



# File Integrity Monitoring (FIM) with Wazuh

Prepared by: [Arshman Abbas](#)

*Air University Islamabad*

# Objective

The objective of this project is to implement File Integrity Monitoring (FIM) using Wazuh to detect unauthorized file modifications, additions, or deletions across monitored endpoints. This project aims to demonstrate the ability to configure, manage, and analyze integrity alerts in a real-world security monitoring environment. By deploying Wazuh, the goal is to ensure data integrity, incident visibility, and compliance readiness through automated log analysis and real-time alerts.

## What is Wazuh?

Wazuh is an open-source security monitoring platform designed to detect threats, monitor system integrity, and ensure regulatory compliance. It functions as a SIEM solution by collecting, parsing, and analyzing data from both endpoints and cloud environments.

It is composed of the following core components:

- **Wazuh Manager:** Collects, correlates, and analyzes security data received from agents.
- **Wazuh Agent:** Installed on endpoints to monitor activity and send event data to manager
- **Elastic Stack:** Enables powerful visualization and analysis through Elasticsearch, Logstash, and Kibana.

## File Integrity Monitoring (FIM) with Wazuh

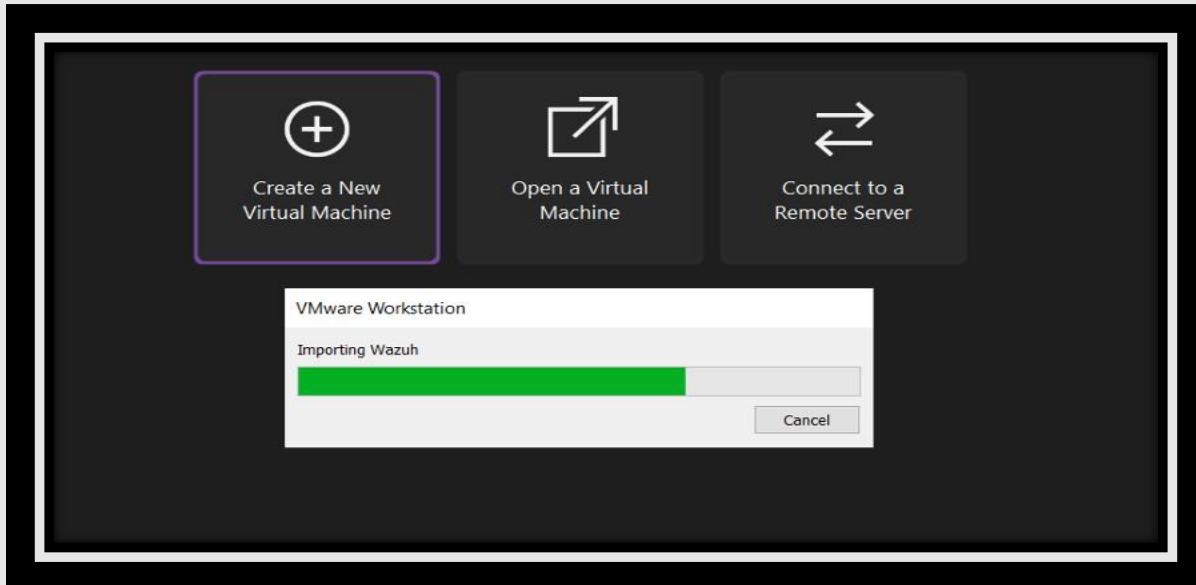
File Integrity Monitoring (FIM) is a crucial cybersecurity mechanism used to detect and track changes to critical files and directories. It helps ensure the integrity and security of systems by identifying unauthorized modifications that could indicate malicious activity or policy violations.

Wazuh integrates FIM as one of its core security features, continuously monitoring file access, creation, modification, and deletion across endpoints. When unexpected or unauthorized changes occur, Wazuh generates real-time alerts, allowing administrators to respond promptly and mitigate potential threats.

Common Use Cases:

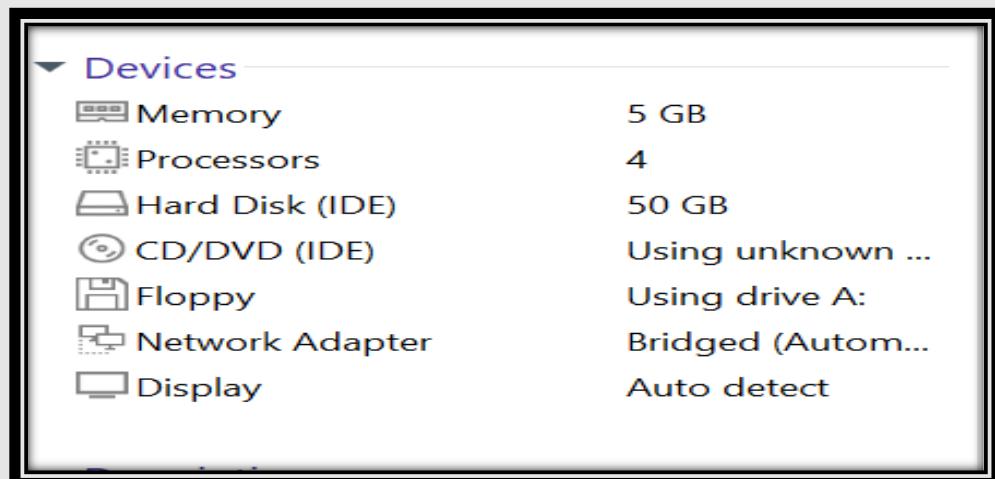
- Detecting unauthorized tampering with configuration or system files.
- Monitoring changes to critical operating system or application files.
- Alerting when website files are altered, indicating possible compromise or defacement attempt

## Step 1 : Installing Wazuh OVA in VMware



To set up the Wazuh environment, the official **Wazuh OVA (Open Virtual Appliance)** file was downloaded and deployed using **VMware Workstation**. The setup process involved the following steps:

1. Downloaded the Wazuh OVA file from the official Wazuh website.
2. Opened **VMware Workstation** and selected the “**Open a Virtual Machine**” option.
3. Imported the OVA file and customized the virtual machine settings, including **RAM** and **CPU allocation**, based on available system resources.
4. Powered on the virtual machine and allowed the Wazuh server to complete its initial boot and configuration process.



## Step 2 : Starting up the Server



Once the virtual machine was up:

- I logged in with the default credentials (username: wazuh-user, password: wazuh).
- Verified services were running (wazuh-manager, dashboard, indexer, etc.).
- Accessed the Wazuh dashboard via web browser using the IP assigned to the VM.
- The ip can be found by command, ip a, ip address, ifconfig.



### Step 3 : Accessing the Dashboard

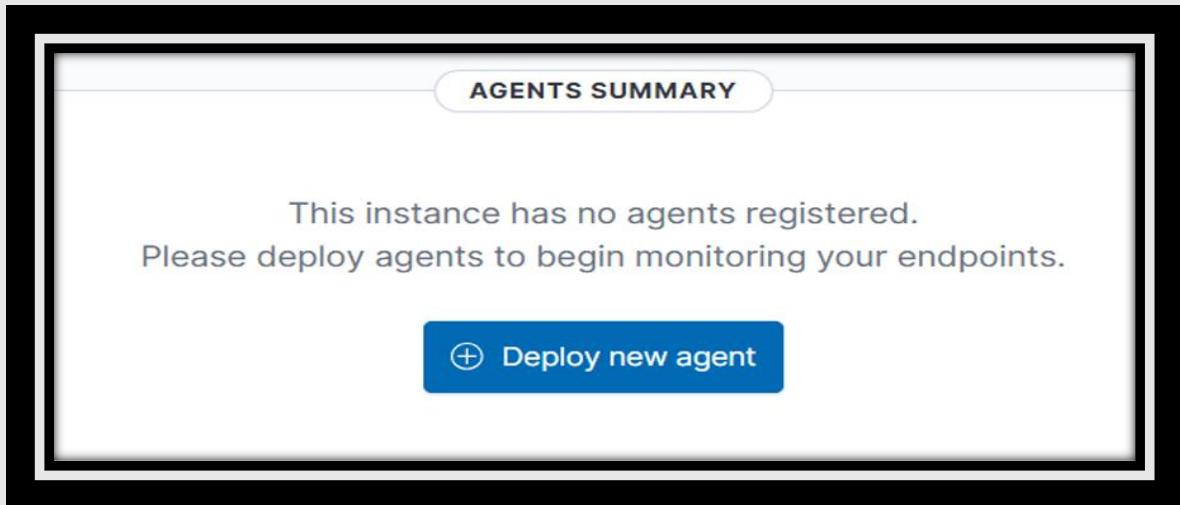
- Opened browser and entered: [https://<Wazuh\\_VM\\_IP>](https://<Wazuh_VM_IP>)
- Logged into Dashboard.
- Navigated to the Wazuh app to access dashboards, rules, alerts, and logs



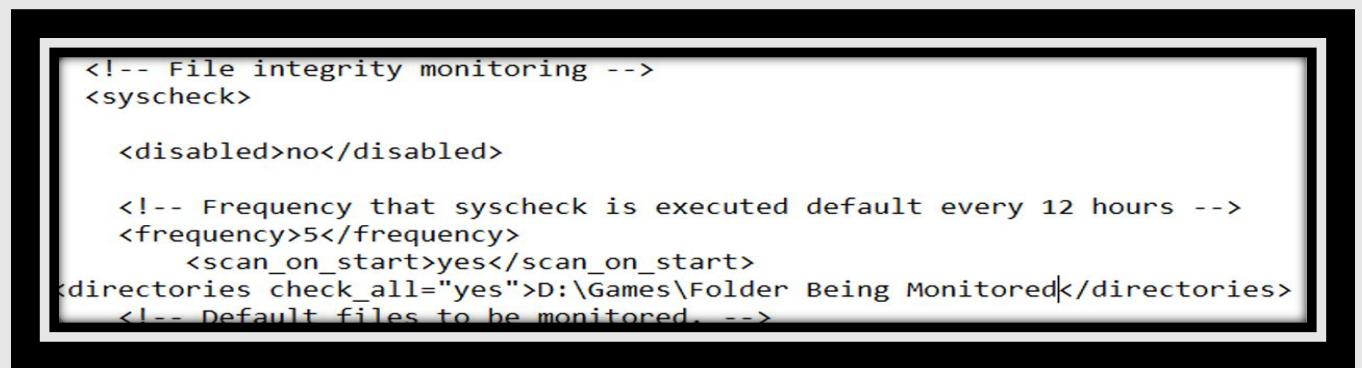
### Step 4: Wazuh Agents Deployment

**Wazuh agents** are lightweight applications installed on endpoints (Windows or Linux) to collect and forward security-related data to the Wazuh manager. The deployment process involved the following steps:

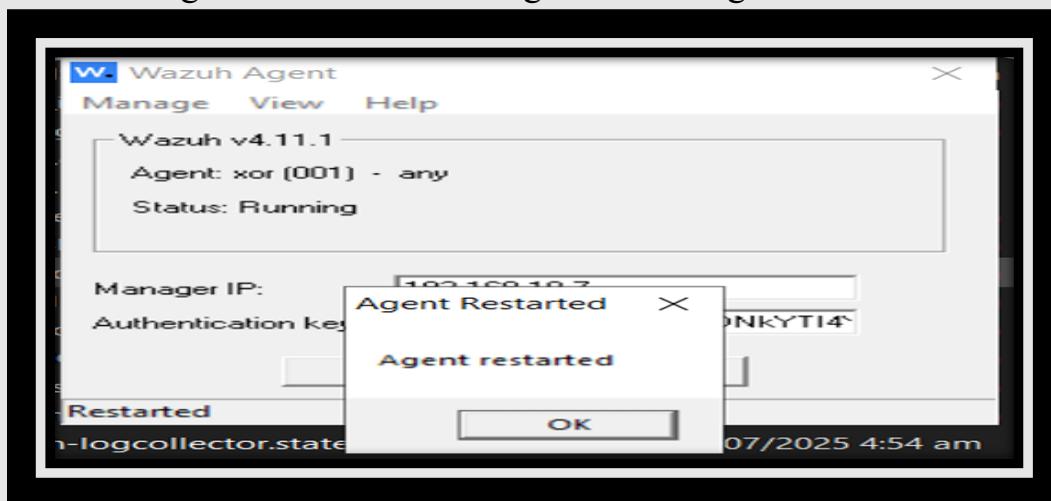
1. Generated an **agent enrollment key** through the Wazuh dashboard.
2. Installed the **Wazuh agent** on a test endpoint, such as a Windows virtual machine.
3. Configured the agent using the **enrollment key** and the **Wazuh server IP** to establish a connection with the manager.
4. Confirmed the agent's successful connection and active status via the Wazuh dashboard.



## Step 5 : Setting Agents



- On the agent machine, I configured the agent to monitor specific folders/files
- Enabled modules like FIM in the agent configuration file (ossec.conf).
- Restarted the agent service after configuration changes.



## Final Step: Viewing Logs in the Dashboard

- After the agent was up and running, logs began populating in the **Wazuh dashboard**.
- I filtered these logs according to:
  - ✓ **Agent name**
  - ✓ **Alert severity**
  - ✓ **File modifications** (for FIM monitoring)
- I explored dashboards such as **Security Events**, **Integrity Monitoring**, and **Agent Overview** to review and analyze the collected data.



## Conclusion

The implementation of File Integrity Monitoring (FIM) using Wazuh successfully demonstrated the ability to detect and respond to unauthorized file changes across monitored endpoints. By deploying agents and configuring the manager, real-time alerts were generated, providing visibility into critical system and configuration files. This project highlights the importance of continuous monitoring for maintaining system integrity, improving security posture, and ensuring compliance with organizational policies.

THE END