8/10/2025

Final Project

CST8808 – Cyber Incident Report

Submitted By :
Adeep Mani
Kanhay Thakore
Mitanshi Solanki
Rajat Mani

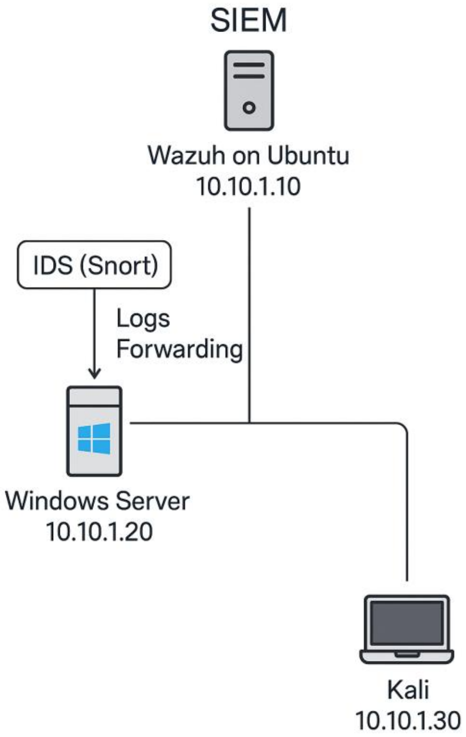
Table Of Contents

Incident Response Plan Implementation

1. Introduction

This report documents the implementation of an Incident Response (IR) plan for CSA271.com, focusing on log monitoring, threat detection, and attack simulation using a SIEM (Wazuh), Snort IDS, and Volatility for memory analysis. The project involved setting up a virtual lab environment with Kali Linux (attacker), Windows Server (IIS/FTP + Snort), and Ubuntu (Wazuh SIEM).

2. Network Topology & Configuration

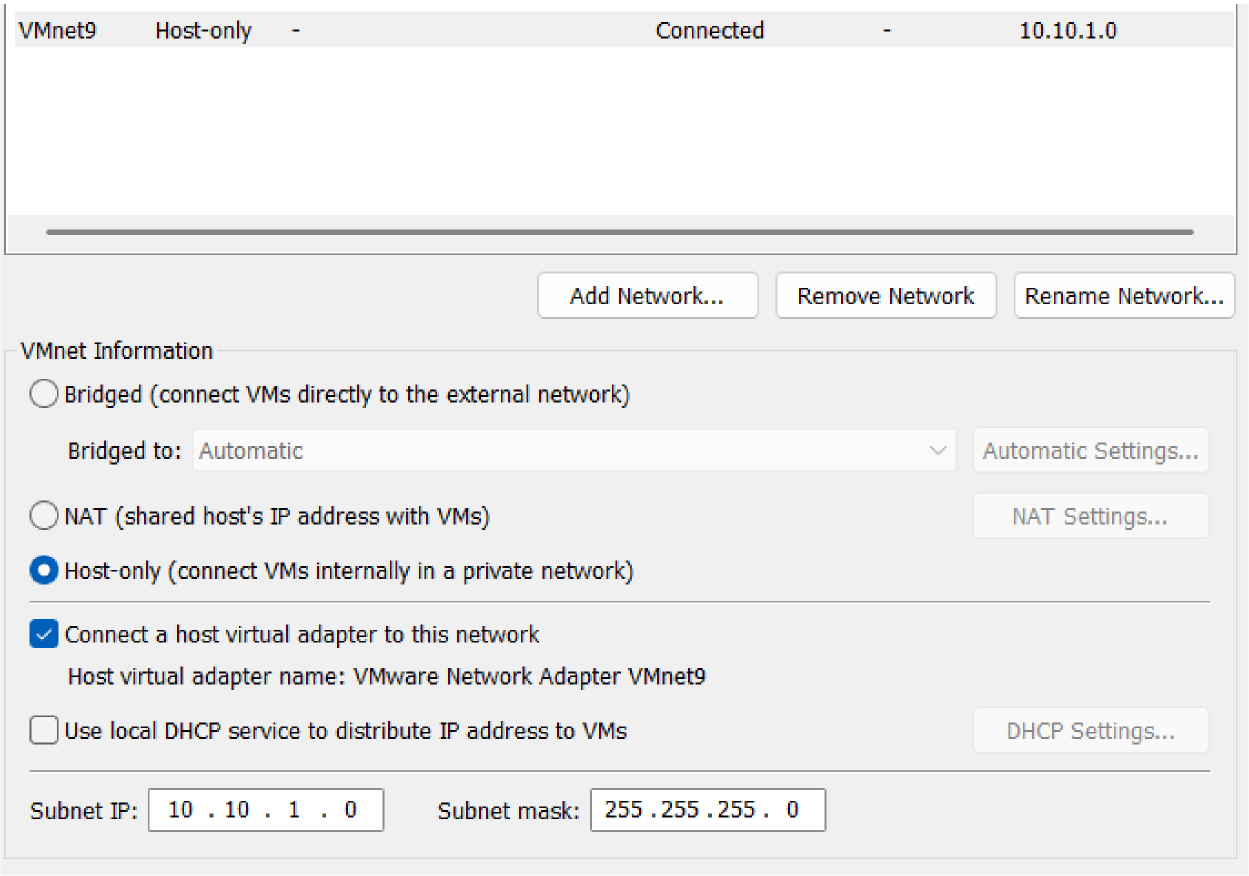


3. VMs Configuration:

IP Configuration

Machine	IP Address	Role
Kali Linux	10.10.1.30	Attack simulation
Windows Server	10.10.1.20	IIS/FTP + Snort IDS
Wazuh (Ubuntu)	10.10.1.10	SIEM (Log collection & alerts)

Configuring Vmware : Setting Subnet IP and Subnet Mask on Vmnet9



VMnet9   Host-only   -   Connected   -   10.10.1.0

VMnet Information

○ Bridged (connect VMs directly to the external network)

Bridged to: Automatic

○ NAT (shared host's IP address with VMs)

● Host-only (connect VMs internally in a private network)

☑ Connect a host virtual adapter to this network

Host virtual adapter name: VMware Network Adapter VMnet9

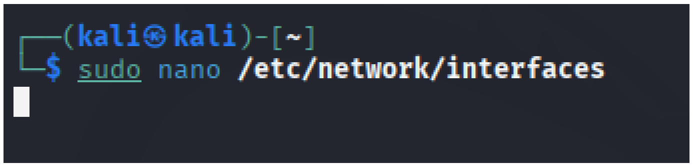
□ Use local DHCP service to distribute IP address to VMs

Subnet IP: 10 . 10 . 1 . 0   Subnet mask: 255 .255 .255 . 0

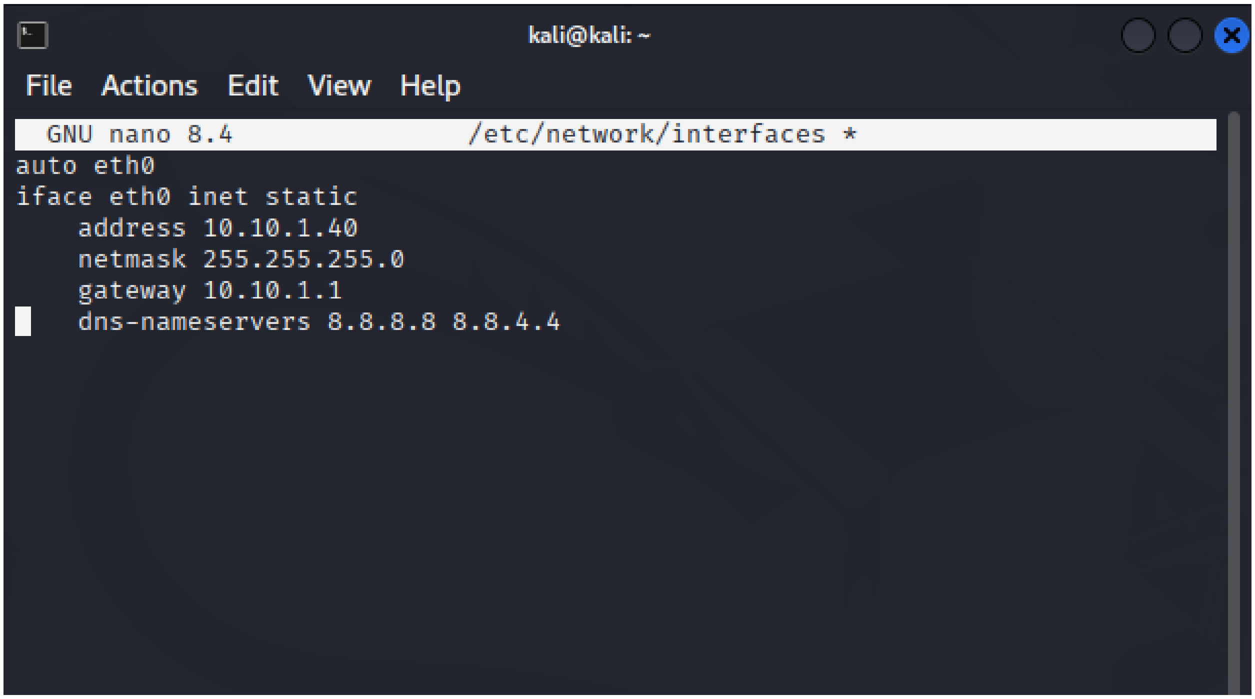
Key Configurations

- **Static IPs** configured on all machines

Kali Linux:



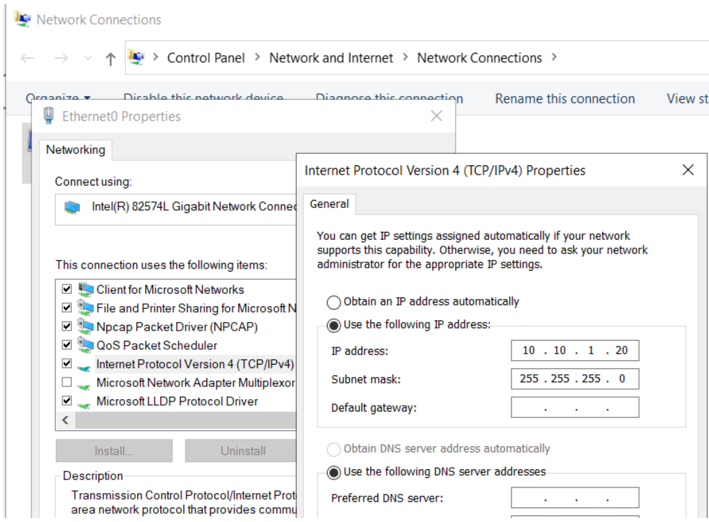
```
┌──(kali㉿kali)-[~]
└─$ sudo nano /etc/network/interfaces
```



```
  GNU nano 8.4               /etc/network/interfaces *
auto eth0
iface eth0 inet static
    address 10.10.1.40
    netmask 255.255.255.0
    gateway 10.10.1.1
    dns-nameservers 8.8.8.8 8.8.4.4
```

Dns-nameservers can be skipped because we are on host only network. It was only set in case we need to switch to NAT for downloading any tools.

Windows Server (IIS):


Network Connections
Control Panel > Network and Internet > Network Connections
Organize
Disable this network device
Diagnose this connection
Rename this connection
View status of this connection
Change settings of this connection
Ethernet0 Properties
Networking
Install...
Uninstall
Internet Protocol Version 4 (TCP/IPv4) Properties
General

Connect using:

Intel(R) 82574L Gigabit Network Connec

This connection uses the following items:

☑ Client for Microsoft Networks
☑ File and Printer Sharing for Microsoft N
☑ Npcap Packet Driver (NPCAP)
☑ QoS Packet Scheduler
☑ Internet Protocol Version 4 (TCP/IPv4)
□ Microsoft Network Adapter Multiplexor
☑ Microsoft LLDP Protocol Driver

Description

Transmission Control Protocol/Internet Prot
area network protocol that provides commu

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

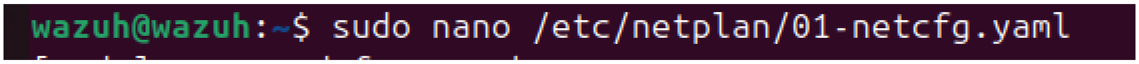
○ Obtain an IP address automatically
● Use the following IP address:

IP address: 10 . 10 . 1 . 20
Subnet mask: 255 . 255 . 255 . 0
Default gateway: . . .

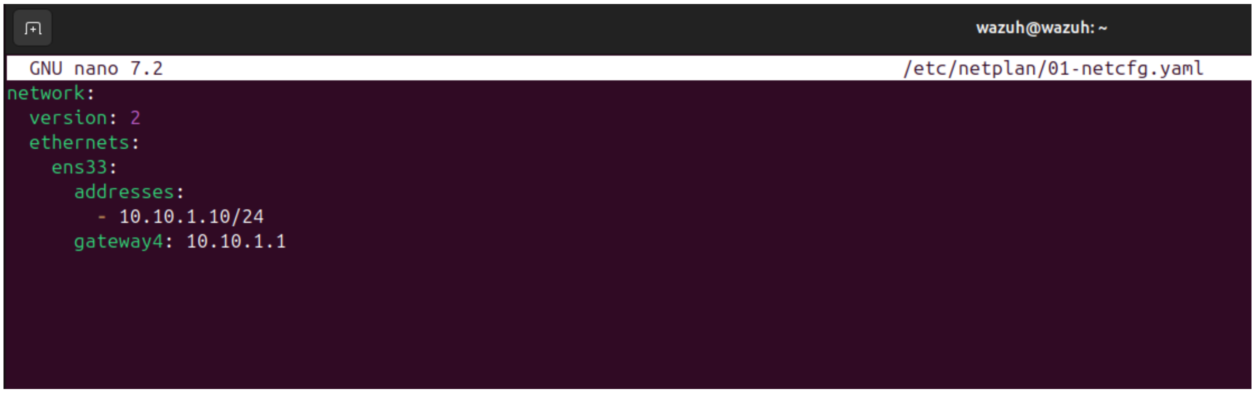
○ Obtain DNS server address automatically
● Use the following DNS server addresses

Preferred DNS server: . . .

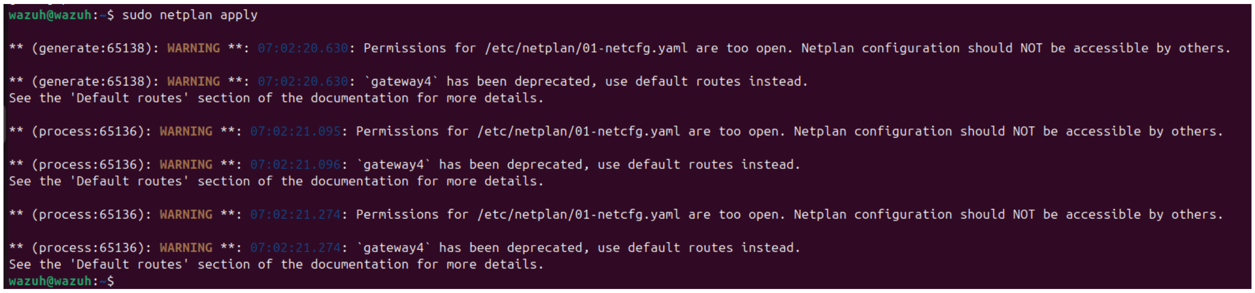
Ubuntu(Wazuh/SIEM):



```
wazuh@wazuh:~$ sudo nano /etc/netplan/01-netcfg.yaml
```

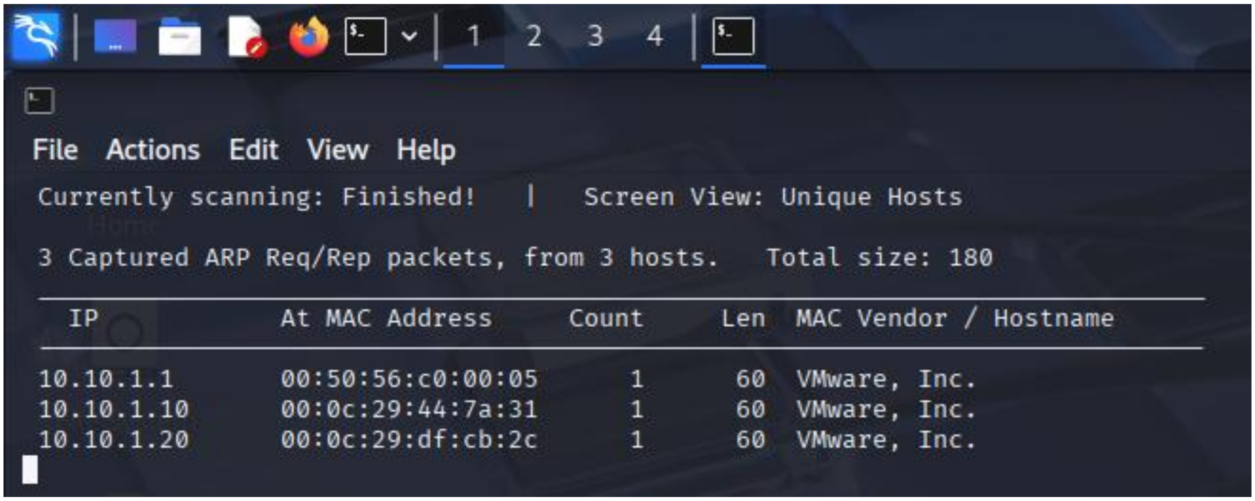


```
  GNU nano 7.2                                        /etc/netplan/01-netcfg.yaml
network:
  version: 2
  ethernets:
    ens33:
      addresses:
        - 10.10.1.10/24
      gateway4: 10.10.1.1
```



```
wazuh@wazuh:~$ sudo netplan apply

** (generate:65138): WARNING **: 07:02:20.630: Permissions for /etc/netplan/01-netcfg.yaml are too open. Netplan configuration should NOT be accessible by others.

** (generate:65138): WARNING **: 07:02:20.630: `gateway4` has been deprecated, use default routes instead.
See the 'Default routes' section of the documentation for more details.

** (process:65136): WARNING **: 07:02:21.095: Permissions for /etc/netplan/01-netcfg.yaml are too open. Netplan configuration should NOT be accessible by others.

** (process:65136): WARNING **: 07:02:21.096: `gateway4` has been deprecated, use default routes instead.
See the 'Default routes' section of the documentation for more details.

** (process:65136): WARNING **: 07:02:21.274: Permissions for /etc/netplan/01-netcfg.yaml are too open. Netplan configuration should NOT be accessible by others.

** (process:65136): WARNING **: 07:02:21.274: `gateway4` has been deprecated, use default routes instead.
See the 'Default routes' section of the documentation for more details.
wazuh@wazuh:~$
```

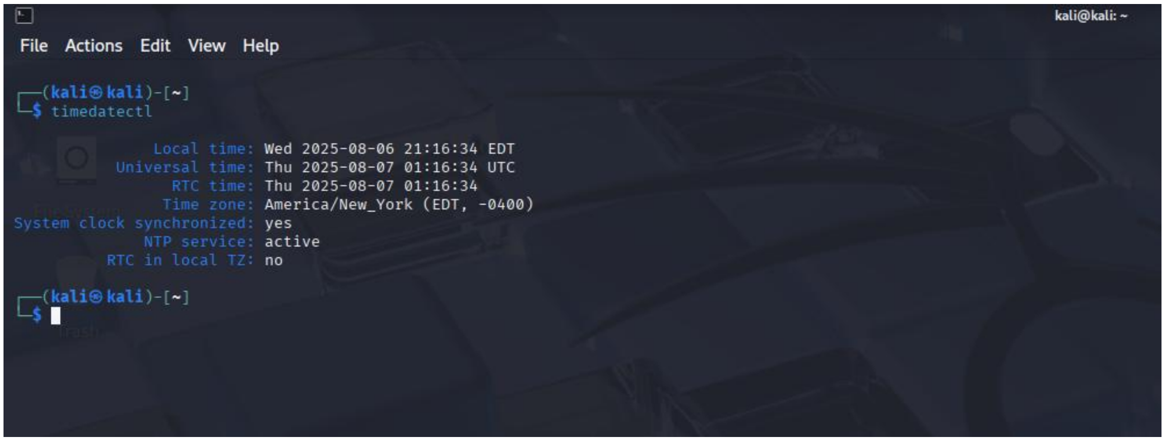
Confirming if machines are on same network:



```
Currently scanning: Finished!   |   Screen View: Unique Hosts

3 Captured ARP Req/Rep packets, from 3 hosts.   Total size: 180
_________________________________________________________________
  IP            At MAC Address     Count     Len  MAC Vendor / Hostname
_________________________________________________________________
10.10.1.1       00:50:56:c0:00:05      1      60  VMware, Inc.
10.10.1.10      00:0c:29:44:7a:31      1      60  VMware, Inc.
10.10.1.20      00:0c:29:df:cb:2c      1      60  VMware, Inc.
```

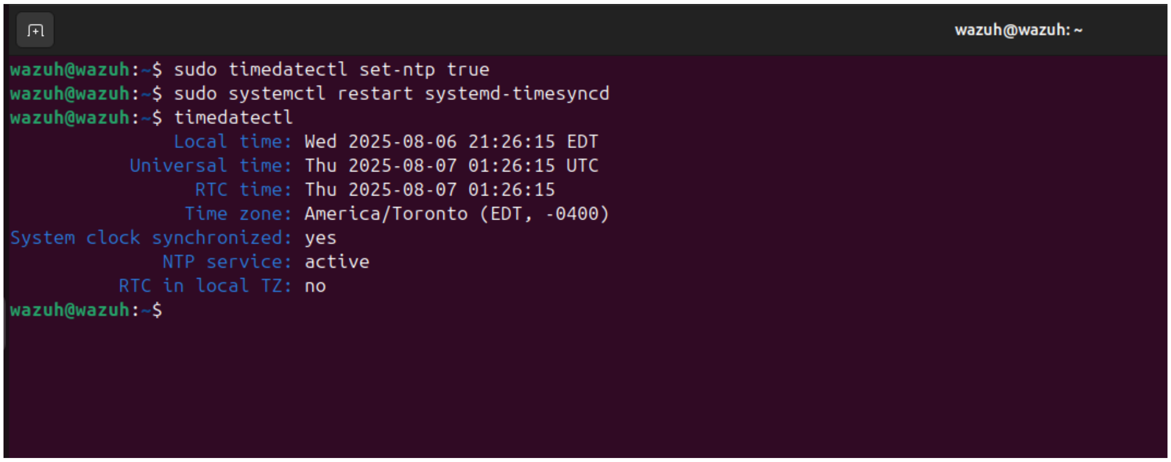
4. NTP Synchronization on all machines:

Kali:



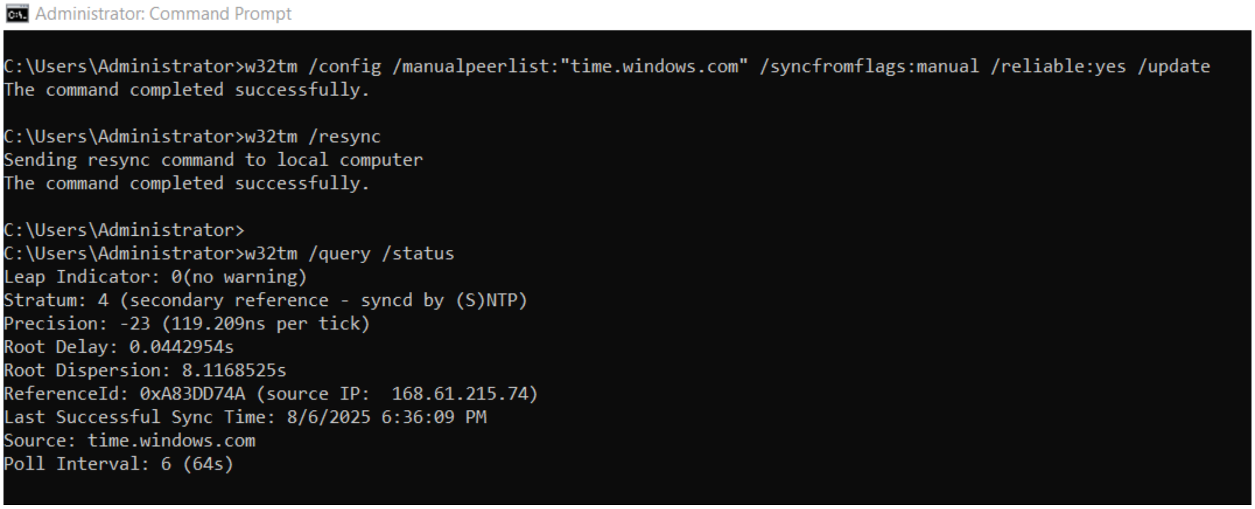
```
┌──(kali㉿kali)-[~]
└─$ timedatectl
               Local time: Wed 2025-08-06 21:16:34 EDT
           Universal time: Thu 2025-08-07 01:16:34 UTC
                 RTC time: Thu 2025-08-07 01:16:34
                Time zone: America/New_York (EDT, -0400)
System clock synchronized: yes
              NTP service: active
          RTC in local TZ: no

┌──(kali㉿kali)-[~]
└─$
```

Ubuntu:



```
wazuh@wazuh:~$ sudo timedatectl set-ntp true
wazuh@wazuh:~$ sudo systemctl restart systemd-timesyncd
wazuh@wazuh:~$ timedatectl
               Local time: Wed 2025-08-06 21:26:15 EDT
           Universal time: Thu 2025-08-07 01:26:15 UTC
                 RTC time: Thu 2025-08-07 01:26:15
                Time zone: America/Toronto (EDT, -0400)
System clock synchronized: yes
              NTP service: active
          RTC in local TZ: no
wazuh@wazuh:~$
```

Windows:



Administrator: Command Prompt

```
C:\Users\Administrator>w32tm /config /manualpeerlist:"time.windows.com" /syncfromflags:manual /reliable:yes /update
The command completed successfully.

C:\Users\Administrator>w32tm /resync
Sending resync command to local computer
The command completed successfully.

C:\Users\Administrator>
C:\Users\Administrator>w32tm /query /status
Leap Indicator: 0(no warning)
Stratum: 4 (secondary reference - syncd by (S)NTP)
Precision: -23 (119.209ns per tick)
Root Delay: 0.0442954s
Root Dispersion: 8.1168525s
ReferenceId: 0xA83DD74A (source IP:  168.61.215.74)
Last Successful Sync Time: 8/6/2025 6:36:09 PM
Source: time.windows.com
Poll Interval: 6 (64s)
```

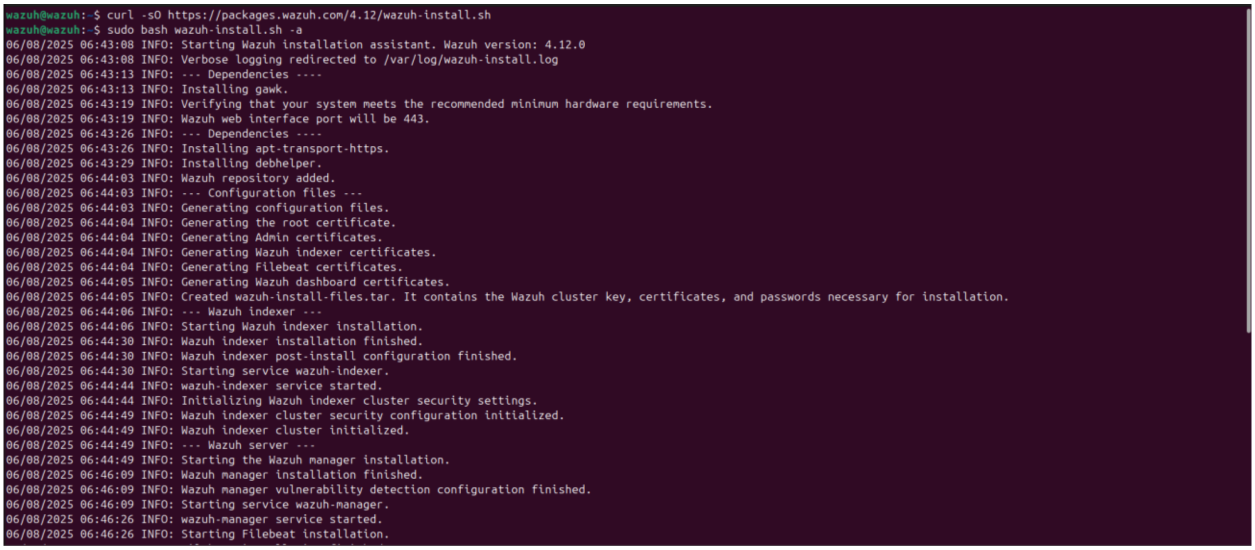
5. SIEM Setup :

SIEM Selection Rationale:

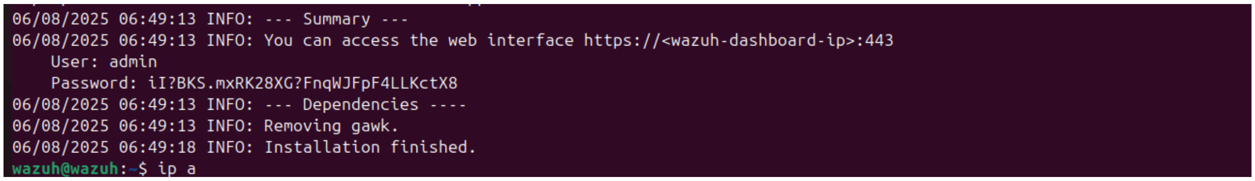
For this project, we selected Wazuh as the SIEM platform instead of alternatives like Splunk, OSSIM, or the ELK Stack. The decision was based on the following factors:

1. **Cost Efficiency** – Wazuh is open-source and free to deploy, avoiding licensing costs associated with Splunk Enterprise or commercial OSSIM implementations, which is ideal for test lab environment.
2. **Feature Set** – Wazuh integrates SIEM, log analysis, and File Integrity Monitoring (FIM) in a single platform. This allowed us to meet both the SIEM and FIM requirements without installing multiple separate tools.
3. **Integration with IDS** – Wazuh can easily ingest alerts from Snort IDS, enabling centralized monitoring of both host-based and network-based events.

Wazuh Installation:



```
wazuh@wazuh:~$ curl -sO https://packages.wazuh.com/4.12/wazuh-install.sh
wazuh@wazuh:~$ sudo bash wazuh-install.sh -a
06/08/2025 06:43:08 INFO: Starting Wazuh installation assistant. Wazuh version: 4.12.0
06/08/2025 06:43:08 INFO: Verbose logging redirected to /var/log/wazuh-install.log
06/08/2025 06:43:13 INFO: --- Dependencies ----
06/08/2025 06:43:13 INFO: Installing gawk.
06/08/2025 06:43:19 INFO: Verifying that your system meets the recommended minimum hardware requirements.
06/08/2025 06:43:19 INFO: Wazuh web interface port will be 443.
06/08/2025 06:43:26 INFO: --- Dependencies ----
06/08/2025 06:43:26 INFO: Installing apt-transport-https.
06/08/2025 06:43:29 INFO: Installing debhelper.
06/08/2025 06:44:03 INFO: Wazuh repository added.
06/08/2025 06:44:03 INFO: --- Configuration files ---
06/08/2025 06:44:03 INFO: Generating configuration files.
06/08/2025 06:44:04 INFO: Generating the root certificate.
06/08/2025 06:44:04 INFO: Generating Admin certificates.
06/08/2025 06:44:04 INFO: Generating Wazuh indexer certificates.
06/08/2025 06:44:04 INFO: Generating Filebeat certificates.
06/08/2025 06:44:05 INFO: Generating Wazuh dashboard certificates.
06/08/2025 06:44:05 INFO: Created wazuh-install-files.tar. It contains the Wazuh cluster key, certificates, and passwords necessary for installation.
06/08/2025 06:44:06 INFO: --- Wazuh indexer ---
06/08/2025 06:44:06 INFO: Starting Wazuh indexer installation.
06/08/2025 06:44:30 INFO: Wazuh indexer installation finished.
06/08/2025 06:44:30 INFO: Wazuh indexer post-install configuration finished.
06/08/2025 06:44:30 INFO: Starting service wazuh-indexer.
06/08/2025 06:44:44 INFO: wazuh-indexer service started.
06/08/2025 06:44:44 INFO: Initializing Wazuh indexer cluster security settings.
06/08/2025 06:44:49 INFO: Wazuh indexer cluster security configuration initialized.
06/08/2025 06:44:49 INFO: Wazuh indexer cluster initialized.
06/08/2025 06:44:49 INFO: --- Wazuh server ---
06/08/2025 06:44:49 INFO: Starting the Wazuh manager installation.
06/08/2025 06:46:09 INFO: Wazuh manager installation finished.
06/08/2025 06:46:09 INFO: Wazuh manager vulnerability detection configuration finished.
06/08/2025 06:46:09 INFO: Starting service wazuh-manager.
06/08/2025 06:46:26 INFO: wazuh-manager service started.
06/08/2025 06:46:26 INFO: Starting Filebeat installation.
```



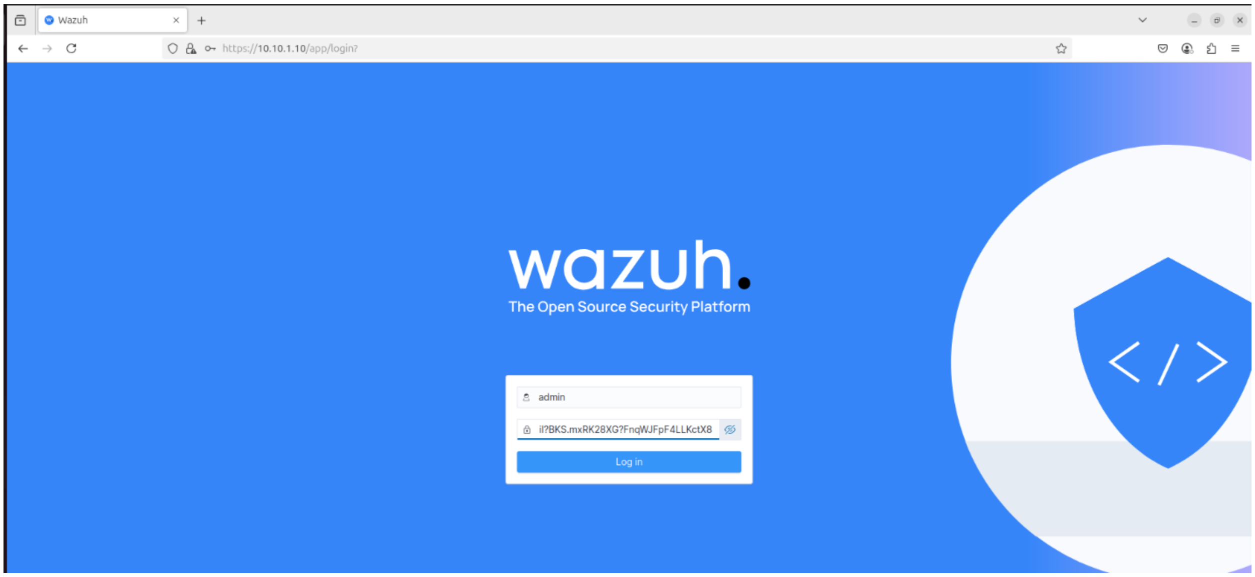
```
06/08/2025 06:49:13 INFO: --- Summary ---
06/08/2025 06:49:13 INFO: You can access the web interface https://<wazuh-dashboard-ip>:443
    User: admin
    Password: iI?BKS.mxRK28XG?FnqWJFpF4LLKctX8
06/08/2025 06:49:13 INFO: --- Dependencies ----
06/08/2025 06:49:13 INFO: Removing gawk.
06/08/2025 06:49:18 INFO: Installation finished.
wazuh@wazuh:~$ ip a
```

User : admin

Password : iI?BKS.mxRK28XG?FnqWJFpF4LLKctX8

Next, we will open the Wazuh Dashboard on our local ip 10.10.1.10 with the above credentials.

Dashboard :



Wazuh

https://10.10.1.10/app/login?

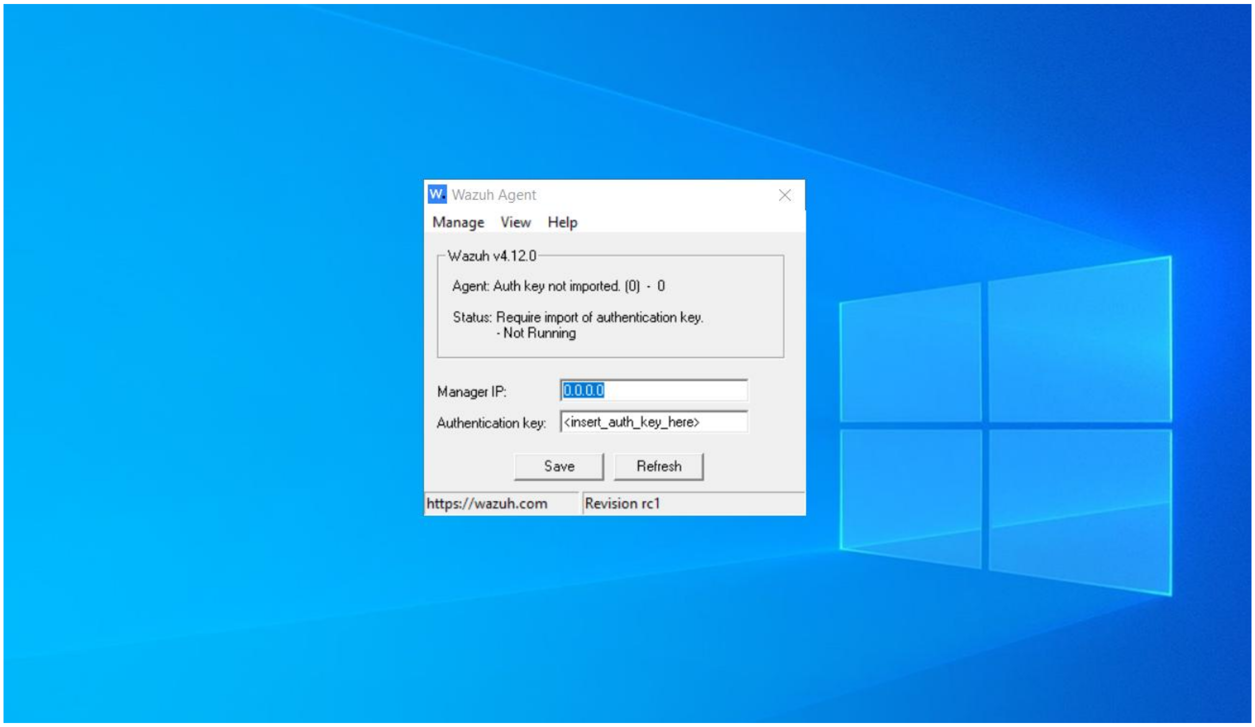
wazuh.

The Open Source Security Platform

admin

il?BKS.mxRK28XG?FnqWJFpF4LLKctX8

Wazuh agent on windows server :



Wazuh Agent

Wazuh v4.12.0

Agent: Auth key not imported. (0) - 0

Status: Require import of authentication key.
- Not Running

Manager IP: 0.0.0.0

Authentication key: <insert_auth_key_here>

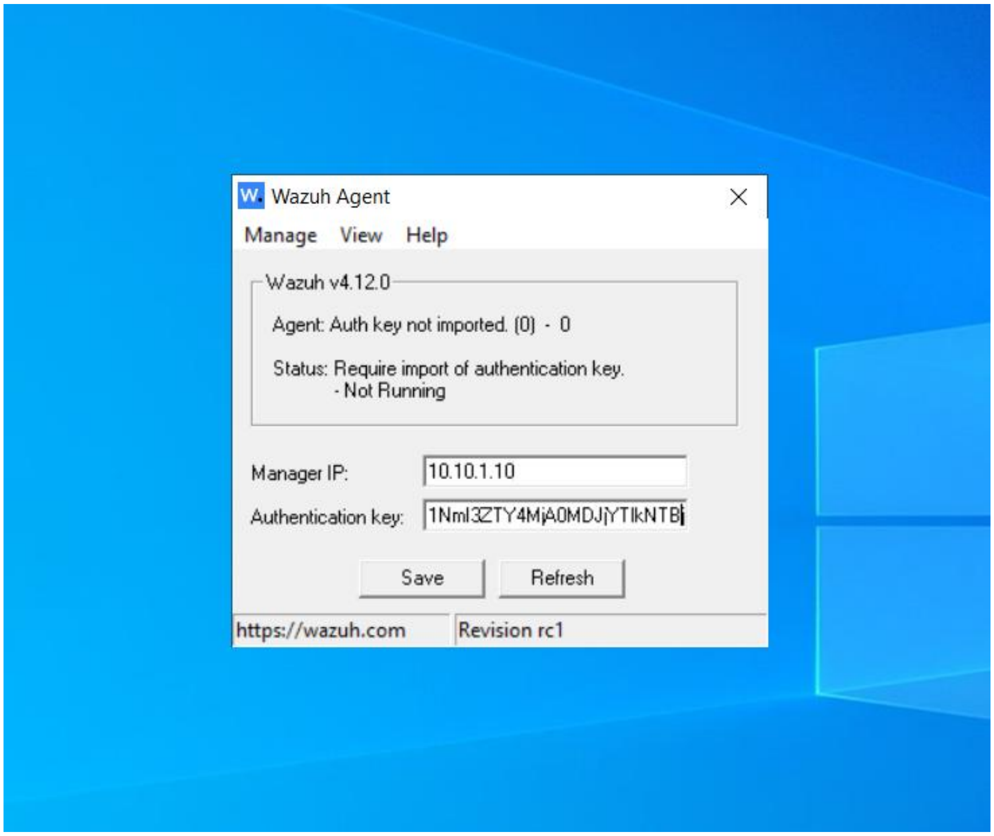
https://wazuh.com

Revision rc1

Wazuh agent will require manager ip (10.10.1.10) and authentication key which we will generate next.

Generating the key for agent:

Wazuh Agent Final Setup:



Wazuh v4.12.0

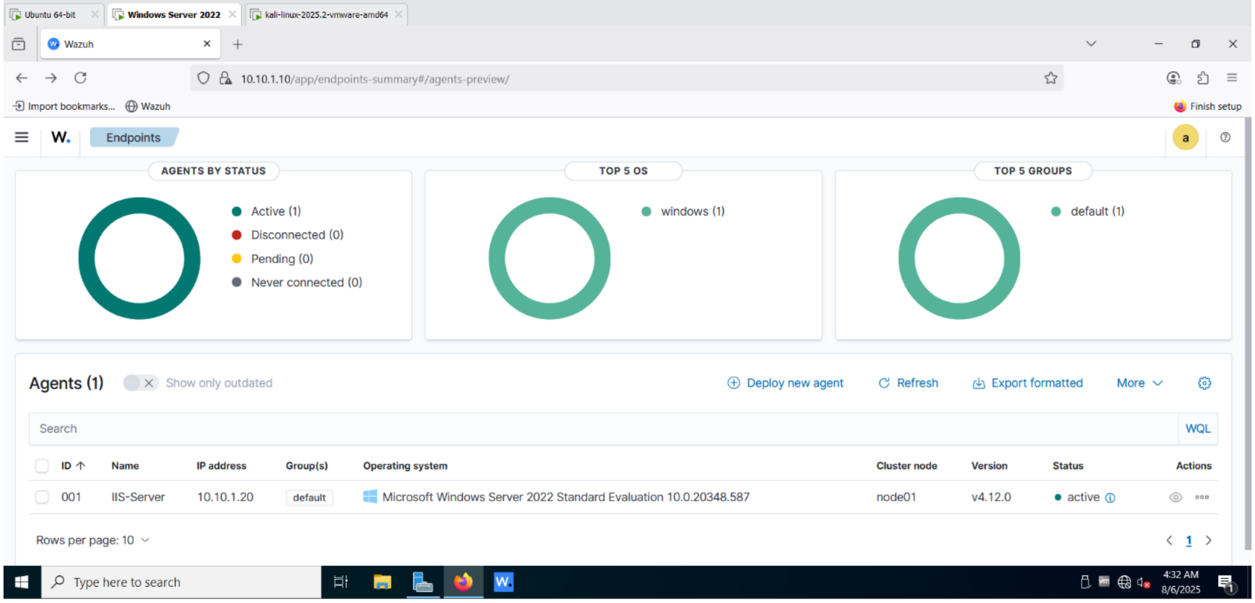
Agent: Auth key not imported. (0) - 0

Status: Require import of authentication key.
- Not Running

Manager IP: 10.10.1.10

Authentication key: 1NmI3ZTY4MjA0MDJjYTlkNTBi

Confirmed agent connection in Wazuh dashboard :



AGENTS BY STATUS

- Active (1)
- Disconnected (0)
- Pending (0)
- Never connected (0)

TOP 5 OS

- windows (1)

TOP 5 GROUPS

- default (1)

Agents (1)

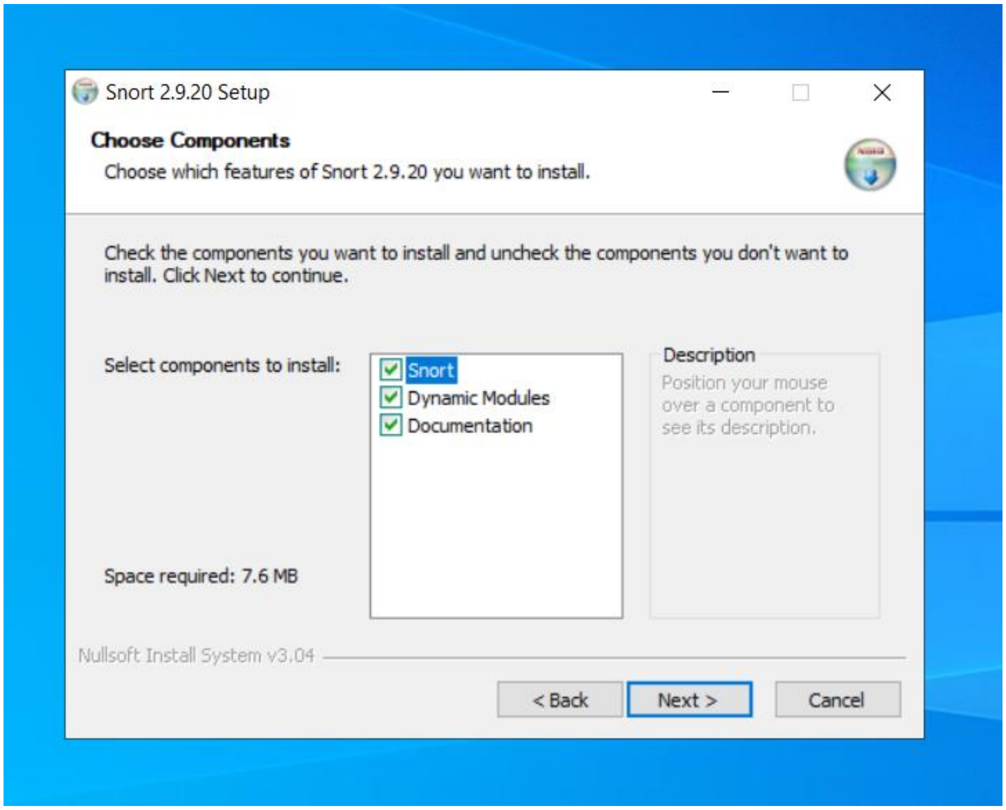
ID ↑	Name	IP address	Group(s)	Operating system	Cluster node	Version	Status	Actions
001	IIS-Server	10.10.1.20	default	Microsoft Windows Server 2022 Standard Evaluation 10.0.20348.587	node01	v4.12.0	● active	

Rows per page: 10

Snort IDS Integration:

Next, we needed an IDS system to forward network-based alerts from the Windows Server to the Wazuh SIEM. In a real-world deployment, IDS is typically hosted on a dedicated appliance or server to avoid resource contention and ensure optimal performance. However, due to the limited resources available in our lab environment for running multiple VMs, we integrated the IDS directly into the same Windows Server hosting IIS. We selected Snort as our IDS because it is open source, widely used, and offers straightforward integration with Wazuh. Snort's extensive rule set and community support allowed us to quickly detect simulated network attacks and forward the corresponding alerts to the SIEM for correlation and visualization.

Snort installation :



Snort 2.9.20 Setup

Choose Components

Choose which features of Snort 2.9.20 you want to install.

Check the components you want to install and uncheck the components you don't want to install. Click Next to continue.

Select components to install:

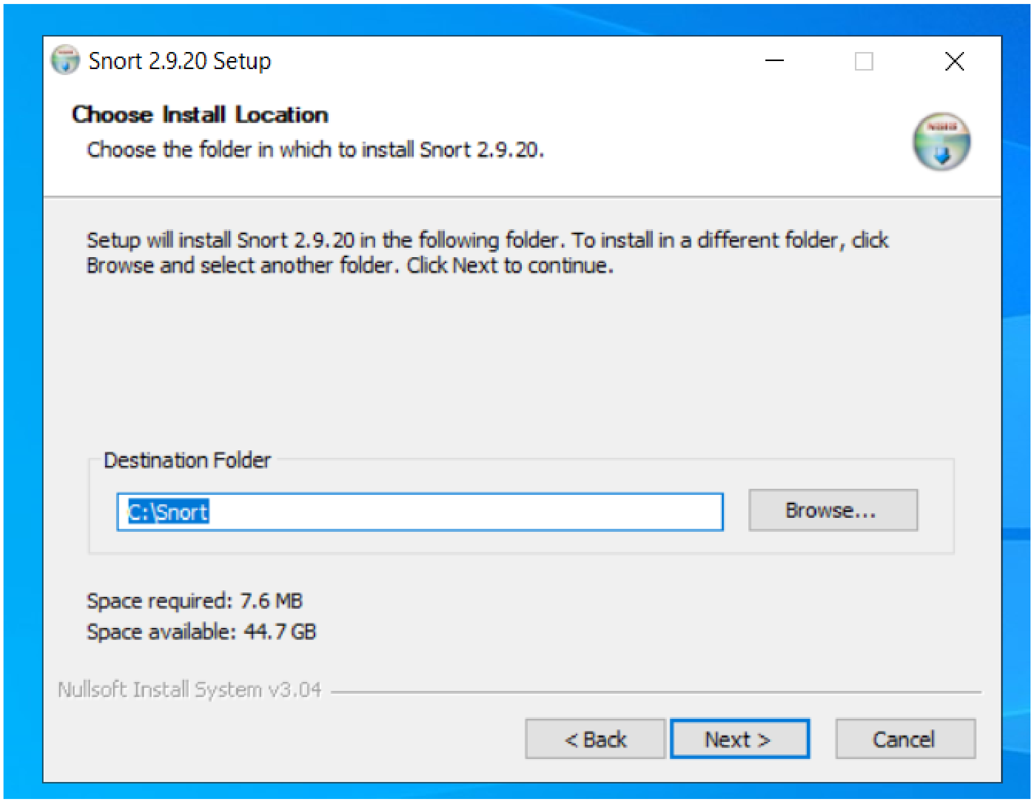
☑ Snort
☑ Dynamic Modules
☑ Documentation

Description

Position your mouse over a component to see its description.

Space required: 7.6 MB

Nullsoft Install System v3.04



Snort 2.9.20 Setup

Choose Install Location

Choose the folder in which to install Snort 2.9.20.

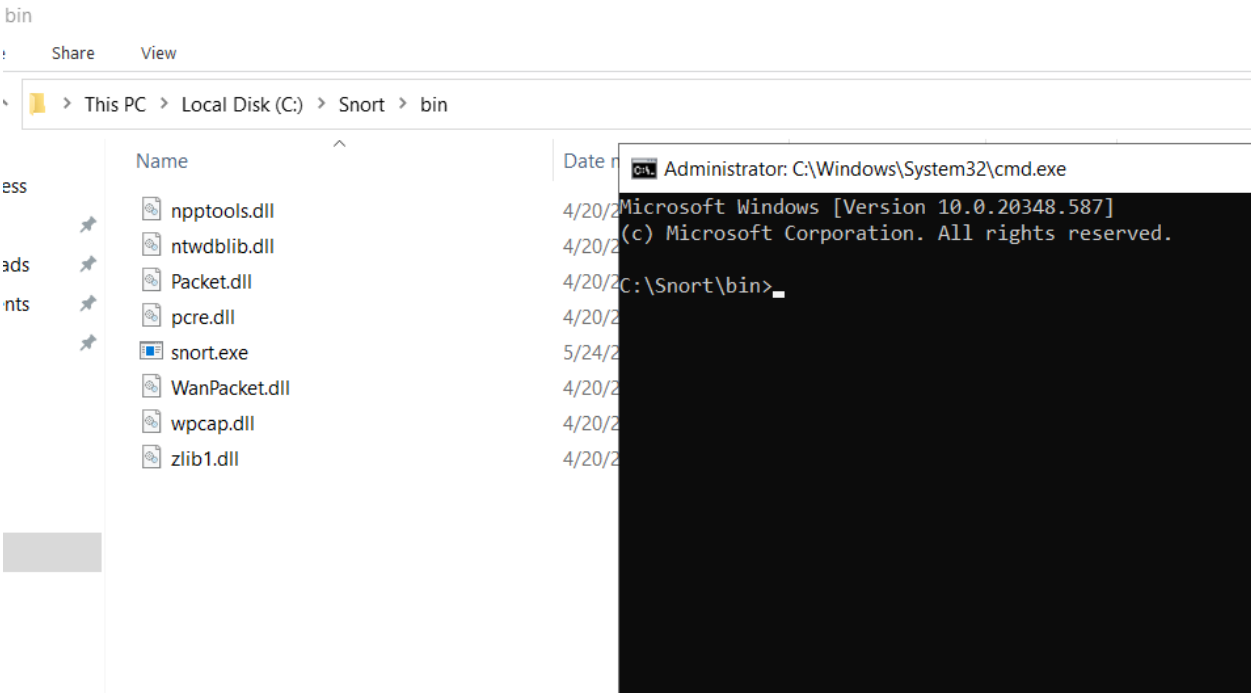
Setup will install Snort 2.9.20 in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.

Destination Folder: C:\Snort

Space required: 7.6 MB
Space available: 44.7 GB

Nullsoft Install System v3.04

Running snort :



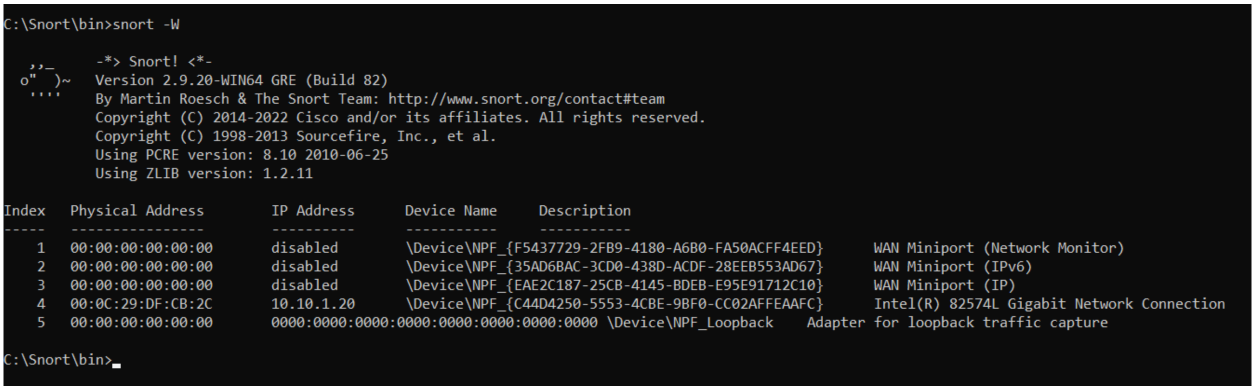
This PC › Local Disk (C:) › Snort › bin

Name	Date m
npptools.dll	4/20/2
ntwdblib.dll	4/20/2
Packet.dll	4/20/2
pcre.dll	4/20/2
snort.exe	5/24/2
WanPacket.dll	4/20/2
wpcap.dll	4/20/2
zlib1.dll	4/20/2

Administrator: C:\Windows\System32\cmd.exe

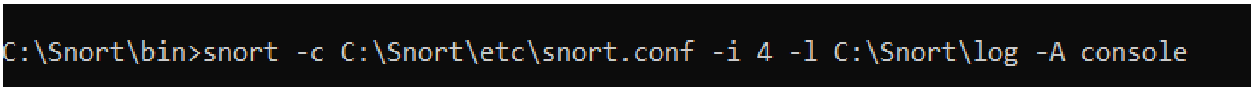
```
Microsoft Windows [Version 10.0.20348.587]
(c) Microsoft Corporation. All rights reserved.

C:\Snort\bin>_
```

Checking the index number of interface:



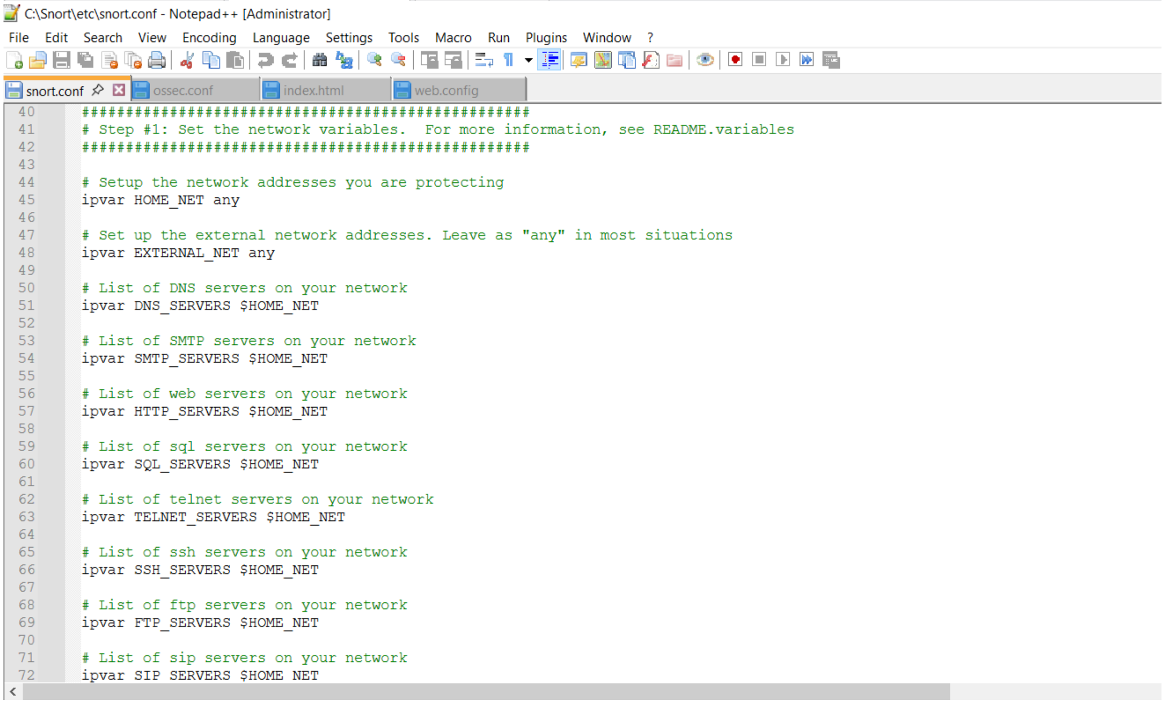
```
C:\Snort\bin>snort -W

   ,,_     -*> Snort! <*-
  o"  )~   Version 2.9.20-WIN64 GRE (Build 82)
   ''''    By Martin Roesch & The Snort Team: http://www.snort.org/contact#team
           Copyright (C) 2014-2022 Cisco and/or its affiliates. All rights reserved.
           Copyright (C) 1998-2013 Sourcefire, Inc., et al.
           Using PCRE version: 8.10 2010-06-25
           Using ZLIB version: 1.2.11

Index  Physical Address        IP Address      Device Name     Description
-----  -----------------       ----------      -----------     -----------
    1  00:00:00:00:00:00       disabled        \Device\NPF_{F5437729-2FB9-4180-A6B0-FA50ACFF4EED}      WAN Miniport (Network Monitor)
    2  00:00:00:00:00:00       disabled        \Device\NPF_{35AD6BAC-3CD0-438D-ACDF-28EEB553AD67}      WAN Miniport (IPv6)
    3  00:00:00:00:00:00       disabled        \Device\NPF_{EAE2C187-25CB-4145-BDEB-E95E91712C10}      WAN Miniport (IP)
    4  00:0C:29:DF:CB:2C       10.10.1.20      \Device\NPF_{C44D4250-5553-4CBE-9BF0-CC02AFFEAAFC}      Intel(R) 82574L Gigabit Network Connection
    5  00:00:00:00:00:00       0000:0000:0000:0000:0000:0000:0000:0000 \Device\NPF_Loopback    Adapter for loopback traffic capture

C:\Snort\bin>_
```

Snort command :



```
C:\Snort\bin>snort -c C:\Snort\etc\snort.conf -i 4 -l C:\Snort\log -A console
```

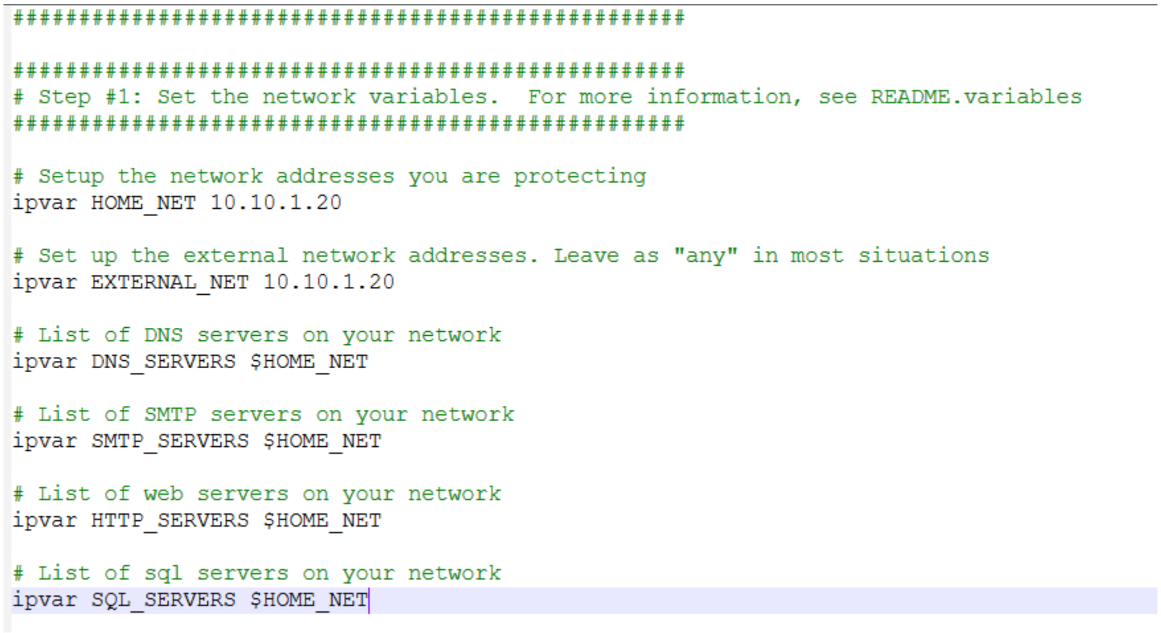
Snort default config:



C:\Snort\etc\snort.conf - Notepad++ [Administrator]
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window ?
snort.conf
ossec.conf
index.html
web.config

```
###################################################
# Step #1: Set the network variables.  For more information, see README.variables
###################################################

# Setup the network addresses you are protecting
ipvar HOME_NET any

# Set up the external network addresses. Leave as "any" in most situations
ipvar EXTERNAL_NET any

# List of DNS servers on your network
ipvar DNS_SERVERS $HOME_NET

# List of SMTP servers on your network
ipvar SMTP_SERVERS $HOME_NET

# List of web servers on your network
ipvar HTTP_SERVERS $HOME_NET

# List of sql servers on your network
ipvar SQL_SERVERS $HOME_NET

# List of telnet servers on your network
ipvar TELNET_SERVERS $HOME_NET

# List of ssh servers on your network
ipvar SSH_SERVERS $HOME_NET

# List of ftp servers on your network
ipvar FTP_SERVERS $HOME_NET

# List of sip servers on your network
ipvar SIP_SERVERS $HOME_NET
```

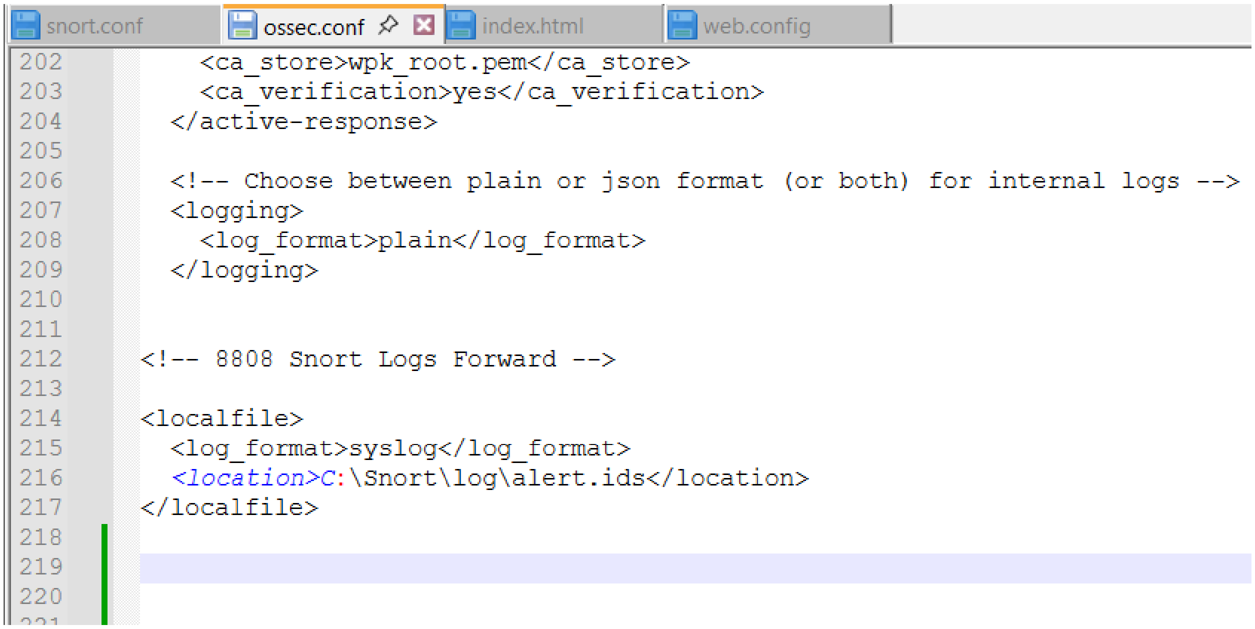
Configured Snort to monitor home_net = 10.10.1.20 :



```
###################################################

###################################################
# Step #1: Set the network variables.  For more information, see README.variables
###################################################

# Setup the network addresses you are protecting
ipvar HOME_NET 10.10.1.20

# Set up the external network addresses. Leave as "any" in most situations
ipvar EXTERNAL_NET 10.10.1.20

# List of DNS servers on your network
ipvar DNS_SERVERS $HOME_NET

# List of SMTP servers on your network
ipvar SMTP_SERVERS $HOME_NET

# List of web servers on your network
ipvar HTTP_SERVERS $HOME_NET

# List of sql servers on your network
ipvar SQL_SERVERS $HOME_NET
```

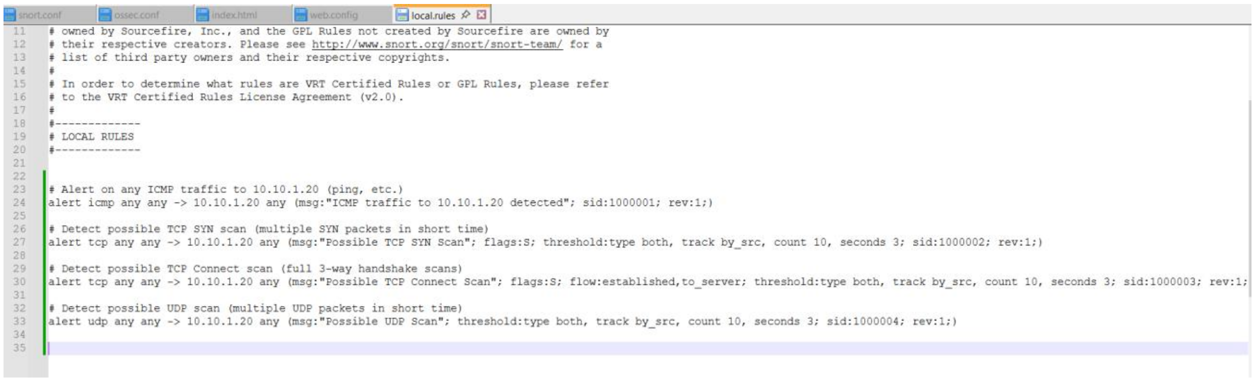
Config can we kept default with “home_net any” as well , but we did it 10.10.1.20 to keep Kali (which is our attacker) out of scope.

Forwarded Snort logs to Wazuh via ossec.conf :



```
      <ca_store>wpk_root.pem</ca_store>
      <ca_verification>yes</ca_verification>
    </active-response>

    <!-- Choose between plain or json format (or both) for internal logs -->
    <logging>
      <log_format>plain</log_format>
    </logging>


  <!-- 8808 Snort Logs Forward -->

  <localfile>
    <log_format>syslog</log_format>
    <location>C:\Snort\log\alert.ids</location>
  </localfile>
```

Updating local.rules:

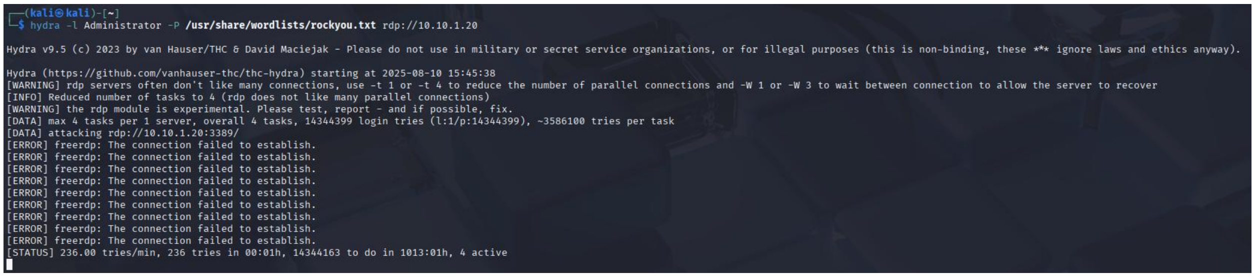


```
# owned by Sourcefire, Inc., and the GPL Rules not created by Sourcefire are owned by
# their respective creators. Please see http://www.snort.org/snort/snort-team/ for a
# list of third party owners and their respective copyrights.
#
# In order to determine what rules are VRT Certified Rules or GPL Rules, please refer
# to the VRT Certified Rules License Agreement (v2.0).
#
#-------------
# LOCAL RULES
#-------------


# Alert on any ICMP traffic to 10.10.1.20 (ping, etc.)
alert icmp any any -> 10.10.1.20 any (msg:"ICMP traffic to 10.10.1.20 detected"; sid:1000001; rev:1;)

# Detect possible TCP SYN scan (multiple SYN packets in short time)
alert tcp any any -> 10.10.1.20 any (msg:"Possible TCP SYN Scan"; flags:S; threshold:type both, track by_src, count 10, seconds 3; sid:1000002; rev:1;)

# Detect possible TCP Connect scan (full 3-way handshake scans)
alert tcp any any -> 10.10.1.20 any (msg:"Possible TCP Connect Scan"; flags:S; flow:established,to_server; threshold:type both, track by_src, count 10, seconds 3; sid:1000003; rev:1;

# Detect possible UDP scan (multiple UDP packets in short time)
alert udp any any -> 10.10.1.20 any (msg:"Possible UDP Scan"; threshold:type both, track by_src, count 10, seconds 3; sid:1000004; rev:1;)


```

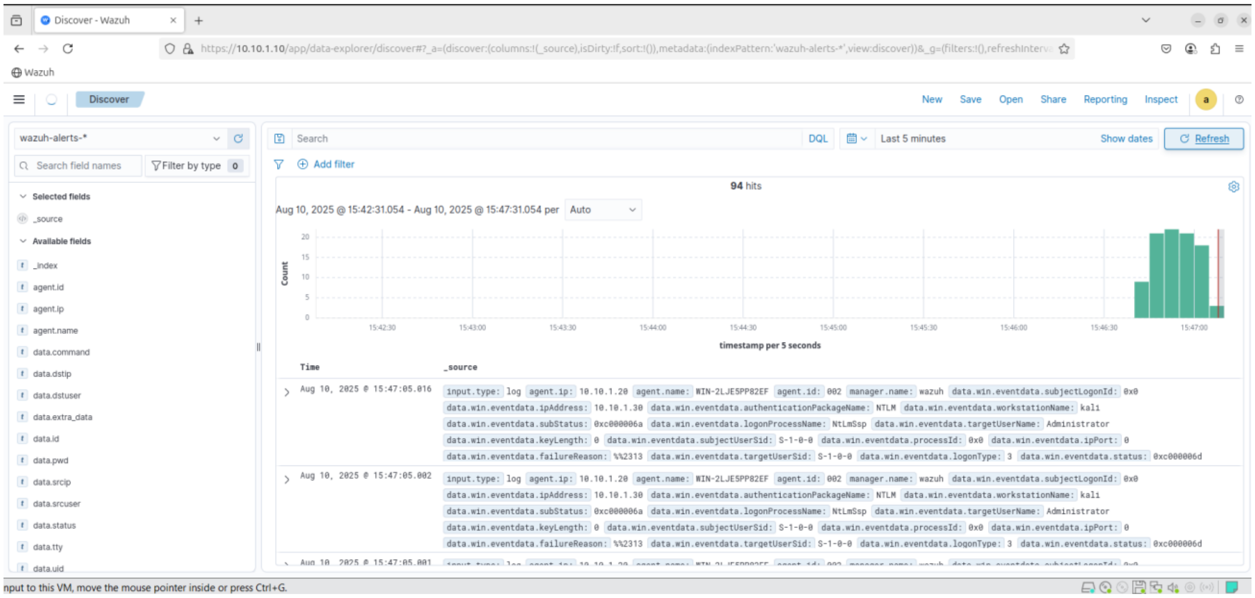
By default, Snort can generate alerts using its preconfigured rule sets without any modifications to the local.rules file. However, we decided to update and customize the local.rules file to ensure that we received precise and relevant alerts for our simulated attacks. This customization allowed us to focus on specific Indicators of Compromise (IoCs) that were part of our project requirements, reducing unnecessary noise and making the alerts in Wazuh more actionable and easier to validate during testing.

6. Attack Simulation & Detection

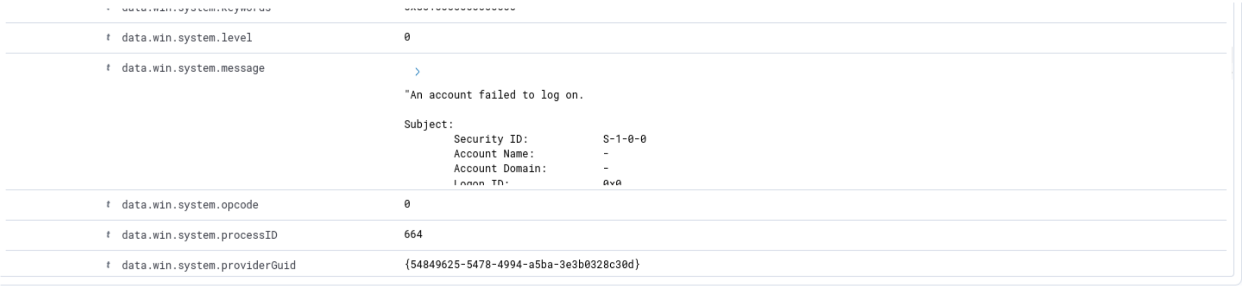
Brute Force Scan Using RDP:



```
┌──(kali㉿kali)-[~]
└─$ hydra -l Administrator -P /usr/share/wordlists/rockyou.txt rdp://10.10.1.20

Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in military or secret service organizations, or for illegal purposes (this is non-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2025-08-10 15:45:38
[WARNING] rdp servers often don't like many connections, use -t 1 or -t 4 to reduce the number of parallel connections and -W 1 or -W 3 to wait between connection to allow the server to recover
[INFO] Reduced number of tasks to 4 (rdp does not like many parallel connections)
[WARNING] the rdp module is experimental. Please test, report - and if possible, fix.
[DATA] max 4 tasks per 1 server, overall 4 tasks, 14344399 login tries (l:1/p:14344399), ~3586100 tries per task
[DATA] attacking rdp://10.10.1.20:3389/
[ERROR] freerdp: The connection failed to establish.
[ERROR] freerdp: The connection failed to establish.
[ERROR] freerdp: The connection failed to establish.
[ERROR] freerdp: The connection failed to establish.
[ERROR] freerdp: The connection failed to establish.
[ERROR] freerdp: The connection failed to establish.
[ERROR] freerdp: The connection failed to establish.
[ERROR] freerdp: The connection failed to establish.
[ERROR] freerdp: The connection failed to establish.
[STATUS] 236.00 tries/min, 236 tries in 00:01h, 14344163 to do in 1013:01h, 4 active
```

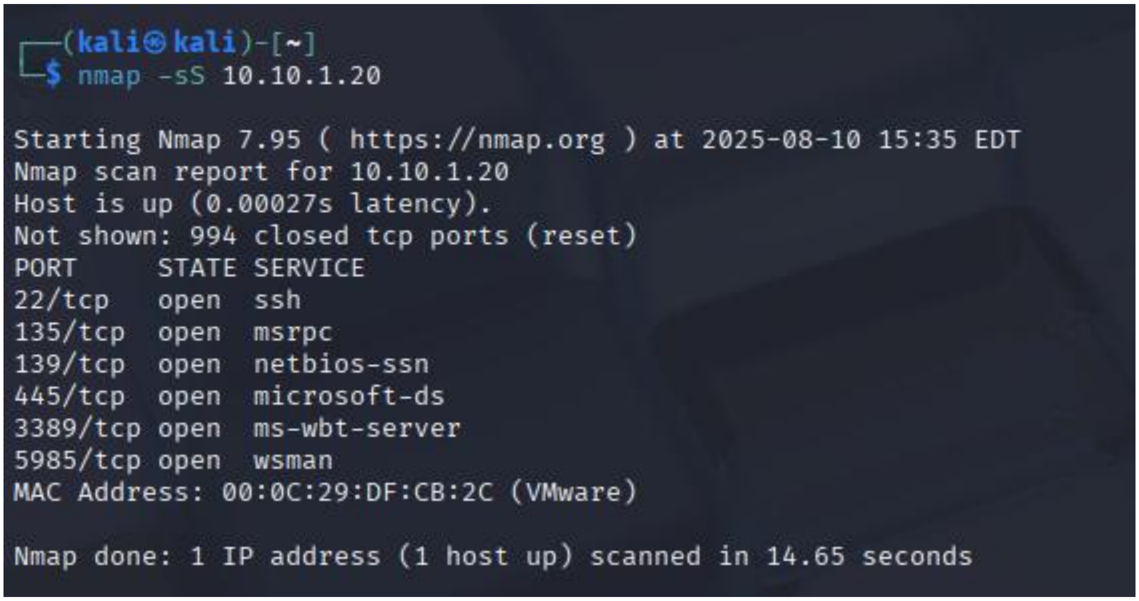


```
Time                          _source
Aug 10, 2025 @ 15:47:05.016   input.type: log agent.ip: 10.10.1.20 agent.name: WIN-2LJE5PP82EF agent.id: 002 manager.name: wazuh data.win.eventdata.subjectLogonId: 0x0
                              data.win.eventdata.ipAddress: 10.10.1.30 data.win.eventdata.authenticationPackageName: NTLM data.win.eventdata.workstationName: kali
                              data.win.eventdata.subStatus: 0xc000006a data.win.eventdata.logonProcessName: NtLmSsp data.win.eventdata.targetUserName: Administrator
                              data.win.eventdata.keyLength: 0 data.win.eventdata.subjectUserSid: S-1-0-0 data.win.eventdata.processId: 0x0 data.win.eventdata.ipPort: 0
                              data.win.eventdata.failureReason: %%2313 data.win.eventdata.targetUserSid: S-1-0-0 data.win.eventdata.logonType: 3 data.win.eventdata.status: 0xc000006d
Aug 10, 2025 @ 15:47:05.002   input.type: log agent.ip: 10.10.1.20 agent.name: WIN-2LJE5PP82EF agent.id: 002 manager.name: wazuh data.win.eventdata.subjectLogonId: 0x0
                              data.win.eventdata.ipAddress: 10.10.1.30 data.win.eventdata.authenticationPackageName: NTLM data.win.eventdata.workstationName: kali
                              data.win.eventdata.subStatus: 0xc000006a data.win.eventdata.logonProcessName: NtLmSsp data.win.eventdata.targetUserName: Administrator
                              data.win.eventdata.keyLength: 0 data.win.eventdata.subjectUserSid: S-1-0-0 data.win.eventdata.processId: 0x0 data.win.eventdata.ipPort: 0
                              data.win.eventdata.failureReason: %%2313 data.win.eventdata.targetUserSid: S-1-0-0 data.win.eventdata.logonType: 3 data.win.eventdata.status: 0xc000006d
Aug 10, 2025 @ 15:47:05.001   input.type: log agent.ip: 10.10.1.20 agent.name: WIN-2LJE5PP82EF agent.id: 002 manager.name: wazuh data.win.eventdata.subjectLogonId: 0x0
```

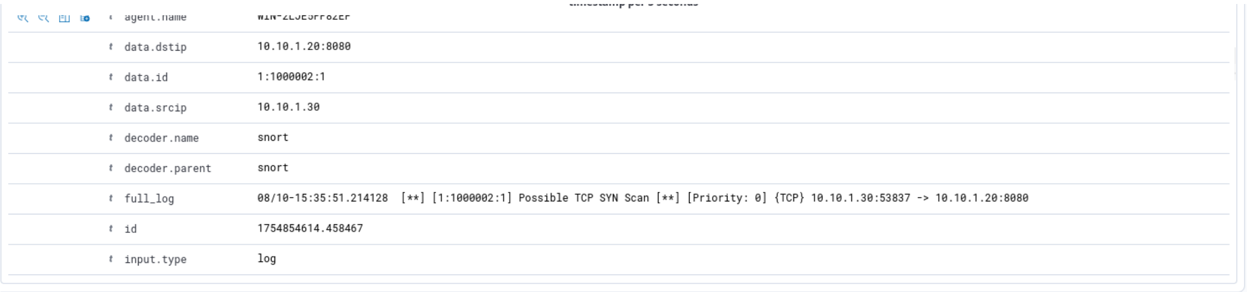


```
t  data.win.system.level          0
t  data.win.system.message        "An account failed to log on.

                                  Subject:
                                          Security ID:            S-1-0-0
                                          Account Name:           -
                                          Account Domain:         -
                                          Logon ID:               0x0
t  data.win.system.opcode         0
t  data.win.system.processID      664
t  data.win.system.providerGuid   {54849625-5478-4994-a5ba-3e3b0328c30d}
```

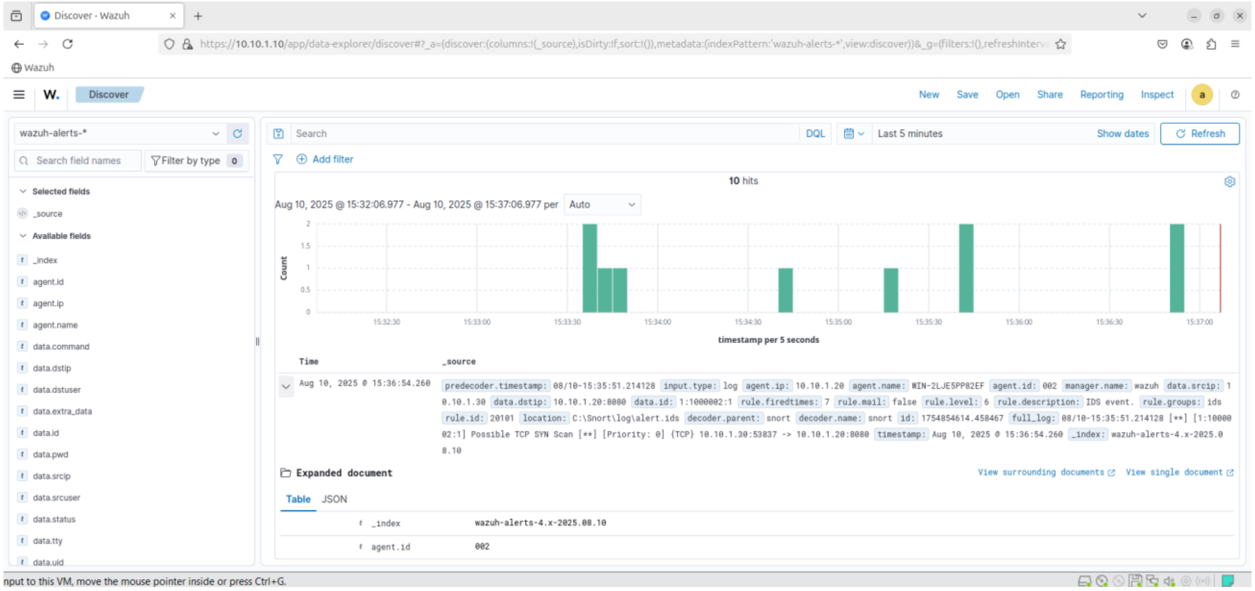
SYN Scan :



```
┌──(kali㉿kali)-[~]
└─$ nmap -sS 10.10.1.20

Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-10 15:35 EDT
Nmap scan report for 10.10.1.20
Host is up (0.00027s latency).
Not shown: 994 closed tcp ports (reset)
PORT     STATE SERVICE
22/tcp   open  ssh
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server
5985/tcp open  wsman
MAC Address: 00:0C:29:DF:CB:2C (VMware)

Nmap done: 1 IP address (1 host up) scanned in 14.65 seconds
```



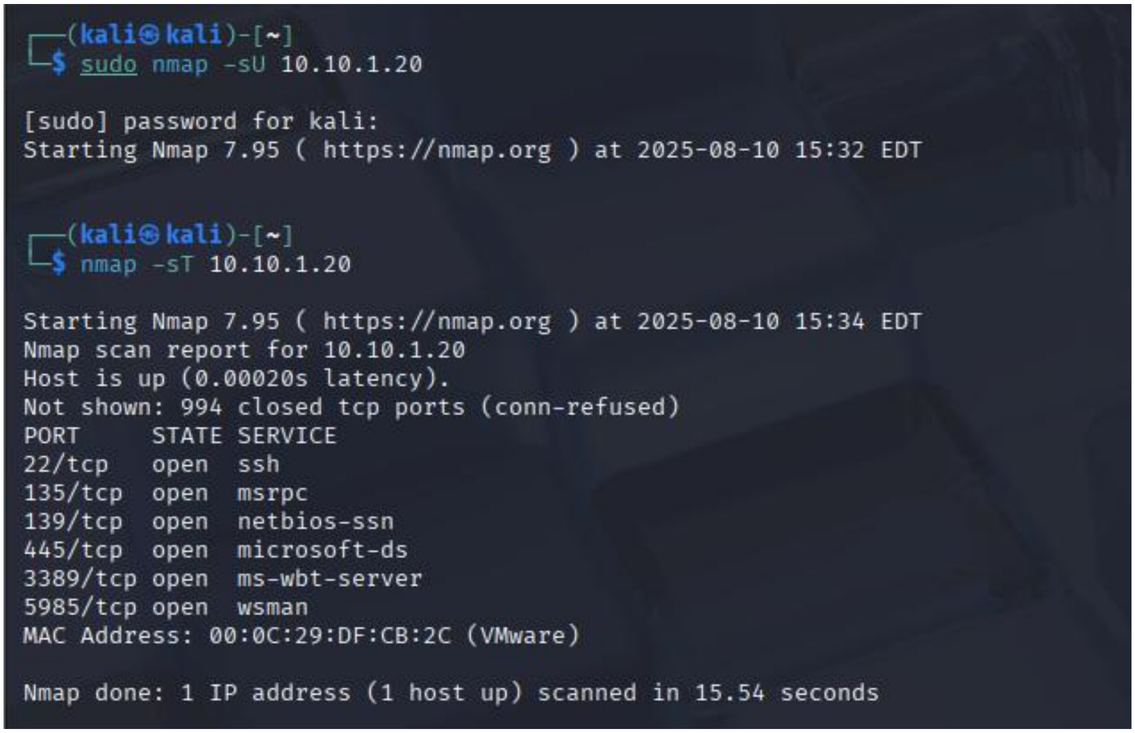
agent.name	WIN-2LJE5PP82EF
data.dstip	10.10.1.20:8080
data.id	1:1000002:1
data.srcip	10.10.1.30
decoder.name	snort
decoder.parent	snort
full_log	08/10-15:35:51.214128 [**] [1:1000002:1] Possible TCP SYN Scan [**] [Priority: 0] {TCP} 10.10.1.30:53837 -> 10.10.1.20:8080
id	1754854614.458467
input.type	log



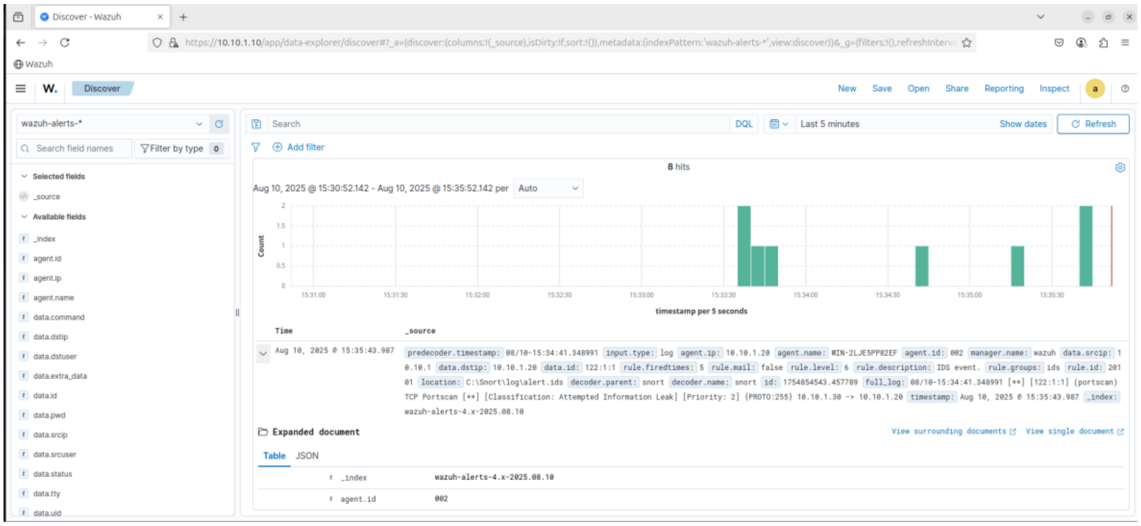
```
Aug 10, 2025 @ 15:36:54.260
predecoder.timestamp: 08/10-15:35:51.214128 input.type: log agent.ip: 10.10.1.20 agent.name: WIN-2LJE5PP82EF agent.id: 002 manager.name: wazuh data.srcip: 10.10.1.30 data.dstip: 10.10.1.20:8080 data.id: 1:1000002:1 rule.firedtimes: 7 rule.mail: false rule.level: 6 rule.description: IDS event. rule.groups: ids rule.id: 20101 location: C:\Snort\log\alert.ids decoder.parent: snort decoder.name: snort id: 1754854614.458467 full_log: 08/10-15:35:51.214128 [**] [1:1000002:1] Possible TCP SYN Scan [**] [Priority: 0] {TCP} 10.10.1.30:53837 -> 10.10.1.20:8080 timestamp: Aug 10, 2025 @ 15:36:54.260 _index: wazuh-alerts-4.x-2025.08.10
```

_index	wazuh-alerts-4.x-2025.08.10
agent.id	002

TCP Scan :

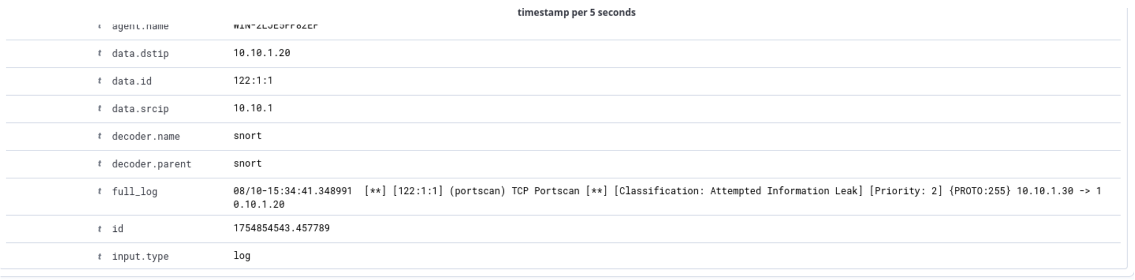


```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sU 10.10.1.20

[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-10 15:32 EDT

┌──(kali㉿kali)-[~]
└─$ nmap -sT 10.10.1.20

Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-10 15:34 EDT
Nmap scan report for 10.10.1.20
Host is up (0.00020s latency).
Not shown: 994 closed tcp ports (conn-refused)
PORT     STATE SERVICE
22/tcp   open  ssh
135/tcp  open  msrpc
139/tcp  open  netbios-ssn
445/tcp  open  microsoft-ds
3389/tcp open  ms-wbt-server
5985/tcp open  wsman
MAC Address: 00:0C:29:DF:CB:2C (VMware)

Nmap done: 1 IP address (1 host up) scanned in 15.54 seconds
```



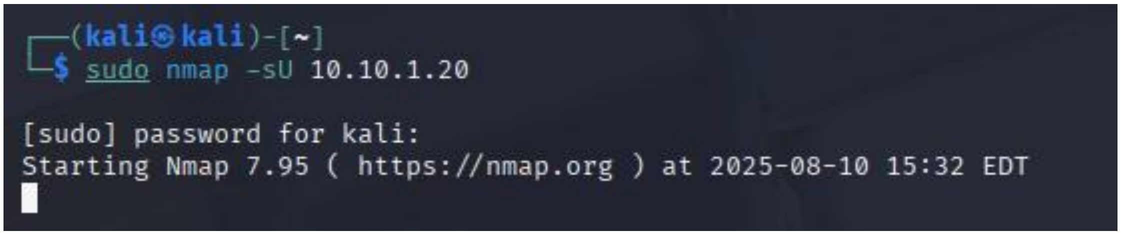
```
Aug 10, 2025 @ 15:35:43.987   predecoder.timestamp: 08/10-15:34:41.348991 input.type: log agent.ip: 10.10.1.20 agent.name: WIN-2LJE5PP82EF agent.id: 002 manager.name: wazuh data.srcip: 1
                              0.10.1 data.dstip: 10.10.1.20 data.id: 122:1:1 rule.firedtimes: 5 rule.mail: false rule.level: 6 rule.description: IDS event. rule.groups: ids rule.id: 201
                              01 location: C:\Snort\log\alert.ids decoder.parent: snort decoder.name: snort id: 1754854543.457789 full_log: 08/10-15:34:41.348991 [**] [122:1:1] (portscan)
                              TCP Portscan [**] [Classification: Attempted Information Leak] [Priority: 2] {PROTO:255} 10.10.1.30 -> 10.10.1.20 timestamp: Aug 10, 2025 @ 15:35:43.987 _index:
                              wazuh-alerts-4.x-2025.08.10
```

_index	wazuh-alerts-4.x-2025.08.10
agent.id	002

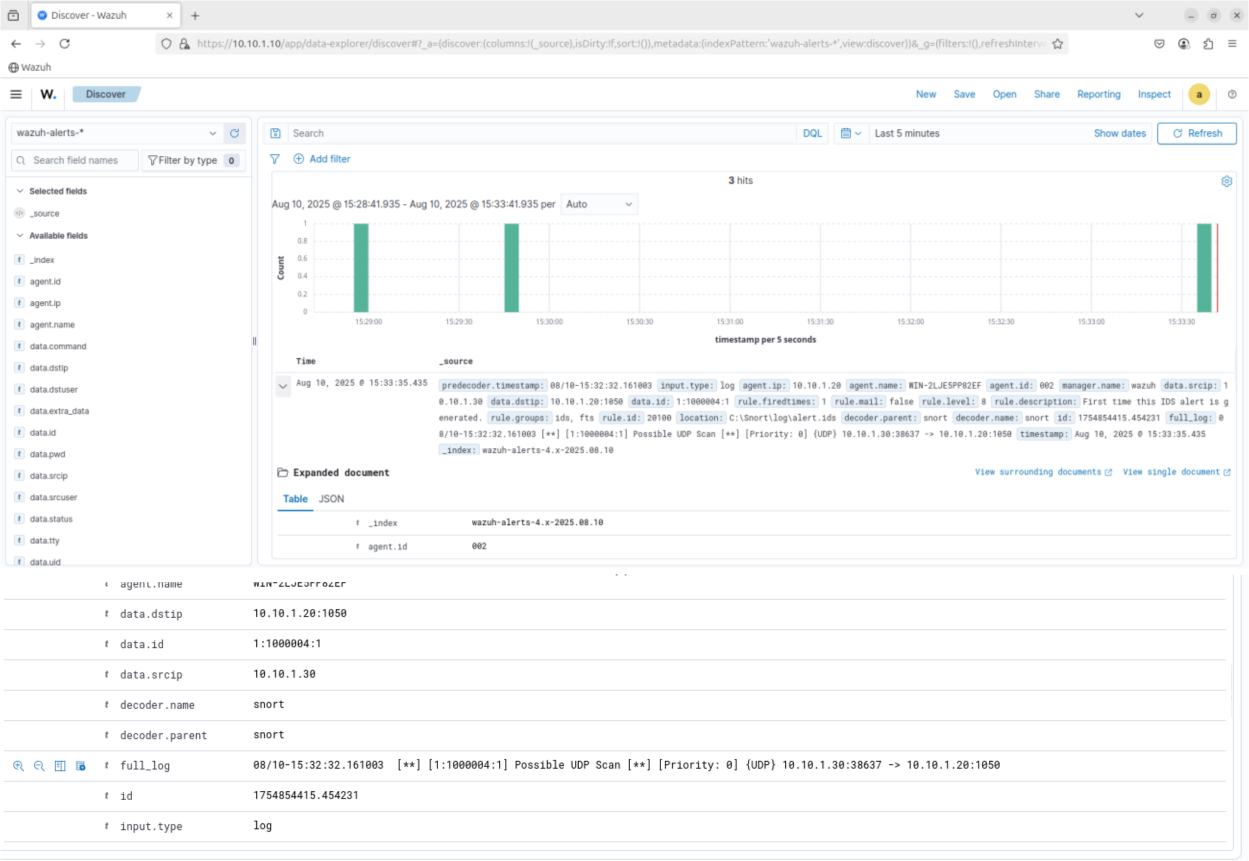


agent.name	WIN-2LJE5PP82EF
data.dstip	10.10.1.20
data.id	122:1:1
data.srcip	10.10.1
decoder.name	snort
decoder.parent	snort
full_log	08/10-15:34:41.348991 [**] [122:1:1] (portscan) TCP Portscan [**] [Classification: Attempted Information Leak] [Priority: 2] {PROTO:255} 10.10.1.30 -> 10.10.1.20
id	1754854543.457789
input.type	log

UDP Scan:



```
┌──(kali㉿kali)-[~]
└─$ sudo nmap -sU 10.10.1.20

[sudo] password for kali:
Starting Nmap 7.95 ( https://nmap.org ) at 2025-08-10 15:32 EDT
```



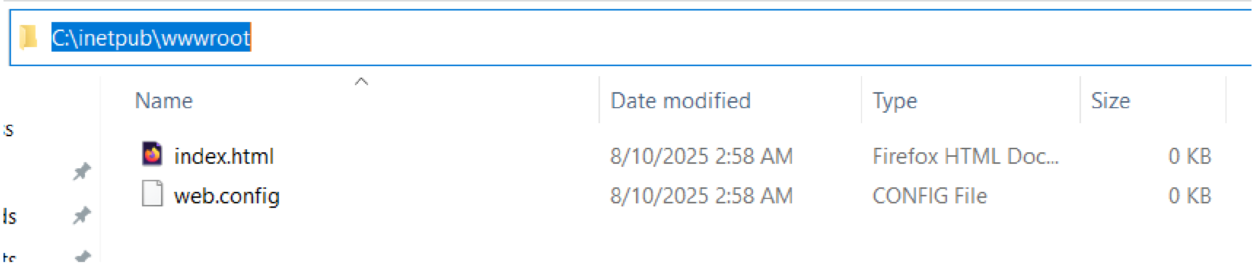
Time	_source
Aug 10, 2025 @ 15:33:35.435	predecoder.timestamp: 08/10-15:32:32.161003 input.type: log agent.ip: 10.10.1.20 agent.name: WIN-2LJE5PP82EF agent.id: 002 manager.name: wazuh data.srcip: 10.10.1.30 data.dstip: 10.10.1.20:1050 data.id: 1:1000004:1 rule.firedtimes: 1 rule.mail: false rule.level: 8 rule.description: First time this IDS alert is generated. rule.groups: ids, fts rule.id: 20100 location: C:\Snort\log\alert.ids decoder.parent: snort decoder.name: snort id: 1754854415.454231 full_log: 08/10-15:32:32.161003 [**] [1:1000004:1] Possible UDP Scan [**] [Priority: 0] {UDP} 10.10.1.30:38637 -> 10.10.1.20:1050 timestamp: Aug 10, 2025 @ 15:33:35.435 _index: wazuh-alerts-4.x-2025.08.10

Expanded document

_index	wazuh-alerts-4.x-2025.08.10
agent.id	002
agent.name	WIN-2LJE5PP82EF
data.dstip	10.10.1.20:1050
data.id	1:1000004:1
data.srcip	10.10.1.30
decoder.name	snort
decoder.parent	snort
full_log	08/10-15:32:32.161003 [**] [1:1000004:1] Possible UDP Scan [**] [Priority: 0] {UDP} 10.10.1.30:38637 -> 10.10.1.20:1050
id	1754854415.454231
input.type	log

7. File Integrity Monitoring (FIM)

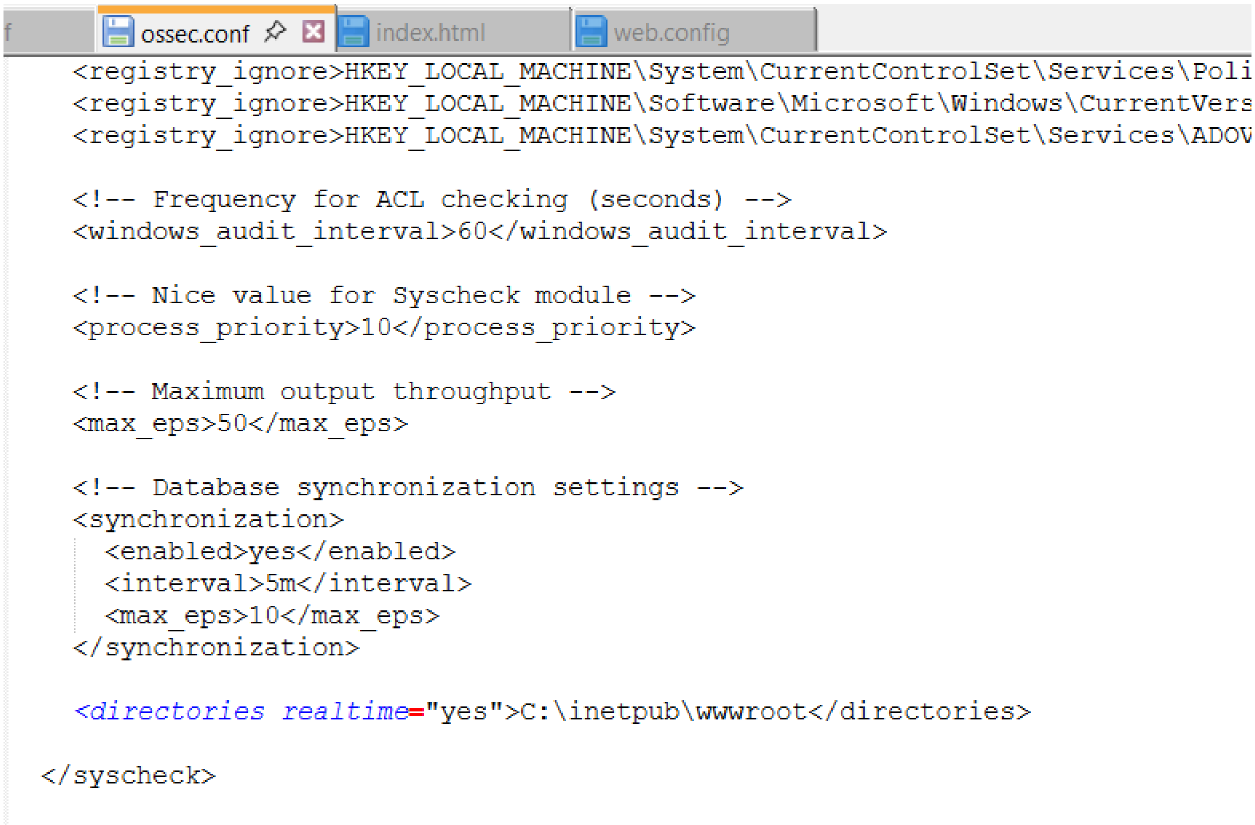
Monitored files: index.html & web.config



C:\inetpub\wwwroot

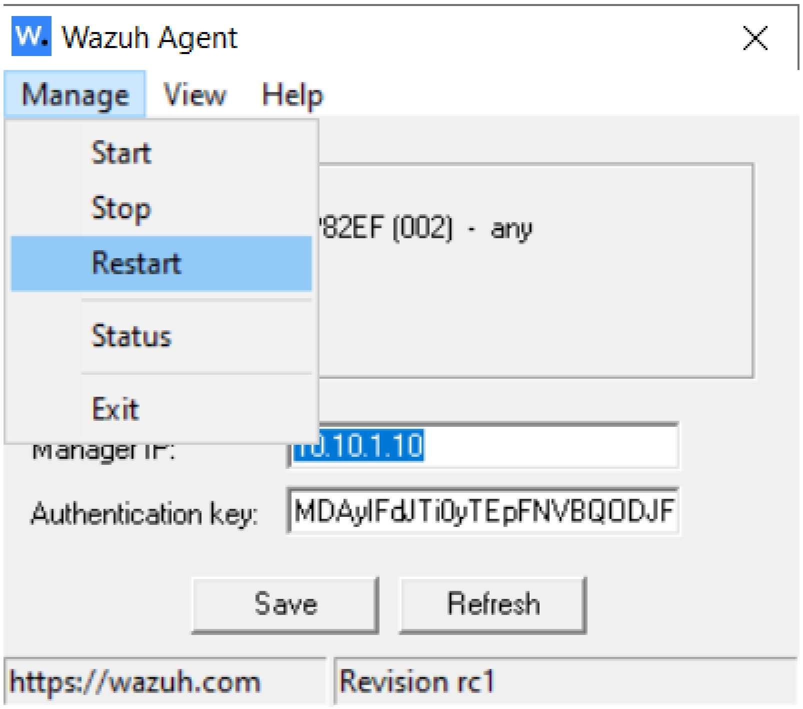
Name	Date modified	Type	Size
index.html	8/10/2025 2:58 AM	Firefox HTML Doc...	0 KB
web.config	8/10/2025 2:58 AM	CONFIG File	0 KB

Configuration: Modified ossec.conf



```
    <registry_ignore>HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\Poli
    <registry_ignore>HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVers
    <registry_ignore>HKEY_LOCAL_MACHINE\System\CurrentControlSet\Services\ADOV

    <!-- Frequency for ACL checking (seconds) -->
    <windows_audit_interval>60</windows_audit_interval>

    <!-- Nice value for Syscheck module -->
    <process_priority>10</process_priority>

    <!-- Maximum output throughput -->
    <max_eps>50</max_eps>

    <!-- Database synchronization settings -->
    <synchronization>
      <enabled>yes</enabled>
      <interval>5m</interval>
      <max_eps>10</max_eps>
    </synchronization>

    <directories realtime="yes">C:\inetpub\wwwroot</directories>

  </syscheck>
```

Restarting Wazuh Agent on Server to take changes:



'82EF (002) - any

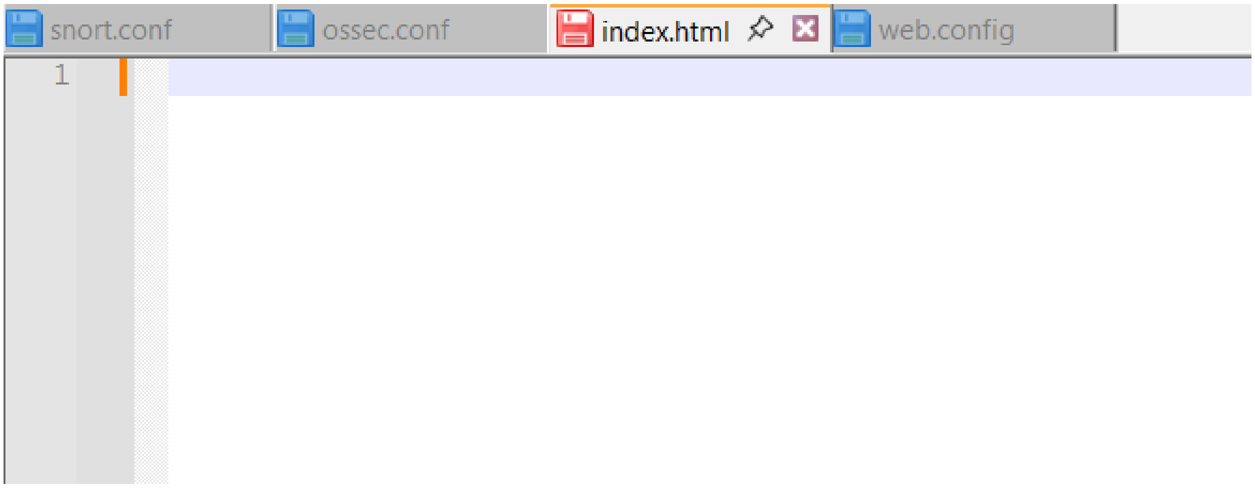
Manager IP: 10.10.1.10

Authentication key: MDAyIFdJTi0yTEpFNVBQODJF

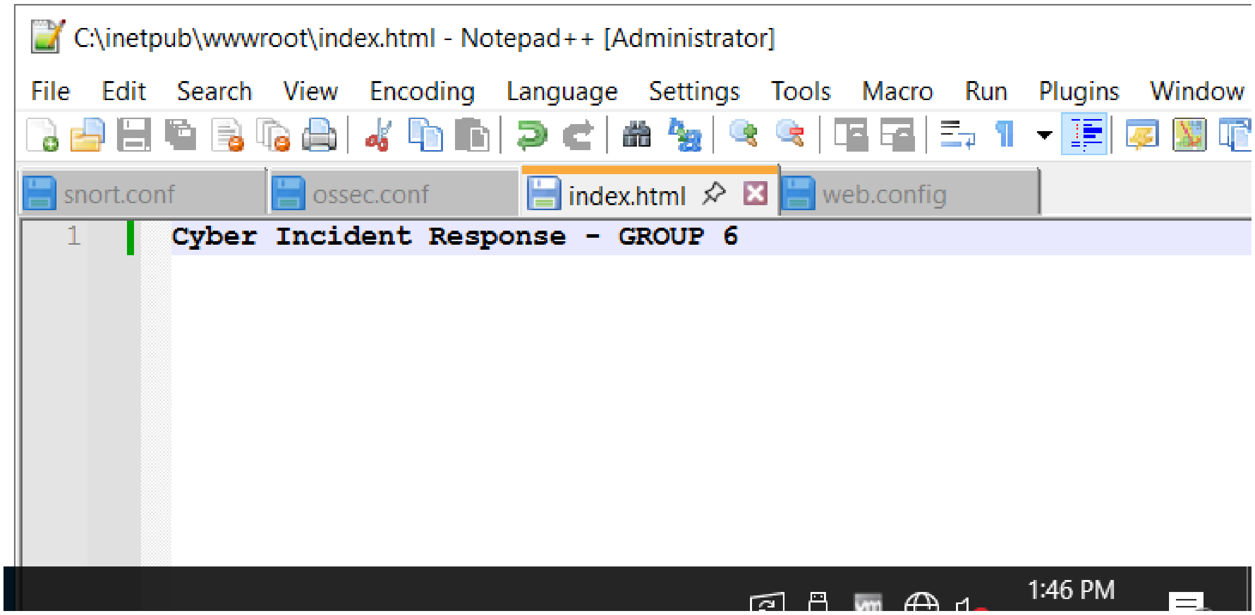
https://wazuh.com

Revision rc1

Test: Currently the index.html is empty

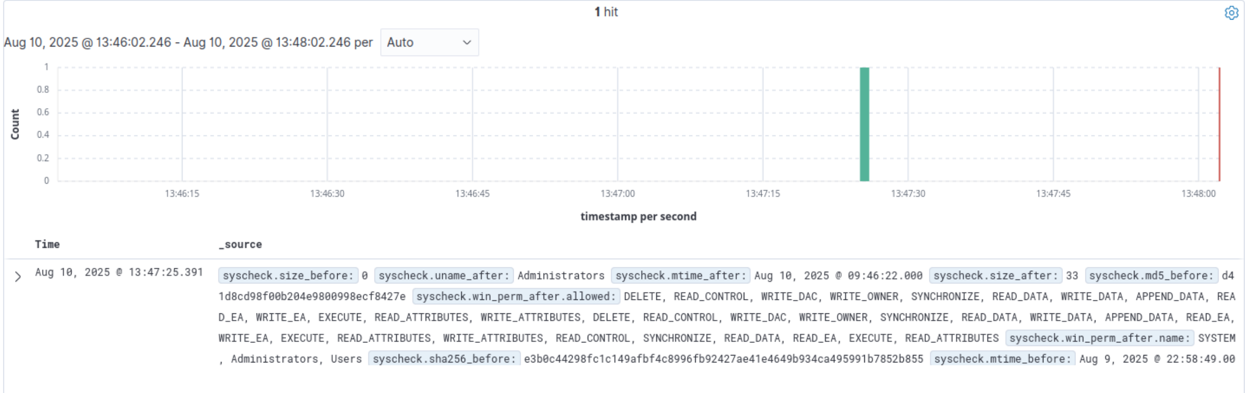


After editing the file :

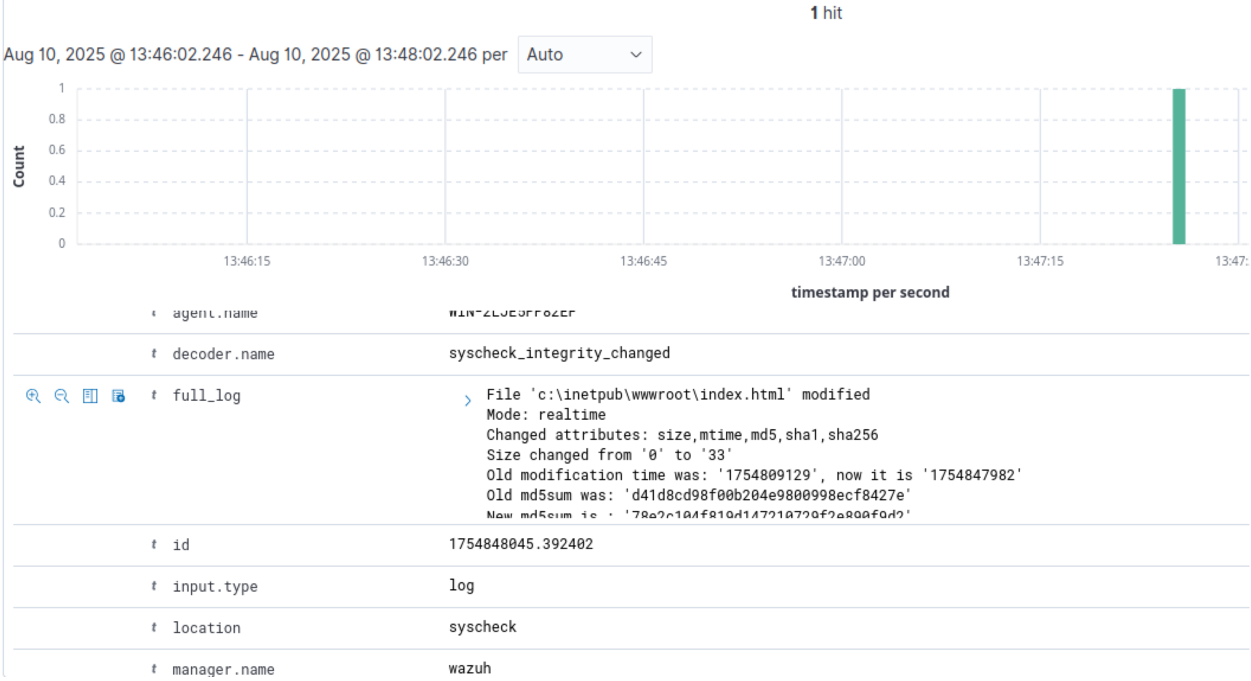

C:\inetpub\wwwroot\index.html - Notepad++ [Administrator]
File Edit Search View Encoding Language Settings Tools Macro Run Plugins Window
snort.conf
ossec.conf
index.html
web.config
1:46 PM

```
Cyber Incident Response - GROUP 6
```

We can see the generated log on Wazuh Dashboard:



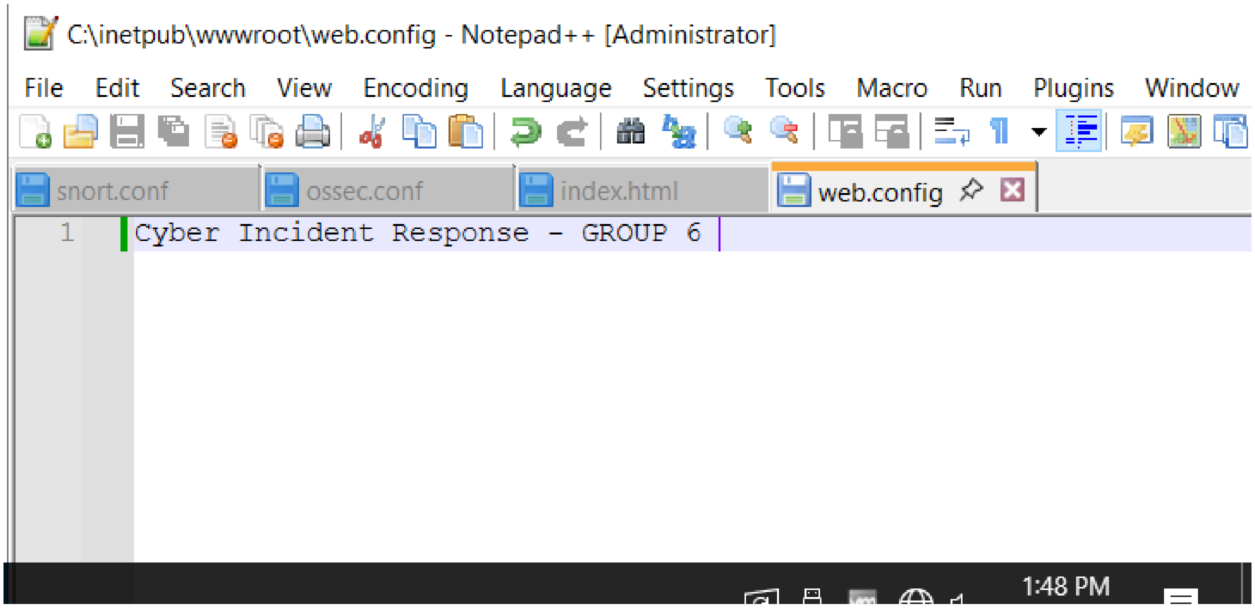
```
Time                      _source
Aug 10, 2025 @ 13:47:25.391  syscheck.size_before: 0 syscheck.uname_after: Administrators syscheck.mtime_after: Aug 10, 2025 @ 09:46:22.000 syscheck.size_after: 33 syscheck.md5_before: d4
1d8cd98f00b204e9800998ecf8427e syscheck.win_perm_after.allowed: DELETE, READ_CONTROL, WRITE_DAC, WRITE_OWNER, SYNCHRONIZE, READ_DATA, WRITE_DATA, APPEND_DATA, REA
D_EA, WRITE_EA, EXECUTE, READ_ATTRIBUTES, WRITE_ATTRIBUTES, DELETE, READ_CONTROL, WRITE_DAC, WRITE_OWNER, SYNCHRONIZE, READ_DATA, WRITE_DATA, APPEND_DATA, READ_EA,
WRITE_EA, EXECUTE, READ_ATTRIBUTES, WRITE_ATTRIBUTES, READ_CONTROL, SYNCHRONIZE, READ_DATA, READ_EA, EXECUTE, READ_ATTRIBUTES syscheck.win_perm_after.name: SYSTEM
, Administrators, Users syscheck.sha256_before: e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855 syscheck.mtime_before: Aug 9, 2025 @ 22:58:49.00
```



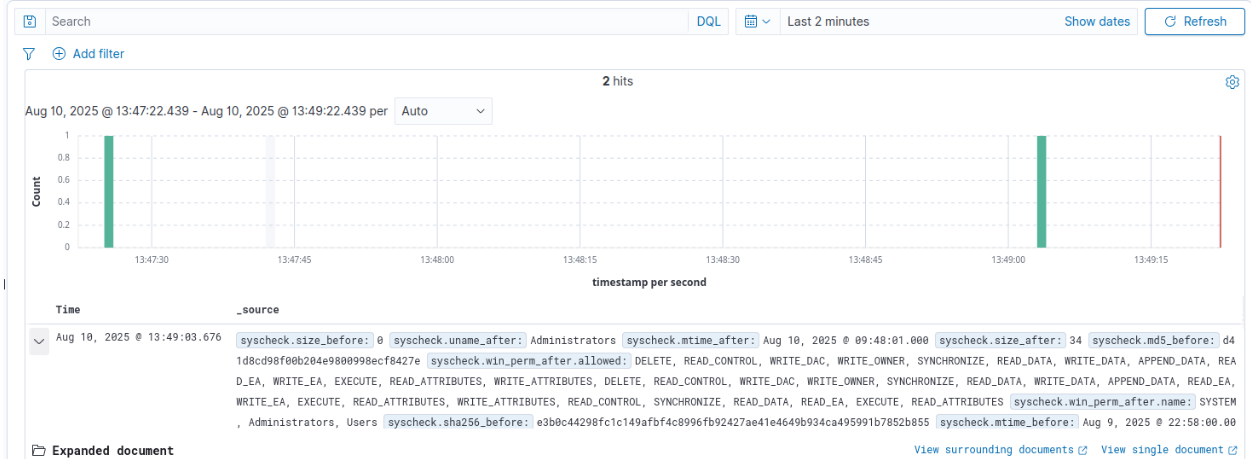
```
t  agent.name       WIN-ZLJE5FF8ZEF
t  decoder.name     syscheck_integrity_changed
t  full_log         File 'c:\inetpub\wwwroot\index.html' modified
                    Mode: realtime
                    Changed attributes: size,mtime,md5,sha1,sha256
                    Size changed from '0' to '33'
                    Old modification time was: '1754809129', now it is '1754847982'
                    Old md5sum was: 'd41d8cd98f00b204e9800998ecf8427e'
                    New md5sum is : '78e2c104f819d147210729f2e890f9d2'
t  id               1754848045.392402
t  input.type       log
t  location         syscheck
t  manager.name     wazuh
```

full_log shows the location where the modification was done. Which is the location of index.html in wwwroot folder. Other information like changed attributes, size(old and new) etc. is also visible.

We will do the same test with web.config file :



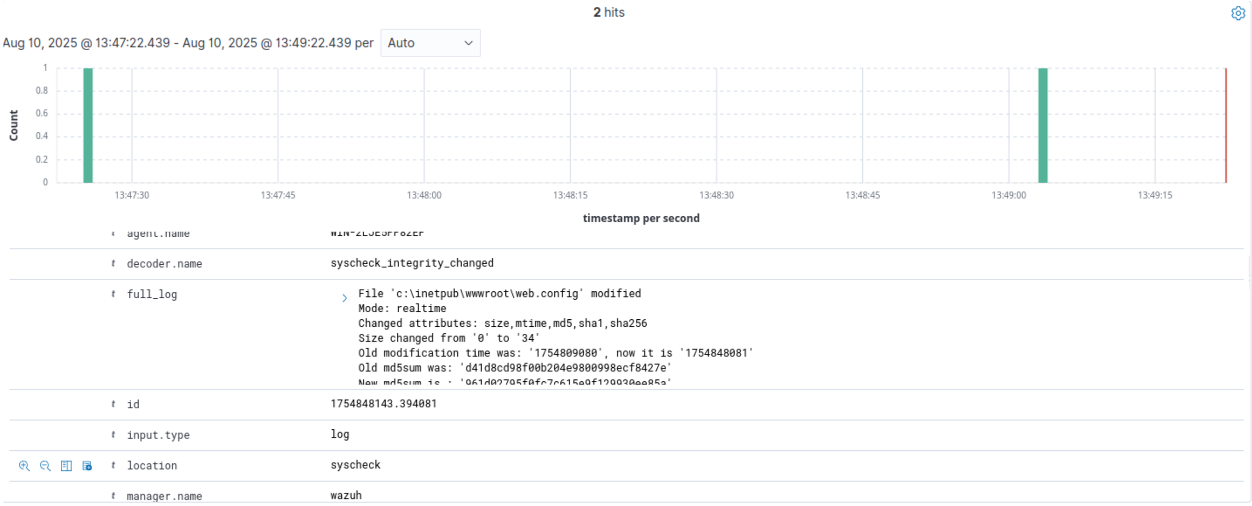
```
Cyber Incident Response - GROUP 6
```



Time	_source
Aug 10, 2025 @ 13:49:03.676	syscheck.size_before: 0 syscheck.uname_after: Administrators syscheck.mtime_after: Aug 10, 2025 @ 09:48:01.000 syscheck.size_after: 34 syscheck.md5_before: d41d8cd98f00b204e9800998ecf8427e syscheck.win_perm_after.allowed: DELETE, READ_CONTROL, WRITE_DAC, WRITE_OWNER, SYNCHRONIZE, READ_DATA, WRITE_DATA, APPEND_DATA, READ_EA, WRITE_EA, EXECUTE, READ_ATTRIBUTES, WRITE_ATTRIBUTES, DELETE, READ_CONTROL, WRITE_DAC, WRITE_OWNER, SYNCHRONIZE, READ_DATA, WRITE_DATA, APPEND_DATA, READ_EA, WRITE_EA, EXECUTE, READ_ATTRIBUTES, WRITE_ATTRIBUTES, READ_CONTROL, SYNCHRONIZE, READ_DATA, READ_EA, EXECUTE, READ_ATTRIBUTES syscheck.win_perm_after.name: SYSTEM, Administrators, Users syscheck.sha256_before: e3b0c44298fc1c149afbf4c8996fb92427ae41e4649b934ca495991b7852b855 syscheck.mtime_before: Aug 9, 2025 @ 22:58:00.00

Expanded document

View surrounding documents View single document

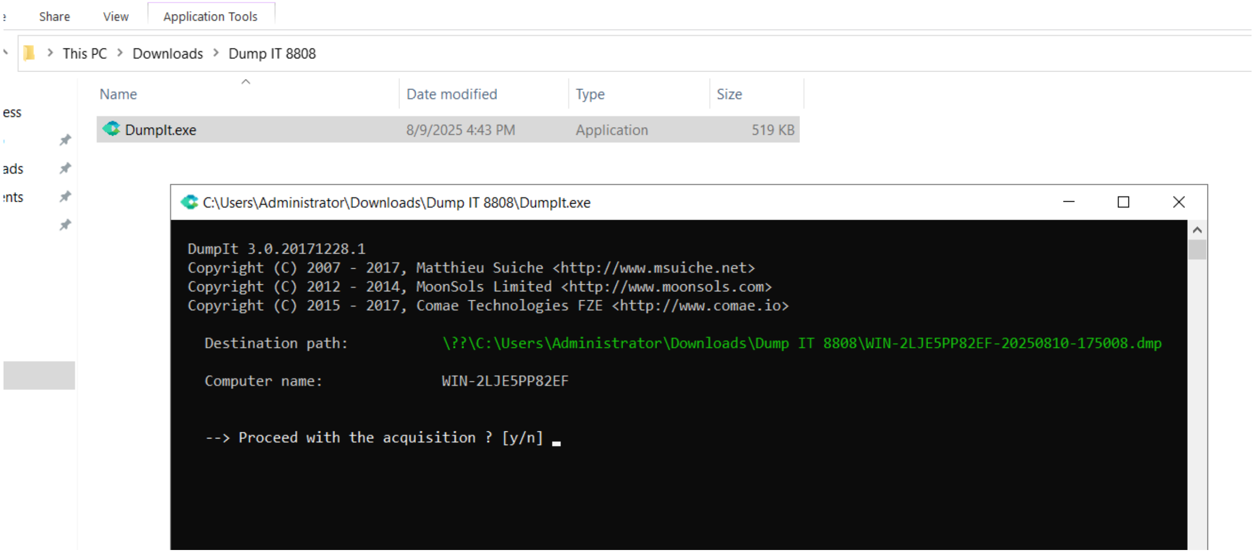


agent.name	WIN-2LJE5FF82EF
decoder.name	syscheck_integrity_changed
full_log	File 'c:\inetpub\wwwroot\web.config' modified Mode: realtime Changed attributes: size,mtime,md5,sha1,sha256 Size changed from '0' to '34' Old modification time was: '1754809080', now it is '1754848081' Old md5sum was: 'd41d8cd98f00b204e9800998ecf8427e' New md5sum is : '961d02795f0fc7c615e9f129930ee85a'
id	1754848143.394081
input.type	log
location	syscheck
manager.name	wazuh

Successful log generated.

8. Memory Analysis with Volatility

Dumped memory using DumpIt:

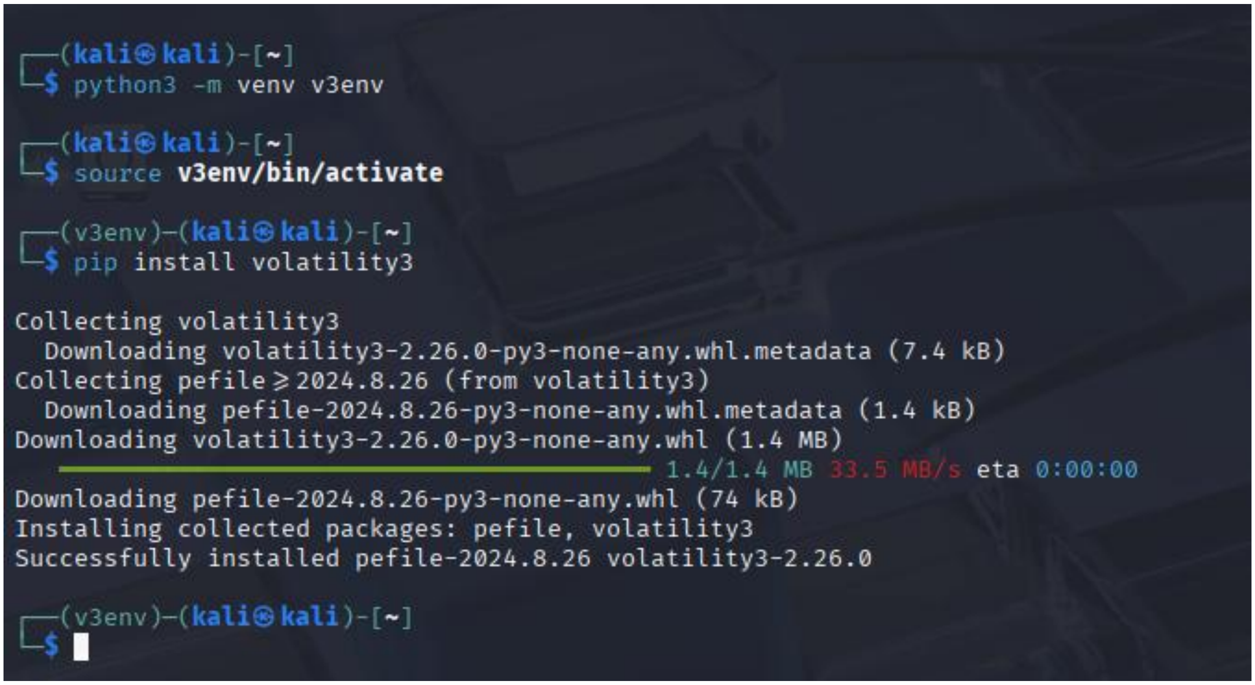


Name	Date modified	Type	Size
DumpIt.exe	8/9/2025 4:43 PM	Application	519 KB

C:\Users\Administrator\Downloads\Dump IT 8808\DumpIt.exe

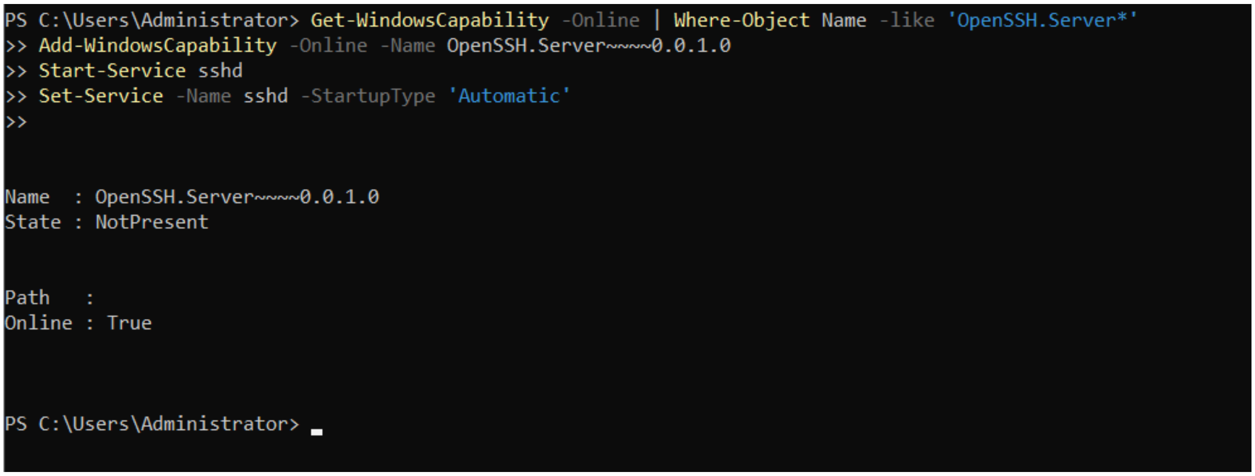
```
DumpIt 3.0.20171228.1
Copyright (C) 2007 - 2017, Matthieu Suiche <http://www.msuiche.net>
Copyright (C) 2012 - 2014, MoonSols Limited <http://www.moonsols.com>
Copyright (C) 2015 - 2017, Comae Technologies FZE <http://www.comae.io>

  Destination path:           \??\C:\Users\Administrator\Downloads\Dump IT 8808\WIN-2LJE5PP82EF-20250810-175008.dmp

  Computer name:              WIN-2LJE5PP82EF


  --> Proceed with the acquisition ? [y/n] _
```

Setting up Volatility:



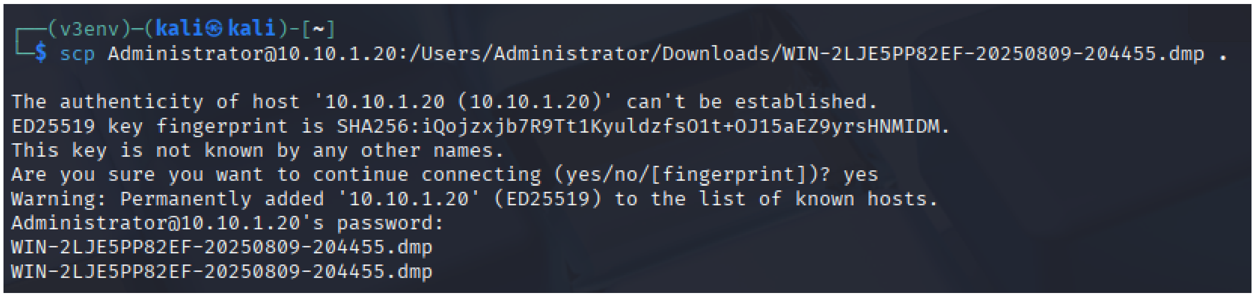
```
┌──(kali㉿kali)-[~]
└─$ python3 -m venv v3env

┌──(kali㉿kali)-[~]
└─$ source v3env/bin/activate

┌──(v3env)─(kali㉿kali)-[~]
└─$ pip install volatility3

Collecting volatility3
  Downloading volatility3-2.26.0-py3-none-any.whl.metadata (7.4 kB)
Collecting pefile>=2024.8.26 (from volatility3)
  Downloading pefile-2024.8.26-py3-none-any.whl.metadata (1.4 kB)
Downloading volatility3-2.26.0-py3-none-any.whl (1.4 MB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 1.4/1.4 MB 33.5 MB/s eta 0:00:00
Downloading pefile-2024.8.26-py3-none-any.whl (74 kB)
Installing collected packages: pefile, volatility3
Successfully installed pefile-2024.8.26 volatility3-2.26.0

┌──(v3env)─(kali㉿kali)-[~]
└─$ █
```

Installing OpenSSH on Windows Server so SCP can work:



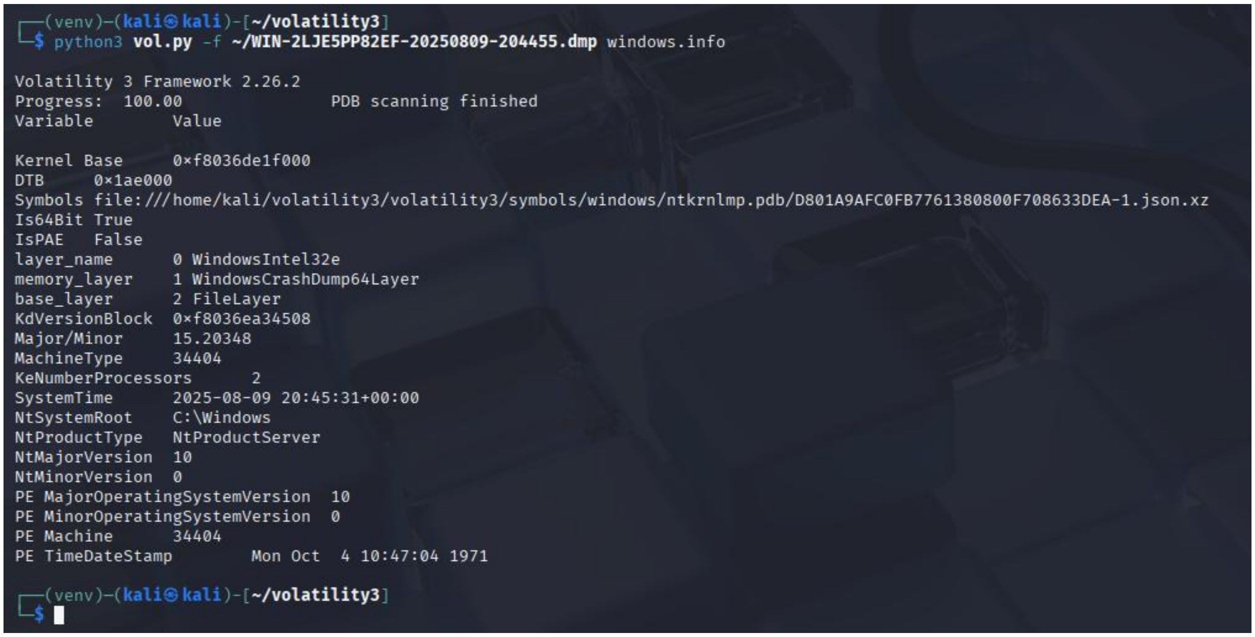
```
PS C:\Users\Administrator> Get-WindowsCapability -Online | Where-Object Name -like 'OpenSSH.Server*'
>> Add-WindowsCapability -Online -Name OpenSSH.Server~~~~0.0.1.0
>> Start-Service sshd
>> Set-Service -Name sshd -StartupType 'Automatic'
>>


Name  : OpenSSH.Server~~~~0.0.1.0
State : NotPresent



Path   :
Online : True



PS C:\Users\Administrator> _
```

Transferred .dmp file to Kali via SCP :



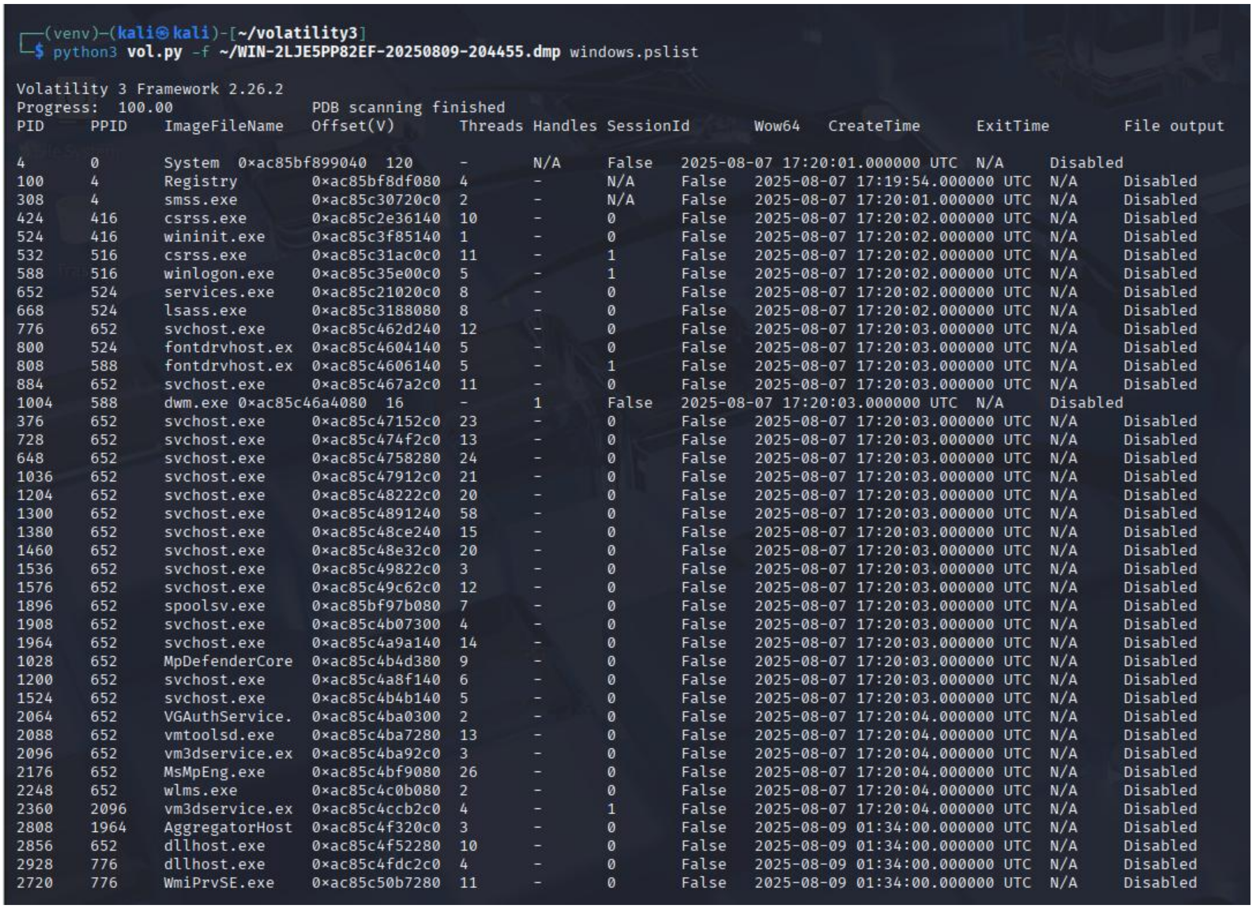
```
┌──(v3env)─(kali㉿kali)-[~]
└─$ scp Administrator@10.10.1.20:/Users/Administrator/Downloads/WIN-2LJE5PP82EF-20250809-204455.dmp .

The authenticity of host '10.10.1.20 (10.10.1.20)' can't be established.
ED25519 key fingerprint is SHA256:iQojzxjb7R9Tt1KyuldzfsO1t+OJ15aEZ9yrsHNMIDM.
This key is not known by any other names.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added '10.10.1.20' (ED25519) to the list of known hosts.
Administrator@10.10.1.20's password:
WIN-2LJE5PP82EF-20250809-204455.dmp
WIN-2LJE5PP82EF-20250809-204455.dmp
```

Command used to inspect .dmp file using volatility:



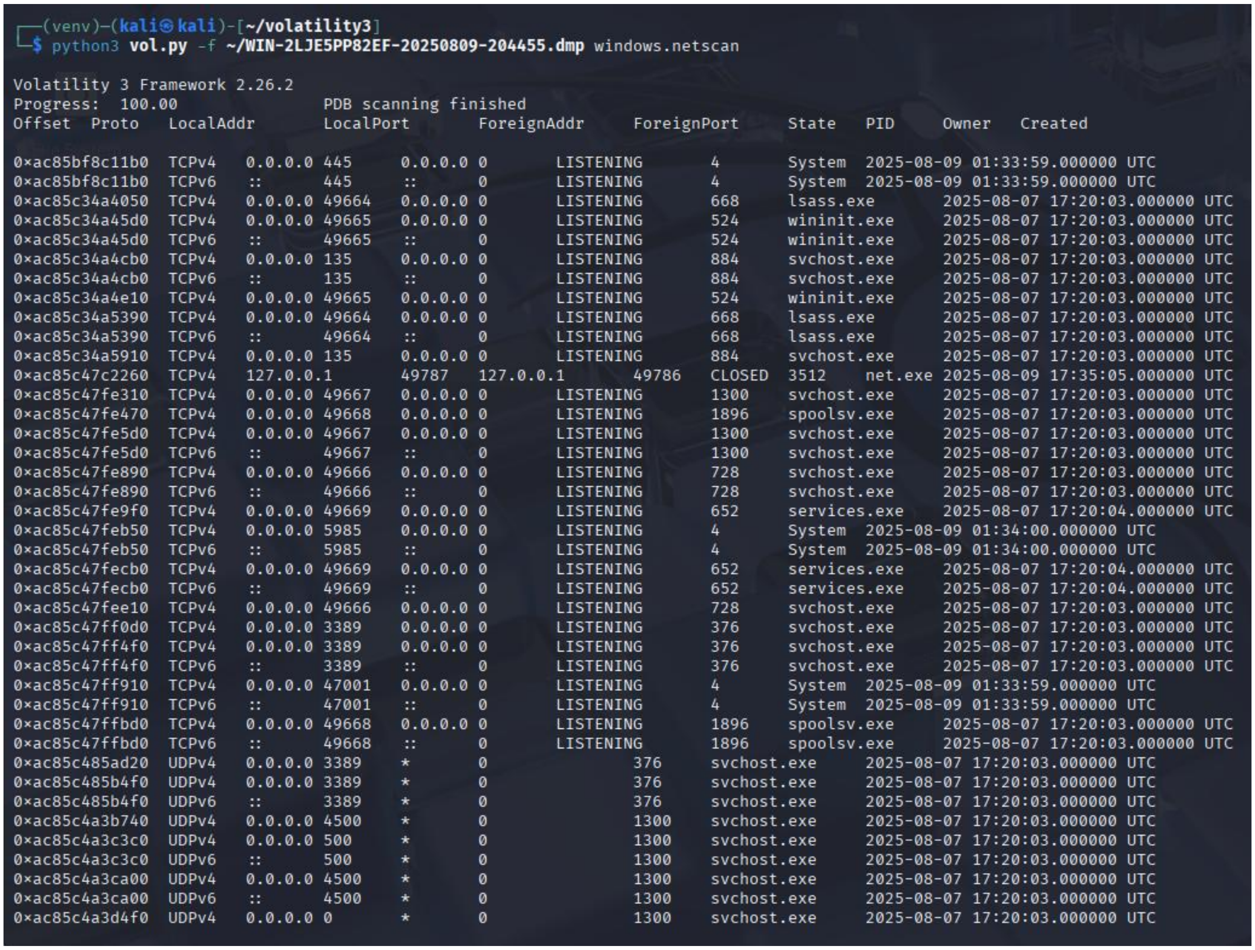
```
┌──(venv)─(kali㉿kali)-[~/volatility3]
└─$ python3 vol.py -f ~/WIN-2LJE5PP82EF-20250809-204455.dmp windows.info

Volatility 3 Framework 2.26.2
Progress:  100.00               PDB scanning finished
Variable        Value

Kernel Base     0xf8036de1f000
DTB     0x1ae000
Symbols file:///home/kali/volatility3/volatility3/symbols/windows/ntkrnlmp.pdb/D801A9AFC0FB7761380800F708633DEA-1.json.xz
Is64Bit True
IsPAE   False
layer_name      0 WindowsIntel32e
memory_layer    1 WindowsCrashDump64Layer
base_layer      2 FileLayer
KdVersionBlock  0xf8036ea34508
Major/Minor     15.20348
MachineType     34404
KeNumberProcessors      2
SystemTime      2025-08-09 20:45:31+00:00
NtSystemRoot    C:\Windows
NtProductType   NtProductServer
NtMajorVersion  10
NtMinorVersion  0
PE MajorOperatingSystemVersion  10
PE MinorOperatingSystemVersion  0
PE Machine      34404
PE TimeDateStamp        Mon Oct  4 10:47:04 1971

┌──(venv)─(kali㉿kali)-[~/volatility3]
└─$ █
```

Windows.pslist dump:

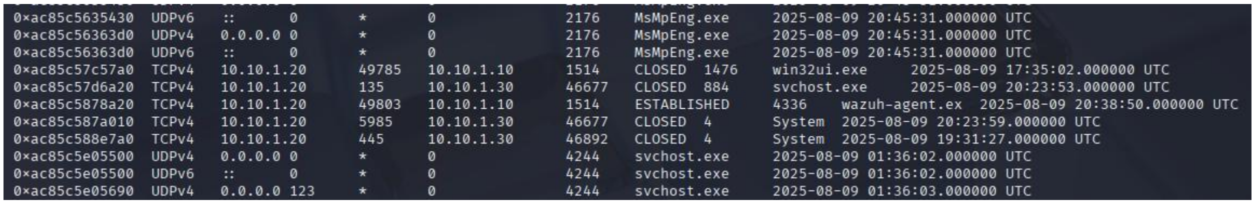


```
┌──(venv)─(kali㉿kali)-[~/volatility3]
└─$ python3 vol.py -f ~/WIN-2LJE5PP82EF-20250809-204455.dmp windows.pslist

Volatility 3 Framework 2.26.2
Progress:  100.00               PDB scanning finished
PID     PPID    ImageFileName   Offset(V)       Threads Handles SessionId       Wow64   CreateTime      ExitTime        File output

4       0       System  0xac85bf899040  120     -       N/A     False   2025-08-07 17:20:01.000000 UTC  N/A     Disabled
100     4       Registry        0xac85bf8df080  4       -       N/A     False   2025-08-07 17:19:54.000000 UTC  N/A     Disabled
308     4       smss.exe        0xac85c30720c0  2       -       N/A     False   2025-08-07 17:20:01.000000 UTC  N/A     Disabled
424     416     csrss.exe       0xac85c2e36140  10      -       0       False   2025-08-07 17:20:02.000000 UTC  N/A     Disabled
524     416     wininit.exe     0xac85c3f85140  1       -       0       False   2025-08-07 17:20:02.000000 UTC  N/A     Disabled
532     516     csrss.exe       0xac85c31ac0c0  11      -       1       False   2025-08-07 17:20:02.000000 UTC  N/A     Disabled
588     516     winlogon.exe    0xac85c35e00c0  5       -       1       False   2025-08-07 17:20:02.000000 UTC  N/A     Disabled
652     524     services.exe    0xac85c21020c0  8       -       0       False   2025-08-07 17:20:02.000000 UTC  N/A     Disabled
668     524     lsass.exe       0xac85c3188080  8       -       0       False   2025-08-07 17:20:02.000000 UTC  N/A     Disabled
776     652     svchost.exe     0xac85c462d240  12      -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
800     524     fontdrvhost.ex  0xac85c4604140  5       -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
808     588     fontdrvhost.ex  0xac85c4606140  5       -       1       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
884     652     svchost.exe     0xac85c467a2c0  11      -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
1004    588     dwm.exe 0xac85c46a4080  16      -       1       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
376     652     svchost.exe     0xac85c47152c0  23      -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
728     652     svchost.exe     0xac85c474f2c0  13      -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
648     652     svchost.exe     0xac85c4758280  24      -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
1036    652     svchost.exe     0xac85c47912c0  21      -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
1204    652     svchost.exe     0xac85c48222c0  20      -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
1300    652     svchost.exe     0xac85c4891240  58      -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
1380    652     svchost.exe     0xac85c48ce240  15      -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
1460    652     svchost.exe     0xac85c48e32c0  20      -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
1536    652     svchost.exe     0xac85c49822c0  3       -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
1576    652     svchost.exe     0xac85c49c62c0  12      -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
1896    652     spoolsv.exe     0xac85bf97b080  7       -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
1908    652     svchost.exe     0xac85c4b07300  4       -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
1964    652     svchost.exe     0xac85c4a9a140  14      -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
1028    652     MpDefenderCore  0xac85c4b4d380  9       -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
1200    652     svchost.exe     0xac85c4a8f140  6       -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
1524    652     svchost.exe     0xac85c4b4b140  5       -       0       False   2025-08-07 17:20:03.000000 UTC  N/A     Disabled
2064    652     VGAuthService.  0xac85c4ba0300  2       -       0       False   2025-08-07 17:20:04.000000 UTC  N/A     Disabled
2088    652     vmtoolsd.exe    0xac85c4ba7280  13      -       0       False   2025-08-07 17:20:04.000000 UTC  N/A     Disabled
2096    652     vm3dservice.ex  0xac85c4ba92c0  3       -       0       False   2025-08-07 17:20:04.000000 UTC  N/A     Disabled
2176    652     MsMpEng.exe     0xac85c4bf9080  26      -       0       False   2025-08-07 17:20:04.000000 UTC  N/A     Disabled
2248    652     wlms.exe        0xac85c4c0b080  2       -       0       False   2025-08-07 17:20:04.000000 UTC  N/A     Disabled
2360    2096    vm3dservice.ex  0xac85c4ccb2c0  4       -       1       False   2025-08-07 17:20:04.000000 UTC  N/A     Disabled
2808    1964    AggregatorHost  0xac85c4f320c0  3       -       0       False   2025-08-09 01:34:00.000000 UTC  N/A     Disabled
2856    652     dllhost.exe     0xac85c4f52280  10      -       0       False   2025-08-09 01:34:00.000000 UTC  N/A     Disabled
2928    776     dllhost.exe     0xac85c4fdc2c0  4       -       0       False   2025-08-09 01:34:00.000000 UTC  N/A     Disabled
2720    776     WmiPrvSE.exe    0xac85c50b7280  11      -       0       False   2025-08-09 01:34:00.000000 UTC  N/A     Disabled
```

Windows.netscan dump:



```
┌──(venv)─(kali㉿kali)-[~/volatility3]
└─$ python3 vol.py -f ~/WIN-2LJE5PP82EF-20250809-204455.dmp windows.netscan

Volatility 3 Framework 2.26.2
Progress:  100.00               PDB scanning finished
Offset  Proto   LocalAddr       LocalPort       ForeignAddr     ForeignPort     State   PID     Owner   Created

0xac85bf8c11b0  TCPv4   0.0.0.0 445     0.0.0.0 0       LISTENING       4       System  2025-08-09 01:33:59.000000 UTC
0xac85bf8c11b0  TCPv6   ::      445     ::      0       LISTENING       4       System  2025-08-09 01:33:59.000000 UTC
0xac85c34a4050  TCPv4   0.0.0.0 49664   0.0.0.0 0       LISTENING       668     lsass.exe       2025-08-07 17:20:03.000000 UTC
0xac85c34a45d0  TCPv4   0.0.0.0 49665   0.0.0.0 0       LISTENING       524     wininit.exe     2025-08-07 17:20:03.000000 UTC
0xac85c34a45d0  TCPv6   ::      49665   ::      0       LISTENING       524     wininit.exe     2025-08-07 17:20:03.000000 UTC
0xac85c34a4cb0  TCPv4   0.0.0.0 135     0.0.0.0 0       LISTENING       884     svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c34a4cb0  TCPv6   ::      135     ::      0       LISTENING       884     svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c34a4e10  TCPv4   0.0.0.0 49665   0.0.0.0 0       LISTENING       524     wininit.exe     2025-08-07 17:20:03.000000 UTC
0xac85c34a5390  TCPv4   0.0.0.0 49664   0.0.0.0 0       LISTENING       668     lsass.exe       2025-08-07 17:20:03.000000 UTC
0xac85c34a5390  TCPv6   ::      49664   ::      0       LISTENING       668     lsass.exe       2025-08-07 17:20:03.000000 UTC
0xac85c34a5910  TCPv4   0.0.0.0 135     0.0.0.0 0       LISTENING       884     svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c47c2260  TCPv4   127.0.0.1       49787   127.0.0.1       49786   CLOSED  3512    net.exe 2025-08-09 17:35:05.000000 UTC
0xac85c47fe310  TCPv4   0.0.0.0 49667   0.0.0.0 0       LISTENING       1300    svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c47fe470  TCPv4   0.0.0.0 49668   0.0.0.0 0       LISTENING       1896    spoolsv.exe     2025-08-07 17:20:03.000000 UTC
0xac85c47fe5d0  TCPv4   0.0.0.0 49667   0.0.0.0 0       LISTENING       1300    svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c47fe5d0  TCPv6   ::      49667   ::      0       LISTENING       1300    svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c47fe890  TCPv4   0.0.0.0 49666   0.0.0.0 0       LISTENING       728     svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c47fe890  TCPv6   ::      49666   ::      0       LISTENING       728     svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c47fe9f0  TCPv4   0.0.0.0 49669   0.0.0.0 0       LISTENING       652     services.exe    2025-08-07 17:20:04.000000 UTC
0xac85c47feb50  TCPv4   0.0.0.0 5985    0.0.0.0 0       LISTENING       4       System  2025-08-09 01:34:00.000000 UTC
0xac85c47feb50  TCPv6   ::      5985    ::      0       LISTENING       4       System  2025-08-09 01:34:00.000000 UTC
0xac85c47fecb0  TCPv4   0.0.0.0 49669   0.0.0.0 0       LISTENING       652     services.exe    2025-08-07 17:20:04.000000 UTC
0xac85c47fecb0  TCPv6   ::      49669   ::      0       LISTENING       652     services.exe    2025-08-07 17:20:04.000000 UTC
0xac85c47fee10  TCPv4   0.0.0.0 49666   0.0.0.0 0       LISTENING       728     svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c47ff0d0  TCPv4   0.0.0.0 3389    0.0.0.0 0       LISTENING       376     svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c47ff4f0  TCPv4   0.0.0.0 3389    0.0.0.0 0       LISTENING       376     svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c47ff4f0  TCPv6   ::      3389    ::      0       LISTENING       376     svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c47ff910  TCPv4   0.0.0.0 47001   0.0.0.0 0       LISTENING       4       System  2025-08-09 01:33:59.000000 UTC
0xac85c47ff910  TCPv6   ::      47001   ::      0       LISTENING       4       System  2025-08-09 01:33:59.000000 UTC
0xac85c47ffbd0  TCPv4   0.0.0.0 49668   0.0.0.0 0       LISTENING       1896    spoolsv.exe     2025-08-07 17:20:03.000000 UTC
0xac85c47ffbd0  TCPv6   ::      49668   ::      0       LISTENING       1896    spoolsv.exe     2025-08-07 17:20:03.000000 UTC
0xac85c485ad20  UDPv4   0.0.0.0 3389    *       0               376     svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c485b4f0  UDPv4   0.0.0.0 3389    *       0               376     svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c485b4f0  UDPv6   ::      3389    *       0               376     svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c4a3b740  UDPv4   0.0.0.0 4500    *       0               1300    svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c4a3c3c0  UDPv4   0.0.0.0 500     *       0               1300    svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c4a3c3c0  UDPv6   ::      500     *       0               1300    svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c4a3ca00  UDPv4   0.0.0.0 4500    *       0               1300    svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c4a3ca00  UDPv6   ::      4500    *       0               1300    svchost.exe     2025-08-07 17:20:03.000000 UTC
0xac85c4a3d4f0  UDPv4   0.0.0.0 0       *       0               1300    svchost.exe     2025-08-07 17:20:03.000000 UTC
```



```
0xac85c5635430  UDPv6   ::      0       *       0               2176    MsMpEng.exe     2025-08-09 20:45:31.000000 UTC
0xac85c56363d0  UDPv4   0.0.0.0 0       *       0               2176    MsMpEng.exe     2025-08-09 20:45:31.000000 UTC
0xac85c56363d0  UDPv6   ::      0       *       0               2176    MsMpEng.exe     2025-08-09 20:45:31.000000 UTC
0xac85c57c57a0  TCPv4   10.10.1.20      49785   10.10.1.10      1514    CLOSED  1476    win32ui.exe     2025-08-09 17:35:02.000000 UTC
0xac85c57d6a20  TCPv4   10.10.1.20      135     10.10.1.30      46677   CLOSED  884     svchost.exe     2025-08-09 20:23:53.000000 UTC
0xac85c5878a20  TCPv4   10.10.1.20      49803   10.10.1.10      1514    ESTABLISHED     4336    wazuh-agent.ex  2025-08-09 20:38:50.000000 UTC
0xac85c587a010  TCPv4   10.10.1.20      5985    10.10.1.30      46677   CLOSED  4       System  2025-08-09 20:23:59.000000 UTC
0xac85c588e7a0  TCPv4   10.10.1.20      445     10.10.1.30      46892   CLOSED  4       System  2025-08-09 19:31:27.000000 UTC
0xac85c5e05500  UDPv4   0.0.0.0 0       *       0               4244    svchost.exe     2025-08-09 01:36:02.000000 UTC
0xac85c5e05500  UDPv6   ::      0       *       0               4244    svchost.exe     2025-08-09 01:36:02.000000 UTC
0xac85c5e05690  UDPv4   0.0.0.0 123     *       0               4244    svchost.exe     2025-08-09 01:36:03.000000 UTC
```

We can see the tcp scan done by 10.10.1.30 (kali) to our windows server (10.10.1.20).

Furthermore, we can also notice that a connection was made by wazuh-agent.ex to 10.10.1.10 (SIEM) , confirming logs we sent in real-time to wazuh server.

9. Conclusion

This project successfully met all the objectives of the CST8808 Final Project by building a functional incident detection and response environment for CSA271.com. Using Wazuh SIEM integrated with Snort IDS, we detected and logged all four required Indicators of Compromise. Brute force login attempts, SYN scans, TCP scans, and UDP scans.

File Integrity Monitoring ensured that unauthorized changes to key web files were immediately flagged, and memory analysis with Volatility confirmed in-memory evidence of the attacks.

Despite resource constraints that prevented us from running a fully dedicated machine for each role, the lab environment was carefully configured to mimic real-world operations while maintaining performance. This allowed us to validate log forwarding, alert generation, and correlation within the SIEM dashboard under realistic attack conditions.

In the end, this project proved that with the right planning and configuration, open-source tools like Wazuh and Snort can deliver robust, enterprise-level security monitoring and incident response without the cost of commercial licenses. Making them both practical and powerful for organizations with limited budgets.

10. References

1. **Wazuh Documentation** – *Installation, configuration, and integration guides*
   Wazuh, Inc. (2025). *Wazuh documentation*. Retrieved from:
   https://documentation.wazuh.com/

2. **Snort Official User Manual** – *Snort configuration, rule writing, and best practices*
   Cisco Systems, Inc. (2025). *Snort 2.x User Manual*. Retrieved from:
   https://www.snort.org/documents

3. **Volatility 3 Framework** – *Memory forensics tool usage*
   Volatility Foundation. (2025). *Volatility 3 documentation*. Retrieved from:
   https://volatility3.readthedocs.io/

4. **Microsoft Windows Server 2016 Documentation** – *Networking, IIS, and NTP setup*
   Microsoft Corporation. (2025). *Windows Server documentation*. Retrieved from:
   https://learn.microsoft.com/en-us/windows-server/

5. **National Institute of Standards and Technology (NIST)** – *Incident Response best practices*
   Cichonski, P., Millar, T., Grance, T., & Scarfone, K. (2012). *Computer Security Incident Handling Guide* (NIST SP 800-61 Rev. 2).
   https://doi.org/10.6028/NIST.SP.800-61r2

6. **Kali Linux Official Documentation** – *Penetration testing and network scanning tools*
   Offensive Security. (2025). *Kali Linux documentation*. Retrieved from:
   https://www.kali.org/docs/

7. **Open Source Security (OSSEC)** – *Log-based intrusion detection concepts*
   Trend Micro, Inc. (2025). *OSSEC documentation*. Retrieved from:
   https://www.ossec.net/docs/