Network Design, Traffic Analysis & SIEM Monitoring

Lab Documentation Report

1. Introduction

1.1 Objective

The challenge relates to the overall process concerning the design as well as protection of a small office network. The focus is on the application of subnetting to optimize IP address usage and logically segment the network to enhance efficiency and security. Substantive to the challenge is the process that entails the analysis of network activities to detect normal network communication patterns, as well as the assessment of potential network threats at the packet level. The challenge further incorporates the application of the concept of a Security Information and Event Management (SIEM) solution that utilizes the Wazuh SIEM platform to continuously monitor the event logs to achieve comprehensive event analysis and overall real-time security insights and monitoring. The overall challenge seeks to collectively enable the learner to gain practical skills relevant to the different contours of network design, network inspection, and security aspects of network threat protection and event correlation.

1.2 Report by:

Name: Shah Devam

Email ID: shahdevam48@gmail.com

Tools used: Cisco Packet Tracer, Linux (Kali linux, Ubuntu), and Wazuh

2. Subnet Design for Small Office Network

2.1 Network Requirements

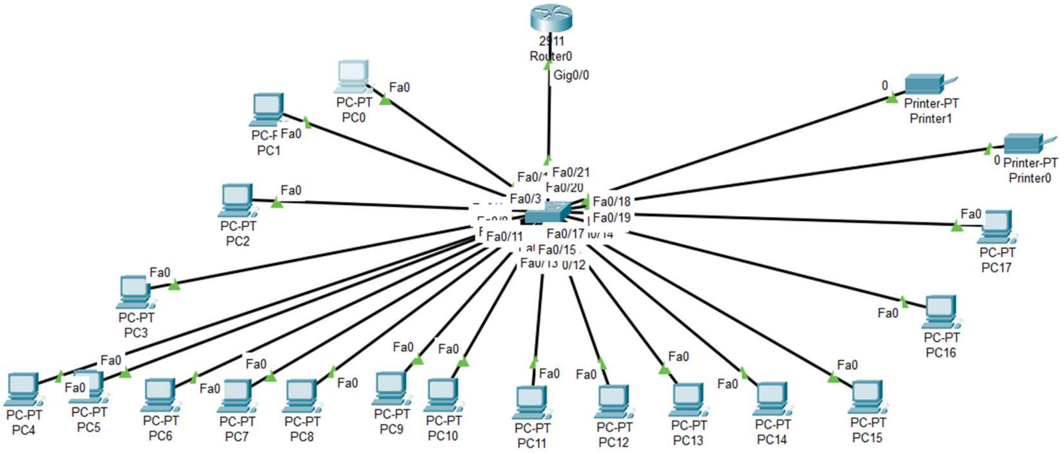
- Total devices: 20
- Environment: Small office

- Addressing: Private IP addressing
- Tool used: Cisco Packet Tracer

2.2 IP Range Selection

The private IP range **192.168.1.0/24** was selected because:

- It provides sufficient IP addresses for current and future expansion
- It is easy to manage and widely used in small office environments
- It allows clean separation from public networks



2.3 Subnet Calculation

PARAMETER	VALUE
CIDR NOTATION	/24
SUBNET MASK	255.255.255.0
TOTAL IPS	256

USABLE IPS	254
NETWORK ADDRESS	192.168.1.0
BROADCAST ADDRESS	192.168.1.255
USABLE RANGE	192.168.1.1 – 192.168.1.254

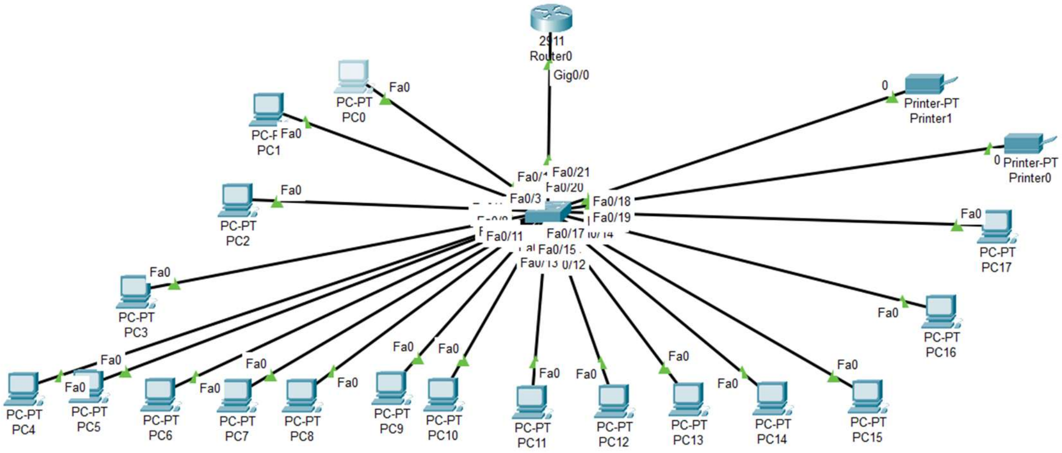
3. Network Topology Implementation (Cisco Packet Tracer)

3.1 Topology Design

The network consists of:

- One router (default gateway)
- One switch
- Multiple end devices (PCs and printers)

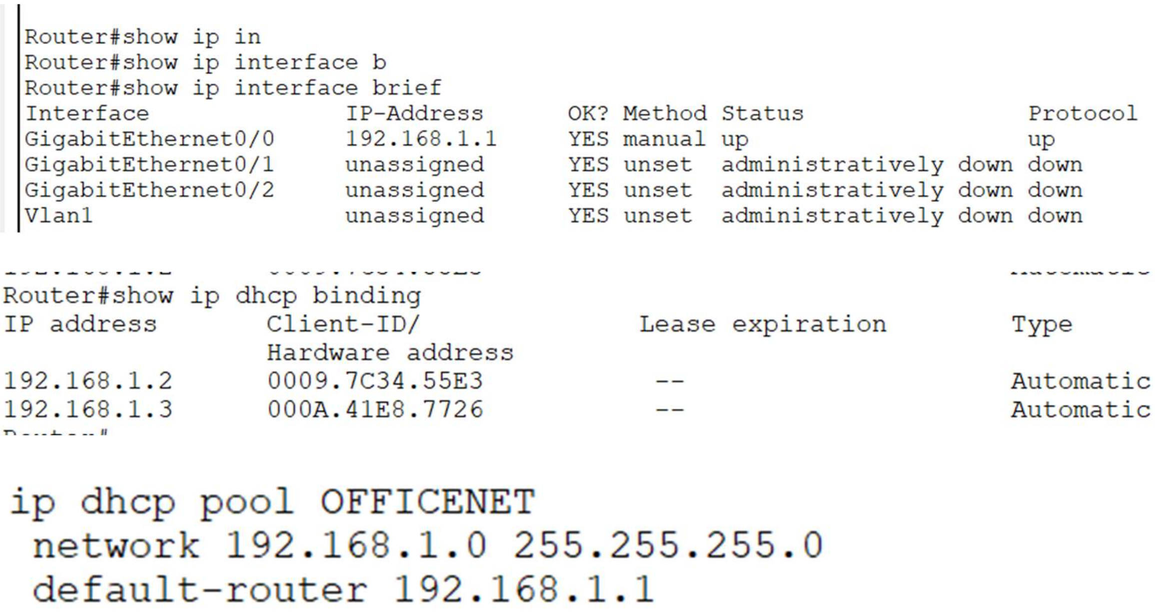
All devices are connected in a star topology via a central switch.



Small office network topology designed using Cisco Packet Tracer

3.2 DHCP Configuration via Router CLI

DHCP was configured on the router to automatically assign IP addresses to client devices, reducing manual configuration errors.



```
Router#show ip in
Router#show ip interface b
Router#show ip interface brief
Interface              IP-Address      OK? Method Status                Protocol
GigabitEthernet0/0     192.168.1.1     YES manual up                    up
GigabitEthernet0/1     unassigned      YES unset  administratively down down
GigabitEthernet0/2     unassigned      YES unset  administratively down down
Vlan1                  unassigned      YES unset  administratively down down
```

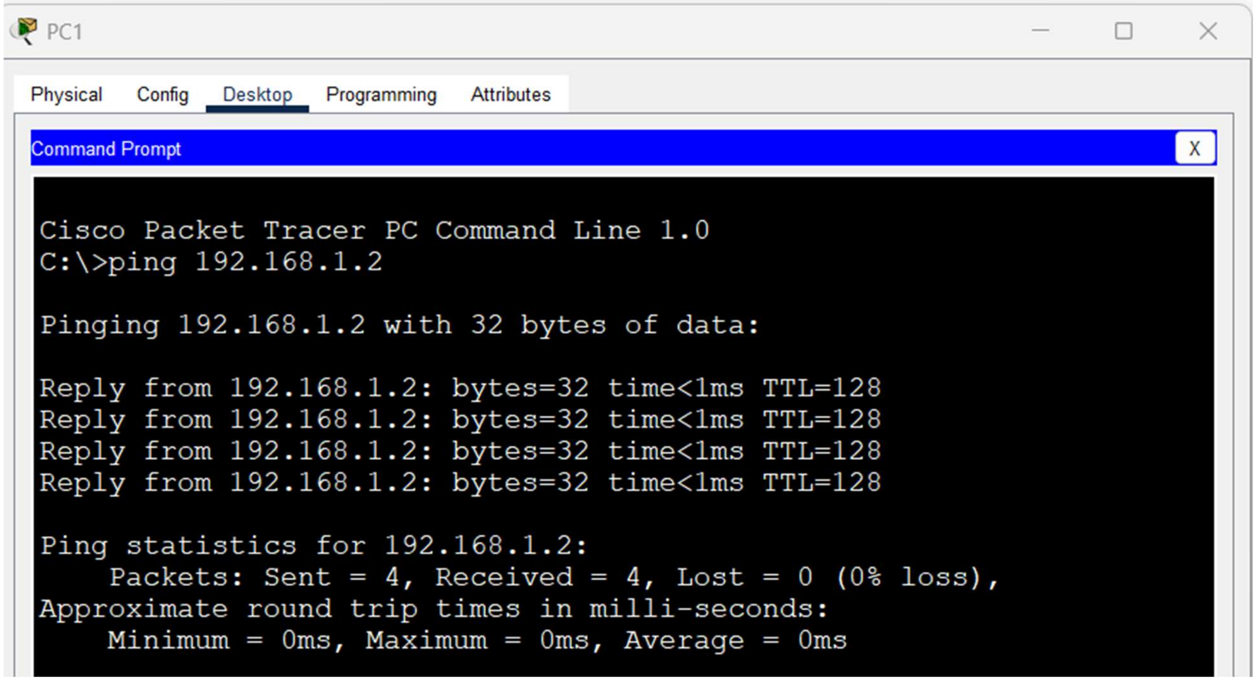
```
Router#show ip dhcp binding
IP address       Client-ID/              Lease expiration        Type
                 Hardware address
192.168.1.2      0009.7C34.55E3           --                     Automatic
192.168.1.3      000A.41E8.7726           --                     Automatic
```

```
ip dhcp pool OFFICENET
 network 192.168.1.0 255.255.255.0
 default-router 192.168.1.1
```

DHCP configuration on router using Cisco CLI

3.3 IP Address Verification

Client devices successfully received IP addresses from the DHCP pool.



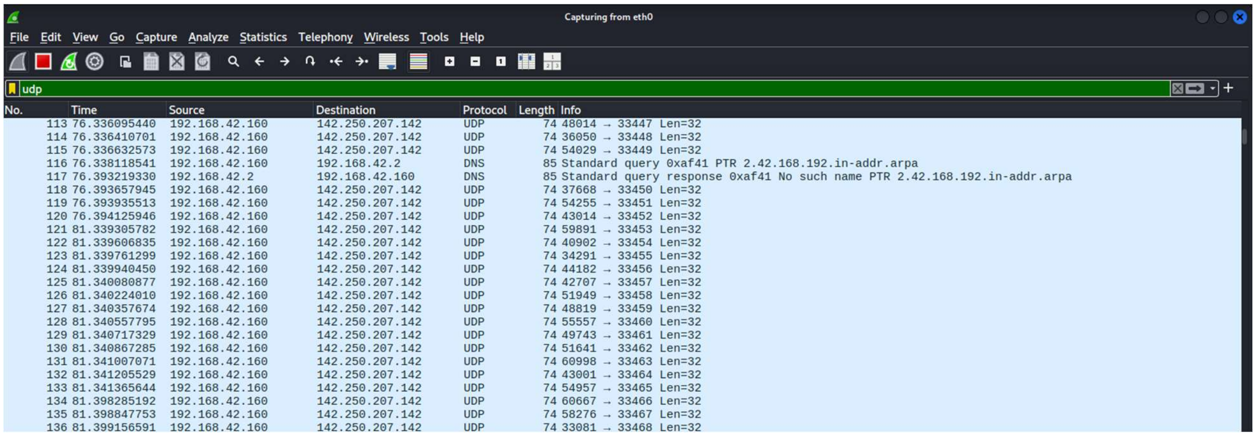
```
Cisco Packet Tracer PC Command Line 1.0
C:\>ping 192.168.1.2

Pinging 192.168.1.2 with 32 bytes of data:

Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128
Reply from 192.168.1.2: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.1.2:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms
```

Successful IP address assignment verified using ipconfig on client device

4. Network Traffic Capture & Analysis

4.1 Traffic Capture Setup

Wireshark was used as a packet sniffing utility to observe and track live communication flow in the environment. The traces captured involved communication made using various protocols like ICMP, which was used to verify connectivity and trace the path of communication in a network; HTTP, which was representative of communication made in a web-based application scenario; and TCP, which was useful in understanding the session establishment and the data transfer process in a communication session. This provided an opportunity to carefully analyze the details of the packets, the flow of communication, and the functionality of various protocols in a very practical way.

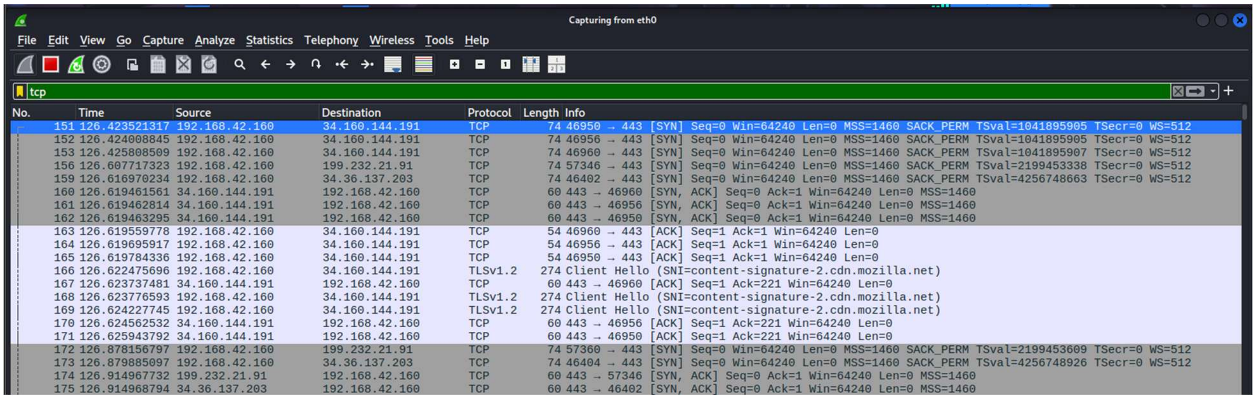

Capturing from eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
udp

```
No.   Time         Source          Destination      Protocol Length Info
  113 76.336095440 192.168.42.160  142.250.207.142  UDP        74 48014 → 33447 Len=32
  114 76.336410701 192.168.42.160  142.250.207.142  UDP        74 36050 → 33448 Len=32
  115 76.336632573 192.168.42.160  142.250.207.142  UDP        74 54029 → 33449 Len=32
  116 76.338118541 192.168.42.160  192.168.42.2     DNS        85 Standard query 0xaf41 PTR 2.42.168.192.in-addr.arpa
  117 76.393219330 192.168.42.2    192.168.42.160   DNS        85 Standard query response 0xaf41 No such name PTR 2.42.168.192.in-addr.arpa
  118 76.393657945 192.168.42.160  142.250.207.142  UDP        74 37668 → 33450 Len=32
  119 76.393935513 192.168.42.160  142.250.207.142  UDP        74 54255 → 33451 Len=32
  120 76.394125946 192.168.42.160  142.250.207.142  UDP        74 43014 → 33452 Len=32
  121 81.339305782 192.168.42.160  142.250.207.142  UDP        74 59891 → 33453 Len=32
  122 81.339606835 192.168.42.160  142.250.207.142  UDP        74 40902 → 33454 Len=32
  123 81.339761299 192.168.42.160  142.250.207.142  UDP        74 34291 → 33455 Len=32
  124 81.339940450 192.168.42.160  142.250.207.142  UDP        74 44182 → 33456 Len=32
  125 81.340080877 192.168.42.160  142.250.207.142  UDP        74 42707 → 33457 Len=32
  126 81.340224010 192.168.42.160  142.250.207.142  UDP        74 51949 → 33458 Len=32
  127 81.340357674 192.168.42.160  142.250.207.142  UDP        74 48819 → 33459 Len=32
  128 81.340557795 192.168.42.160  142.250.207.142  UDP        74 55557 → 33460 Len=32
  129 81.340717329 192.168.42.160  142.250.207.142  UDP        74 49743 → 33461 Len=32
  130 81.340867285 192.168.42.160  142.250.207.142  UDP        74 51641 → 33462 Len=32
  131 81.341007071 192.168.42.160  142.250.207.142  UDP        74 60998 → 33463 Len=32
  132 81.341205529 192.168.42.160  142.250.207.142  UDP        74 43001 → 33464 Len=32
  133 81.341365644 192.168.42.160  142.250.207.142  UDP        74 54957 → 33465 Len=32
  134 81.398285192 192.168.42.160  142.250.207.142  UDP        74 60667 → 33466 Len=32
  135 81.398847753 192.168.42.160  142.250.207.142  UDP        74 58276 → 33467 Len=32
  136 81.399156591 192.168.42.160  142.250.207.142  UDP        74 33081 → 33468 Len=32
```

Live network traffic capture initiated in Wireshark

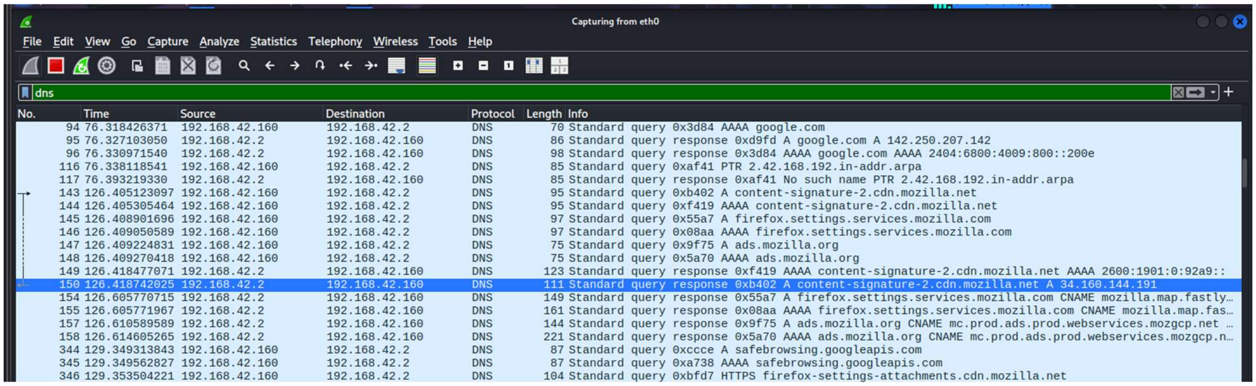
4.2 Protocol Analysis

Captured traffic was filtered to analyze different protocols such as TCP, UDP, and ICMP.

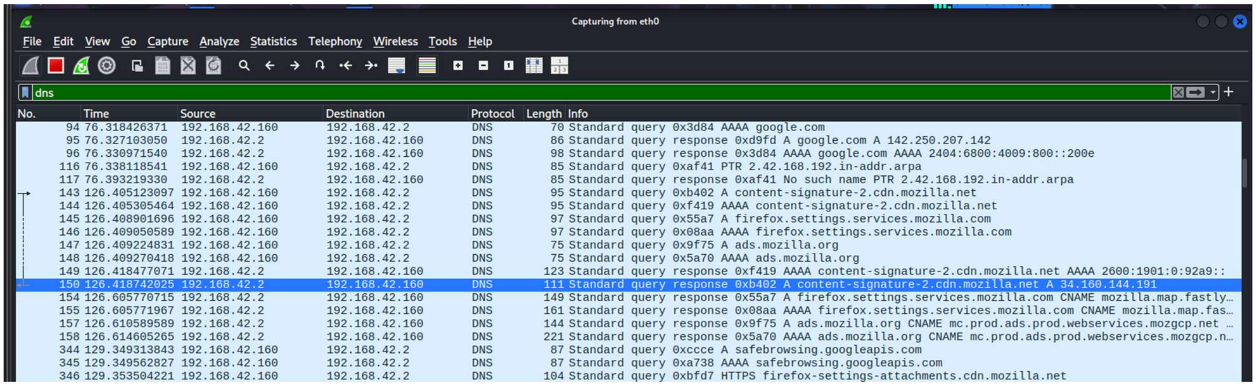

Capturing from eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
tcp

```
No.   Time          Source          Destination      Protocol Length Info
  151 126.423521317 192.168.42.160  34.160.144.191   TCP        74 46950 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1041895905 TSecr=0 WS=512
  152 126.424008845 192.168.42.160  34.160.144.191   TCP        74 46956 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1041895905 TSecr=0 WS=512
  153 126.425808509 192.168.42.160  34.160.144.191   TCP        74 46960 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=1041895907 TSecr=0 WS=512
  156 126.607717323 192.168.42.160  199.232.21.91    TCP        74 57346 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=2199453338 TSecr=0 WS=512
  159 126.616970234 192.168.42.160  34.36.137.203    TCP        74 46402 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=4256748663 TSecr=0 WS=512
  160 126.619461561 34.160.144.191  192.168.42.160   TCP        60 443 → 46960 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
  161 126.619462814 34.160.144.191  192.168.42.160   TCP        60 443 → 46956 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
  162 126.619463295 34.160.144.191  192.168.42.160   TCP        60 443 → 46950 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
  163 126.619559778 192.168.42.160  34.160.144.191   TCP        54 46960 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
  164 126.619695917 192.168.42.160  34.160.144.191   TCP        54 46956 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
  165 126.619784336 192.168.42.160  34.160.144.191   TCP        54 46950 → 443 [ACK] Seq=1 Ack=1 Win=64240 Len=0
  166 126.622475696 192.168.42.160  34.160.144.191   TLSv1.2   274 Client Hello (SNI=content-signature-2.cdn.mozilla.net)
  167 126.623737481 34.160.144.191  192.168.42.160   TCP        60 443 → 46960 [ACK] Seq=1 Ack=221 Win=64240 Len=0
  168 126.623776593 192.168.42.160  34.160.144.191   TLSv1.2   274 Client Hello (SNI=content-signature-2.cdn.mozilla.net)
  169 126.624227745 192.168.42.160  34.160.144.191   TLSv1.2   274 Client Hello (SNI=content-signature-2.cdn.mozilla.net)
  170 126.624562532 34.160.144.191  192.168.42.160   TCP        60 443 → 46956 [ACK] Seq=1 Ack=221 Win=64240 Len=0
  171 126.625943792 34.160.144.191  192.168.42.160   TCP        60 443 → 46950 [ACK] Seq=1 Ack=221 Win=64240 Len=0
  172 126.878156797 192.168.42.160  199.232.21.91    TCP        74 57360 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=2199453609 TSecr=0 WS=512
  173 126.879885097 192.168.42.160  34.36.137.203    TCP        74 46404 → 443 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM TSval=4256748926 TSecr=0 WS=512
  174 126.914967732 199.232.21.91   192.168.42.160   TCP        60 443 → 57346 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
  175 126.914968794 34.36.137.203   192.168.42.160   TCP        60 443 → 46402 [SYN, ACK] Seq=0 Ack=1 Win=64240 Len=0 MSS=1460
```

TCP Capture



```
No.    Time          Source          Destination     Protocol Length Info
    94 76.318426371  192.168.42.160  192.168.42.2    DNS       70 Standard query 0x3d84 AAAA google.com
    95 76.327103050  192.168.42.2    192.168.42.160  DNS       86 Standard query response 0xd9fd A google.com A 142.250.207.142
    96 76.330971540  192.168.42.2    192.168.42.160  DNS       98 Standard query response 0x3d84 AAAA google.com AAAA 2404:6800:4009:800::200e
   116 76.338118541  192.168.42.160  192.168.42.2    DNS       85 Standard query 0xaf41 PTR 2.42.168.192.in-addr.arpa
   117 76.393219330  192.168.42.2    192.168.42.160  DNS       85 Standard query response 0xaf41 No such name PTR 2.42.168.192.in-addr.arpa
   143 126.405123097 192.168.42.160  192.168.42.2    DNS       95 Standard query 0xb402 A content-signature-2.cdn.mozilla.net
   144 126.405305464 192.168.42.160  192.168.42.2    DNS       95 Standard query 0xf419 AAAA content-signature-2.cdn.mozilla.net
   145 126.408901696 192.168.42.160  192.168.42.2    DNS       97 Standard query 0x55a7 A firefox.settings.services.mozilla.com
   146 126.409050589 192.168.42.160  192.168.42.2    DNS       97 Standard query 0x08aa AAAA firefox.settings.services.mozilla.com
   147 126.409224831 192.168.42.160  192.168.42.2    DNS       75 Standard query 0x9f75 A ads.mozilla.org
   148 126.409270418 192.168.42.160  192.168.42.2    DNS       75 Standard query 0x5a70 AAAA ads.mozilla.org
   149 126.418477071 192.168.42.2    192.168.42.160  DNS      123 Standard query response 0xf419 AAAA content-signature-2.cdn.mozilla.net AAAA 2600:1901:0:92a9::
   150 126.418742025 192.168.42.2    192.168.42.160  DNS      111 Standard query response 0xb402 A content-signature-2.cdn.mozilla.net A 34.160.144.191
   154 126.605770715 192.168.42.2    192.168.42.160  DNS      149 Standard query response 0x55a7 A firefox.settings.services.mozilla.com CNAME mozilla.map.fastly…
   155 126.605771967 192.168.42.2    192.168.42.160  DNS      161 Standard query response 0x08aa AAAA firefox.settings.services.mozilla.com CNAME mozilla.map.fas…
   157 126.610589589 192.168.42.2    192.168.42.160  DNS      144 Standard query response 0x9f75 A ads.mozilla.org CNAME mc.prod.ads.prod.webservices.mozgcp.net …
   158 126.614605265 192.168.42.2    192.168.42.160  DNS      221 Standard query response 0x5a70 AAAA ads.mozilla.org CNAME mc.prod.ads.prod.webservices.mozgcp.n…
   344 129.349313843 192.168.42.160  192.168.42.2    DNS       87 Standard query 0xccce A safebrowsing.googleapis.com
   345 129.349562827 192.168.42.160  192.168.42.2    DNS       87 Standard query 0xa738 AAAA safebrowsing.googleapis.com
   346 129.353504221 192.168.42.160  192.168.42.2    DNS      104 Standard query 0xbfd7 HTTPS firefox-settings-attachments.cdn.mozilla.net
```



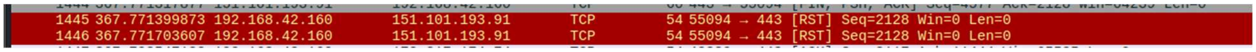
Capturing from eth0
File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help
dns

```
No.    Time          Source          Destination     Protocol Length Info
    94 76.318426371  192.168.42.160  192.168.42.2    DNS       70 Standard query 0x3d84 AAAA google.com
    95 76.327103050  192.168.42.2    192.168.42.160  DNS       86 Standard query response 0xd9fd A google.com A 142.250.207.142
    96 76.330971540  192.168.42.2    192.168.42.160  DNS       98 Standard query response 0x3d84 AAAA google.com AAAA 2404:6800:4009:800::200e
   116 76.338118541  192.168.42.160  192.168.42.2    DNS       85 Standard query 0xaf41 PTR 2.42.168.192.in-addr.arpa
   117 76.393219330  192.168.42.2    192.168.42.160  DNS       85 Standard query response 0xaf41 No such name PTR 2.42.168.192.in-addr.arpa
   143 126.405123097 192.168.42.160  192.168.42.2    DNS       95 Standard query 0xb402 A content-signature-2.cdn.mozilla.net
   144 126.405305464 192.168.42.160  192.168.42.2    DNS       95 Standard query 0xf419 AAAA content-signature-2.cdn.mozilla.net
   145 126.408901696 192.168.42.160  192.168.42.2    DNS       97 Standard query 0x55a7 A firefox.settings.services.mozilla.com
   146 126.409050589 192.168.42.160  192.168.42.2    DNS       97 Standard query 0x08aa AAAA firefox.settings.services.mozilla.com
   147 126.409224831 192.168.42.160  192.168.42.2    DNS       75 Standard query 0x9f75 A ads.mozilla.org
   148 126.409270418 192.168.42.160  192.168.42.2    DNS       75 Standard query 0x5a70 AAAA ads.mozilla.org
   149 126.418477071 192.168.42.2    192.168.42.160  DNS      123 Standard query response 0xf419 AAAA content-signature-2.cdn.mozilla.net AAAA 2600:1901:0:92a9::
   150 126.418742025 192.168.42.2    192.168.42.160  DNS      111 Standard query response 0xb402 A content-signature-2.cdn.mozilla.net A 34.160.144.191
   154 126.605770715 192.168.42.2    192.168.42.160  DNS      149 Standard query response 0x55a7 A firefox.settings.services.mozilla.com CNAME mozilla.map.fastly…
   155 126.605771967 192.168.42.2    192.168.42.160  DNS      161 Standard query response 0x08aa AAAA firefox.settings.services.mozilla.com CNAME mozilla.map.fas…
   157 126.610589589 192.168.42.2    192.168.42.160  DNS      144 Standard query response 0x9f75 A ads.mozilla.org CNAME mc.prod.ads.prod.webservices.mozgcp.net …
   158 126.614605265 192.168.42.2    192.168.42.160  DNS      221 Standard query response 0x5a70 AAAA ads.mozilla.org CNAME mc.prod.ads.prod.webservices.mozgcp.n…
   344 129.349313843 192.168.42.160  192.168.42.2    DNS       87 Standard query 0xccce A safebrowsing.googleapis.com
   345 129.349562827 192.168.42.160  192.168.42.2    DNS       87 Standard query 0xa738 AAAA safebrowsing.googleapis.com
   346 129.353504221 192.168.42.160  192.168.42.2    DNS      104 Standard query 0xbfd7 HTTPS firefox-settings-attachments.cdn.mozilla.net
```

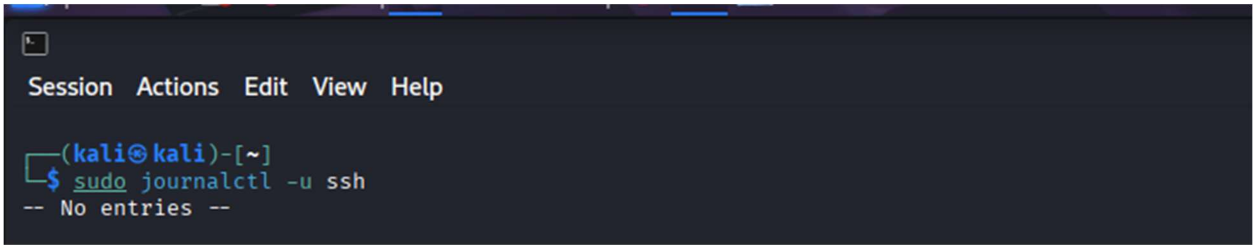
DNS

4.3 Top Talkers & Conversations

The statistical analysis capabilities that are inherent within Wireshark software were employed in analyzing the captured data in search of devices that were generating the most communications. Through such information as packets and protocol analysis as well as connection statistics, there was an ability to determine those computers in the network that were generating the most data in terms of communications. The software analysis not only enabled identification but also enabled visible interpretation regarding patterns and possible points within networks that could be conclusive in detecting errors in communications.



```
  1445 367.771399873 192.168.42.160      151.101.193.91      TCP        54 55094 → 443 [RST] Seq=2128 Win=0 Len=0
  1446 367.771703607 192.168.42.160      151.101.193.91      TCP        54 55094 → 443 [RST] Seq=2128 Win=0 Len=0
```



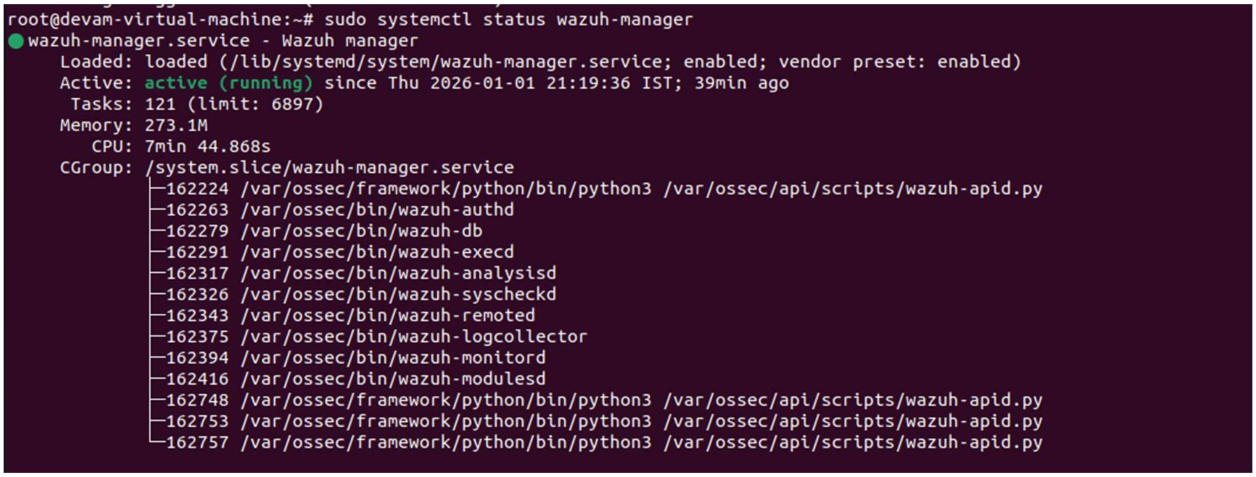
```
┌──(kali㉿kali)-[~]
└─$ sudo journalctl -u ssh
-- No entries --
```

Kali checking SSH

5. SIEM Implementation Using Wazuh

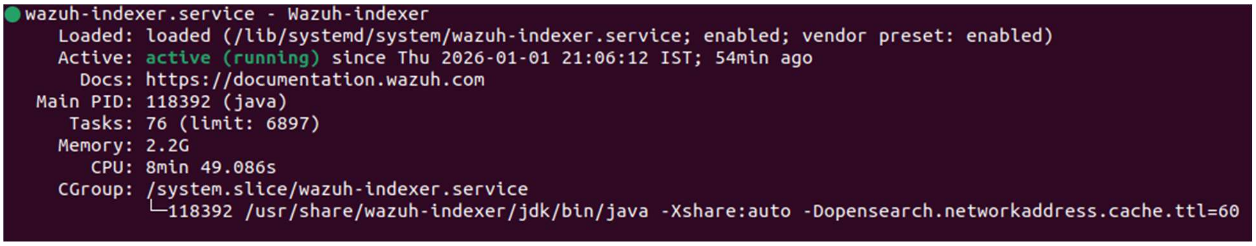
5.1 Wazuh Agent & Manager Setup

The Wazuh agent was deployed on an intended monitored system and installed, ensuring the capture of endpoint activity and its forwarding for centralized analysis. After installation, the agent successfully registered itself with the Wazuh Manager and established a secure channel with it for the collection of logs, forwarding of events, and correlation based on rules. This established channel allowed the continuous sending of security data by the monitored system to the Wazuh platform for real-time visibility into the data, the detection of anomalies, and the generation of alerts according to predefined policies. It had been successfully integrated and corroborated that the SIEM infrastructure was working as it should, with the agent playing an active role in comprehensive security monitoring and incident detection.



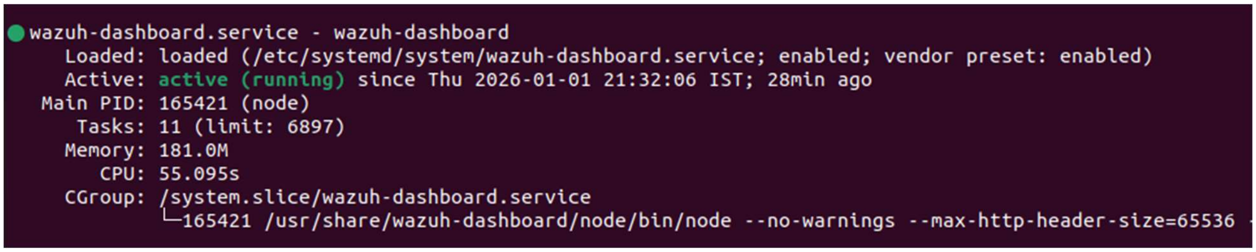
```
root@devam-virtual-machine:~# sudo systemctl status wazuh-manager
● wazuh-manager.service - Wazuh manager
     Loaded: loaded (/lib/systemd/system/wazuh-manager.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2026-01-01 21:19:36 IST; 39min ago
      Tasks: 121 (limit: 6897)
     Memory: 273.1M
        CPU: 7min 44.868s
     CGroup: /system.slice/wazuh-manager.service
             ├─162224 /var/ossec/framework/python/bin/python3 /var/ossec/api/scripts/wazuh-apid.py
             ├─162263 /var/ossec/bin/wazuh-authd
             ├─162279 /var/ossec/bin/wazuh-db
             ├─162291 /var/ossec/bin/wazuh-execd
             ├─162317 /var/ossec/bin/wazuh-analysisd
             ├─162326 /var/ossec/bin/wazuh-syscheckd
             ├─162343 /var/ossec/bin/wazuh-remoted
             ├─162375 /var/ossec/bin/wazuh-logcollector
             ├─162394 /var/ossec/bin/wazuh-monitord
             ├─162416 /var/ossec/bin/wazuh-modulesd
             ├─162748 /var/ossec/framework/python/bin/python3 /var/ossec/api/scripts/wazuh-apid.py
             ├─162753 /var/ossec/framework/python/bin/python3 /var/ossec/api/scripts/wazuh-apid.py
             └─162757 /var/ossec/framework/python/bin/python3 /var/ossec/api/scripts/wazuh-apid.py
```

Wazuh manager is Active



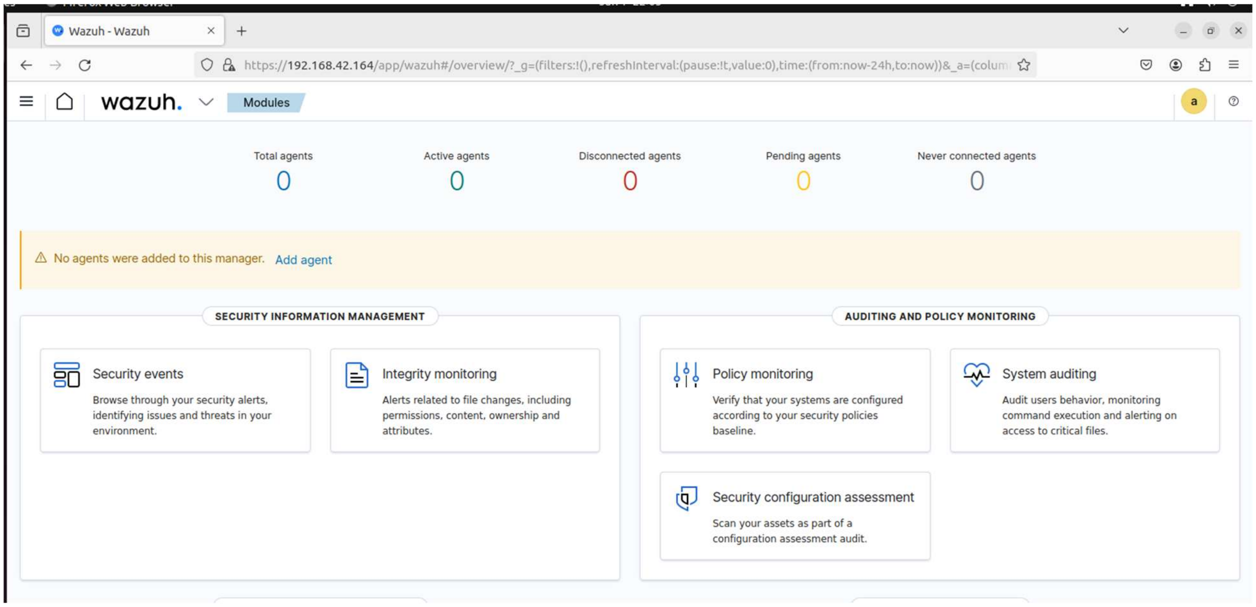
```
● wazuh-indexer.service - Wazuh-indexer
     Loaded: loaded (/lib/systemd/system/wazuh-indexer.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2026-01-01 21:06:12 IST; 54min ago
       Docs: https://documentation.wazuh.com
   Main PID: 118392 (java)
      Tasks: 76 (limit: 6897)
     Memory: 2.2G
        CPU: 8min 49.086s
     CGroup: /system.slice/wazuh-indexer.service
             └─118392 /usr/share/wazuh-indexer/jdk/bin/java -Xshare:auto -Dopensearch.networkaddress.cache.ttl=60
```

Wazuh Indexer is Active



```
● wazuh-dashboard.service - wazuh-dashboard
     Loaded: loaded (/etc/systemd/system/wazuh-dashboard.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2026-01-01 21:32:06 IST; 28min ago
   Main PID: 165421 (node)
      Tasks: 11 (limit: 6897)
     Memory: 181.0M
        CPU: 55.095s
     CGroup: /system.slice/wazuh-dashboard.service
             └─165421 /usr/share/wazuh-dashboard/node/bin/node --no-warnings --max-http-header-size=65536
```

Wazuh Dashboard is Active


Wazuh - Wazuh
https://192.168.42.164/app/wazuh#/overview/?_g=(filters:!(),refreshInterval:(pause:!t,value:0),time:(from:now-24h,to:now))&_a=(colum
wazuh.
Modules
a

Total agents 0

Active agents 0

Disconnected agents 0

Pending agents 0

Never connected agents 0

⚠ No agents were added to this manager. Add agent

SECURITY INFORMATION MANAGEMENT

Security events
Browse through your security alerts, identifying issues and threats in your environment.

Integrity monitoring
Alerts related to file changes, including permissions, content, ownership and attributes.

AUDITING AND POLICY MONITORING

Policy monitoring
Verify that your systems are configured according to your security policies baseline.

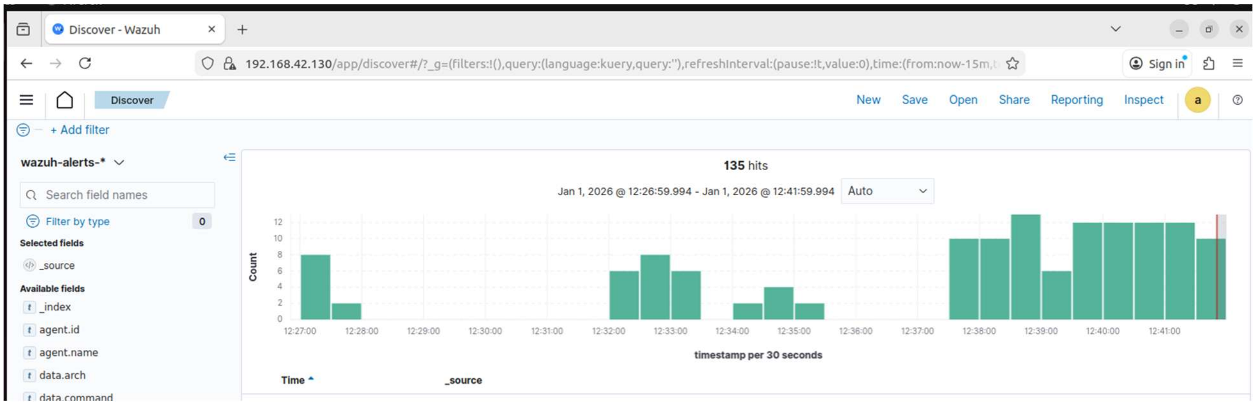
System auditing
Audit users behavior, monitoring command execution and alerting on access to critical files.

Security configuration assessment
Scan your assets as part of a configuration assessment audit.

Wazuh agent is running active

5.2 Log Collection Verification

System logs, including authentication events, were successfully ingested into the SIEM platform.



192.168.42.130/app/discover#/?_g=(filters:!(),query:(language:kuery,query:''),refreshInterval:(pause:!t,value:0),time:(from:now-15m,t

135 hits

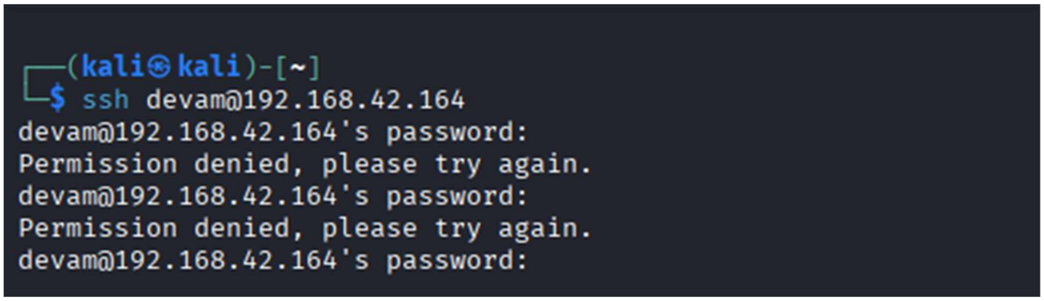
Jan 1, 2026 @ 12:26:59.994 - Jan 1, 2026 @ 12:41:59.994

Total number of hits after various attempts

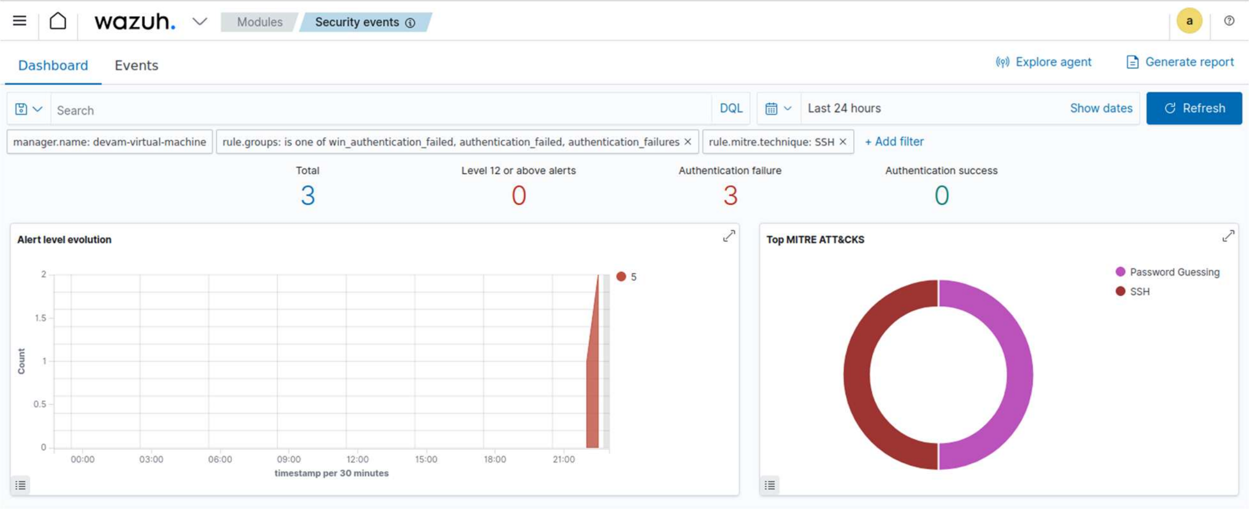
6. Security Incident Simulation (SSH Failed Login)

6.1 Attack Simulation

Multiple events of brute-force SSH login attempts were started on the target system to generate the brute-force attack. This controlled event was designed to mimic how a malicious attacker would repeatedly attempt to gain unauthorized access by systematically trying different username and password combinations. These events generated actual data for the monitoring tools and security platforms to detect suspicious login patterns, trigger alerts, and demonstrate the importance of intrusion detection mechanisms in safeguarding networked systems.



```
┌──(kali㉿kali)-[~]
└─$ ssh devam@192.168.42.164
devam@192.168.42.164's password:
Permission denied, please try again.
devam@192.168.42.164's password:
Permission denied, please try again.
devam@192.168.42.164's password:
```



Last 24 hours

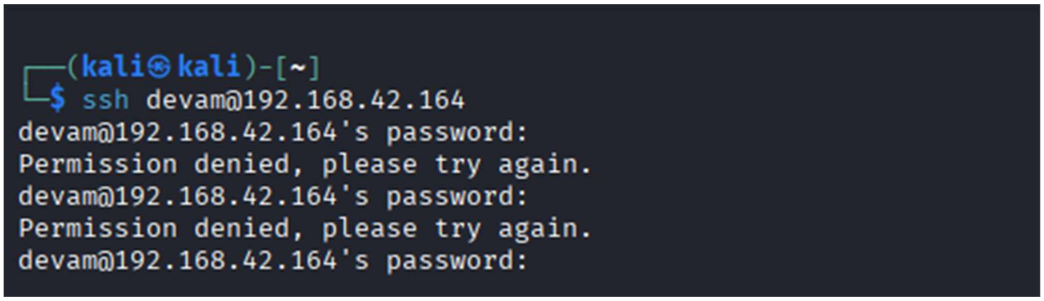
manager.name: devam-virtual-machine | rule.groups: is one of win_authentication_failed, authentication_failed, authentication_failures | rule.mitre.technique: SSH

Total	Level 12 or above alerts	Authentication failure	Authentication success
3	0	3	0

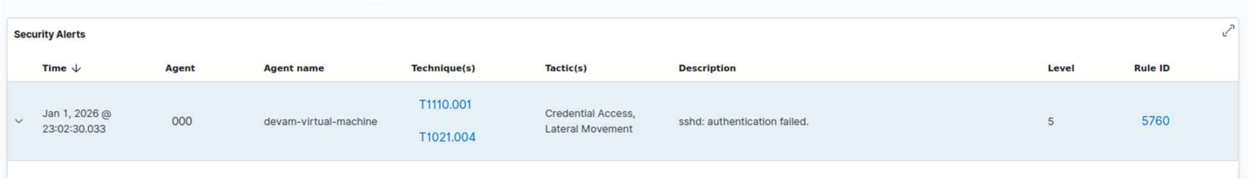
Failed SSH login attempts generated from attacker system

6.2 Alert Detection in SIEM

Wazuh detected the repeated failed login attempts and generated security alerts.

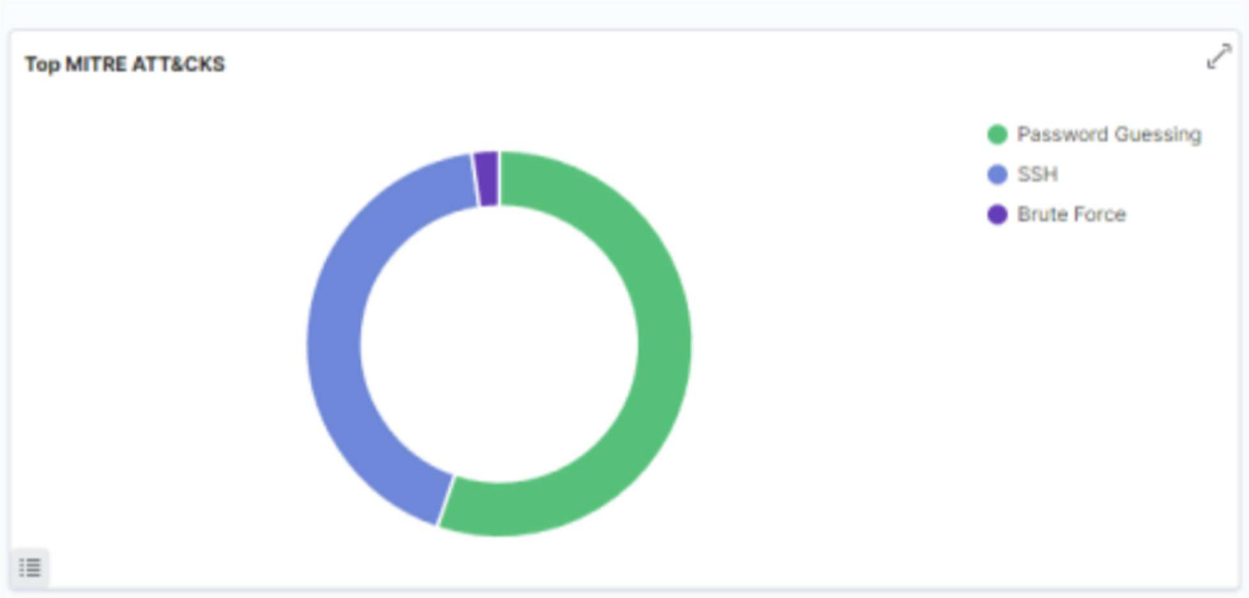


```
┌──(kali㉿kali)-[~]
└─$ ssh devam@192.168.42.164
devam@192.168.42.164's password:
Permission denied, please try again.
devam@192.168.42.164's password:
Permission denied, please try again.
devam@192.168.42.164's password:
```



Security Alerts

Time ↓	Agent	Agent name	Technique(s)	Tactic(s)	Description	Level	Rule ID
Jan 1, 2026 @ 23:02:30.033	000	devam-virtual-machine	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: authentication failed.	5	5760



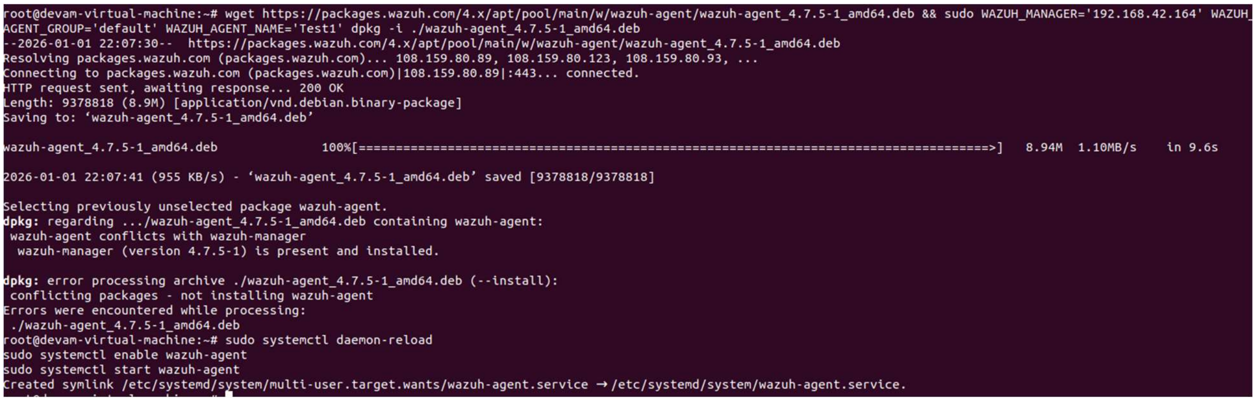
Burte Force and SSH attempts

7. Custom Alert Rule Configuration

7.1 Rule Creation

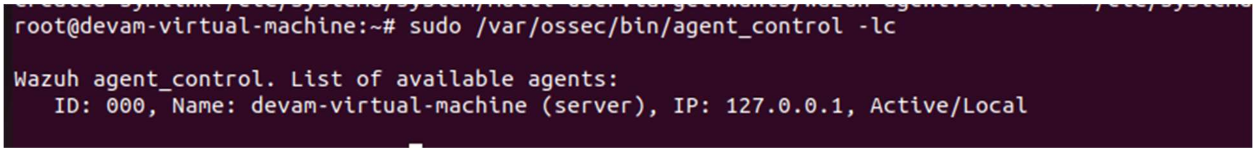
A custom detection rule has been deployed where the system automatically triggers an alert whenever there are several failed SSH authentications detected within a short time frame.

It enables the detection of a possible brute-force or illegal access attempt based on repeated failed login attempts, which can then be escalated to a security alert.



```
root@devam-virtual-machine:~# wget https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.7.5-1_amd64.deb && sudo WAZUH_MANAGER='192.168.42.164' WAZUH_
AGENT_GROUP='default' WAZUH_AGENT_NAME='Test1' dpkg -i ./wazuh-agent_4.7.5-1_amd64.deb
--2026-01-01 22:07:30--  https://packages.wazuh.com/4.x/apt/pool/main/w/wazuh-agent/wazuh-agent_4.7.5-1_amd64.deb
Resolving packages.wazuh.com (packages.wazuh.com)... 108.159.80.89, 108.159.80.123, 108.159.80.93, ...
Connecting to packages.wazuh.com (packages.wazuh.com)|108.159.80.89|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 9378818 (8.9M) [application/vnd.debian.binary-package]
Saving to: ‘wazuh-agent_4.7.5-1_amd64.deb’

wazuh-agent_4.7.5-1_amd64.deb             100%[==================================================================================>]   8.94M  1.10MB/s    in 9.6s

2026-01-01 22:07:41 (955 KB/s) - ‘wazuh-agent_4.7.5-1_amd64.deb’ saved [9378818/9378818]

Selecting previously unselected package wazuh-agent.
dpkg: regarding .../wazuh-agent_4.7.5-1_amd64.deb containing wazuh-agent:
 wazuh-agent conflicts with wazuh-manager
  wazuh-manager (version 4.7.5-1) is present and installed.

dpkg: error processing archive ./wazuh-agent_4.7.5-1_amd64.deb (--install):
 conflicting packages - not installing wazuh-agent
Errors were encountered while processing:
 ./wazuh-agent_4.7.5-1_amd64.deb
root@devam-virtual-machine:~# sudo systemctl daemon-reload
sudo systemctl enable wazuh-agent
sudo systemctl start wazuh-agent
Created symlink /etc/systemd/system/multi-user.target.wants/wazuh-agent.service → /etc/systemd/system/wazuh-agent.service.
```

Created Custom rule for wazuh and added with help of CLI

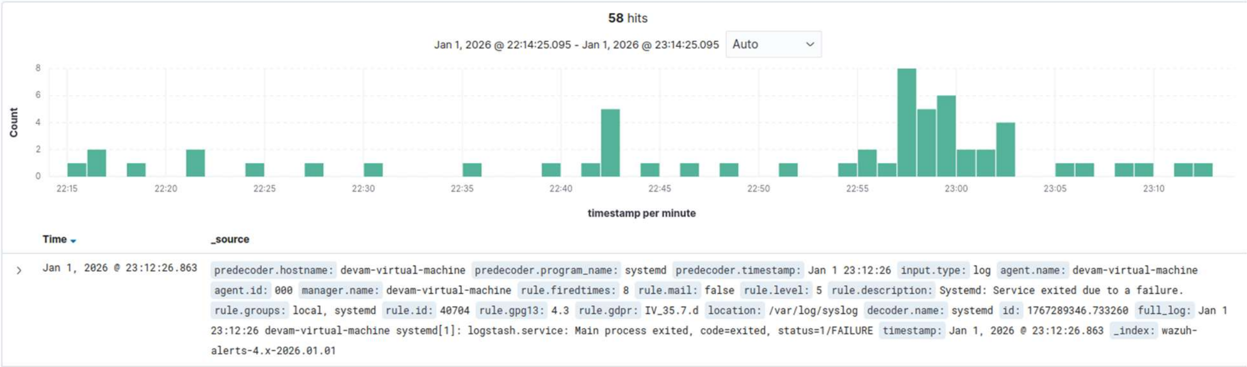


```
root@devam-virtual-machine:~# sudo /var/ossec/bin/agent_control -lc

Wazuh agent_control. List of available agents:
   ID: 000, Name: devam-virtual-machine (server), IP: 127.0.0.1, Active/Local
```

Agent has been added

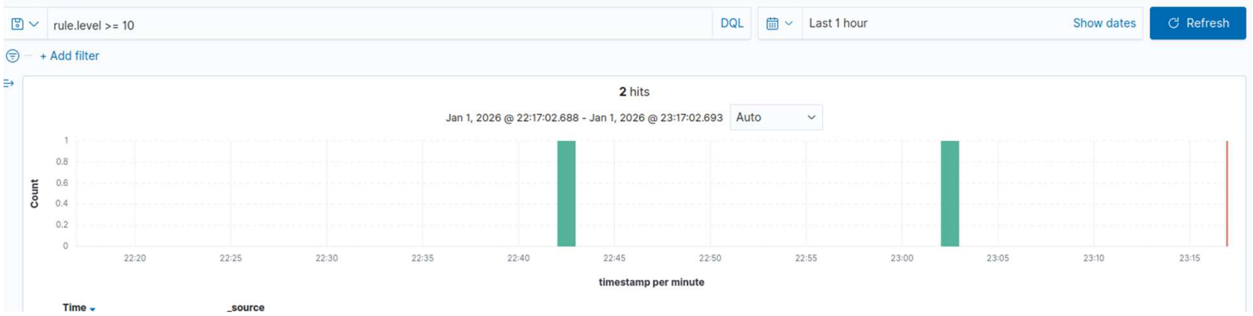
7.2 Rule Validation

The rule successfully triggered alerts during testing.



```
Jan 1, 2026 @ 23:12:26.863   predecoder.hostname: devam-virtual-machine predecoder.program_name: systemd predecoder.timestamp: Jan 1 23:12:26 input.type: log agent.name: devam-virtual-machine
                             agent.id: 000 manager.name: devam-virtual-machine rule.firedtimes: 8 rule.mail: false rule.level: 5 rule.description: Systemd: Service exited due to a failure.
                             rule.groups: local, systemd rule.id: 40704 rule.gpg13: 4.3 rule.gdpr: IV_35.7.d location: /var/log/syslog decoder.name: systemd id: 1767289346.733260 full_log: Jan 1
                             23:12:26 devam-virtual-machine systemd[1]: logstash.service: Main process exited, code=exited, status=1/FAILURE timestamp: Jan 1, 2026 @ 23:12:26.863 _index: wazuh-
                             alerts-4.x-2026.01.01
```

Total number of hits during complete process



rule.level >= 10

HIGH LEVEL HITS (SSH)

8. Challenges and How They Were Overcome

8.1 Encrypted Traffic Analysis

Challenge: The encrypted traffic could not be analyzed at a payload-level inspection.

Solution: There was a shift in focus of the analysis towards the metadata, which included IP, ports, duration of session, and number of packets, that was obtained through the use of Wireshark or Python scripts.

8.2 Security Event Monitoring with Wazuh

Problem: Identifying related abnormal activities across multiple systems in real time.

Solution: The Wazuh agent was successfully installed on the monitored computers and established a connection with the Wazuh Manager. With this configuration, it was possible to have a centralized collection of logs, event correlation, and alerting regarding possible anomalies, like repeated login failures. The use of Wazuh as a SIEM tool provided better insight into possible dangers.

9. Conclusion

This exercise provided in-depth learning in many key areas of networking security. It required practical design work in subnets to optimize the allocation of IP addresses, thorough analysis of the traffic in the networks to identify trends and abnormalities, and the use of SIEM tools to provide real-time security scanning in case of possible dangers to the networks. The use of a combination of industry-leading tools like Packet Tracer to develop networks for simulations in the networks, Wireshark to capture the information contained in individual packets in networks for thorough analysis, and Wazuh to offer log analysis to provide in-depth security scanning in the networks brought out the need for a multi-layered security approach to be in place at all times in a given organization or enterprise. It was clear that the solution to securing networks in the modern enterprise depends upon a mix of technical expertise and foresight.