

Project Title

Wazuh-Based SIEM and File Integrity Monitoring Home Lab

Project Description

This project focuses on building a complete **Security Information and Event Management (SIEM)** and **File Integrity Monitoring (FIM)** environment using **Wazuh**. The setup includes deploying a Wazuh Manager on an Ubuntu Virtual Machine and configuring a Wazuh Agent on a Windows host system.

The project demonstrates how logs, system events, and file-change activities from a Windows machine can be collected, analyzed, and visualized in real time using the Wazuh Dashboard. It also includes configuring file integrity monitoring to detect unauthorized file modifications, deletions, and creations. The system provides real-time alerts and serves as a practical demonstration of endpoint monitoring, intrusion detection, and security automation — essential skills in cybersecurity and SOC operations.

The setup consists of two main components:

- 1.Wazuh Manager installed on an Ubuntu virtual machine
- 2.Wazuh Agent installed on a Windows host machine

The Wazuh Manager collects logs and security data from the agent, processes it, and displays insights through the Wazuh Dashboard. The Windows agent continuously monitors activities such as logins, system events, configuration changes, and any modification to sensitive folders.

A major part of this project is implementing File Integrity Monitoring (FIM). A specific directory on the Windows system was configured for real-time monitoring. Any file creation, deletion, or modification in this folder triggers alerts in the Wazuh dashboard. This demonstrates how SIEM tools help detect suspicious activity or unauthorized access.

Overall, this project provides hands-on experience with endpoint security, log analysis, threat detection, and real-time security operations — skills crucial in cybersecurity and SOC analyst roles.

Objective of the Project

To build and configure a real SIEM environment using Wazuh.

To understand how security logs are collected and analyzed.

To implement File Integrity Monitoring for detecting unauthorized changes.

To simulate a basic SOC workflow and observe real-time alerts.

To gain practical exposure to SIEM dashboards and security event correlation.

Scope of the Project

Installation and configuration of Wazuh Manager on Ubuntu.

Deployment and registration of the Wazuh Agent on Windows.

Monitoring Windows logs, events, system health, and processes.

Implementation of real-time File Integrity Monitoring (FIM).

Testing by generating file changes and observing alert generation.

Viewing logs and alerts through the Wazuh Dashboard.

Understanding how SIEM tools detect unusual activity.

Problem Statement

Modern endpoints face threats such as unauthorized file access, malware activity, insider misuse, and configuration tampering. Organizations need automated systems that continuously monitor endpoints, detect suspicious actions, and generate security alerts.

This project solves this problem by implementing a Wazuh-based SIEM and FIM setup to monitor a Windows machine in real time.

Use Case Model

Log Monitoring: Collect Windows event logs and detect abnormal patterns.

File Integrity Monitoring: Detect changes to sensitive files or directories.

Intrusion Detection: Identify suspicious activity such as failed logins or privilege misuse.

Alerting: Real-time notifications for security violations.

Dashboard Analysis: SOC-like visual analysis of system events and alerts.

Key Features Implemented

SIEM setup with Wazuh

Windows agent onboarding

Generated agent connection keys

Real-time file and folder monitoring

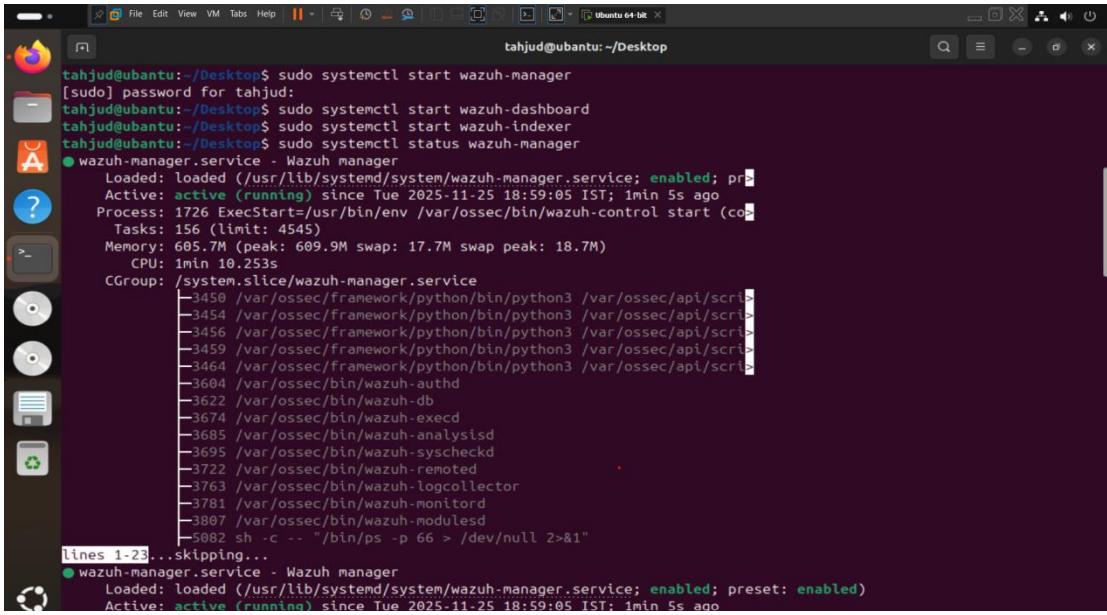
Event log collection

Wazuh dashboard usage

Real-time alerts for file creation/deletion/modification

Hands-on SOC analysis workflow

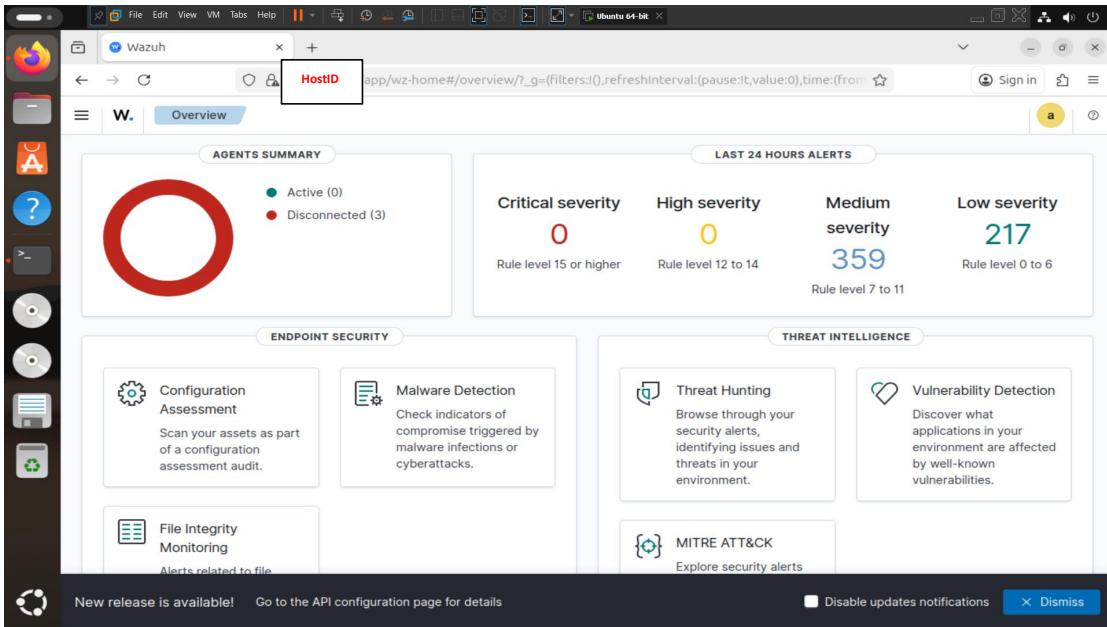
1. Install wazuh in Ubuntu



```
tahjud@ubuntu:~/Desktop$ sudo systemctl start wazuh-manager
[sudo] password for tahjud:
tahjud@ubuntu:~/Desktop$ sudo systemctl start wazuh-dashboard
tahjud@ubuntu:~/Desktop$ sudo systemctl start wazuh-indexer
tahjud@ubuntu:~/Desktop$ sudo systemctl status wazuh-manager
● wazuh-manager.service - Wazuh manager
   Loaded: loaded (/usr/lib/systemd/system/wazuh-manager.service; enabled; pres>
   Active: active (running) since Tue 2025-11-25 18:59:05 IST; 1min 5s ago
     Process: 1726 ExecStart=/usr/bin/env /var/ossec/bin/wazuh-control start (co>
   Tasks: 156 (limit: 4545)
  Memory: 605.7M (peak: 609.9M swap: 17.7M swap peak: 18.7M)
    CPU: 1min 10.253s
  CGroup: /system.slice/wazuh-manager.service
          ├─3450 /var/ossec/framework/python/bin/python3 /var/ossec/api/script>
          ├─3454 /var/ossec/framework/python/bin/python3 /var/ossec/api/script>
          ├─3456 /var/ossec/framework/python/bin/python3 /var/ossec/api/script>
          ├─3459 /var/ossec/framework/python/bin/python3 /var/ossec/api/script>
          ├─3464 /var/ossec/framework/python/bin/python3 /var/ossec/api/script>
          ├─3604 /var/ossec/bin/wazuh-authd
          ├─3622 /var/ossec/bin/wazuh-db
          ├─3674 /var/ossec/bin/wazuh-execd
          ├─3685 /var/ossec/bin/wazuh-analysisd
          ├─3695 /var/ossec/bin/wazuh-syscheckd
          ├─3722 /var/ossec/bin/wazuh-remoted
          ├─3763 /var/ossec/bin/wazuh-logcollector
          ├─3781 /var/ossec/bin/wazuh-monitord
          ├─3807 /var/ossec/bin/wazuh-modulesd
          └─5082 sh -c -- "/bin/ps -p 66 > /dev/null 2>&1"
lines 1-23...skipping...
● wazuh-manager.service - Wazuh manager
   Loaded: loaded (/usr/lib/systemd/system/wazuh-manager.service; enabled; preset: enabled)
   Active: active (running) since Tue 2025-11-25 18:59:05 IST; 1min 5s ago
```

On your Ubuntu server, open a browser and go to: <https://localhost>

1. Accept any browser security warning due to the self-signed certificate.
2. Log in using the credentials displayed at the end of the installation script.

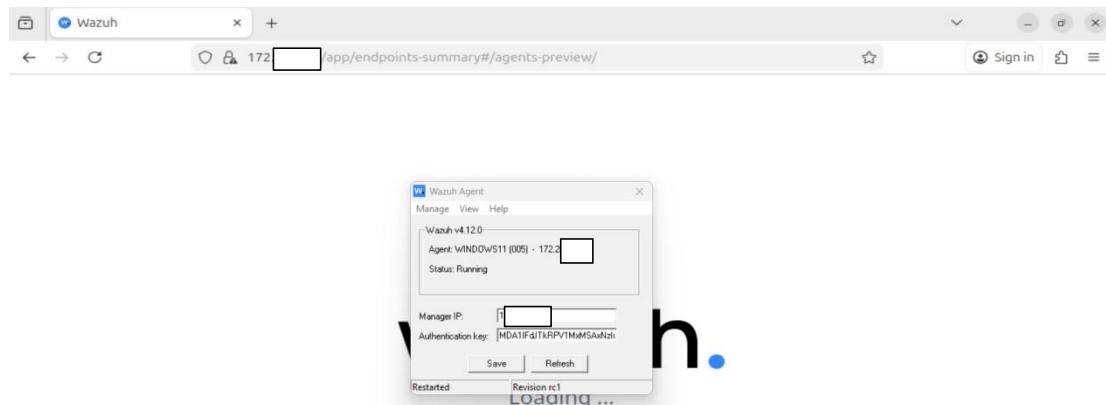


The screenshot shows the Wazuh dashboard interface. At the top, there's a navigation bar with a search bar containing "HostID". Below the navigation, the main dashboard is divided into several sections:

- OVERVIEW**: Shows a summary of agents: Active (0) and Disconnected (3). It also displays the number of alerts over the last 24 hours categorized by severity: Critical (0), High (0), Medium (359), and Low (217).
- ENDPOINT SECURITY**: Includes cards for Configuration Assessment (Scan your assets as part of a configuration assessment audit), Malware Detection (Check indicators of compromise triggered by malware infections or cyberattacks), Threat Hunting (Browse through your security alerts, identifying issues and threats in your environment), and Vulnerability Detection (Discover what applications in your environment are affected by well-known vulnerabilities).
- THREAT INTELLIGENCE**: Includes a card for MITRE ATT&CK (Explore security alerts).

At the bottom of the dashboard, there's a notification bar stating "New release is available! Go to the API configuration page for details" with a "Dismiss" button.

Install the Wazuh Agent (Windows Host)



Registering the Agent with the Manager (sudo /var/ossec/bin/manage_agents)

```
*****
* Wazuh v4.12.0 Agent manager. *
* The following options are available: *
*****
(A)dd an agent (A).
(E)xtract key for an agent (E).
(L)ist already added agents (L).
(R)emove an agent (R).
(Q)uit.
Choose your action: A,E,L,R or Q: E

available agents:
ID: 007, Name: Windows11, IP: 172.20.10.10
ID: 005, Name: WINDOWS11, IP: 172.20.10.10
ID: 004, Name: WINDOWS10, IP: 172.20.10.10
ID: 006, Name: TAHJUD, IP: any
Provide the ID of the agent to extract the key (or '\q' to quit): 005

agent key information for '005' is:
MDA1IFdJTkRPV1MxMSAxNzIuMjAuMTAuMyA2YTQ4YzIzOTc10ThlNDcwMjdkMzQ0ZGVhNTc0NDU4ZTQwNzE2MzFlZWm2NjJkMDI4MDI5ZTlj0WQ4NDM0Njhj

* Press ENTER to return to the main menu.
```

File Integrity Monitoring on Windows Wazuh supports real-time monitoring of file and folder changes using Syscheck.

Edit Agent Configuration Open the following configuration file: C:\Program Files (x86)\ossec-agent\ossec.conf



The screenshot shows a Windows Notepad window with the file 'ossec.conf' open. The content of the file is a XML configuration for the Wazuh agent. It includes sections for clients, server addresses, log analysis locations, and event log formats. The XML uses various tags like <client>, <server>, <config_profile>, <crypto_method>, <notify_time>, <time_reconnect>, <auto_restart>, <client_buffer>, <log_format>, and <location>. The code is well-formatted with indentation and line breaks.

```
<!--
Wazuh - Agent - Default configuration for Windows
More info at: https://documentation.wazuh.com
Mailing list: https://groups.google.com/forum/#!forum/wazuh
-->

<ossec_config>

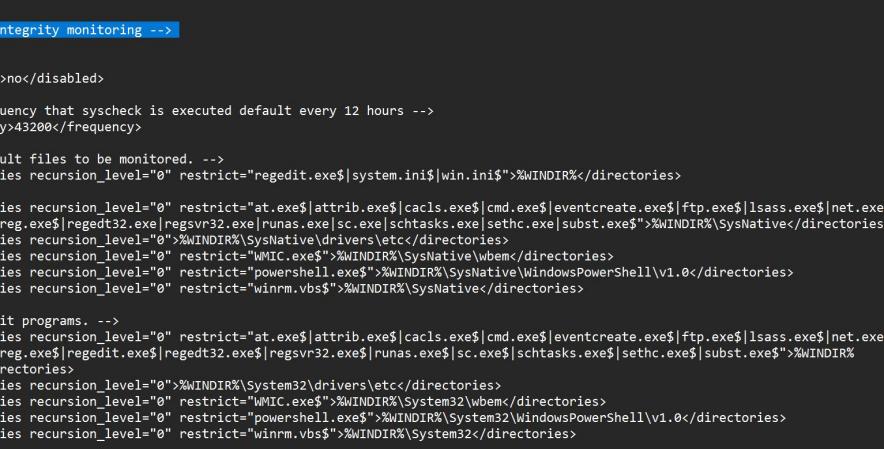
  <client>
    <server>
      <address>>172. [REDACTED] /address<
      <port>5145</port>
      <protocol>tcp</protocol>
    </server>
    <config_profile>windows, windows10</config_profile>
    <crypto_method>aes</crypto_method>
    <notify_time>10</notify_time>
    <time_reconnect>60</time_reconnect>
    <auto_restart>yes</auto_restart>
  </client>

  <!-- Agent buffer options -->
  <client_buffer>
    <disabled>no</disabled>
    <queue_size>50000</queue_size>
    <events_per_second>500</events_per_second>
  </client_buffer>

  <!-- Log analysis -->
  <localfile>
    <locations>application</location>
    <log_format>eventchannel</log_format>
  </localfile>

  <localfile>
    <location>Security</location>
    <log_format>eventchannel</log_format>
    <query>EventID>5145 and EventID != 5156 and EventID != 5447 and
        EventID != 4656 and EventID != 4658 and EventID != 4663 and EventID != 4669 and
        EventID != 4670 and EventID != 4690 and EventID != 4703 and EventID != 4907 and
        EventID != 4908</query>
  </localfile>

<!-- Col. 1  Col. 2  Col. 3  Col. 4  Col. 5  Col. 6  Col. 7  Col. 8  Col. 9  Col. 10 -->
<ln1>Col 1  9643 characters  Plain text  100%  Windows (CRLF)  UTF-8
```



The screenshot shows a Windows Notepad window displaying an XML configuration file for the Ossec Host Agent. The file defines various monitoring rules, including system integrity checks, scheduled tasks, and specific file monitoring for system files like cmd.exe and powershell.exe.

```
<ossec.conf>
  <File>
    <Edit> View
    <!-- Monitoring interval -->
    <interval>12h</interval>
    <skip_nfs>yes</skip_nfs>
  </sca>

  <!-- File integrity monitoring -->
  <syscheck>

    <disabled>no</disabled>

    <!-- Frequency that syscheck is executed default every 12 hours -->
    <frequency>43200</frequency>

    <!-- Default files to be monitored. -->
    <directories recursion_level="0" restrict="regedit.exe$|system.ini$|win.ini$">%WINDIR%</directories>

    <directories recursion_level="0" restrict="at.exe$|attrib.exe$|cacls.exe$|cmd.exe$|eventcreate.exe$|ftp.exe$|lsass.exe$|net.exe$|net1.exe
    $|netsh.exe$|reg.exe$|regedit32.exe$|regsvr32.exe$|runas.exe$|sc.exe$|schtasks.exe$|sethc.exe$|subst.exe$">%WINDIR%\SysNative</directories>
    <directories recursion_level="0">%WINDIR%\SysNative\drivers\etc</directories>
    <directories recursion_level="0" restrict="WMIC.exe$">%WINDIR%\SysNative\wbem</directories>
    <directories recursion_level="0" restrict="powershell.exe$">%WINDIR%\SysNative\WindowsPowerShell\v1.0</directories>
    <directories recursion_level="0" restrict="winrm.vbs$">%WINDIR%\SysNative</directories>

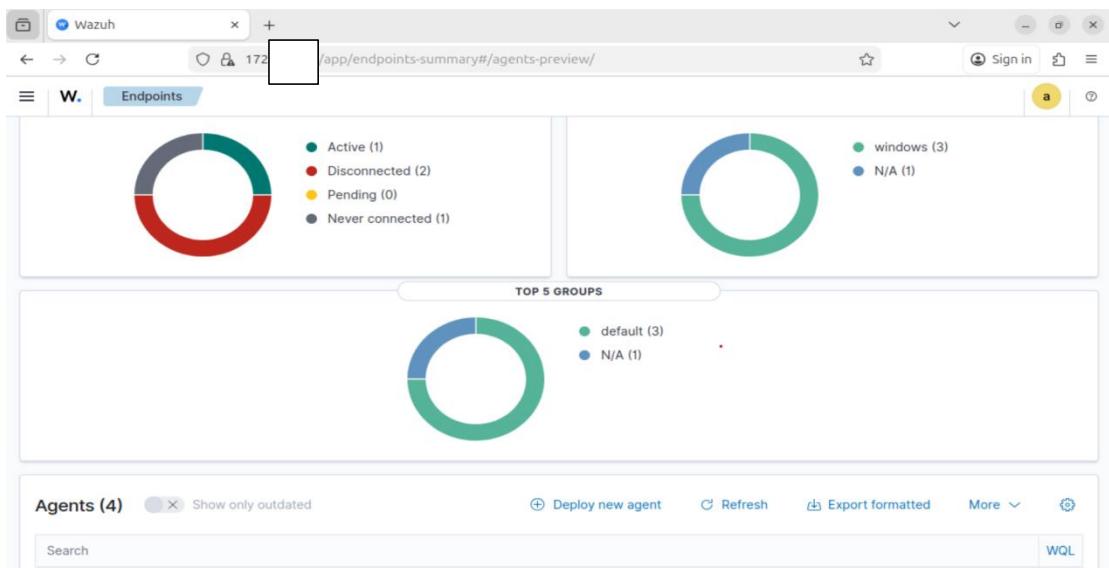
    <!-- 32-bit programs. -->
    <directories recursion_level="0" restrict="at.exe$|attrib.exe$|cacls.exe$|cmd.exe$|eventcreate.exe$|ftp.exe$|lsass.exe$|net.exe$|net1.exe
    $|netsh.exe$|reg.exe$|regedit.exe$|regedit32.exe$|regsvr32.exe$|runas.exe$|sc.exe$|schtasks.exe$|sethc.exe$|subst.exe$">%WINDIR%
    \System32</directories>
    <directories recursion_level="0">%WINDIR%\System32\drivers\etc</directories>
    <directories recursion_level="0" restrict="WMIC.exe$">%WINDIR%\System32\wbem</directories>
    <directories recursion_level="0" restrict="powershell.exe$">%WINDIR%\System32\WindowsPowerShell\v1.0</directories>
    <directories recursion_level="0" restrict="winrm.vbs$">%WINDIR%\System32</directories>

  <directories realtime="yes">%PROGRAMDATA%\Microsoft\Windows\Start Menu\Programs\Startup</directories>
</File>
```

```

ossec.conf
File Edit View
<!-- Default files to be monitored. -->
<directories recursion_level="0" restrict="regedit.exe$|system.ini$|win.ini$">%WINDIR%</directories>
<directories recursion_level="0" restrict="at.exe$|attrib.exe$|cacls.exe$|cmd.exe$|eventcreate.exe$|ftp.exe$|lsass.exe$|net.exe$|net1.exe
$|netsh.exe$|reg.exe$|regedit32.exe$|regsvr32.exe$|runas.exe$|sc.exe$|schtasks.exe$|sethc.exe$|subst.exe$">%WINDIR%\SysNative</directories>
<directories recursion_level="0">%WINDIR%\SysNative\drivers\etc</directories>
<directories recursion_level="0" restrict="WMIC.exe$">%WINDIR%\SysNative\wbem</directories>
<directories recursion_level="0" restrict="powershell.exe$">%WINDIR%\SysNative\WindowsPowerShell\v1.0</directories>
<directories recursion_level="0" restrict="winrm.vbs$">%WINDIR%\SysNative</directories>
<!-- 32-bit programs. -->
<directories recursion_level="0" restrict="at.exe$|attrib.exe$|cacls.exe$|cmd.exe$|eventcreate.exe$|ftp.exe$|lsass.exe$|net.exe$|net1.exe
$|netsh.exe$|reg.exe$|regedit.exe$|regedit32.exe$|regsvr32.exe$|runas.exe$|sc.exe$|schtasks.exe$|sethc.exe$|subst.exe$">%WINDIR%
\System32</directories>
<directories recursion_level="0" restrict="WMIC.exe$">%WINDIR%\System32\wbem</directories>
<directories recursion_level="0" restrict="powershell.exe$">%WINDIR%\System32\WindowsPowerShell\v1.0</directories>
<directories recursion_level="0" restrict="winrm.vbs$">%WINDIR%\System32</directories>
<directories realtime="yes">%PROGRAMDATA%\Microsoft\Windows\Start Menu\Programs\Startup</directories>
<directories realtime="yes">C:\Users\TAHJUD TAHA NOOR\Documents\wazuh-test</directories>
<ignore>%PROGRAMDATA%\Microsoft\Windows\Start Menu\Programs\Startup\desktop.ini</ignore>
<ignore type="sregex">.log$|.htm$|.jpg$|.png$|.chm$|.pnf$|.evtx$</ignore>
<!-- Windows registry entries to monitor. -->
<windows_registry>HKEY_LOCAL_MACHINE\Software\Classes\batfile</windows_registry>
<windows_registry>HKEY_LOCAL_MACHINE\Software\Classes\cmdfile</windows_registry>
<windows_registry>HKEY_LOCAL_MACHINE\Software\Classes\comfile</windows_registry>
<windows_registry>HKEY_LOCAL_MACHINE\Software\Classes\exefile</windows_registry>
<windows_registry>HKEY_LOCAL_MACHINE\Software\Classes\biffile</windows_registry>

```

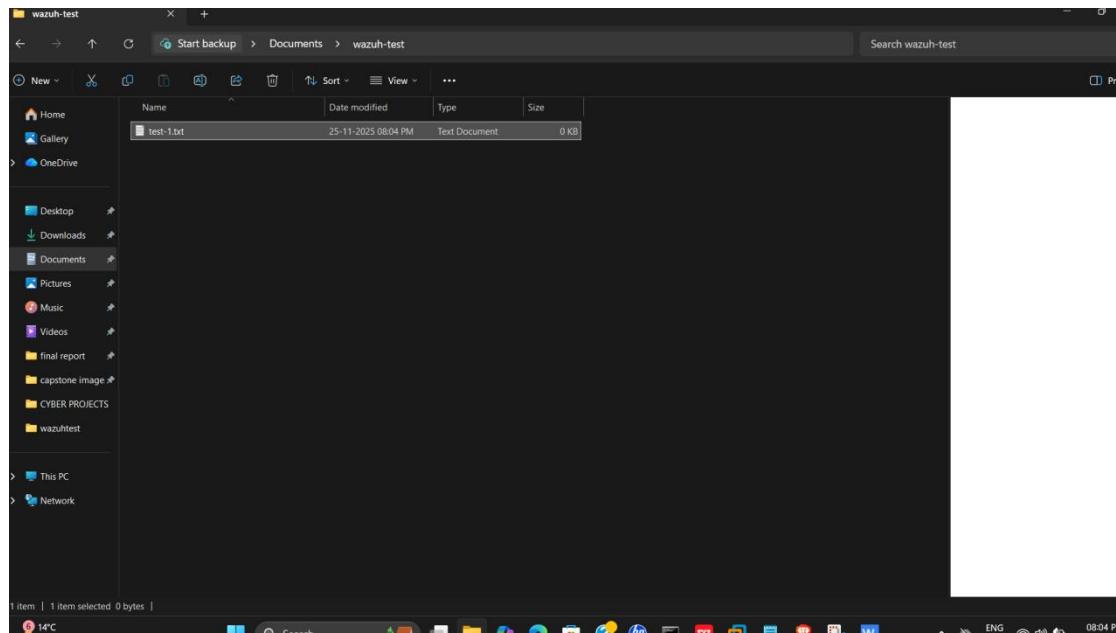


Screenshot of the Wazuh web interface showing the agent summary for a Windows 11 host (ID: 005). The agent is active, connected via IP address 172.21.1.120, version Wazuh v4.12.0, and group default. The operating system is Microsoft Windows 11 Home Single Language 10.0.26200.7171, running on cluster node node01 since Nov 25, 2025, at 18:10:10.000. The last keep alive was on Nov 25, 2025, at 19:36:26.000.

The interface includes three main cards:

- Events count evolution:** A line chart showing event counts over the last 24 hours, ranging from 0 to 500 events per 30-minute interval.
- MITRE ATT&CK:** A card displaying the top tactics: Defense Evasion (1), Initial Access (1), Persistence (1), and Privilege Escalation (1).
- Compliance:** A donut chart showing compliance status across PCI DSS categories: 2.2 (484), 10.2.5 (1), and 10.6.1 (1).

Create a folder and add a text file then delete it



This is the detail of the file deleted in host

The screenshot shows the Wazuh File Integrity Monitoring interface. The top navigation bar includes 'File', 'Edit', 'View', 'VM', 'Tabs', 'Help', and a search bar with the URL '172.24.1.1/app/file-integrity-monitoring#/overview/?tab=fim&tabView=events&agentId=005'. A yellow notification badge with the letter 'a' is visible.

The main area has tabs for 'Dashboard', 'Inventory', and 'Events'. The 'Events' tab is selected, showing a search bar with filters: 'manager.name: ubuntu', 'rule.groups: syscheck', and 'agent.id: 005'. Below the search bar is a chart showing 'Count' over time from 21:00 to 06:00, with a single data point at 21:00. The chart title is 'Nov 24, 2025 @ 20:28:31.257'.

On the left, there's a sidebar with icons for Home, Overview, Agents, Dashboards, Metrics, Events, Logs, Config, and Help.

The right side displays 'Document Details' for a specific event. It shows a table of log entries:

Table	JSON
_index	wazuh-alerts-4.x-2025.11.25
agent.id	005
agent.ip	172.24.1.1
agent.name	WINDOWS11
decoder.name	syscheck_deleted
full_log	File 'c:\users\tahjud taha noor\documents\bt ech\wazuh\test1.txt' deleted Mode: realtime
id	1764082701.5027439
input.type	log
location	syscheck
manager.name	ubuntu
rule.description	File deleted.
# rule.firedtimes	1
rule.gdpr	II_5.1.f
rule.gpp13	4.11
rule.message	ossec syscheck syscheck entry deleted evm