National College of Ireland

Configuration Manual

MSc Research Project
Cybersecurity

Siddhesh Subhash Aher
Student ID: x23297867

School of Computing
National College of Ireland

Supervisor: Khadija Hafeez

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name:	Siddhesh Subhash Aher
Student ID:	X23297867
Programme:	Masters in Cybersecurity **Year:** 2024-25
Module:	Practicum
Lecturer:	Khadija Hafeez
Submission Due Date:	11/08/2025
Project Title:	Configuration Manual
Word Count:	1124 **Page Count:** 12

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.
ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature:	Siddhesh Subhash Aher
Date:	11/08/2025

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Design and Implementation of a Behavior based Trust Engine for Insider Threat Detection in Zero Trust Environments

Configuration Manual

Siddhesh Subhash Aher

x23297867

1 Introduction

This guide is a full step-by-step tutorial on setting up, configuring and testing the Trust Engine project. It includes a description of hardware, virtual machine (VM) configuration, lab network details, installation of the Wazuh server as well as agents, simulating an attack, receiving alerts and running Trust Engine and verifying the output.

2 Hardware Requirements

- Processor: Quad core 2.5 GHz or higher
- RAM: Minimum 8 GB (16 GB recommended)
- Storage: At least 50 GB free disk space
- Network: NAT Network

3 Virtual Machine Setup

1) Download the softwares

   a) Virtualbox : Get the latest version of Virtualbox from this link https://www.virtualbox.org/wiki/Downloads (Oracle, 2025)
   b) Ubuntu server ISO (22.04 LTS) : Download the current ubuntu server from the official site https://ubuntu.com/download/server (Canonical, 2025)
   c) Kali Linux : Download the Kali Linux VM ISO image from the official link https://www.kali.org/get-kali/#kali-platforms (Offensive Security, 2025)
   d) WAZUH OVA: Download the official Wazuh OVA (includes Wazuh Manager/Indexer/Dashboard 4.12.x; base OS: Amazon Linux 2023). https://documentation.wazuh.com/current/deployment-options/virtual-machine/virtual-machine.html (Wazuh Inc., 2025a)

2) Install the Virtualbox and import the downloaded VM images and server

a) Installed Oracle VirtualBox (latest version) along with the Extension Pack on the host machine. (Oracle, 2025)
b) Downloaded the required VM images from official sources:
c) Wazuh OVA (Wazuh Inc., 2025a)
d) Ubuntu Server ISO as it will act as main server and Trust Engine host. (Canonical, 2025).
e) Kali Linux VirtualBox Image it will act as client machine that is insider. (Offensive Security, 2025)
f) Imported the images into VirtualBox:
g) Wazuh OVA → File → Import Appliance → selected .ova file.
h) Kali Linux → Extracted .7z archive → Machine → Add → selected .vbox file.
i) Ubuntu Server → Created new VM → attached downloaded .iso for installation.
a) Started all three machines successfully and confirmed they boot without errors.

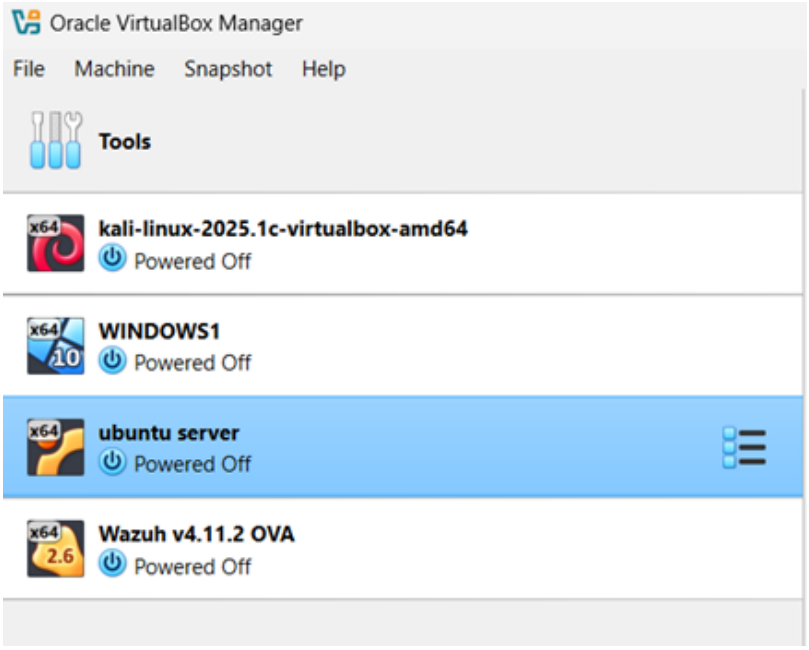

Figure 2: VM Setup

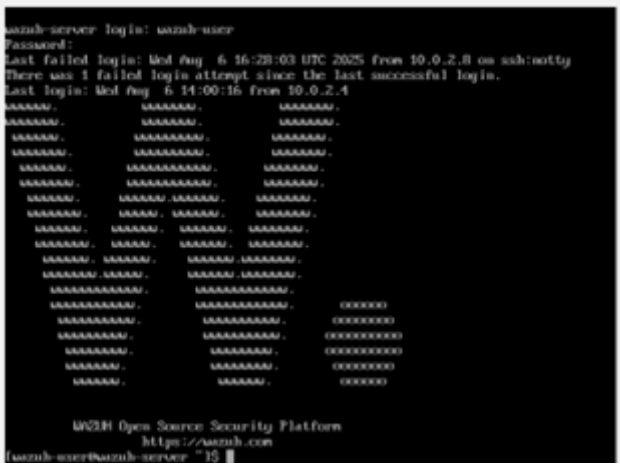

Figure 1: Wazuh Server Running

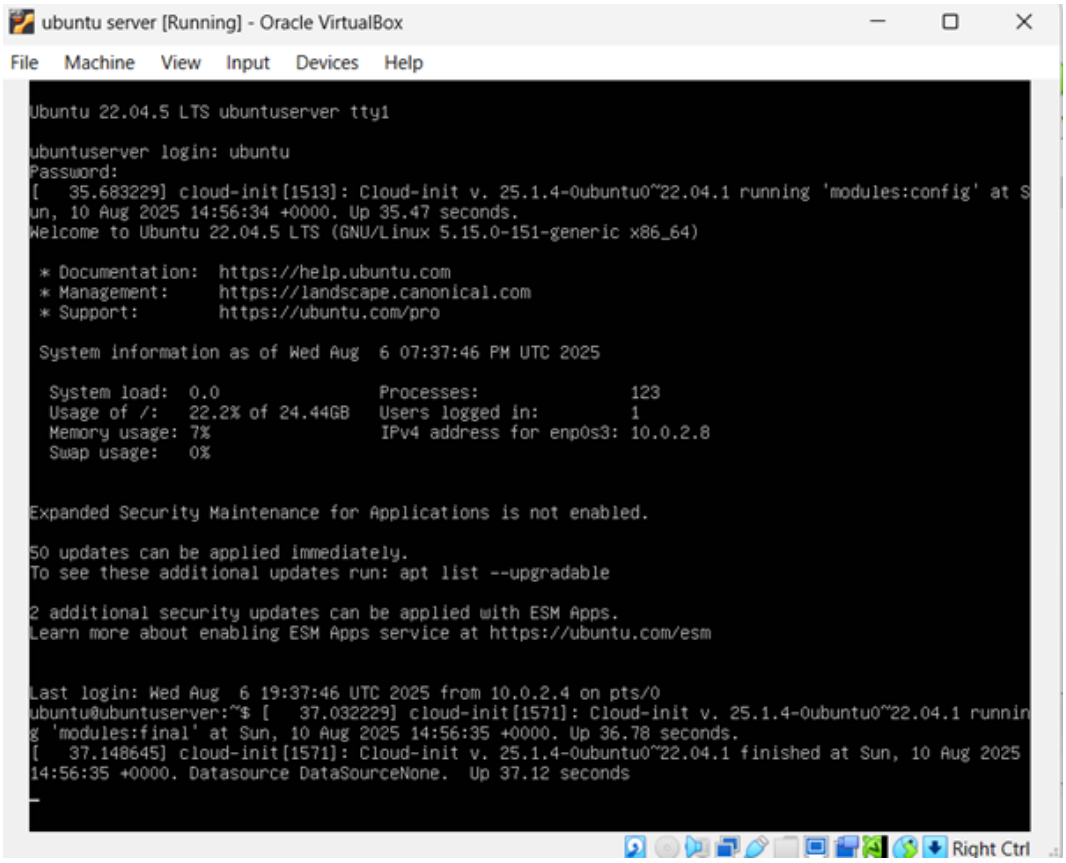

Figure 3: Ubuntu Running Succesfully

4 Setup a Network

1) Create NAT Network in VirtualBox (Oracle, 2025)
    a) Open VirtualBox.
    b) Go to File → Tools → Network Manager → NAT Networks.
    c) Click Create.
    d) Set:
    e) Name: NATnetwork1
    f) Network CIDR: 10.0.2.0/24
    g) Click OK to save.

2) Attatch each VM to the NAT network
    Repeat for each VM (**Wazuh OVA**, **Ubuntu Trust Engine**, **Kali Attacker**):
    a) Right-click the VM → **Settings** → **Network**.
    b) Under **Adapter 1**, select:
        a. **Attached to:** NAT Network
        b. **Name:** NATnetwork1

c) Click **OK**.

3) Ping all machines with each other IP to ensure their connectivity.

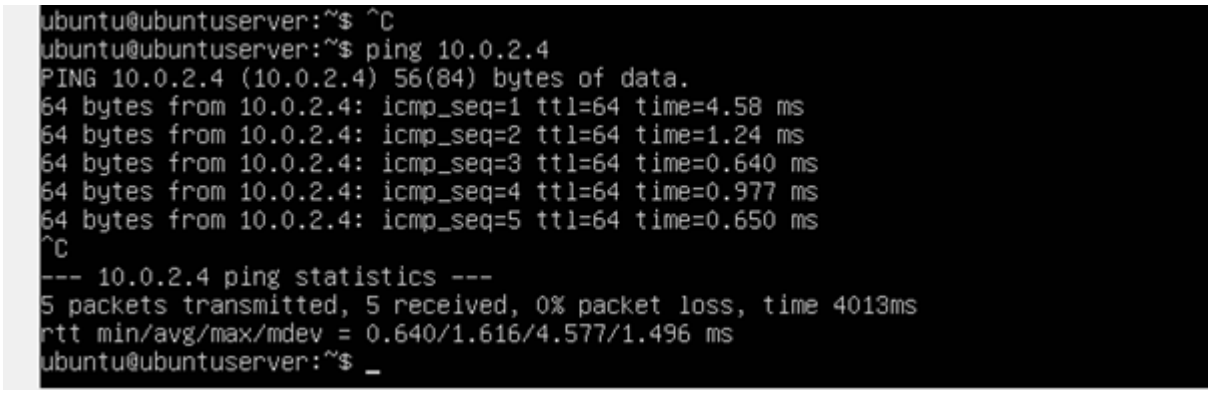

Figure 4: Ubuntu and Kali Connected

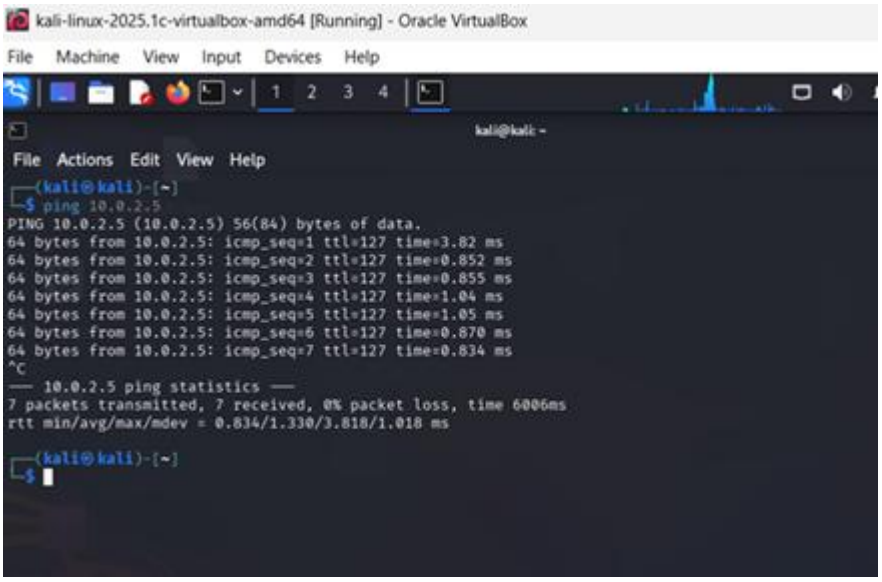

Figure 5: Wazuh and Kali connected

5 START WAZUH SERVER

1) Select the **Wazuh-Manager** VM and click **Start**.
2) At the login prompt:
   login: wazuh-user
   password: wazuh
3) Verify Wazuh services are running

   sudo systemctl status wazuh-manager

4) It its not running then start it

sudo systemctl start wazuh-manager

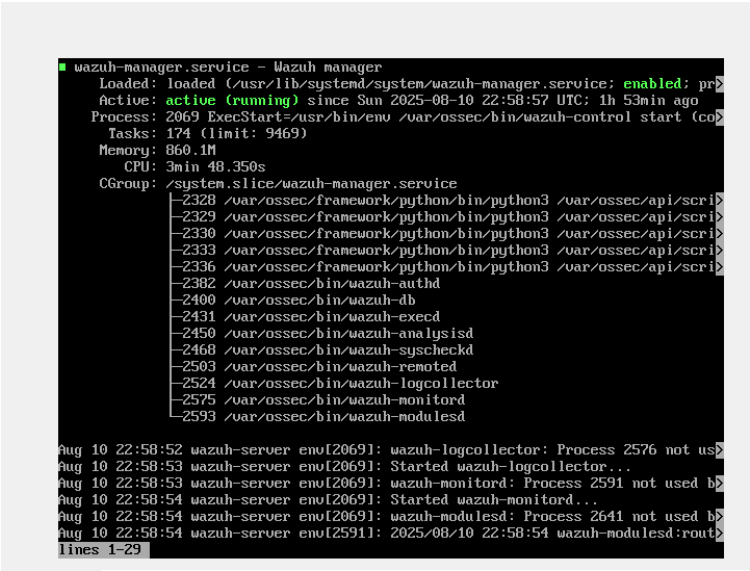

Figure 6: Wazuh Manager Service Running

5) Access Wazuh Dashboard from Kali Browser
   a) From **Kali Linux (10.0.2.4)**:
      Open Firefox/Chromium.
   b) Enter the Wazuh dashboard URL: https://10.0.2.5
   c) You'll see a certificate warning (self-signed).
   d) Accept the risk and continue.
   e) Log in with the dashboard credentials:
      User: admin
      Password: admin

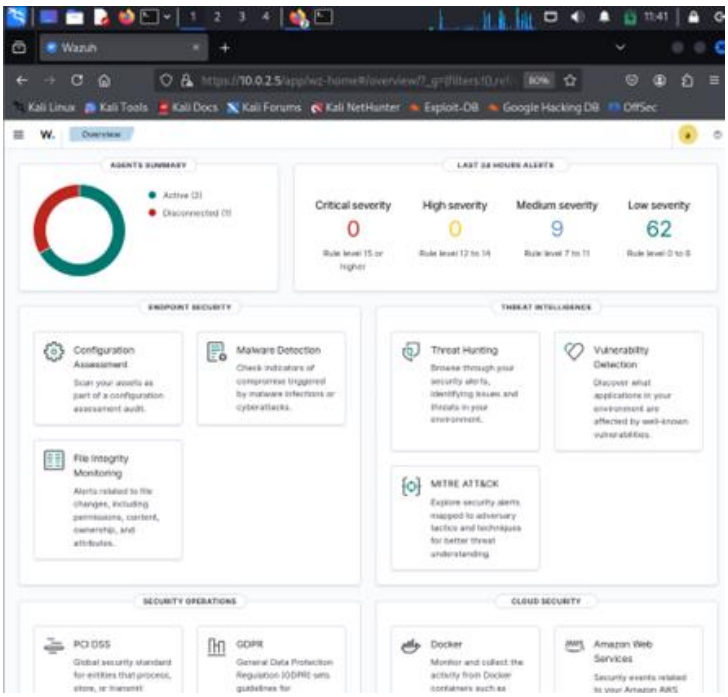

Figure 7: Wazuh Accessible through Dashboard

6 INSTALL WAZUH AGENT

1) Install agent on Kali Linux

a) Open the Wazuh Dashboard from Kali Browser https://10.0.2.5

b) Log in with **admin** credentials.

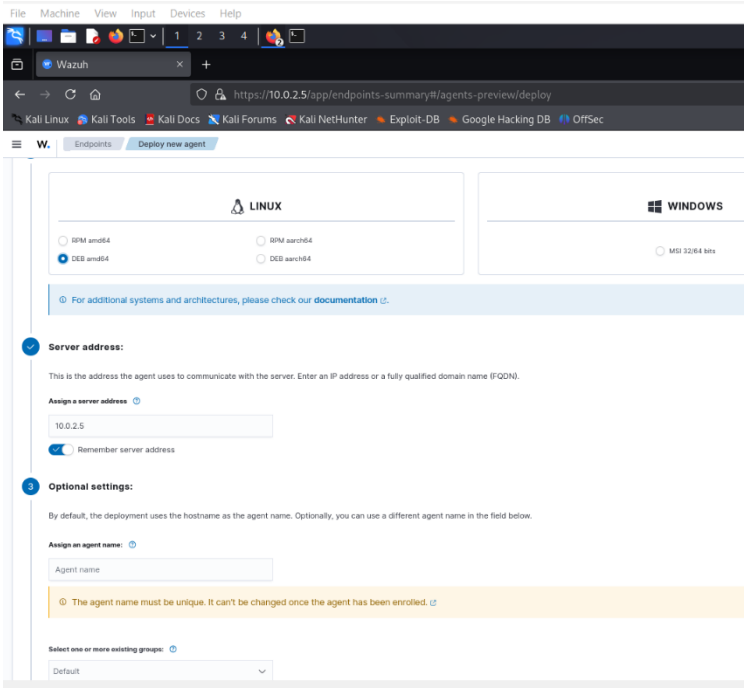

Figure 8: Setup to install agent

c) Navigate to:

Menu → Agents → Deploy new agent

d) Select:

Operating system: Linux

Package: Debian

e) Enter:

Wazuh Manager IP: 10.0.2.5

Agent name: Kali

f) The GUI will now display a **pre-generated installation script**

```
curl -so wazuh-agent.deb https://packages.wazuh.com/4.x/apt/wazuh-agent_4.x.x-1_amd64.deb \
&& sudo dpkg -i ./wazuh-agent.deb \
&& sudo WAZUH_MANAGER='10.0.2.5' WAZUH_AGENT_NAME='Kali' /var/ossec/bin/agent-auth -m 10.0.2.5 \
&& sudo systemctl enable wazuh-agent \
&& sudo systemctl start wazuh-agent
```

g) Copy the script from the Wazuh Dashboard and paste it into Kali's terminal

```
┌──(kali㉿kali)-[~]
└─$ sudo systemctl status wazuh-agent.service
[sudo] password for kali:
● wazuh-agent.service - Wazuh agent
     Loaded: loaded (/usr/lib/systemd/system/wazuh-agent.service; enabled; preset: disabled)
     Active: active (running) since Sun 2025-08-10 18:55:05 EDT; 2h 3min ago
 Invocation: dae2bbb120364e66957346d13b585e13
    Process: 767 ExecStart=/usr/bin/env /var/ossec/bin/wazuh-control start (code=exited, status=0/SUCCESS)
      Tasks: 32 (limit: 2210)
     Memory: 59.3M (peak: 1G, swap: 5M, swap peak: 5M)
        CPU: 1min 23.002s
     CGroup: /system.slice/wazuh-agent.service
             ├─812 /var/ossec/bin/wazuh-execd
             ├─820 /var/ossec/bin/wazuh-agentd
             ├─836 /var/ossec/bin/wazuh-syscheckd
             ├─849 /var/ossec/bin/wazuh-logcollector
             └─918 /var/ossec/bin/wazuh-modulesd

Aug 10 18:55:00 kali env[767]: Deleting PID file '/var/ossec/var/run/wazuh-syscheckd-847.pid' not used...
Aug 10 18:55:00 kali env[767]: Deleting PID file '/var/ossec/var/run/wazuh-agentd-828.pid' not used...
Aug 10 18:55:00 kali env[767]: Deleting PID file '/var/ossec/var/run/wazuh-execd-820.pid' not used...
Aug 10 18:55:00 kali env[767]: Started wazuh-execd...
Aug 10 18:55:01 kali env[767]: Started wazuh-agentd...
Aug 10 18:55:01 kali env[767]: Started wazuh-syscheckd...
Aug 10 18:55:02 kali env[767]: Started wazuh-logcollector...
Aug 10 18:55:03 kali env[767]: Started wazuh-modulesd...
Aug 10 18:55:05 kali env[767]: Completed.
Aug 10 18:55:05 kali systemd[1]: Started wazuh-agent.service - Wazuh agent.
```

Figure 9: Agent installed and active on Kali

h) Wait for the installation to complete and for the agent to start

2) Install Agent on Ubuntu server (Trustengine Host)

a) Same as of Kali Installation

```
ubuntu@ubuntuserver:~/trust-engine$ sudo systemctl status wazuh-agent.service
● wazuh-agent.service - Wazuh agent
     Loaded: loaded (/lib/systemd/system/wazuh-agent.service; enabled; vendor preset: enabled)
     Active: active (running) since Sun 2025-08-10 23:02:51 UTC; 1h 58min ago
    Process: 690 ExecStart=/usr/bin/env /var/ossec/bin/wazuh-control start (code=exited, status=0/SUCCESS)
      Tasks: 35 (limit: 7221)
     Memory: 656.6M
        CPU: 55.143s
     CGroup: /system.slice/wazuh-agent.service
             ├─783 /var/ossec/bin/wazuh-execd
             ├─791 /var/ossec/bin/wazuh-agentd
             ├─838 /var/ossec/bin/wazuh-syscheckd
             ├─871 /var/ossec/bin/wazuh-logcollector
             └─965 /var/ossec/bin/wazuh-modulesd

Aug 10 23:02:44 ubuntuserver env[690]: Deleting PID file '/var/ossec/var/run/wazuh-syscheckd-838.pid' not used...
Aug 10 23:02:44 ubuntuserver env[690]: Deleting PID file '/var/ossec/var/run/wazuh-agentd-804.pid' not used...
Aug 10 23:02:44 ubuntuserver env[690]: Deleting PID file '/var/ossec/var/run/wazuh-execd-795.pid' not used...
Aug 10 23:02:44 ubuntuserver env[690]: Started wazuh-execd...
Aug 10 23:02:45 ubuntuserver env[690]: Started wazuh-agentd...
Aug 10 23:02:46 ubuntuserver env[690]: Started wazuh-syscheckd...
Aug 10 23:02:47 ubuntuserver env[690]: Started wazuh-logcollector...
Aug 10 23:02:49 ubuntuserver env[690]: Started wazuh-modulesd...
Aug 10 23:02:51 ubuntuserver env[690]: Completed.
Aug 10 23:02:51 ubuntuserver systemd[1]: Started Wazuh agent.
```

Figure 10: Agent installed and active on Ubuntu

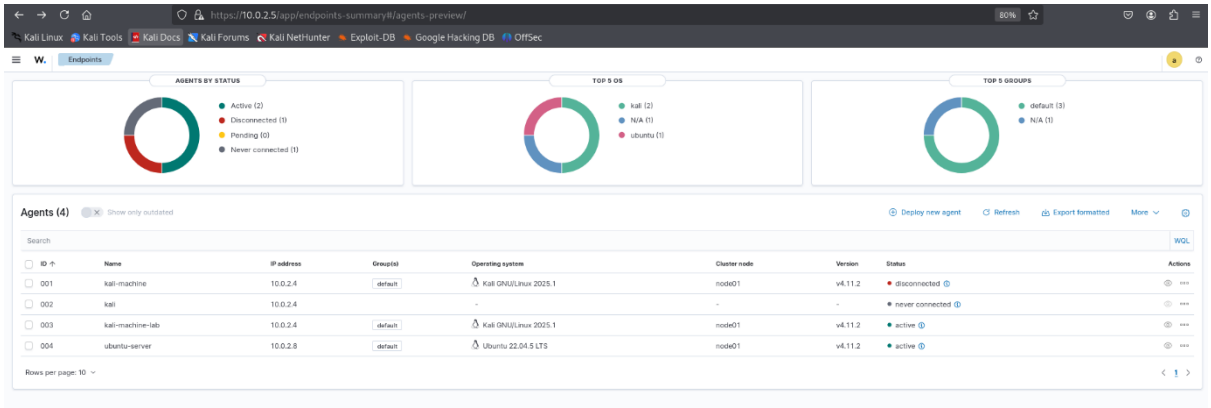

Figure 11: Agents verified and receiving logs

7 CONFIGURATION TO DETECT AND BLOCK SSH BRUTE FORCE

1) Create the rule in Wazuh that is local_rules.xml for detecting ssh
2) Edit ossec.conf on Wazuh Manager (Wazuh Inc., 2025b) (van Hauser and The Hacker's Choice, 2025)

   sudo nano /var/ossec/etc/ossec.conf

3) Inside the <ossec_config> block, find or create an <active-response> section for SSH brute force.

```
<active-response>
  <command>firewall-drop</command>
  <location>local</location>
  <rules_id>5710</rules_id>
  <timeout>60</timeout>
</active-response>
```

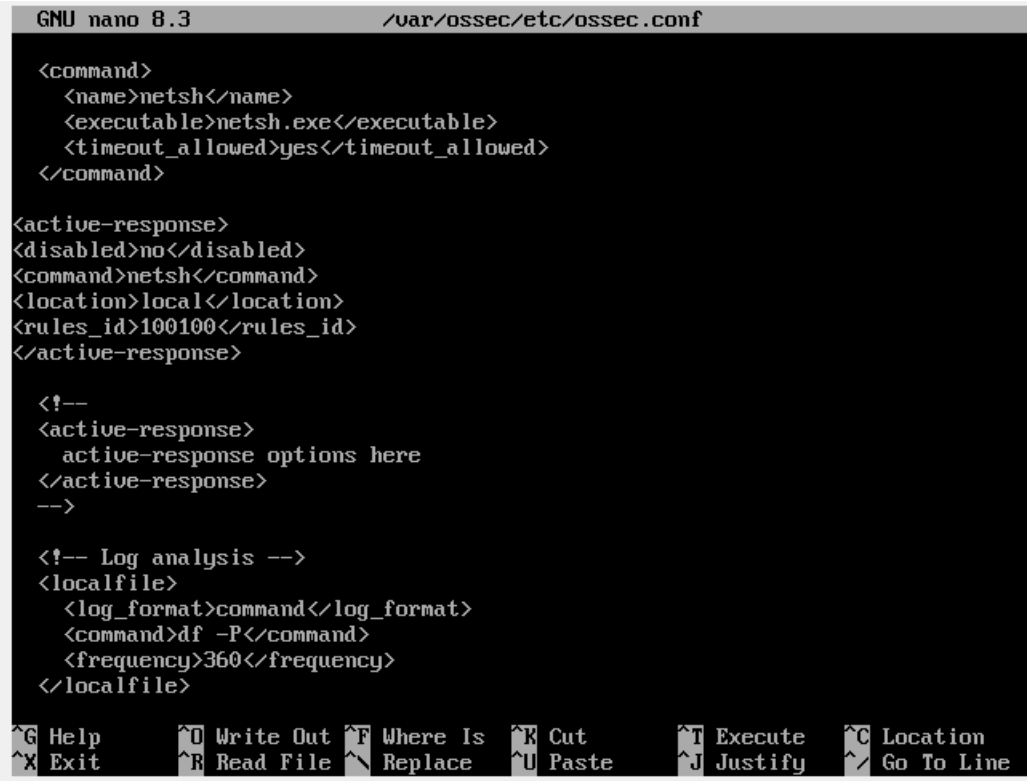

Figure 12: Changes in .conf file

4) Restart Wazuh Manager and Agent
5) Test the SSH Brute Force Detection
   From Kali Attacker (10.0.2.4)

ssh ubuntu@10.0.2.8

type wrong passwords to let the alert trigger

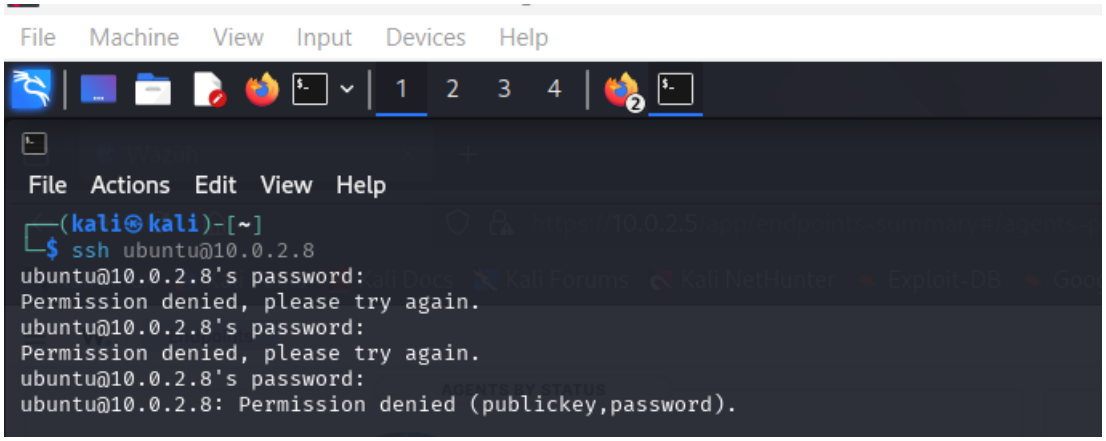

Figure 13: Dummy SSH attack

6) Fetch Alerts for Trust Engine Processing

Before simulating the attack, ensure the **Ubuntu Trust Engine** host has the latest alerts file from the Wazuh Manager.

From the **Ubuntu Trust Engine** VM:

scp wazuh-user@10.0.2.5:/home/wazuh-user/shared-logs/alerts.json ~/trust-engine/alerts.json

Copies the alerts.json file from the Wazuh Manager's shared-logs directory.

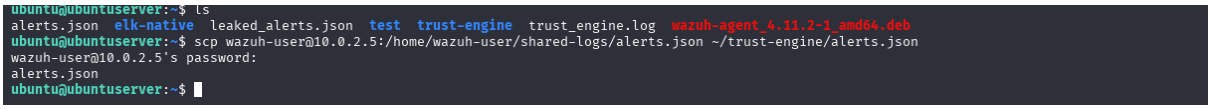

Figure 14: Syncing alerts

7) Run the trust engine on Ubuntu Server

sudo python3 trust-engine/trust_engine_from_json.py

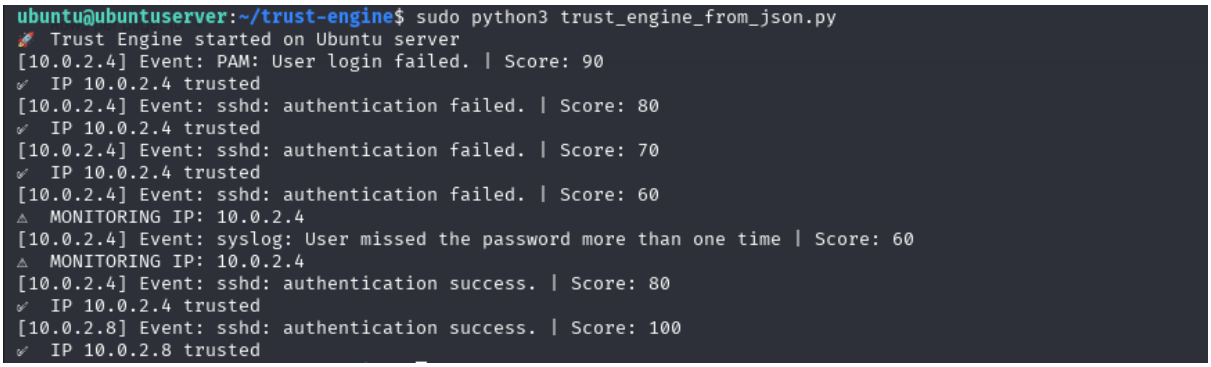

Figure 15: Result after execution of Trust Engine

8) Check the Ubuntu server IP tables there you will get the list of IP's which are blocked

sudo iptables -L

```
ubuntu@ubuntuserver:~$ sudo iptables -L INPUT -n --line-numbera
[sudo] password for ubuntu:
iptables v1.8.7 (nf_tables): unknown option "--line-numbera"
Try `iptables -h' or 'iptables --help' for more information.
ubuntu@ubuntuserver:~$ sudo iptables -L INPUT -n --line-numbers
Chain INPUT (policy ACCEPT)
num  target     prot opt source               destination
1    DROP       all  --  10.0.2.4             0.0.0.0/0
```

Figure 16: IP block succesffuly in IP tables

8 CONFIGURATION TO DETECT AND BLOCK SENSITIVE FILE ACCESS

1) Set the customized rule in wazuh local rules
2) Edit the ossec.conf file
   Add

<directories realtime="yes" report_changes="yes" check_all="yes" whodata="yes">/trust-engine</directories>

3) Restart Wazuh Manager and Agent
4) Try to access trust engine file in the trust engine directory
5) Fetch the generated alerts in wazuh with file in ubuntu server
   scp wazuh-user@10.0.2.5:/home/wazuh-user/shared-logs/alerts.json ~/trust-engine/alerts.json

```
ubuntu@ubuntuserver:~/trust-engine$ sudo nano trust_engine_from_json.py
ubuntu@ubuntuserver:~/trust-engine$ scp wazuh-user@10.0.2.5:/home/wazuh-user/shared-logs/alerts.json ~/trust-engine
/alerts.json
wazuh-user@10.0.2.5's password:
alerts.json                                   100%   16KB   2.9MB/s   00:00
```

Figure 17: Sensitive file accessed

```
🚀 Trust Engine started on Ubuntu server
[10.0.2.8] Event: sshd: authentication success. | Rule ID: 5715 | Score: 100
✔ IP 10.0.2.8 trusted
[10.0.2.8] Event: PAM: User login failed. | Rule ID: 5503 | Score: 90
✔ IP 10.0.2.8 trusted
[10.0.2.8] Event: sshd: authentication failed. | Rule ID: 5760 | Score: 80
✔ IP 10.0.2.8 trusted
[10.0.2.8] Event: sshd: authentication success. | Rule ID: 5715 | Score: 100
✔ IP 10.0.2.8 trusted
[10.0.2.4] Event: sshd: authentication success. | Rule ID: 5715 | Score: 100
✔ IP 10.0.2.4 trusted
[10.0.2.4] Event: Integrity checksum changed. | Rule ID: 550 | Score: 90
✔ IP 10.0.2.4 trusted
[10.0.2.4] Event: Integrity checksum changed. | Rule ID: 550 | Score: 80
✔ IP 10.0.2.4 trusted
[10.0.2.4] Event: Integrity checksum changed. | Rule ID: 550 | Score: 70
✔ IP 10.0.2.4 trusted
[10.0.2.4] Event: Integrity checksum changed. | Rule ID: 550 | Score: 60
⚠ MONITORING IP: 10.0.2.4
[10.0.2.4] Event: Integrity checksum changed. | Rule ID: 550 | Score: 50
⚠ MONITORING IP: 10.0.2.4
[10.0.2.8] Event: sshd: authentication success. | Rule ID: 5715 | Score: 100
✔ IP 10.0.2.8 trusted
```

Figure 18: Trust Score after running trust engine

6) Run the trust engine to get desired output
7) Check the IP tables to get the status of blocked IP
   sudo iptables -L

```
ubuntu@ubuntuserver:~$ sudo iptables -L INPUT -n --line-numbera
[sudo] password for ubuntu:
iptables v1.8.7 (nf_tables): unknown option "--line-numbera"
Try `iptables -h' or 'iptables --help' for more information.
ubuntu@ubuntuserver:~$ sudo iptables -L INPUT -n --line-numbers
Chain INPUT (policy ACCEPT)
num  target     prot opt source               destination
1    DROP       all  --  10.0.2.4             0.0.0.0/0
```

Figure 17: IP blocked

9 REFERENCES

Oracle. (2025) Oracle VM VirtualBox Downloads. Available at: https://www.virtualbox.org/wiki/Downloads (Accessed: 11 August 2025).

Canonical. (2025) Ubuntu Server Downloads. Available at: https://ubuntu.com/download/server (Accessed: 11 August 2025).

Offensive Security. (2025) Get Kali Linux – Official Kali Linux Downloads. Available at: https://www.kali.org/get-kali/#kali-platforms (Accessed: 11 August 2025).

Wazuh Inc. (2025) Deploying the Wazuh OVA. Available at: https://documentation.wazuh.com/current/deployment-options/virtual-machine/virtual-machine.html (Accessed: 11 August 2025).

Wazuh Inc. (2025) Active Response. Available at: https://documentation.wazuh.com/current/user-manual/capabilities/active-response/index.html (Accessed: 11 August 2025).

Wazuh Inc. (2025) File Integrity Monitoring. Available at: https://documentation.wazuh.com/current/user-manual/capabilities/file-integrity/index.html (Accessed: 11 August 2025).

Oracle. (2025) Virtual Networking – NAT Networks. Available at: https://www.virtualbox.org/manual/ch06.html (Accessed: 11 August 2025).

van Hauser, T. and The Hacker's Choice (2025) THC-Hydra – Online Password Cracking Tool. Available at: https://github.com/vanhauser-thc/thc-hydra (Accessed: 11 August 2025).