watch for changes to high-value objects (e.g., Admin groups)

# Service and Task Scheduling

Event ID	Description	Why It Matters
7045	New service installed	Common persistence method
4697	Service installation	Another service-related event for detection
106 (Task Scheduler)	Task created	Used for persistence or executing payloads

# Object Access and File Changes

Event ID	Description	Why It Matters
4663	Object accessed	Can track access to sensitive files or registry keys (needs proper auditing setup)
4656	Handle to object requested	Precedes 4663, useful for detecting access attempts

# Logoff and Session Tracking

Event ID	Description	Why It Matters
4634	Logoff	Correlate with 4624 for session duration
4647	User-initiated logoff	Helps distinguish user from system actions
4778, 4779	RDP session reconnect/disconnect	Useful in tracing remote access sessions

# References & Resources

## **Microsoft “Appendix L – Events to Monitor”**

A high-priority catalog of Windows Event IDs—including critical, high, and medium-critical events—directly from Microsoft guidance.

- <https://learn.microsoft.com/en-us/windows-server/identity/ad-ds/plan/appendix-l--events-to-monitor>

## **Ultimate Windows Security – Security Log Encyclopedia**

A comprehensive, browsable encyclopedia covering every Windows Event ID with descriptions and context.

- <https://www.ultimatewindowssecurity.com/securitylog/encyclopedia/>



# Exporting Windows Event Logs with wevtutil

<https://t.me/learningnets>




A man with glasses and a beard is shown in profile, looking at a computer screen in a dark room. The background is dimly lit, showing the glow of the screen and some office equipment. The overall mood is professional and focused.

# Lab Demo: Exporting Windows Event Logs with wevtutil

<https://t.me/learningnets>






# Parsing Windows Event Logs with EvtxECmd

<https://t.me/learningnets>



A man with glasses is shown in profile, looking at a computer screen in a dark room. The screen displays some code or data. The overall scene is dimly lit, with the primary light source being the screen.

# Lab Demo: Parsing Windows Event Logs with EvtxECmd

<https://t.me/learningnets>



A man with glasses and a beard is shown in profile, looking at a computer screen in a dark room. The screen displays some data or code, though it's not clearly legible. The overall atmosphere is professional and focused.

# Analyzing Windows Event Logs with Timeline Explorer

<https://t.me/learningnets>



# What is Timeline Explorer?

**Timeline Explorer** is a *GUI tool* created by Eric Zimmerman, designed to help digital forensics and incident response professionals analyze time-based forensic data—especially CSV files generated from parsed logs and system artifacts.


Eric Zimmerman's Tools: <https://ericzimmerman.github.io/#!index.md>

Eric is the creator of EvtxECmd, Registry Explorer, Timeline Explorer and a boatload of other tools

# What It's Used For

Timeline Explorer is primarily used to:


- + Build and review forensic timelines
  - + Visualize a chronological sequence of events from logs, registry data, file system activity, etc.
- + Correlate artifacts from multiple sources
  - + Combine exported data from tools like:
    - + EvtxECmd (Windows event logs)
    - + MFTECmd, RECmd, PECmd (for NTFS, registry, and prefetch)
    - + KAPE (as a triage artifact extractor)
- + Filter and search timeline data
  - + Apply filters across timestamps, file paths, event types, or keywords to spot attacker activity
- + Identify suspicious patterns over time
  - + Highlight logon spikes, script execution, persistence setup, etc.

A man with glasses is shown in profile, looking at a computer screen in a dark room. The screen displays some graphical data, possibly a timeline or logs. The overall scene is dimly lit, with the primary light source being the screen.

# Lab Demo: Analyzing Windows Event Logs with Timeline Explorer

<https://t.me/learningnets>




A man with glasses is shown in profile, looking at a computer screen in a dark room. The background is dimly lit, showing the glow of the screen and some office equipment. The overall mood is professional and focused.

# Windows Log Analysis with Chainsaw & Sigma

<https://t.me/learningnets>



A man with glasses is shown in profile, looking at a computer screen in a dark room. The screen displays some data or code. The overall scene is dimly lit, with the primary light source being the computer screen.

# Lab Demo: Windows Log Analysis with Chainsaw & Sigma

<https://t.me/learningnets>





# Introduction to Network Analysis In IR

<https://t.me/learningnets>



# What is Network Analysis?

**Network analysis** in *incident response* is the process of examining network traffic and communications to **detect, investigate**, and **understand** suspicious or malicious activity within an organization's environment.

It involves **reviewing various forms of network data** ranging from **raw packets** to **high-level logs**, in order to trace the actions of attackers, identify compromised systems, and determine how an incident unfolded.

# Importance of Network Analysis in IR

In the Detection & Analysis phase of the incident response lifecycle, network analysis serves as a powerful tool to:

- + **Detect Intrusions Early:** Identify signs of compromise, such as unauthorized connections, malware communication, or abnormal traffic patterns.
- + **Validate Alerts:** Confirm whether alerts from IDS/IPS, firewalls, or SIEM systems are true positives.
- + **Scope the Incident:** Determine which systems communicated with the attacker or were involved in lateral movement.
- + **Understand Attacker Tactics:** Reveal command and control (C2) channels, data exfiltration methods, and privilege escalation attempts.
- + **Support Attribution and Reporting:** Provide evidence that supports incident timelines, root cause analysis, and incident documentation.

**Think of network analysis as the “black box recorder” of an incident; it often holds the most objective record of what happened over the wire.**

# Types of Network Data Sources

Different types of network data provide varying levels of detail and insight:

- + Packet Captures (PCAPs)
  - + Raw, full-content network traffic.
  - + Can reconstruct entire sessions (e.g., emails, file transfers).
  - + High fidelity but storage-intensive.
- + Flow Data (NetFlow, sFlow, IPFIX)
  - + Metadata about communications: IPs, ports, protocols, byte counts.
  - + Great for identifying large data transfers, scanning, or beaconing.
  - + Lower fidelity but scalable and less storage-heavy.
- + Protocol Metadata (e.g., Zeek logs)
  - + Layer-7 insights: DNS queries, HTTP requests, TLS handshakes.
  - + Helps identify misuse of legitimate services (e.g., DNS tunneling).

# Types of Network Data Sources

Different types of network data provide varying levels of detail and insight:

- + Network Logs
  - + IDS/IPS alerts (Suricata, Snort), firewall logs, web proxy logs.
  - + Provide context around blocked or allowed traffic, rule matches.
- + Name Resolution and Authentication Logs
  - + DNS, DHCP, and Kerberos logs provide asset identification and movement tracking.

# Goals of Network Analysis in IR

Incident responders use network analysis to:

- + **Identify Indicators of Compromise (IOCs):**
  - + IP addresses, domains, protocol anomalies, signatures.
- + **Detect Malicious Behavior Patterns:**
  - + Beaconsing, scanning, lateral movement, tunneling, and exfiltration.
- + **Correlate Events Across Time and Systems:**
  - + Build timelines, connect initial access to subsequent actions.
- + **Assess Incident Impact:**
  - + Determine if sensitive data was accessed or exfiltrated.
  - + Understand the scale and duration of attacker activity.
- + **Enhance Detection Capabilities:**
  - + Use insights from analysis to create new alerts or signatures for proactive defense.



# Network-Based Analysis Methodology

<https://t.me/learningnets>



# Network-Based Analysis Methodology

## 1 - Preparation and Baseline Understanding

- + Understand the normal network behavior and topology.
- + Ensure access to network monitoring tools, logs, and packet captures.
- + Verify logging policies (e.g., what traffic is captured, where logs are stored, retention period).

## 2 - Alert Triage and Scoping

- + Start with an alert or anomaly (e.g., from IDS, SIEM, or user report).
- + Identify initial indicators (IP addresses, ports, protocols, timestamps).
- + Scope the affected systems, users, and network segments.

# Network-Based Analysis Methodology

## 3 - Data Collection

- + Pull relevant network artifacts:
  - + Packet captures (PCAP)
  - + Flow records (NetFlow, sFlow)
  - + IDS/IPS alerts
  - + DNS, web proxy, and firewall logs
- + **Time-box the data:** narrow collection around key timestamps.

## 4 - Traffic Analysis

- + Examine communications for:
  - + Unusual destinations (geolocation, threat intel match)
  - + Abnormal ports or protocols
  - + Beaconsing or persistence patterns
  - + Large data transfers (potential exfiltration)
- + Reconstruct sessions to see full conversations.

# Network-Based Analysis Methodology

## 5 - Correlation & Contextualization

- + Cross-reference findings with:
  - + Host-based logs
  - + Threat intelligence feeds
  - + User activity
- + Build a timeline of events to understand attacker objectives and movements.

## 6 - Documentation & Reporting

- + Document all findings with timestamps, evidence, and conclusions.
- + Provide impact assessment and recommendations.
- + Share findings with other IR teams (e.g., for remediation or threat hunting).



# Network Data Types & Analysis Tools

<https://t.me/learningnets>



# Introduction: Network Data Types & Analysis Tools

*In the field of incident response, time is critical*, and network traffic is often one of the first and richest sources of evidence available when a potential security event is detected.

*Whether responding to a phishing attack, investigating data exfiltration, or tracing lateral movement across systems, network-based evidence provides a timeline, context, and often a complete picture of an attacker's actions.*

To effectively detect, analyze, and respond to threats, incident responders must be proficient in identifying what network data is available, where it's located, and how to interpret it using the right tools.

This includes knowing the difference between full packet captures and summarized flow records, understanding how to leverage logs from firewalls and proxies, and using specialized tools to make sense of raw or structured data.

# Packet Captures (PCAPs)

## What Are They?

- + PCAPs are raw network traffic data captured at the packet level.
- + Each packet includes full headers and payloads, allowing complete session reconstruction (e.g., file downloads, chat messages, credentials in plaintext).

## **Common Tools:**

- + Wireshark – Graphical tool for deep packet inspection
- + tcpdump – Command-line tool for capturing and filtering packets
- + Tshark – Wireshark's CLI counterpart for automated parsing
- + NetworkMiner – Focuses on session and artifact reconstruction

# Packet Captures (PCAPs)

## Use Cases in Incident Response:

- + Identify malware delivery mechanisms (e.g., malicious HTTP downloads)
- + Detect command and control (C2) channels
- + Reconstruct attacker behavior during lateral movement or data exfiltration

## **Considerations:**

- + Storage-intensive
- + Requires access to key network chokepoints or pre-configured capture

# NetFlow / sFlow / IPFIX

## What Are They?

- + These are flow-based telemetry that summarize network communications without capturing content. Each flow record typically includes:
  - + 5-tuple (source IP, destination IP, source port, destination port, protocol)
  - + Timestamps
  - + Packet and byte counts

## **Common Tools:**

- + nfdump/nfsen – CLI and web interface for querying NetFlow data
- + SiLK (CERT NetSA) – Powerful flow analysis and forensics suite
- + Elastiflow – Visualize flow data in the Elastic stack

# NetFlow / sFlow / IPFIX

## Use Cases in Incident Response:

- + Identify beaconing or regular interval connections
- + Detect scanning or brute force attempts
- + Spot unusual traffic spikes or large data transfers

## **Considerations:**

- + Lower fidelity—no payload or application-layer data
- + Must be enabled and tuned on routers/switches

# Firewall and IDS/IPS Logs

## What Are They?

- + Firewalls: Record allowed/blocked traffic, access control enforcement, and sometimes application-layer events
- + IDS/IPS: Monitor and alert on patterns that match known threats (signatures) or anomalies

## **Common Tools:**

- + Firewall logs: From Palo Alto, Fortinet, Cisco ASA, Check Point
- + IDS/IPS: Suricata, Snort, Cisco Firepower
- + SIEM integration: Splunk, Elastic, QRadar for log aggregation and alerting

# Firewall and IDS/IPS Logs

## Use Cases in Incident Response:

- + Triage alerts about potential attacks (e.g., exploit attempts)
- + Correlate log events with flow or PCAP data
- + Investigate lateral movement or failed login attempts

## **Considerations:**

- + Signature-based detection can miss novel or stealthy attacks
- + High false positive rate if not tuned properly

# Proxy and DNS Logs

## What Are They?

- + Proxy logs: Capture HTTP/HTTPS requests, including destination URLs, domains, user agents
- + DNS logs: Track domain name lookups and their resolving IPs, critical for detecting C2 and phishing

## **Common Tools:**

- + Blue Coat, Squid Proxy – For web traffic inspection
- + Zeek – For logging DNS queries and HTTP sessions
- + Windows DNS Server logs, BIND logs – Native resolution logs

# Proxy and DNS Logs

## Use Cases in Incident Response:

- + Detect DNS tunneling or domain generation algorithms (DGAs)
- + Investigate user browsing behavior leading up to an incident
- + Identify malicious or newly registered domains

## **Considerations:**

- + DNS over HTTPS (DoH) can bypass traditional DNS logging
- + May require decryption visibility to inspect HTTPS traffic via proxies



# Analyzing PCAPs with Wireshark

<https://t.me/learningnets>





# Lab Demo: Analyzing PCAPs with Wireshark

<https://t.me/learningnets>





# Network File Carving with Wireshark

<https://t.me/learningnets>



# Network File Carving

**Network file carving** is the process of *extracting files from raw network traffic captures* (typically in PCAP format).


It allows incident responders and forensic analysts to reconstruct files that were transferred across the network, often without needing access to the original endpoints.

# How Network File Carving Works

When files are transferred over protocols like HTTP, FTP, SMB, or SMTP, the file contents are embedded within the network packets. By analyzing and reassembling those packets, you can "carve out" the complete file.

This is commonly used in:

- + Malware analysis – Extracting malicious payloads delivered via phishing or drive-by downloads.
- + Data exfiltration investigations – Recovering files an attacker sent out of the network.
- + Intrusion analysis – Identifying unauthorized file transfers or dropped files.



# Lab Demo: Network File Carving with Wireshark

<https://t.me/learningnets>





# Investigating Network Scans

<https://t.me/learningnets>





# Lab Demo: Investigating Network Scans

<https://t.me/learningnets>





# Investigating Network Attacks

<https://t.me/learningnets>



# Scenario

A company's internal network is under serious attacks. The IDS at every Department are throwing various alerts of unknown network attacks.

You have been called to find out what's going on, explain what type of attacks are being conducted and how it may be possible to mitigate them.

The network engineers at each department installed a packet capturing device and presented you with the captured files.

## Goals

- + Examine the traffic generated by the various network attacks
- + Determine the attack and its potential impact
- + Find the machine where the attack originated from



# Lab Demo: Investigating Network Attacks

<https://t.me/learningnets>



# Incident Response: Analysis - Summary

# Key Concepts - Recap

- + Bridging detection & analysis
- + Evidence triage & collection
- + Endpoint forensics & live vs. dead-box
- + Log & network analysis with real-world tools



## Learning Outcomes Recap

- + Conduct deep endpoint & network analysis
- + Interpret PCAPs & system logs
- + Apply forensic techniques to real incidents
- + Use tools like EvtxECmd, Splunk & Wireshark effectively

# Next Steps

- + Review *Incident Response: Detection* if not yet completed
- + Explore advanced forensics or threat hunting paths
- + Practice tool workflows in lab environments

**THANKS FOR WATCHING!**

<https://t.me/learningnets>



*EXPERTS AT MAKING YOU AN EXPERT*



<https://t.me/learningnets>