

Wazuh SIEM Setup & Configuration

Introduction

This project entails the implementation of a **Security Information and Event Management (SIEM)** system to gain hands-on experience with **log collection, analysis, and threat detection**. The objective is to understand how SIEM tools operate and how they support **security monitoring and incident response**.

What is a SIEM?

A SIEM is a security platform that helps organizations **detect, analyze, and respond to cybersecurity threats in real-time** by collecting and correlating logs from across their IT environment.

The SIEM used in this project is Wazuh.

What is Wazuh?

Wazuh is a **free, open-source SIEM platform** that helps monitor and protect systems by collecting, analyzing, and visualizing security data.

Why Wazuh?

Wazuh is ideal because it's a **powerful, cost-free SIEM** that offers **real-time threat detection**, log analysis, and comprehensive security monitoring—making it perfect for **learning environments, labs, and small-scale deployments**.

System Requirements

The following resources were used to set up the Wazuh SIEM lab environment:

- **Virtualization Platform:** VMware Workstation Pro
- **Operating System:** Ubuntu Server (latest LTS version)
- **Virtual Machine Specs:**
 - **RAM:** 4 GB
 - **Storage:** 40 GB
 - **CPU:** 2 cores (recommended minimum)
- **Internet Access:** Required for installing Wazuh components and updates

Wazuh Server Installation

Installing Ubuntu Server

- Downloaded the latest **Ubuntu Server LTS ISO**.
- Created a new virtual machine in **VMware Workstation Pro** with:
 - 4 GB RAM
 - 40 GB storage
 - 2 CPU cores
- Attached the ISO and followed the guided installation process.
- Assigned a **static IP address** during or after installation for consistent connectivity.

Installing Wazuh

Executed the following commands in the Ubuntu terminal:

```
curl -sO https://packages.wazuh.com/4.7/wazuh-install.sh
```

```
chmod +x wazuh-install.sh
```

```
sudo ./wazuh-install.sh -a --ignore-check
```

- The -a flag installs all components (Wazuh manager, dashboard, etc.).
- The --ignore-check flag bypasses the OS check to allow running on Ubuntu Server.
- The installation took approximately **10–20 minutes**.

Accessing the Wazuh Dashboard

- Once installed, accessed the Wazuh dashboard via browser at: **https://<Wazuh-Server-IP>**
- Used the default credentials provided during setup:
 - **Username:** admin
 - **Password:** (found in ~/wazuh-install-files/passwords.txt)
- Upon first login, verified the dashboard was active and functional.


Windows Agent Deployment


After successful setup of wazuh, I decided to deploy the first agent on my host computer (windows) to test the functionalities of the SIEM. I followed the following steps:


1. **Package selection** – first, the target OS is selected in order for a command to be generated for installation of the wazuh agent.

Deploy new agent

1 Select the package to download and install on your system:

**LINUX**
☐ RPM amd64 ☐ RPM aarch64
☐ DEB amd64 ☐ DEB aarch64

**WINDOWS**
☐ MSI 32/64 bits

**macOS**
☐ Intel
☐ Apple silicon

[For additional systems and architectures, please check our documentation.](#)

2. **Server address selection** – this step involves inputting the IP address of the wazuh server.

2 Server address:

This is the address the agent uses to communicate with the server. Enter an IP address or a fully qualified domain name (FDQN).

Assign a server address: [?](#)

3. **Optional settings** - this step involves creating a name for the agent.

3 Optional settings:

By default, the deployment uses the hostname as the agent name. Optionally, you can use a different agent name in the field below.

Assign an agent name: [?](#)

[?](#) The agent name must be unique. It can't be changed once the agent has been enrolled. [?](#)

4. **Installation command** – this step involves obtaining the command that installs the agent on the target machine. This command is inserted in the powershell with administrator clearance.

4 Run the following commands to download and install the agent:

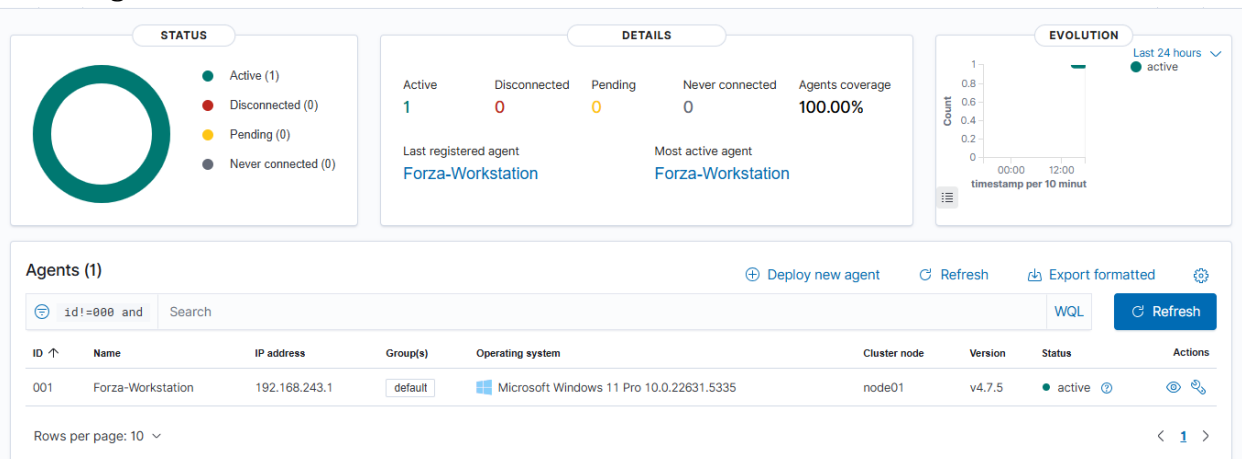
```
Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.7.5-1.msi -OutFile  
${env.tmp}\wazuh-agent; msexec.exe /i ${env.tmp}\wazuh-agent /q WAZUH_MANAGER='192.168.1.1'  
WAZUH_AGENT_NAME='forza' WAZUH_REGISTRATION_SERVER='192.168.1.1'
```

Requirements

- You will need administrator privileges to perform this installation.
- PowerShell 3.0 or greater is required.

Keep in mind you need to run this command in a Windows PowerShell terminal.

5. **Start the agent** – this is the final step. The command NET START WazuhSvc is used to start the agent. Upon successful startup, the device will appear on the dashboard under 'agents'



Conclusion

This was a very informative project. It was fairly simple as there are numerous tutorials online. I look forward to interacting with this system further to maximize its functionalities.

Potential future projects with wazuh

1. CIS Benchmark Hardening

Use Wazuh's CIS module to evaluate system compliance and gradually increase your CIS security score.

2. **Custom Rule Creation**

Develop custom detection rules to identify specific attack patterns or unusual system behavior.

3. **File Integrity Monitoring Expansion**

Monitor additional directories across multiple endpoints, including sensitive config files and scripts.

4. **VirusTotal Integration Testing**

Perform detailed testing using known malicious files and observe VirusTotal responses in Wazuh.

5. **Integration with Suricata or Zeek**

Link Wazuh to network intrusion detection tools to analyze packet-level traffic and correlate alerts.

6. **Multi-Endpoint Deployment**

Expand the lab environment to include more Windows, Linux, or cloud-based endpoints and observe centralized log management.

7. **Automated Threat Response**

Test and implement Active Response rules that trigger actions like blocking IPs or stopping malicious processes.

8. **Scheduled Reporting**

Configure automated email or dashboard reports summarizing alert trends, compliance issues, or system changes.

9. **Cloud Monitoring Simulation**

Connect virtual cloud instances or simulate a cloud workload to understand how Wazuh handles cloud log monitoring.

10. **SIEM + SOAR Integration**

Experiment with integrating Wazuh into a broader security stack including SOAR platforms like TheHive or Cortex for incident management.