

Wazuh EDR: TryHackMe

Name: Daniel Mwendwa Mwithui

ADM NO. CS-SA04-23080

Program: Security Analyst

Date of submission: 28th July 2023

Introduction

In this report, we will explore the fundamentals of Wazuh, a powerful open source EDR designed to enhance threat detection and response capabilities within organizations. Through this assignment, we aim to provide an overview of Wazuh and its essential features, including log analysis, vulnerability assessment, and security event correlation. We will delve into the process of installing Wazuh agents on different operating systems, configuring log collection, and utilizing the Wazuh API for seamless integration with other security tools. Additionally, we will explore how to generate reports in Wazuh, which help visualize security data and aid in decision-making to bolster cybersecurity measures.

Endpoint detection and response (EDR)

Endpoint Detection and Response (EDR) is a cybersecurity technology that focuses on detecting and responding to advanced threats and attacks on individual endpoints within a network. EDR solutions are designed to provide real-time monitoring, threat detection, investigation, and response capabilities at the endpoint level.

The primary goal of EDR is to detect malicious activities and potential security breaches that traditional antivirus software might miss. It operates by continuously collecting and analyzing endpoint data, including system events, processes, files, network connections, and user activities. This data is then compared against known patterns of malicious behavior or indicators of compromise (IOCs) to identify suspicious activities. Once an anomaly or potential threat is detected, EDR tools can take various actions, such as isolating the compromised endpoint from the network, blocking malicious processes, quarantining files, and alerting security personnel for further investigation and remediation.

Wazuh is an open-source security platform that incorporates EDR functionality along with intrusion detection, log analysis, security event correlation, and compliance management. It was initially created as a fork of the OSSEC (Open Source HIDS Security) project and has evolved into a more extensive and robust platform for security monitoring and threat detection. Wazuh integrates with various data sources, including logs, events, and alerts generated by endpoints, firewalls, intrusion detection systems, and other security-related devices and applications. The platform centrally collects and analyzes this data, allowing security teams to gain comprehensive visibility into the security posture of their environment.

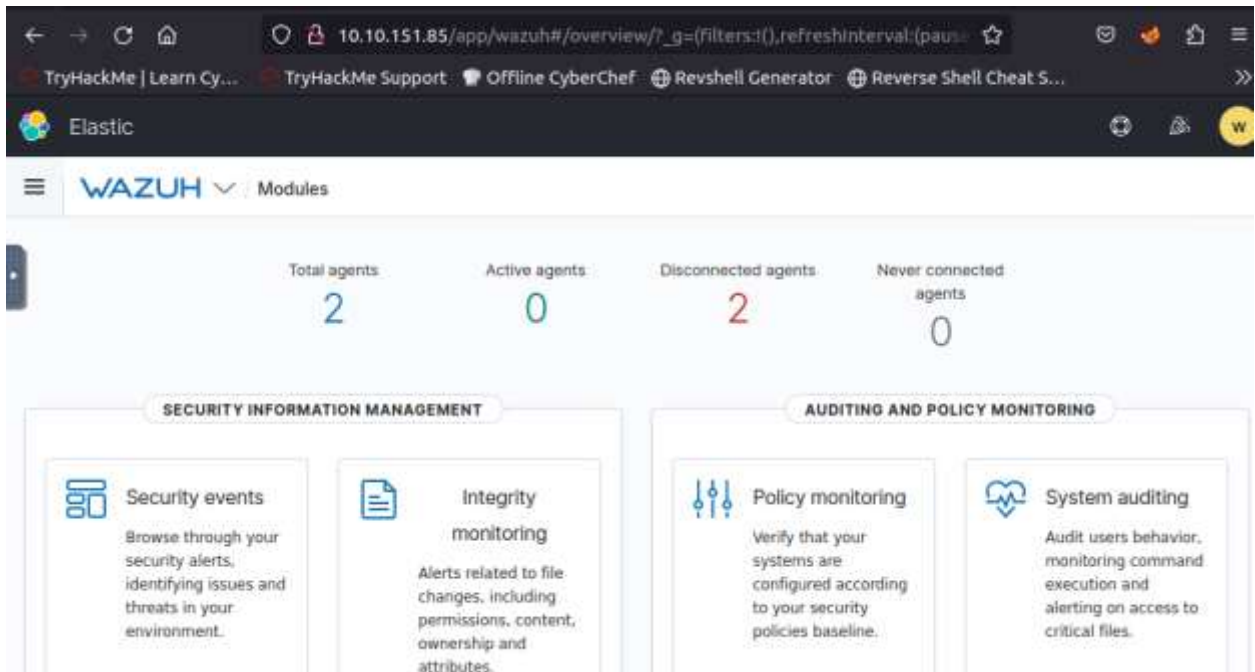
Task 1: Wazuh Agents

Wazuh agents are lightweight software components that are installed on individual endpoints (e.g., computers, servers) within a network to collect security-related data and send it to the central Wazuh manager for analysis and monitoring.

To install Wazuh agents:

- Download the appropriate agent package for your operating system from the Wazuh website.
- Install the agent on the endpoint by running the installer or following the provided instructions.
- Configure the agent to point to the IP address or hostname of the Wazuh manager.
- Start the Wazuh agent service to begin sending security data to the central manager for analysis and response.

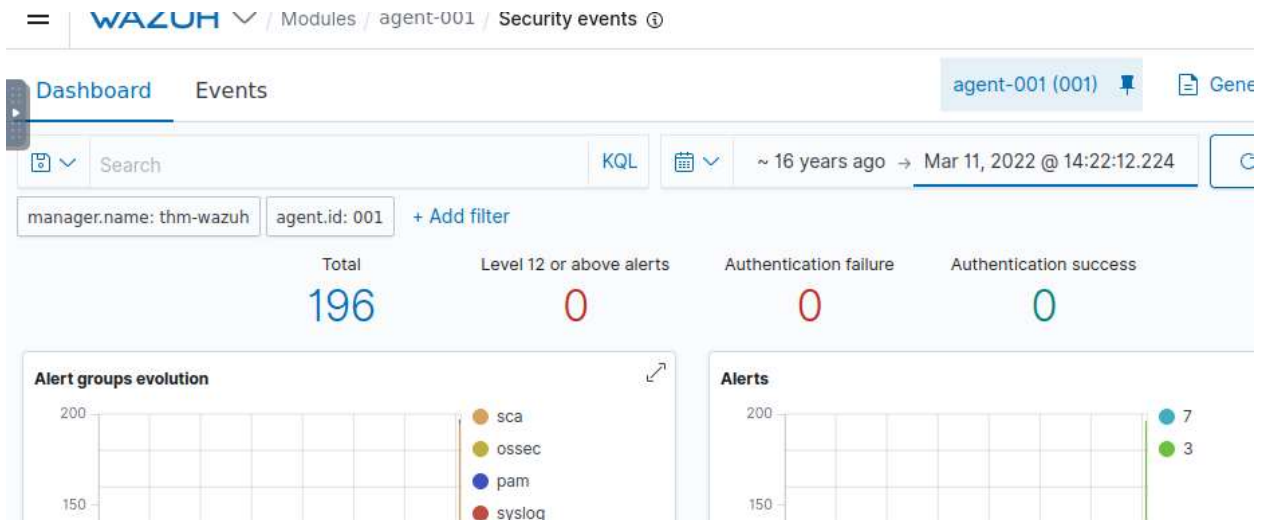
In this section we are going to login to Wazuh using the given credentials and navigate to Wazuh > Agents.



Task 2: Wazuh Vulnerability Assessment & Security Events

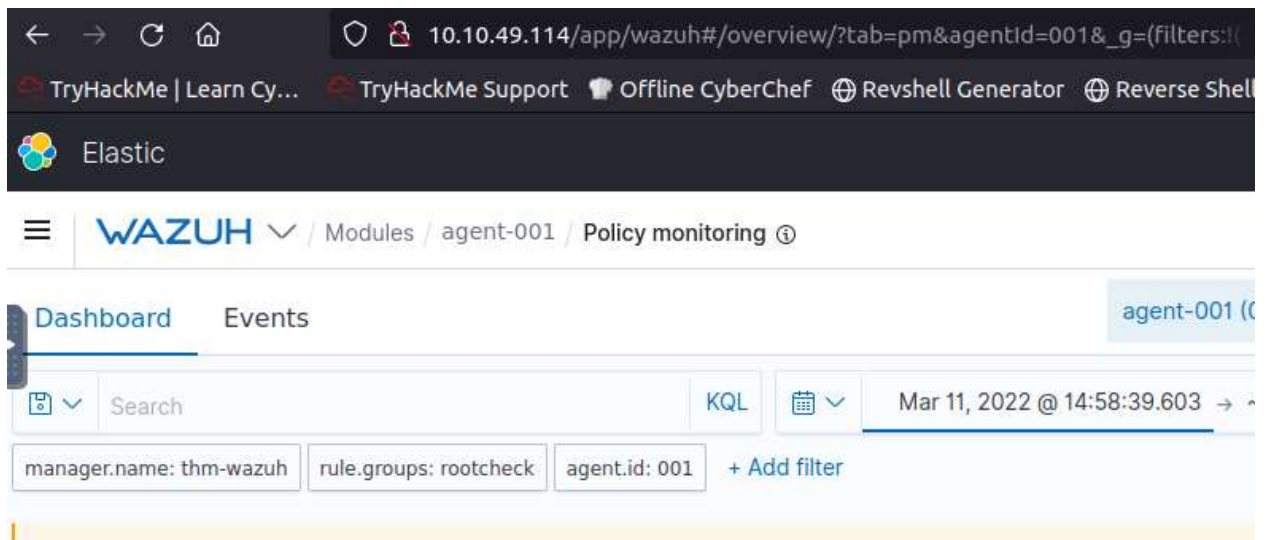
Wazuh provides Vulnerability Assessment and Security Events capabilities to enhance cybersecurity:

- **Vulnerability Assessment:** Wazuh can scan endpoints and network devices for known vulnerabilities, identifying potential weaknesses in the system that attackers could exploit.
- **Security Events:** Wazuh continuously monitors and analyzes security-related events, such as log data, system events, and network activities, to detect suspicious or malicious behavior and trigger real-time alerts for investigation and response.



Task 3: Wazuh Policy Auditing

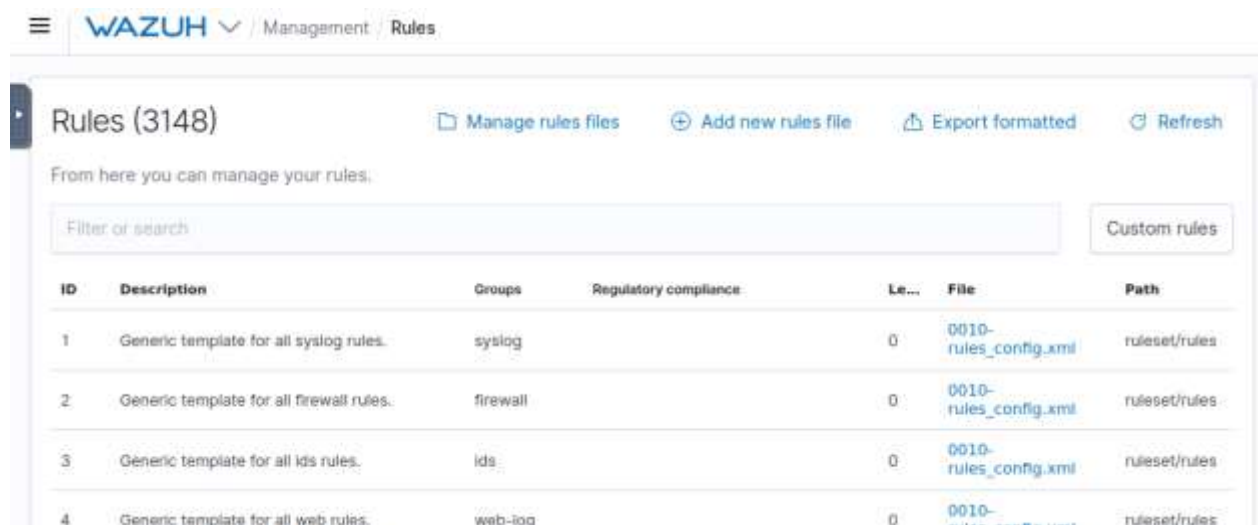
Wazuh policy refers to the set of rules and configurations that govern the behavior and security practices of the Wazuh platform within an organization. These policies determine how Wazuh collects, analyzes, and responds to security-related data from different sources, such as endpoints, firewalls, and servers. The policies in Wazuh can be customized to align with an organization's specific security requirements and compliance standards. They dictate which events are logged, what types of activities trigger alerts, and how the system responds to different security incidents. By defining and fine-tuning Wazuh policies, organizations can tailor the platform's capabilities to their unique security needs, ensuring a proactive approach to threat detection and a swift response to potential security breaches.



Task 4: Monitoring Logons with Wazuh

Monitoring logons with Wazuh allows organizations to track and analyze user authentication activities across their network. By collecting and analyzing logon events from various endpoints, servers, and network devices, Wazuh helps detect suspicious or unauthorized login attempts, providing crucial insights into potential security threats.

Wazuh can monitor both successful and failed logon events, identifying anomalies and patterns that may indicate unauthorized access or credential misuse. The platform can alert security teams in real-time when it detects suspicious logon behavior, enabling rapid response and investigation to mitigate potential risks.



Task 5: Collecting Window Logs with Wazuh

The Wazuh agent on Windows can be configured to collect both Sysmon and Event Viewer logs and forward them to the Wazuh manager for analysis. Once the Wazuh manager receives the Windows log data, it can analyze the information in real-time, correlating events and detecting anomalies, potential threats, and patterns indicative of security breaches.

Sysmon is a powerful Windows system monitoring tool that can be utilized to collect detailed information about process creations, network connections, and other system-level activities. These logs provide valuable visibility into potentially malicious behavior and advanced threats. Event Viewer on the other hand is a built-in Windows tool that logs a wide range of system events, such as security events, application events, and system events. These logs contain crucial information about user activities, login attempts, changes to the system, and other critical events.

Rules (84)						
From here you can manage your rules.						
<div> <div>search: sysmon X</div> <div>Filter or search</div> <div>Custom rules</div> </div>						
ID	Description	Groups	Regulatory compliance	Le...	File	Path
60004	Group of Windows rules for the Sysmon channel	windows		0	0575-win-base_rules.xml	ruleset/rules
61600	Windows Sysmon informational event	windows, sysmon		0	0595-win-sysmon_rules.xml	ruleset/rules
61601	Windows Sysmon warning event	windows, sysmon	GPG13	0	0595-win-sysmon_rules.xml	ruleset/rules

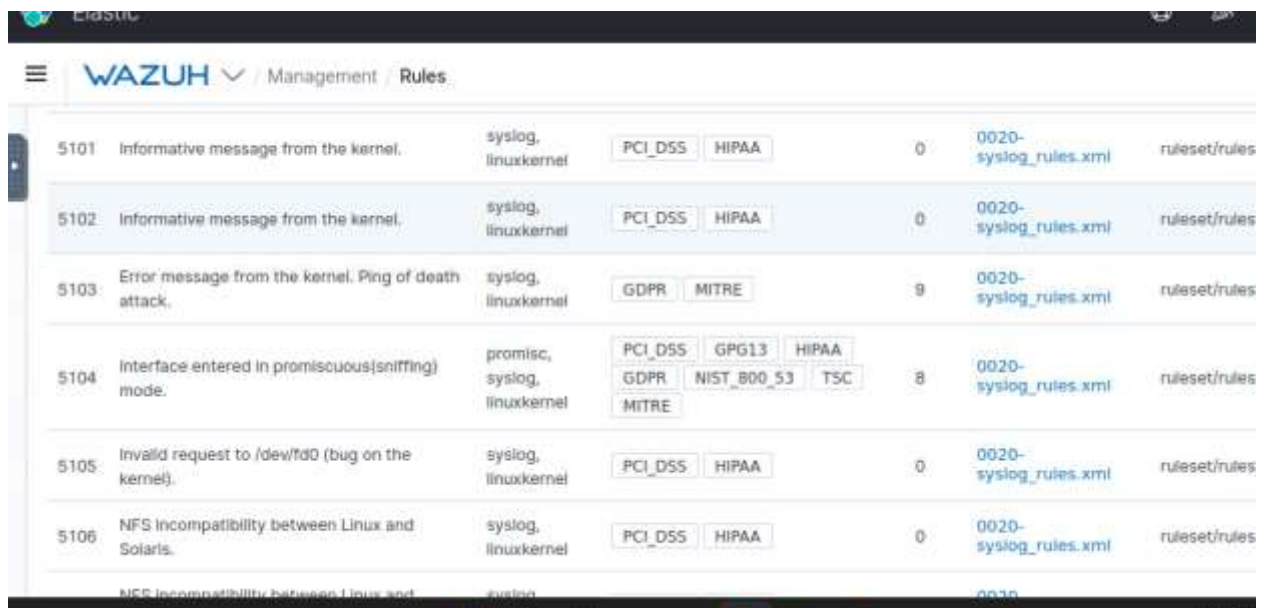
Task 6: Collecting Linux Logs with Wazuh

Collecting Linux logs with Wazuh involves deploying the Wazuh agent on Linux systems to gather and forward relevant log data to the central Wazuh manager. The Wazuh agent on Linux can collect a wide variety of logs from different sources, including:

- **System Logs:** These logs, typically located in the directory `/var/log`, contain important information about system events, hardware, and kernel messages. Common system logs include `messages`, `syslog`, `auth.log`, and `kern.log`.
- **Authentication Logs:** Linux maintains logs related to user authentication and access attempts. These logs, such as `secure`, `auth.log`, or `messages`, can be critical in identifying unauthorized login attempts or suspicious activities.
- **Application Logs:** Logs generated by various applications can be collected, providing insights into the activities of specific software and potential security issues.
- **Web Server Logs:** If applicable, logs from web servers like Apache or Nginx can be collected to monitor web traffic and detect suspicious or potentially malicious requests.

- **Database Logs:** Logs from databases like MySQL or PostgreSQL can be monitored to identify unusual database access patterns or potential SQL injection attempts.

To fine-tune the behavior of the Wazuh agent and specify which logs to collect and how to handle them, configuration rules can be defined. These rules are stored in the directory `/var/ossec/etc/rules` on the Wazuh agent. The rules are written in XML format and are highly customizable, allowing organizations to adapt the log collection and analysis process to their specific security needs.



ID	Description	Log Source	Frameworks	Score	File	Category
5101	Informative message from the kernel.	syslog, linuxkernel	PCI_DSS, HIPAA	0	0020-syslog_rules.xml	ruleset/rules
5102	Informative message from the kernel.	syslog, linuxkernel	PCI_DSS, HIPAA	0	0020-syslog_rules.xml	ruleset/rules
5103	Error message from the kernel. Ping of death attack.	syslog, linuxkernel	GDPR, MITRE	9	0020-syslog_rules.xml	ruleset/rules
5104	Interface entered in promiscuous(sniffing) mode.	promisc, syslog, linuxkernel	PCI_DSS, GPG13, HIPAA, GDPR, NIST_B00_53, TSC, MITRE	8	0020-syslog_rules.xml	ruleset/rules
5105	Invalid request to /dev/fd0 (bug on the kernel).	syslog, linuxkernel	PCI_DSS, HIPAA	0	0020-syslog_rules.xml	ruleset/rules
5106	NFS incompatibility between Linux and Solaris.	syslog, linuxkernel	PCI_DSS, HIPAA	0	0020-syslog_rules.xml	ruleset/rules

Task 7: Auditing Commands on Linux with Wazuh

Auditing commands on Linux with Wazuh involves monitoring and recording user activities and command executions on the system. By deploying the Wazuh agent on Linux systems, organizations can track and analyze command-line activities, enabling enhanced security monitoring and threat detection. This helps to identify potentially malicious or unauthorized actions and strengthens the overall cybersecurity posture on Linux environments. Auditd, which is used in this section, is a user space utility that interacts with

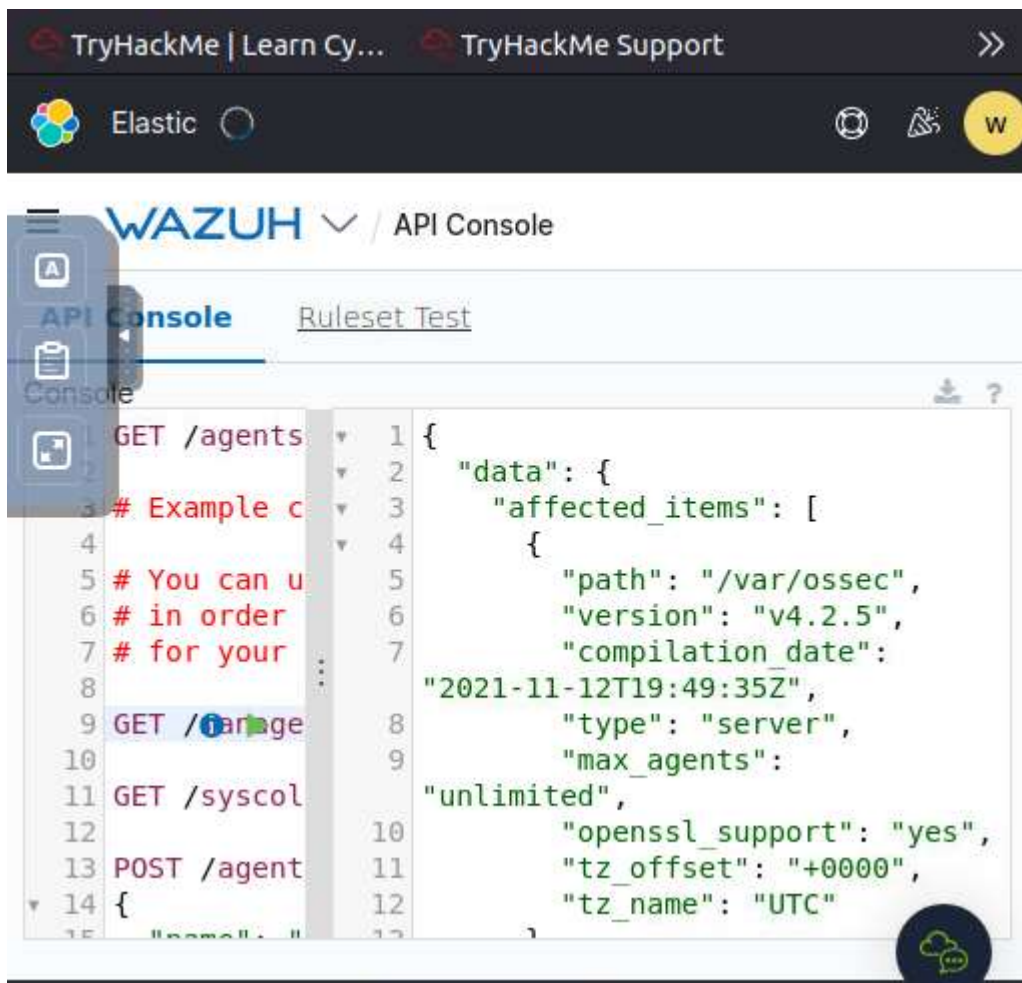
the Linux Audit Framework, providing detailed logging and reporting capabilities for various system events, including command executions. By enabling **auditd**, organizations can create audit rules that specify which system events to monitor, such as command executions, file access, user logins, and more. For the specific task of auditing commands, you can configure **auditd** to track the execution of specific binaries or all commands executed by users. When a monitored event occurs, **auditd** records the event details in the system's audit log files, typically located in **/var/log/audit/audit.log**. These logs contain essential information, including the timestamp, event type, user information, command executed, and more.

Task 8: Wazuh API

The Wazuh API (Application Programming Interface) is a set of endpoints that allows external applications and services to interact with and access the data and functionalities of the Wazuh manager. It enables developers and security teams to integrate Wazuh's capabilities into their own applications, scripts, or security workflows.

With the Wazuh API, users can perform various tasks programmatically, such as:

- Querying and retrieving security events and alerts from the Wazuh manager.
- Managing Wazuh agents, including adding new agents, updating agent configurations, and removing agents.
- Fetching vulnerability assessment reports from integrated vulnerability scanners.
- Interacting with custom decoders and rules to fine-tune the behavior of Wazuh.



Task 9: Generating Reports with Wazuh

Generating reports with Wazuh involves leveraging the Wazuh manager's reporting functionality to compile and present security-related data in a structured and organized manner. Here's a brief overview of how to generate reports with Wazuh:

- **Accessing the Wazuh Manager:** First, log in to the Wazuh manager's web interface, where you can access the reporting feature.

- **Selecting Report Type:** Choose the type of report you want to generate from the available options. Wazuh offers various report types, such as alerts summary, compliance reports, vulnerability assessment reports, and custom reports.
- **Customizing Report Parameters:** Depending on the selected report type, you can specify parameters such as the time range, data filters, and report format.
- **Generating the Report:** Once you have configured the report parameters, initiate the report generation process. Wazuh will process the data based on your settings and generate the report accordingly.
- **Viewing and Exporting the Report:** After the report is generated, you can view it within the Wazuh web interface or export it in different formats like PDF, CSV, or JSON for further analysis or sharing with stakeholders.

Task 10: Loading Sample Data on Wazuh

Loading sample data on Wazuh involves using pre-configured datasets to simulate security events and populate your Wazuh environment with test data. These sample datasets contain various types of security-related events, such as alerts, logs, and vulnerabilities, allowing users to evaluate and familiarize themselves with the Wazuh platform's capabilities without having real-world security incidents.

To load sample data on Wazuh:

- **Accessing Wazuh Manager:** Log in to the Wazuh manager's web interface, where you can manage and configure your Wazuh environment.

- **Enabling Sample Data:** In the Wazuh manager's settings or configurations, look for an option to enable sample data or load demo datasets.
- **Importing Sample Data:** Follow the provided instructions to import the sample data into your Wazuh manager. This process may involve uploading sample log files or configuring virtual agents that generate simulated events.
- **Exploring Sample Data:** Once the sample data is loaded, you can explore the various types of security events and alerts within the Wazuh interface. This allows you to test the functionality of the platform, including threat detection, incident response, and reporting.

By loading sample data on Wazuh, users can practice and experiment with the platform's features in a safe and controlled environment. It is particularly useful for training, testing, and understanding how Wazuh works, helping organizations make better use of the platform to strengthen their cybersecurity defenses.

Here is completion for this module:

Link: <https://tryhackme.com/room/wazuhct>



Conclusion

Through this report and our experience with Wazuh on TryHackMe, we have learned the significance of a robust cybersecurity platform like Wazuh in safeguarding organizations from evolving cyber threats. Wazuh's capabilities in monitoring endpoints, analyzing logs, and correlating security events enable security teams to detect and respond to potential incidents swiftly. Moreover, the Wazuh API offers flexibility for customization and integration with existing security infrastructure, optimizing security operations and fostering a proactive approach to cybersecurity. By generating reports in Wazuh, we have gained valuable insights into security trends and vulnerabilities, enabling us to make informed decisions to strengthen our network defenses. Overall, this module has been enlightening, highlighting the importance of continuous monitoring and vigilance in maintaining a secure and resilient digital environment.

