Security Monitoring & Incident Report

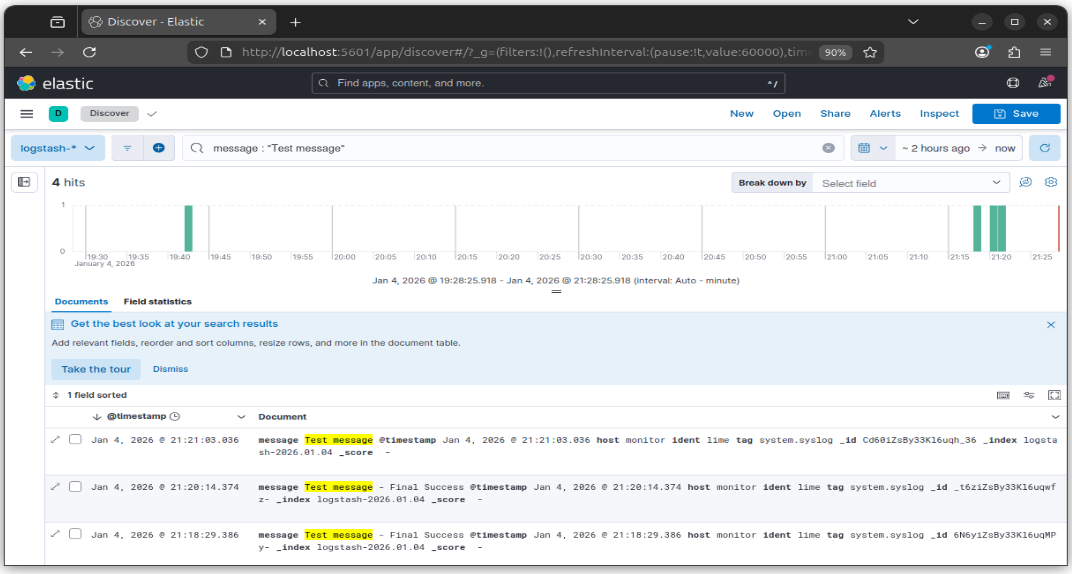
Introduction

1. Log Management
2. Security Tools
3. Log Analysis
4. Document Security Events
5. Monitoring Dashboards
6. Key Learnings
7. Conclusion
8. References

Log Management

1. Log Collection Pipeline

Set up Fluentd on Ubuntu to collect Syslog. Test by generating logs with `logger "Test message"`. Forward logs to Elastic SIEM and verify receipt



Discover - Elastic
http://localhost:5601/app/discover#/?_g=(filters:!(),refreshInterval:(pause:!t,value:60000),tim
elastic
Find apps, content, and more.
Discover
New
Open
Share
Alerts
Inspect
Save
logstash-*
~ 2 hours ago → now
Break down by
Select field
Documents
Field statistics
Get the best look at your search results
Add relevant fields, reorder and sort columns, resize rows, and more in the document table.
Take the tour
Dismiss
1 field sorted
@timestamp
Document

```
message : "Test message"
```

4 hits

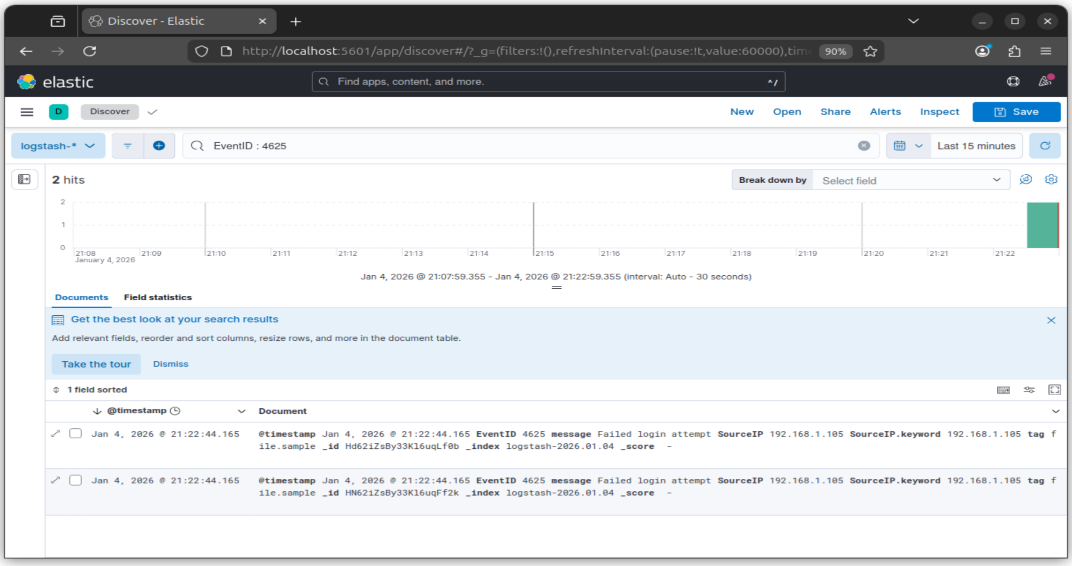
Jan 4, 2026 @ 19:28:25.918 - Jan 4, 2026 @ 21:28:25.918 (interval: Auto - minute)

```
Jan 4, 2026 @ 21:21:03.036   message Test message @timestamp Jan 4, 2026 @ 21:21:03.036 host monitor ident lime tag system.syslog _id Cd60iZsBy33K16uqh_36 _index logstash-2026.01.04 _score -
Jan 4, 2026 @ 21:20:14.374   message Test message - Final Success @timestamp Jan 4, 2026 @ 21:20:14.374 host monitor ident lime tag system.syslog _id _t6ziZsBy33K16uqwfz- _index logstash-2026.01.04 _score -
Jan 4, 2026 @ 21:18:29.386   message Test message - Final Success @timestamp Jan 4, 2026 @ 21:18:29.386 host monitor ident lime tag system.syslog _id 6N6yiZsBy33K16uqMPy- _index logstash-2026.01.04 _score -
```

Figure 1: Fluentd to Elastic SIEM

2. KQL Query Practice

In Elastic SIEM, write a KQL query to find Event ID 4625



Discover - Elastic
http://localhost:5601/app/discover#/?_g=(filters:!(),refreshInterval:(pause:!t,value:60000),tim
elastic
Find apps, content, and more.
Discover
New
Open
Share
Alerts
Inspect
Save
logstash-*
Last 15 minutes
Break down by
Select field
Documents
Field statistics
Get the best look at your search results
Add relevant fields, reorder and sort columns, resize rows, and more in the document table.
Take the tour
Dismiss
1 field sorted
@timestamp
Document

```
EventID : 4625
```

2 hits

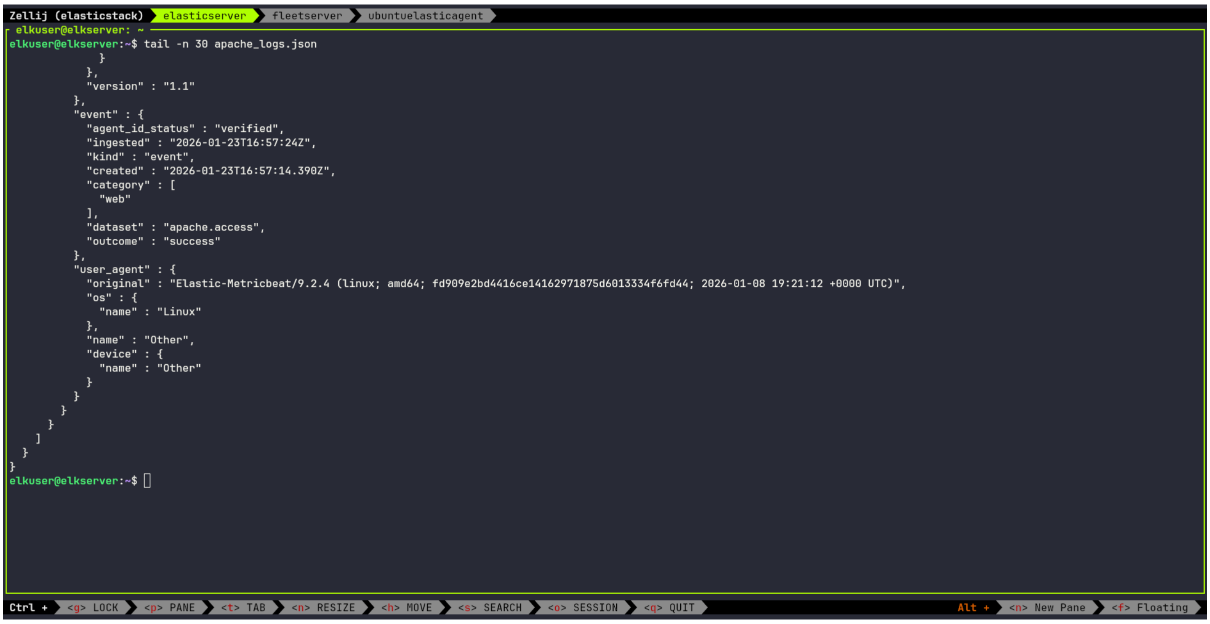
Jan 4, 2026 @ 21:07:59.355 - Jan 4, 2026 @ 21:22:59.355 (interval: Auto - 30 seconds)

```
Jan 4, 2026 @ 21:22:44.165   @timestamp Jan 4, 2026 @ 21:22:44.165 EventID 4625 message Failed login attempt SourceIP 192.168.1.105 SourceIP.keyword 192.168.1.105 tag file.sample _id Hd62iZsBy33K16uqLf0b _index logstash-2026.01.04 _score -
Jan 4, 2026 @ 21:22:44.165   @timestamp Jan 4, 2026 @ 21:22:44.165 EventID 4625 message Failed login attempt SourceIP 192.168.1.105 SourceIP.keyword 192.168.1.105 tag file.sample _id HN62iZsBy33K16uqFf2k _index logstash-2026.01.04 _score -
```

Figure 2: KQL Query for Audit Failure

3. Normalization Exercise

Convert an Apache access log to JSON. Save output to a file and check the format



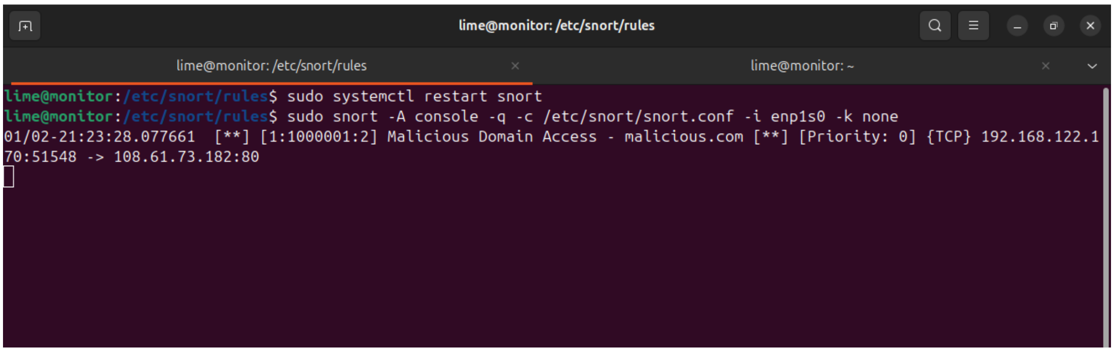
```
elkuser@elkserver:~$ tail -n 30 apache_logs.json
              }
            },
            "version" : "1.1"
          },
          "event" : {
            "agent_id_status" : "verified",
            "ingested" : "2026-01-23T16:57:24Z",
            "kind" : "event",
            "created" : "2026-01-23T16:57:14.390Z",
            "category" : [
              "web"
            ],
            "dataset" : "apache.access",
            "outcome" : "success"
          },
          "user_agent" : {
            "original" : "Elastic-Metricbeat/9.2.4 (linux; amd64; fd909e2bd4416ce14162971875d6013334f6fd44; 2026-01-08 19:21:12 +0000 UTC)",
            "os" : {
              "name" : "Linux"
            },
            "name" : "Other",
            "device" : {
              "name" : "Other"
            }
          }
        }
      }
    ]
  }
}
elkuser@elkserver:~$
```

Figure 3: Raw Apache Access Logs to JSON

Security Tools

1. Snort Rule Testing

Write a rule to detect HTTP requests to "malicious.com":

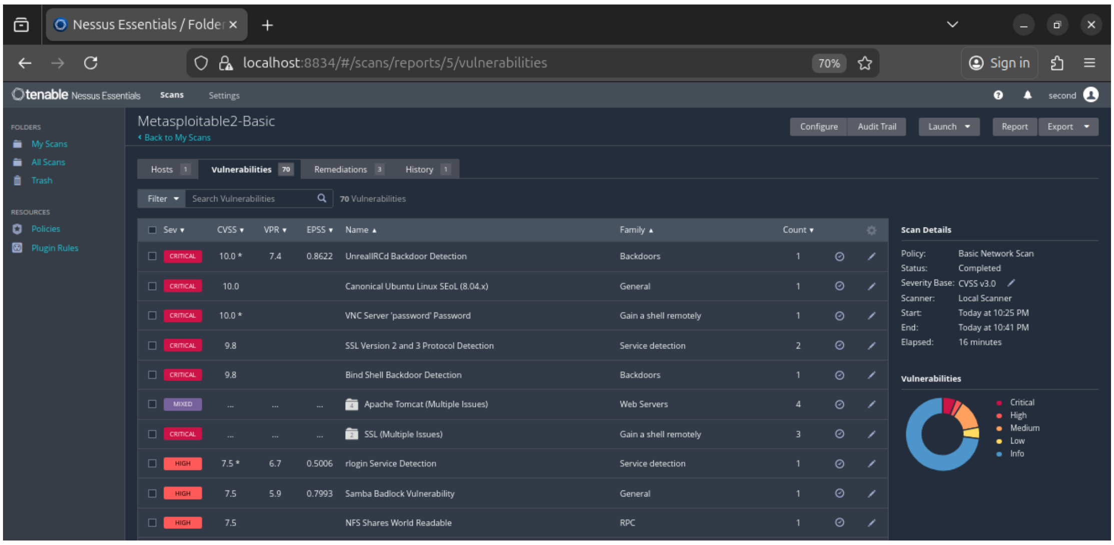

lime@monitor: /etc/snort/rules
lime@monitor: /etc/snort/rules
lime@monitor: ~

```
lime@monitor:/etc/snort/rules$ sudo systemctl restart snort
lime@monitor:/etc/snort/rules$ sudo snort -A console -q -c /etc/snort/snort.conf -i enp1s0 -k none
01/02-21:23:28.077661  [**] [1:1000001:2] Malicious Domain Access - malicious.com [**] [Priority: 0] {TCP} 192.168.122.1
70:51548 -> 108.61.73.182:80
```

Figure 4: Snort Rule for Malicious Domain

2. Nessus Scan

Run Nessus Essentials against Metasploitable2 and list the top 3 vulnerabilities by CVSS score



Metasploitable2-Basic

‹ Back to My Scans

70 Vulnerabilities

Sev ▾	CVSS ▾	VPR ▾	EPSS ▾	Name ▴	Family ▴	Count ▾
CRITICAL	10.0 *	7.4	0.8622	UnrealIRCd Backdoor Detection	Backdoors	1
CRITICAL	10.0			Canonical Ubuntu Linux SEoL (8.04.x)	General	1
CRITICAL	10.0 *			VNC Server 'password' Password	Gain a shell remotely	1
CRITICAL	9.8			SSL Version 2 and 3 Protocol Detection	Service detection	2
CRITICAL	9.8			Bind Shell Backdoor Detection	Backdoors	1
MIXED	...	...	...	4 Apache Tomcat (Multiple Issues)	Web Servers	4
CRITICAL	...	...	...	2 SSL (Multiple Issues)	Gain a shell remotely	3
HIGH	7.5 *	6.7	0.5006	rlogin Service Detection	Service detection	1
HIGH	7.5	5.9	0.7993	Samba Badlock Vulnerability	General	1
HIGH	7.5			NFS Shares World Readable	RPC	1

Scan Details

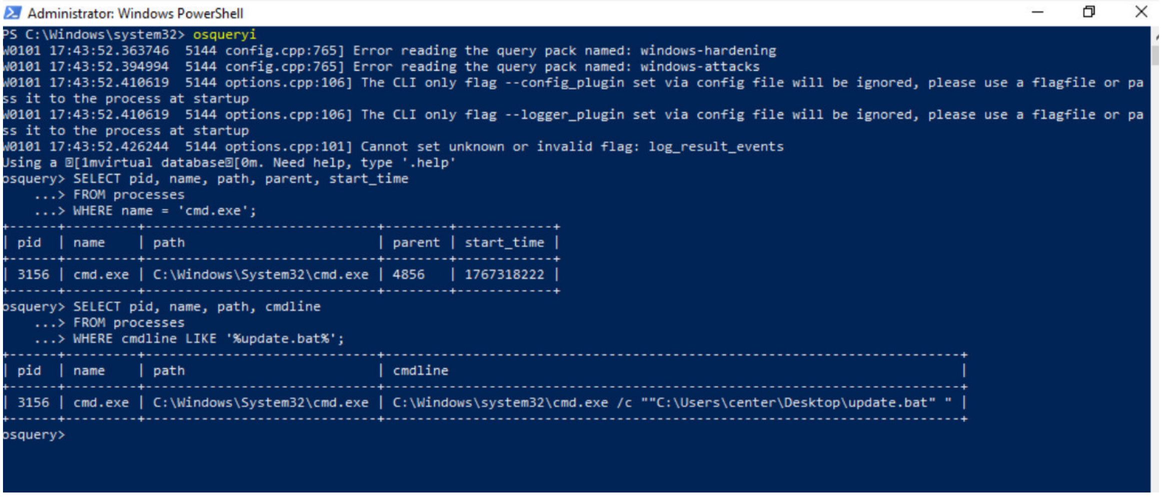
Policy:	Basic Network Scan
Status:	Completed
Severity Base:	CVSS v3.0
Scanner:	Local Scanner
Start:	Today at 10:25 PM
End:	Today at 10:41 PM
Elapsed:	16 minutes

Vulnerabilities

Figure 5: Nessus Vulnerability Scan Results

3. Osquery Monitoring

Install Osquery on Windows VM. Query running processes (SELECT * FROM processes;) and simulate a malicious process with a harmless batch file



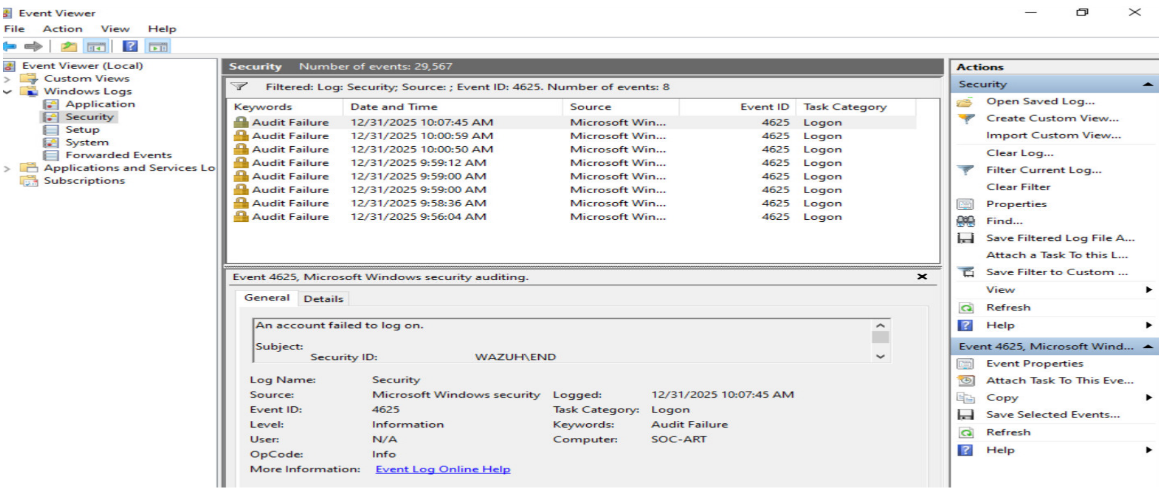
```
PS C:\Windows\system32> osqueryi
W0101 17:43:52.363746  5144 config.cpp:765] Error reading the query pack named: windows-hardening
W0101 17:43:52.394994  5144 config.cpp:765] Error reading the query pack named: windows-attacks
W0101 17:43:52.410619  5144 options.cpp:106] The CLI only flag --config_plugin set via config file will be ignored, please use a flagfile or pa
ss it to the process at startup
W0101 17:43:52.410619  5144 options.cpp:106] The CLI only flag --logger_plugin set via config file will be ignored, please use a flagfile or pa
ss it to the process at startup
W0101 17:43:52.426244  5144 options.cpp:101] Cannot set unknown or invalid flag: log_result_events
Using a [1mvirtual database[0m. Need help, type '.help'
osquery> SELECT pid, name, path, parent, start_time
    ...> FROM processes
    ...> WHERE name = 'cmd.exe';
+------+---------+-----------------------------+--------+------------+
| pid  | name    | path                        | parent | start_time |
+------+---------+-----------------------------+--------+------------+
| 3156 | cmd.exe | C:\Windows\System32\cmd.exe | 4856   | 1767318222 |
+------+---------+-----------------------------+--------+------------+
osquery> SELECT pid, name, path, cmdline
    ...> FROM processes
    ...> WHERE cmdline LIKE '%update.bat%';
+------+---------+-----------------------------+------------------------------------------------------------------------+
| pid  | name    | path                        | cmdline                                                                |
+------+---------+-----------------------------+------------------------------------------------------------------------+
| 3156 | cmd.exe | C:\Windows\System32\cmd.exe | C:\Windows\system32\cmd.exe /c ""C:\Users\center\Desktop\update.bat" " |
+------+---------+-----------------------------+------------------------------------------------------------------------+
osquery>
```

Figure 6: Querying System Processes via Osquery

Log Analysis Practice

1. Windows Event Viewer

Filter for Event ID 4625 or 7045



Security Number of events: 29,567

Filtered: Log: Security; Source: ; Event ID: 4625. Number of events: 8

Keywords	Date and Time	Source	Event ID	Task Category
Audit Failure	12/31/2025 10:07:45 AM	Microsoft Win...	4625	Logon
Audit Failure	12/31/2025 10:00:59 AM	Microsoft Win...	4625	Logon
Audit Failure	12/31/2025 10:00:50 AM	Microsoft Win...	4625	Logon
Audit Failure	12/31/2025 9:59:12 AM	Microsoft Win...	4625	Logon
Audit Failure	12/31/2025 9:59:00 AM	Microsoft Win...	4625	Logon
Audit Failure	12/31/2025 9:59:00 AM	Microsoft Win...	4625	Logon
Audit Failure	12/31/2025 9:58:36 AM	Microsoft Win...	4625	Logon
Audit Failure	12/31/2025 9:56:04 AM	Microsoft Win...	4625	Logon

Event 4625, Microsoft Windows security auditing.

An account failed to log on.

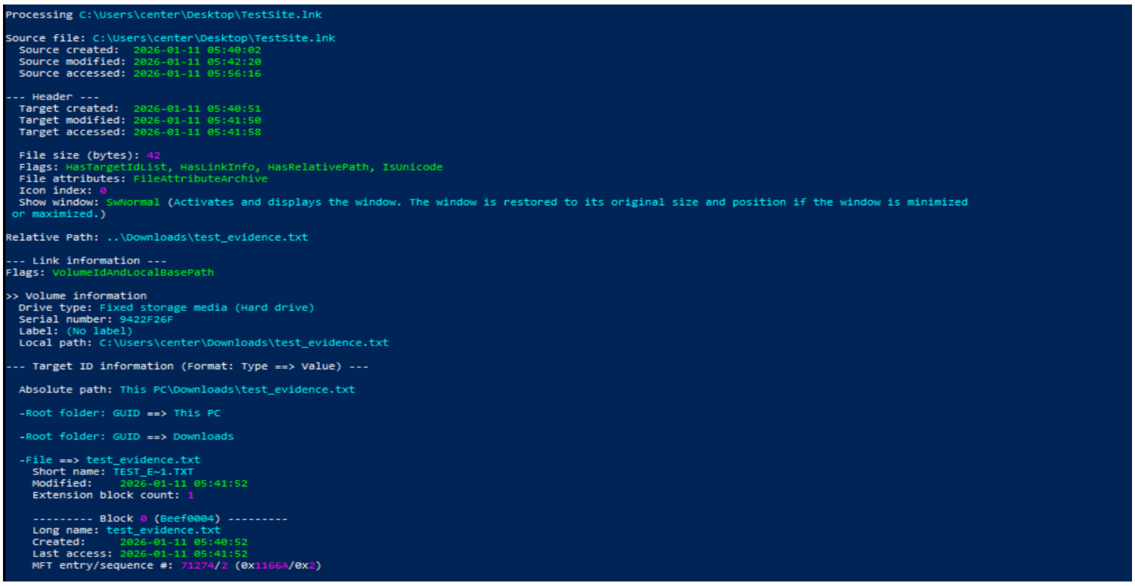
Subject:
Security ID: WAZUH\END

Log Name:	Security		
Source:	Microsoft Windows security	Logged:	12/31/2025 10:07:45 AM
Event ID:	4625	Task Category:	Logon
Level:	Information	Keywords:	Audit Failure
User:	N/A	Computer:	SOC-ART
OpCode:	Info		
More Information:	Event Log Online Help		

Figure 7: Filtering for Authentication Failures

2. Zimmerman Tools

Use Eric Zimmerman's Tools to parse Windows Shortcut (LNK) files for malicious content



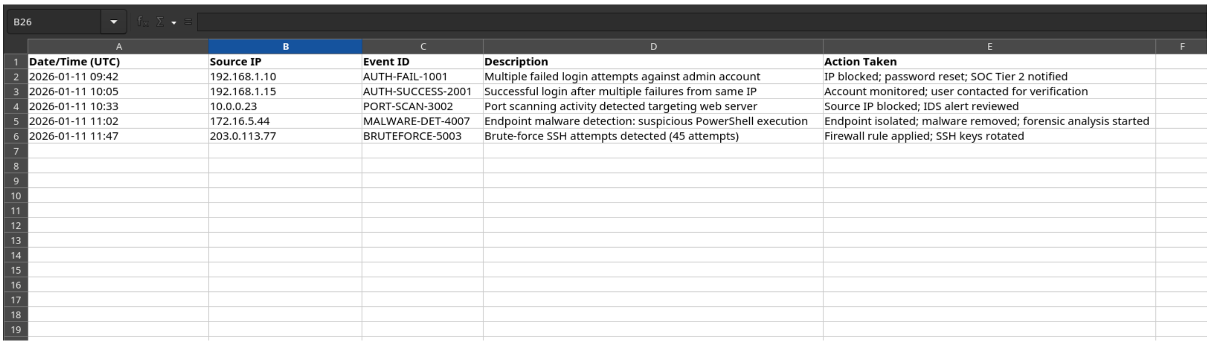
```
Processing C:\Users\center\Desktop\TestSite.lnk

Source file: C:\Users\center\Desktop\TestSite.lnk
  Source created:  2026-01-11 05:40:02
  Source modified: 2026-01-11 05:42:20
  Source accessed: 2026-01-11 05:56:16

--- Header ---
  Target created:  2026-01-11 05:40:51
  Target modified: 2026-01-11 05:41:50
  Target accessed: 2026-01-11 05:41:58

  File size (bytes): 42
  Flags: HasTargetIdList, HasLinkInfo, HasRelativePath, IsUnicode
  File attributes: FileAttributeArchive
  Icon index: 0
  Show window: SwNormal (Activates and displays the window. The window is restored to its original size and position if the window is minimized
 or maximized.)

Relative Path: ..\Downloads\test_evidence.txt

--- Link information ---
Flags: VolumeIdAndLocalBasePath

>> Volume information
  Drive type: Fixed storage media (Hard drive)
  Serial number: 9422F26F
  Label: (No label)
  Local path: C:\Users\center\Downloads\test_evidence.txt

--- Target ID information (Format: Type ==> Value) ---

  Absolute path: This PC\Downloads\test_evidence.txt

  -Root folder: GUID ==> This PC

  -Root folder: GUID ==> Downloads

  -File ==> test_evidence.txt
    Short name: TEST_E~1.TXT
    Modified:     2026-01-11 05:41:52
    Extension block count: 1

    --------- Block 0 (Beef0004) ---------
    Long name: test_evidence.txt
    Created:     2026-01-11 05:40:52
    Last access: 2026-01-11 05:41:52
    MFT entry/sequence #: 71274/2 (0x1166A/0x2)
```

Figure 8: Parsing LNK Files for Malicious Indicators

Document Security Events

1. Mock Event

Create template with Date/Time | Source IP | Event ID | Description | Action Taken

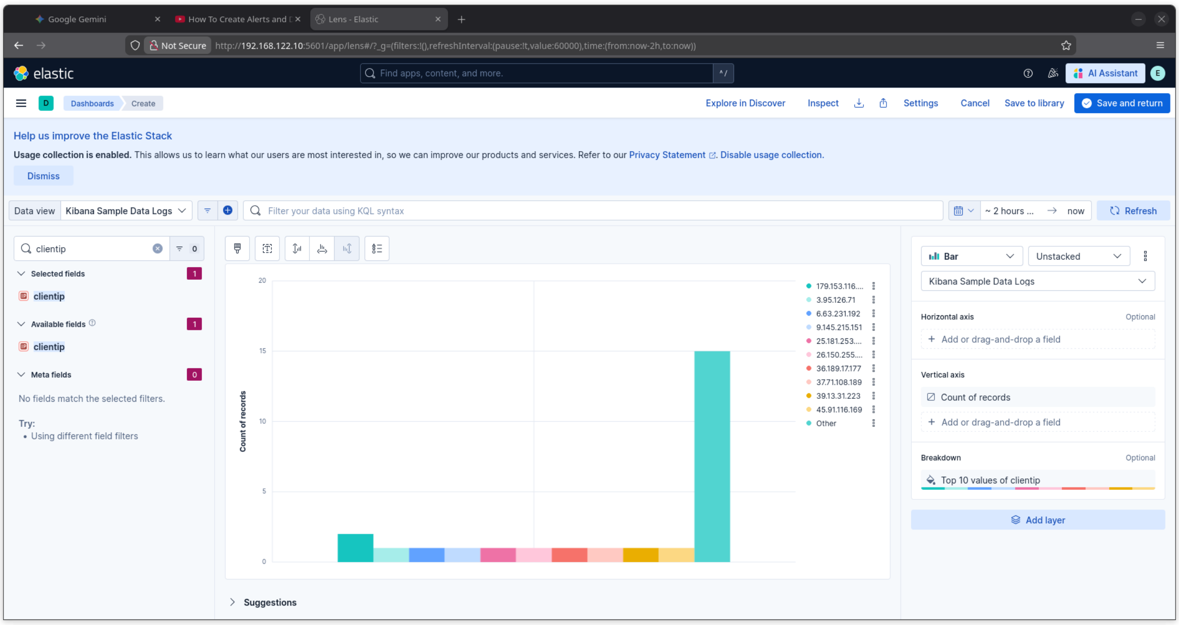


Date/Time (UTC)	Source IP	Event ID	Description	Action Taken
2026-01-11 09:42	192.168.1.10	AUTH-FAIL-1001	Multiple failed login attempts against admin account	IP blocked; password reset; SOC Tier 2 notified
2026-01-11 10:05	192.168.1.15	AUTH-SUCCESS-2001	Successful login after multiple failures from same IP	Account monitored; user contacted for verification
2026-01-11 10:33	10.0.0.23	PORT-SCAN-3002	Port scanning activity detected targeting web server	Source IP blocked; IDS alert reviewed
2026-01-11 11:02	172.16.5.44	MALWARE-DET-4007	Endpoint malware detection: suspicious PowerShell execution	Endpoint isolated; malware removed; forensic analysis started
2026-01-11 11:47	203.0.113.77	BRUTEFORCE-5003	Brute-force SSH attempts detected (45 attempts)	Firewall rule applied; SSH keys rotated

Figure 9: Mock Event Documentation

Monitoring Dashboards

1. Top 10 source IPs generating alerts



http://192.168.122.10:5601/app/lens#/?_g=(filters:!(),refreshInterval:(pause:!t,value:60000),time:(from:now-2h,to:now))

Help us improve the Elastic Stack

Usage collection is enabled. This allows us to learn what our users are most interested in, so we can improve our products and services. Refer to our Privacy Statement. Disable usage collection.

Data view: Kibana Sample Data Logs

No fields match the selected filters.

Try:
- Using different field filters

Bar | Unstacked

Kibana Sample Data Logs

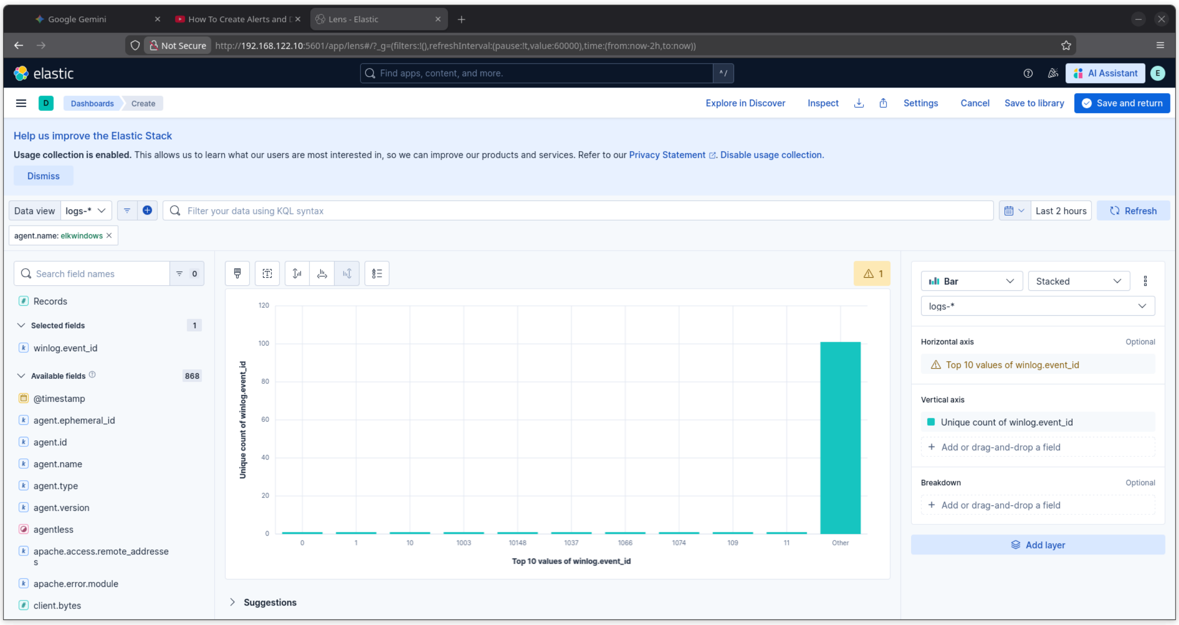
Horizontal axis (Optional): + Add or drag-and-drop a field

Vertical axis: Count of records

Breakdown (Optional): Top 10 values of clientip

Figure 10: Top 10 Source IP Distribution

2. Frequency of critical Event Ids

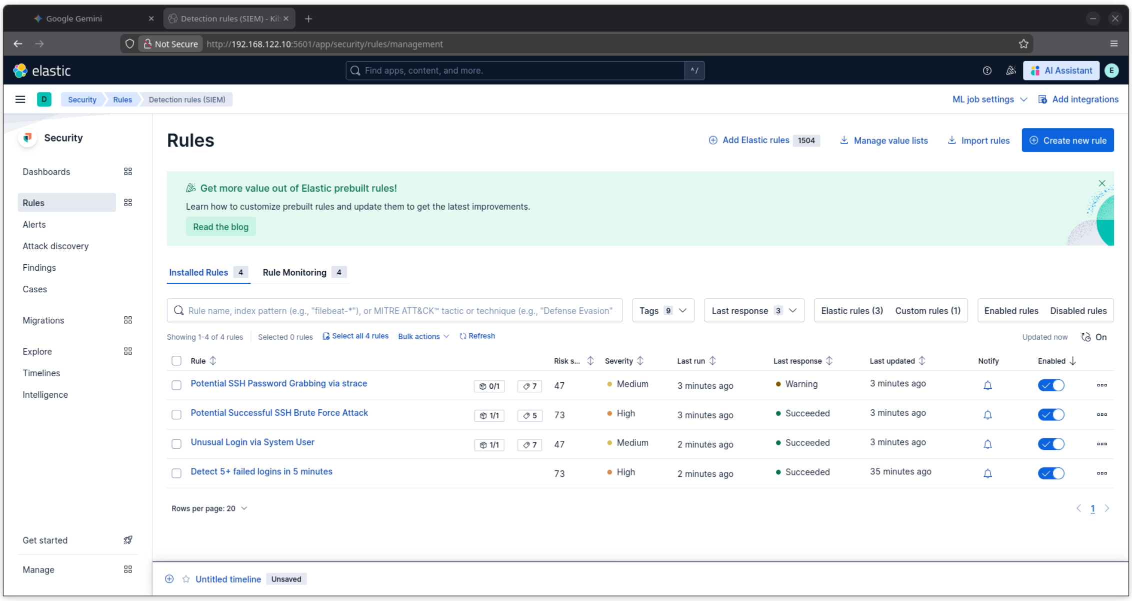


Help us improve the Elastic Stack

Usage collection is enabled. This allows us to learn what our users are most interested in, so we can improve our products and services. Refer to our Privacy Statement. Disable usage collection.

Figure 11: Top 10 Event ID Distribution

3. Prebuilt Dashboard



Get more value out of Elastic prebuilt rules!

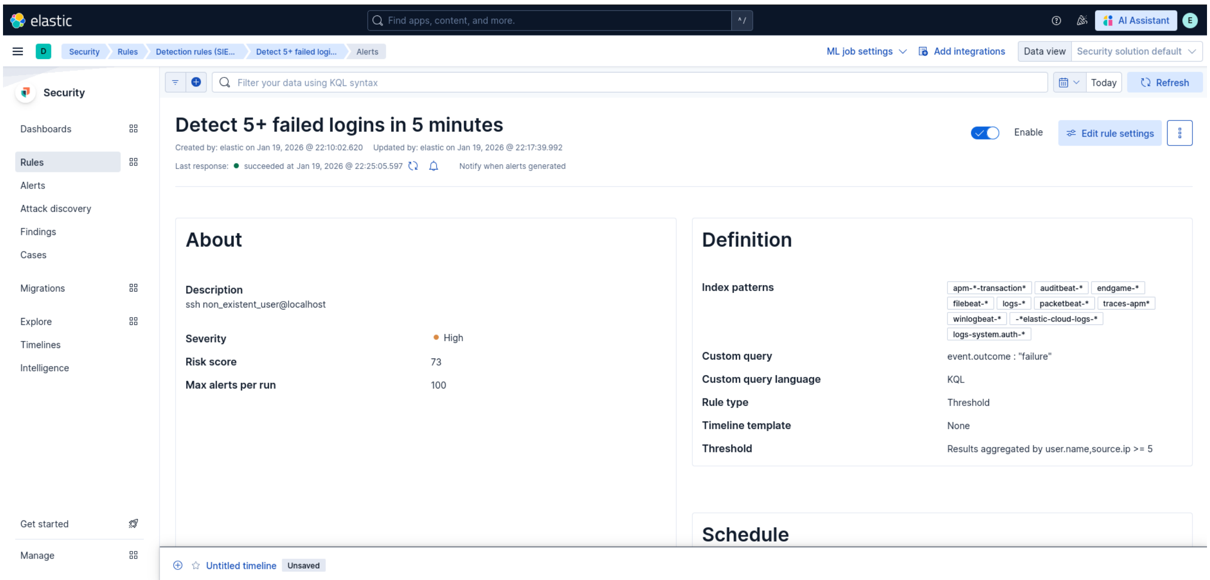
Learn how to customize prebuilt rules and update them to get the latest improvements.

Rule			Risk s...	Severity	Last run	Last response	Last updated	Notify	Enabled
Potential SSH Password Grabbing via strace	0/1	7	47	● Medium	3 minutes ago	● Warning	3 minutes ago		✓
Potential Successful SSH Brute Force Attack	1/1	5	73	● High	3 minutes ago	● Succeeded	3 minutes ago		✓
Unusual Login via System User	1/1	7	47	● Medium	2 minutes ago	● Succeeded	3 minutes ago		✓
Detect 5+ failed logins in 5 minutes			73	● High	2 minutes ago	● Succeeded	35 minutes ago		✓

Figure 12: Sigma-Based Behavioral Analytics

4. Configure Alert Rules

In Elastic SIEM: Create: Rule: "Detect 5+ failed logins in 5 minutes", Index: security- login-*, Condition: count > 5



Detect 5+ failed logins in 5 minutes

Created by: elastic on Jan 19, 2026 @ 22:10:02.620 Updated by: elastic on Jan 19, 2026 @ 22:17:39.992

Last response: ● succeeded at Jan 19, 2026 @ 22:25:05.597 Notify when alerts generated

About

Description
ssh non_existent_user@localhost

Severity	● High
Risk score	73
Max alerts per run	100

Definition

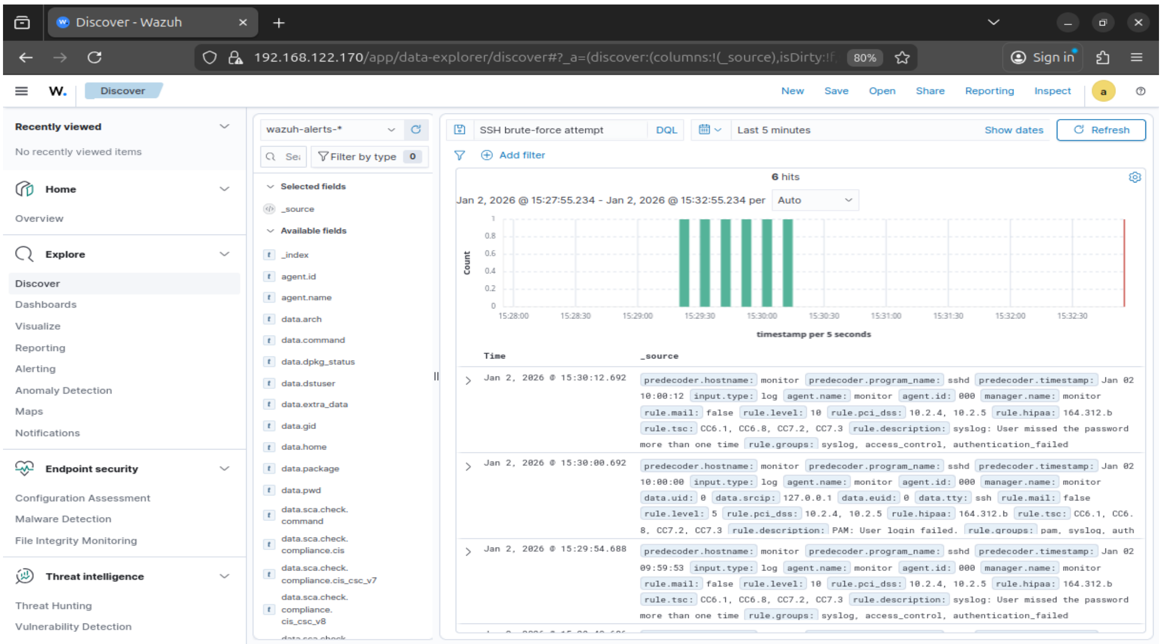
Index patterns	apm-*-transaction* auditbeat-* endgame-* filebeat-* logs-* packetbeat-* traces-apm* winlogbeat-* -*elastic-cloud-logs-* logs-system.auth-*
Custom query	event.outcome : "failure"
Custom query language	KQL
Rule type	Threshold
Timeline template	None
Threshold	Results aggregated by user.name,source.ip >= 5

Schedule

Figure 11: Failed Login Threshold Rule

5. Alert Rule in Wazuh

Create a rule in Wazuh to detect 3+ failed logins in 2 minutes



SSH brute-force attempt DQL Last 5 minutes

6 hits

Jan 2, 2026 @ 15:27:55.234 - Jan 2, 2026 @ 15:32:55.234 per Auto

```
Time                          _source
Jan 2, 2026 @ 15:30:12.692    predecoder.hostname: monitor predecoder.program_name: sshd predecoder.timestamp: Jan 02
                              10:00:12 input.type: log agent.name: monitor agent.id: 000 manager.name: monitor
                              rule.mail: false rule.level: 10 rule.pci_dss: 10.2.4, 10.2.5 rule.hipaa: 164.312.b
                              rule.tsc: CC6.1, CC6.8, CC7.2, CC7.3 rule.description: syslog: User missed the password
                              more than one time rule.groups: syslog, access_control, authentication_failed
Jan 2, 2026 @ 15:30:00.692    predecoder.hostname: monitor predecoder.program_name: sshd predecoder.timestamp: Jan 02
                              10:00:00 input.type: log agent.name: monitor agent.id: 000 manager.name: monitor
                              data.uid: 0 data.srcip: 127.0.0.1 data.euid: 0 data.tty: ssh rule.mail: false
                              rule.level: 5 rule.pci_dss: 10.2.4, 10.2.5 rule.hipaa: 164.312.b rule.tsc: CC6.1, CC6.
                              8, CC7.2, CC7.3 rule.description: PAM: User login failed. rule.groups: pam, syslog, auth
Jan 2, 2026 @ 15:29:54.688    predecoder.hostname: monitor predecoder.program_name: sshd predecoder.timestamp: Jan 02
                              09:59:53 input.type: log agent.name: monitor agent.id: 000 manager.name: monitor
                              rule.mail: false rule.level: 10 rule.pci_dss: 10.2.4, 10.2.5 rule.hipaa: 164.312.b
                              rule.tsc: CC6.1, CC6.8, CC7.2, CC7.3 rule.description: syslog: User missed the password
                              more than one time rule.groups: syslog, access_control, authentication_failed
```

Figure 12: Wazuh Brute-Force Detection Rule

Key Learnings

1. Integration of Automated Detection Engineering: Transitioning from manual log review to automated alerting is a fundamental requirement for modern security operations. By implementing **Sigma rules** and custom **Wazuh/Elastic threshold logic**, the project successfully transformed raw telemetry into actionable intelligence. This implementation ensures that high-frequency threats, such as SSH brute-force attacks (Event ID 4625), are flagged in real-time, significantly reducing the mean time to detect (MTTD) without requiring constant human oversight.

2. Forensic Visibility through Structured Telemetry: Visibility serves as the primary foundation for defensive operations. Utilizing **Osquery** for Windows endpoint monitoring and **Zimmerman's LECmd** for LNK file analysis provided deep visibility into system-level activity. This approach demonstrates a commitment to "forensic readiness," moving beyond simple log collection to the parsing of complex artifacts. These methods are essential for identifying indicators of malicious persistence and lateral movement that traditional logging might overlook.

3. Operational Efficiency through Data Transformation: Effective security monitoring is predicated on standardized, machine-readable data. The conversion of **Apache logs to JSON** and the subsequent development of **Kibana/Grafana dashboards** highlight the critical role of data normalization. Structured data enables the high-speed, complex visualizations—such as Top 10 Source IP distributions and event frequency heatmaps—necessary for identifying anomalies and emerging threats at scale.

Conclusion

The project moved beyond passive observation by deploying and testing active security controls. By writing custom Snort IDS rules to flag malicious domains and engineering Wazuh/Elastic threshold alerts to detect brute-force patterns (Event ID 4625), the environment was proven capable of identifying threats in real-time. The integration of Nessus vulnerability scanning and Osquery process monitoring ensured a defense-in-depth posture, allowing for a full security operations workflow from initial detection and triage to documented incident response.

References

1. **Elastic Stack (Kibana/SIEM):** https://www.elastic.co/guide/index.html
2. **Wazuh (Open Source XDR**): https://documentation.wazuh.com/
3. **Snort IDS/IPS:** https://www.snort.org/documents
4. **Osquery:** https://osquery.io/
5. **Eric Zimmerman's Tools:** https://ericzimmerman.github.io/
6. **Nessus Essentials:** https://www.tenable.com/products/nessus/nessus-essentials
7. **Fluentd:** https://docs.fluentd.org/
8. **Logstash:** https://www.elastic.co/guide/en/logstash/current/index.html
9. **Metasploitable2:** https://sourceforge.net/projects/metasploitable/