

BÁO CÁO

Giải pháp SIEM && XDR sử dụng mã nguồn mở Wazuh

Người thực hiện:

Nguyễn Ngọc Anh

Email: ngocanhnguyen99.xyz@gmail.com

Hà Nội - 2024

MỤC LỤC

DANH MỤC CÁC CHỮ VIẾT TẮT	4
DANH MỤC CÁC HÌNH VẼ	5
TÓM TẮT NỘI DUNG	6
LỜI NÓI ĐẦU	7
CHƯƠNG 1: GIỚI THIỆU.....	8
1.1 Tổng quan về SIEM và XDR	8
1.1.1 SIEM (Security Information and Event Management).....	8
1.1.2 XDR (Extended Detection and Response)	9
1.1.3 So sánh giữa SIEM và XDR	10
1.2 Vai trò của Wazuh trong lĩnh vực bảo mật	11
1.2.1 Quản lý sự kiện bảo mật (SIEM)	12
1.2.2 Phát hiện xâm nhập (HIDS/NIDS)	13
1.2.3 Giám sát bảo mật liên tục	13
1.2.4 Tích hợp với Elastic Stack để phân tích dữ liệu bảo mật	13
1.2.5 Đáp ứng các yêu cầu tuân thủ bảo mật	14
1.2.6 Phát hiện và phản ứng mở rộng (XDR)	17
1.3 Mục tiêu và phạm vi dự án	17
1.3.1 Mục tiêu của dự án	17
1.3.2 Phạm vi của dự án	17
CHƯƠNG 2: CƠ SỞ LÝ THUYẾT	19
2.1 Khái niệm SIEM (Security Information and Event Management)	19
2.2 Khái niệm XDR (Extended Detection and Response)	20
2.3 Giới thiệu về Wazuh.....	23
2.4 So sánh giữa các giải pháp SIEM/XDR khác và Wazuh	26
CHƯƠNG 3: PHÂN TÍCH YÊU CẦU	27
3.1 Yêu cầu kỹ thuật.....	27
3.2 Yêu cầu bảo mật	30
3.3 Yêu cầu hệ thống và triển khai	33
3.4 Yêu cầu tích hợp với các hệ thống khác.....	36
CHƯƠNG 4: TRIỂN KHAI VÀ THỰC NGHIỆM	40
4.1 Triển khai Wazuh Server và Agents	40

4.1.1 Cài đặt Wazuh	40
4.1.2 Triển khai giám sát các Agent	52
4.2 Cấu hình Wazuh phát hiện cuộc tấn công Brute-Force.....	54
4.3 Cấu hình Wazuh phát hiện các cuộc tấn công SQL Injection	58
4.4 Cấu hình Wazuh chặn địa chỉ IP độc hại truy cập đến Web Server.....	61
4.5 Tích hợp VirusTotal để phát hiện và xóa các phần mềm độc hại	73
CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN	102
5.1 Tóm tắt kết quả đạt được.....	102
5.2 Những khó khăn và thách thức trong quá trình triển khai.....	102
5.3 Hướng phát triển và cải thiện trong tương lai	103
TÀI LIỆU THAM KHẢO	105

DANH MỤC CÁC CHỮ VIẾT TẮT

AD	Active Directory
API	Application Programming Interface
AWS	Amazon Web Services
FIM	File Integrity Monitoring
GDPR	General Data Protection Regulation
GRC	Governance, risk management and compliance
HIDS	Host-based Intrusion Detection System
IAM	Identity Access Management
IDS	Intrusion Detection System
IPS	Intrusion Prevention System
MFA	Multi-Factor Authentication
NIDS	Network Intrusion Detection System
OAuth	Open standard for Authorization
PCI-DSS	Payment Card Industry Data Security Standard
RBAC	Role-Based Access Control
RDP	Remote Desktop Protocol
SAML	Security Assertion Markup Language
SIEM	Security Information and Event Management
SIM	Security Information Management
SOAR	Security Orchestration, Automation and Response
SSH	Secure Shell
SSO	Single Sign-On
TCP	Transmission Control Protocol
TLS/SSL	Transport Layer Security/ Secure Sockets Layer
UBA	User Behavior Analytics
XDR	Extended Detection and Response

DANH MỤC CÁC HÌNH VẼ

Hình 1: Các thành phần và khả năng của SIEM	8
Hình 2: Lớp bảo mật XDR	10
Hình 3: Sự khác nhau giữa SIEM và XDR	11
Hình 4: Giao diện web của Wazuh.....	12
Hình 5: Nâng cao phân tích bảo mật tích hợp Wazuh với Elastic Stack	14
Hình 6:Tiêu chuẩn GDPR bảo vệ dữ liệu cá nhân	15
Hình 7: Các đối tượng cần tuân thủ HIPAA	16
Hình 8: Các mục tiêu giám sát PCI-DSS	16
Hình 9: So sánh giữa các giải pháp SIEM/XDR khác và Wazuh	26
Hình 10: Mô hình triển khai thực nghiệm.....	40

TÓM TẮT NỘI DUNG

Báo cáo "**Giải pháp SIEM && XDR sử dụng mã nguồn mở Wazuh**" trình bày quá trình nghiên cứu, triển khai và đánh giá hệ thống bảo mật dựa trên nền tảng mã nguồn mở Wazuh, với mục tiêu xây dựng một giải pháp **SIEM** (Security Information and Event Management) và **XDR** (Extended Detection and Response) tối ưu, tiết kiệm chi phí nhưng vẫn đảm bảo tính hiệu quả trong giám sát và bảo vệ hệ thống mạng.

Nội dung báo cáo bao gồm các phần chính như sau:

- 1. Tổng quan về SIEM và XDR:* Giới thiệu khái niệm, vai trò và tầm quan trọng của các giải pháp SIEM và XDR trong an ninh mạng. Báo cáo cũng so sánh các hệ thống SIEM/XDR khác và làm rõ ưu điểm của Wazuh.
- 2. Giới thiệu về Wazuh:* Trình bày chi tiết về nền tảng Wazuh, một giải pháp mã nguồn mở tích hợp nhiều tính năng quan trọng như quản lý sự kiện bảo mật, phát hiện các mối đe dọa và khả năng phản ứng tự động.
- 3. Yêu cầu kỹ thuật và triển khai hệ thống:* Phân tích các yêu cầu về kỹ thuật, bảo mật, hệ thống và tích hợp cần thiết để triển khai Wazuh trong môi trường doanh nghiệp. Các bước triển khai Wazuh Server và Agents, cũng như cấu hình các chính sách bảo mật, được mô tả chi tiết trong phần này.
- 4. Triển khai và thực nghiệm:* Mô tả quá trình triển khai thực tế hệ thống Wazuh, bao gồm những khó khăn và thách thức đã gặp phải, từ việc tích hợp hệ thống đến tối ưu hóa khả năng phân tích log và quản lý mối đe dọa.
- 5. Kết quả đạt được và hướng phát triển:* Tổng kết các kết quả đạt được sau khi triển khai hệ thống, đánh giá hiệu quả trong việc giám sát, phát hiện và xử lý các mối đe dọa an ninh mạng. Phần này cũng đề xuất các hướng phát triển và cải thiện hệ thống trong tương lai, bao gồm tối ưu hóa hiệu suất, tích hợp thêm các giải pháp bảo mật khác, và cải thiện khả năng phản ứng tự động.

LỜI NÓI ĐẦU

Trong bối cảnh công nghệ thông tin phát triển mạnh mẽ, an ninh mạng đã trở thành một yếu tố vô cùng quan trọng đối với mọi tổ chức, từ các doanh nghiệp đến các cơ quan chính phủ. Hệ thống mạng của các tổ chức ngày càng phức tạp, đi kèm với sự gia tăng của các cuộc tấn công mạng, đòi hỏi các giải pháp bảo mật toàn diện và hiệu quả hơn để giám sát, phát hiện và phản ứng nhanh chóng với các mối đe dọa.

Với sự xuất hiện của các công nghệ như **SIEM** (Security Information and Event Management) và **XDR** (Extended Detection and Response), các doanh nghiệp có thể theo dõi và quản lý các sự kiện an ninh một cách tập trung, phát hiện sớm những bất thường và bảo vệ tài sản thông tin một cách chủ động. Tuy nhiên, chi phí triển khai các giải pháp thương mại thường khá cao, đòi hỏi đầu tư lớn vào hạ tầng và giấy phép phần mềm.

Trong bối cảnh đó, mã nguồn mở đã trở thành một lựa chọn hấp dẫn nhờ tính linh hoạt và tiết kiệm chi phí. Wazuh – một nền tảng mã nguồn mở nổi bật trong lĩnh vực giám sát an ninh, kết hợp giữa khả năng thu thập, phân tích log và quản lý các mối đe dọa, đã được nhiều tổ chức tin dùng. Dự án "**Giải pháp SIEM && XDR sử dụng mã nguồn mở Wazuh**" không chỉ nhằm nghiên cứu và triển khai Wazuh như một công cụ bảo mật mạnh mẽ, mà còn giúp xây dựng một mô hình bảo mật tối ưu dựa trên nhu cầu thực tế của tổ chức, từ đó nâng cao khả năng phát hiện và xử lý sự cố an ninh mạng.

CHƯƠNG 1: GIỚI THIỆU

1.1 Tổng quan về SIEM và XDR

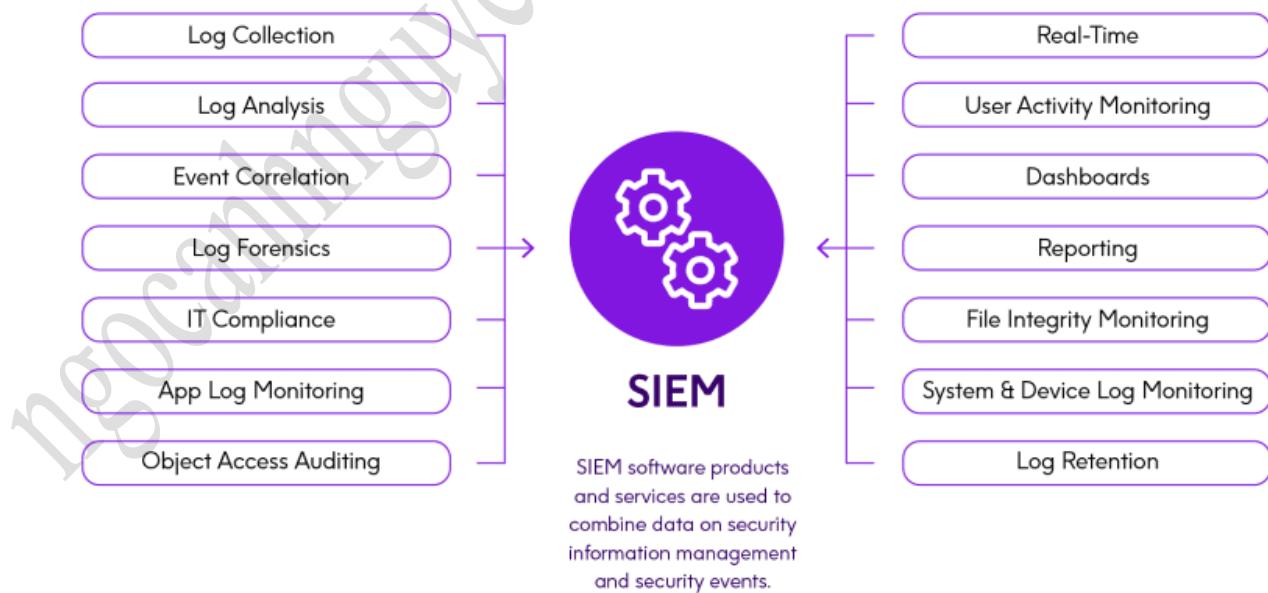
SIEM (Security Information and Event Management) và *XDR (Extended Detection and Response)* là hai giải pháp bảo mật quan trọng trong lĩnh vực quản lý và phát hiện mối đe dọa an ninh mạng. Cả hai đều có mục tiêu chính là giúp các tổ chức phát hiện, phân tích và phản ứng nhanh chóng với các sự cố bảo mật, nhưng cách thức tiếp cận và công nghệ sử dụng lại khác nhau.

1.1.1 SIEM (Security Information and Event Management)

SIEM là một hệ thống quản lý thông tin và sự kiện bảo mật, kết hợp hai yếu tố chính:

- *SIM (Security Information Management)*: Thu thập, lưu trữ và phân tích dữ liệu nhật ký từ nhiều nguồn khác nhau, bao gồm máy chủ, thiết bị mạng, và ứng dụng. Từ đó, nó giúp tổ chức duy trì nhật ký hoạt động và tuân thủ các quy định bảo mật.

- *SEM (Security Event Management)*: Tập trung vào việc phân tích các sự kiện bảo mật thời gian thực, nhằm phát hiện các hành vi bất thường hoặc nguy cơ bảo mật.



Hình 1: Các thành phần và khả năng của SIEM

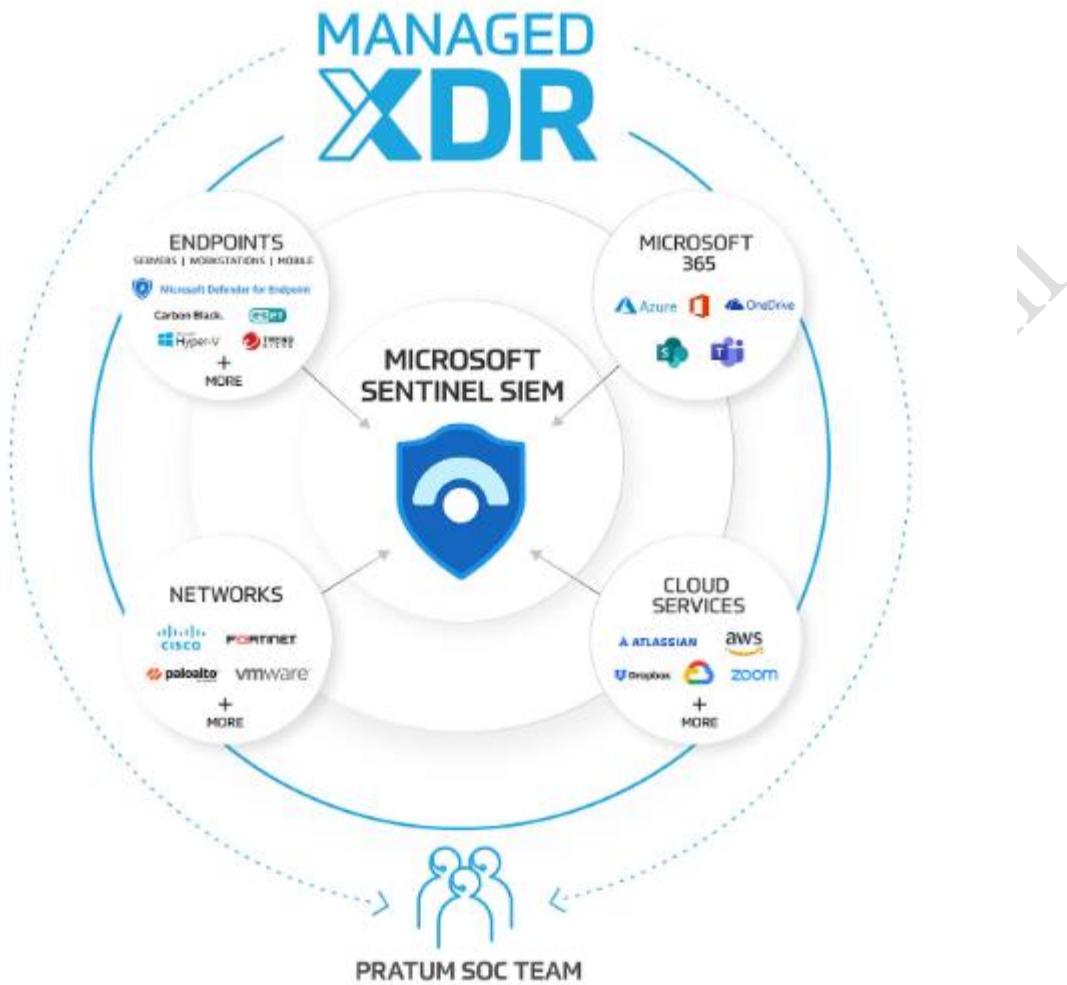
SIEM hoạt động dựa trên việc thu thập dữ liệu từ nhiều nguồn khác nhau trong hệ thống và sau đó phân tích, so sánh dữ liệu để phát hiện các mối đe dọa hoặc sự kiện bất thường. Các tính năng chính của SIEM bao gồm:

- *Thu thập và tập trung dữ liệu* từ nhiều hệ thống khác nhau.
- *Phân tích nhật ký* để tìm kiếm các hành vi bất thường.
- *Phát hiện mối đe dọa* dựa trên các luật và quy tắc đã được thiết lập.
- *Báo cáo và cảnh báo* khi có sự cố bảo mật xảy ra.
- *Quản lý tuân thủ* giúp đảm bảo tổ chức tuân thủ các tiêu chuẩn và quy định bảo mật như GDPR, HIPAA, PCI-DSS.

1.1.2 XDR (Extended Detection and Response)

XDR là một giải pháp bảo mật hiện đại và mở rộng hơn so với SIEM. Trong khi SIEM chủ yếu tập trung vào việc thu thập và phân tích nhật ký, XDR cung cấp khả năng phản hồi và quản lý mối đe dọa toàn diện trên nhiều lớp bảo mật khác nhau, bao gồm:

- *Endpoint (Thiết bị đầu cuối)*
- *Network (Mạng)*
- *Email*
- *Cloud*
- *Ứng dụng*



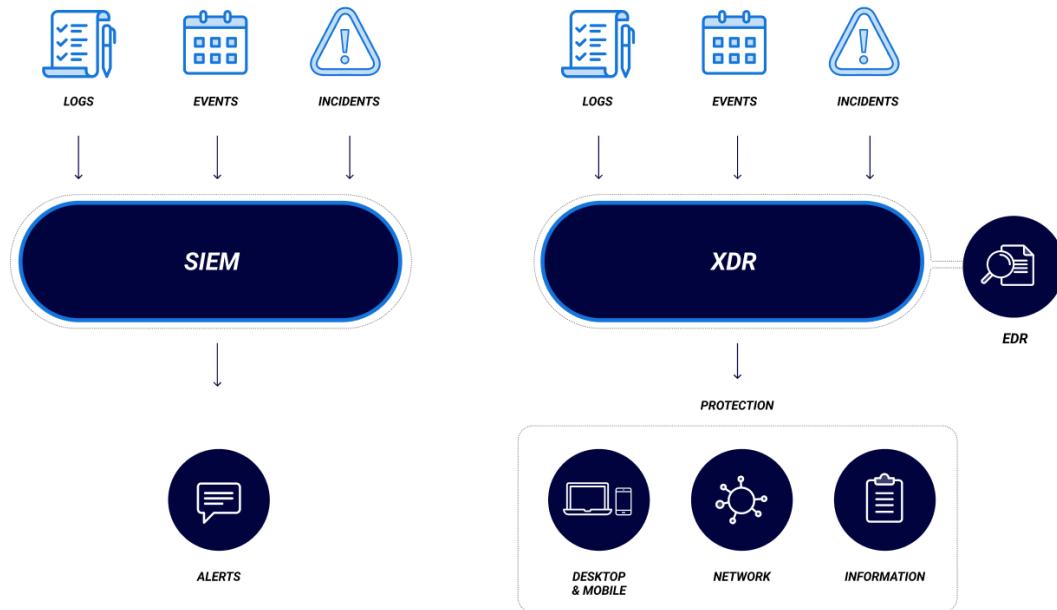
Hình 2: Lớp bảo mật XDR

Các điểm nổi bật của XDR bao gồm:

- *Tự động phát hiện và phản hồi* các mối đe dọa dựa trên trí tuệ nhân tạo (AI) và học máy (Machine Learning).
- *Tích hợp nhiều lớp bảo mật* để cung cấp tầm nhìn toàn diện về môi trường bảo mật của tổ chức.
- *Phân tích sâu các mối đe dọa* từ nhiều nguồn khác nhau, giúp nâng cao khả năng phát hiện và giảm thiểu rủi ro.
- *Phản ứng tự động* đối với các sự cố bảo mật, giảm thời gian phản hồi và xử lý mối đe dọa.

1.1.3 So sánh giữa SIEM và XDR

SIEM VS. XDR



Hình 3: Sự khác nhau giữa SIEM và XDR

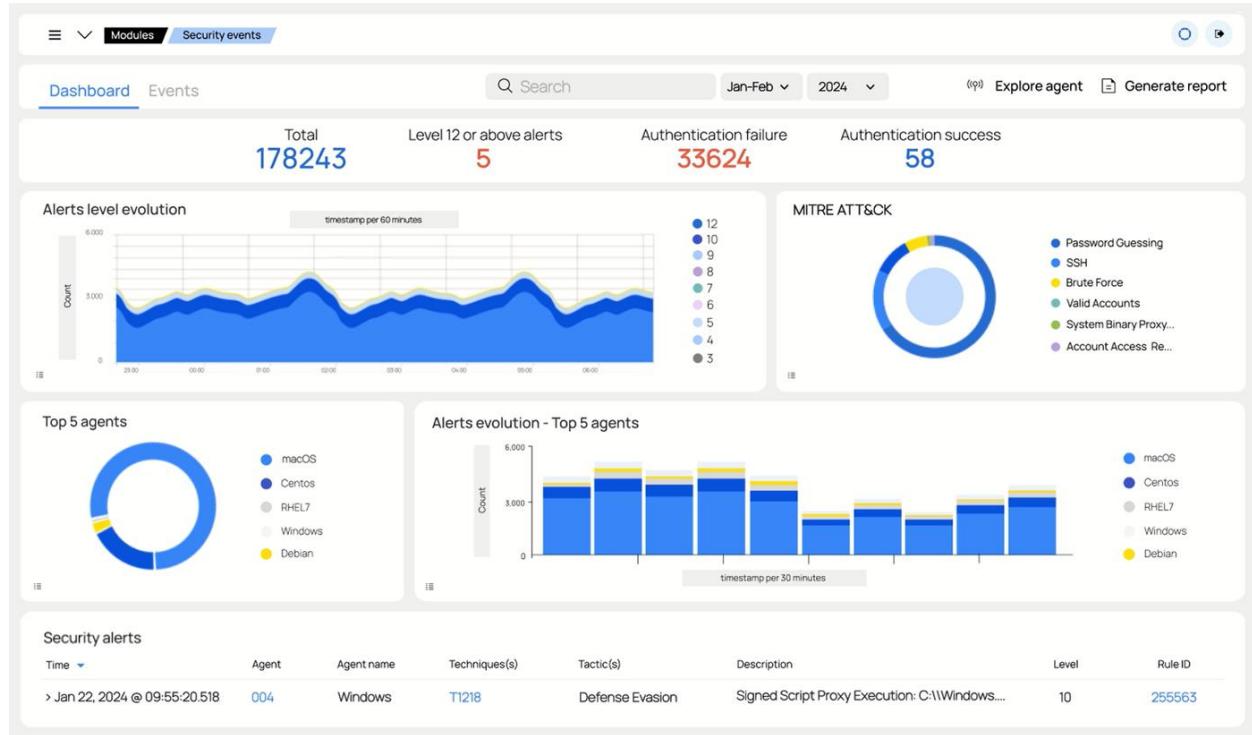
- *SIEM* tập trung vào việc thu thập và phân tích dữ liệu từ nhiều hệ thống và ứng dụng, hỗ trợ tuân thủ và phát hiện mối đe dọa thông qua quy tắc và quy định.
- *XDR* là giải pháp toàn diện hơn, tích hợp các chức năng của SIEM và mở rộng với khả năng phản ứng tự động, kết hợp nhiều lớp bảo mật để tối ưu hóa khả năng phát hiện và phản ứng trước các mối đe dọa phức tạp.

SIEM và XDR đều có giá trị riêng trong môi trường bảo mật hiện đại. SIEM chủ yếu phục vụ việc tuân thủ và báo cáo sự kiện, trong khi XDR giúp tối ưu hóa quy trình phát hiện và phản ứng trước các mối đe dọa phức tạp.

1.2 Vai trò của Wazuh trong lĩnh vực bảo mật

Wazuh là một nền tảng mã nguồn mở mạnh mẽ, được thiết kế để cung cấp các giải pháp bảo mật toàn diện, chủ yếu tập trung vào quản lý sự kiện và phát hiện các mối đe dọa bảo mật. Wazuh đóng vai trò quan trọng trong việc giúp các tổ chức

xây dựng và duy trì hệ thống bảo mật chủ động thông qua khả năng giám sát, phát hiện, phân tích, và phản ứng với các sự cố an ninh mạng.



Hình 4: Giao diện web của Wazuh

1.2.1 Quản lý sự kiện bảo mật (SIEM)

Wazuh hoạt động như một giải pháp *SIEM* (*Security Information and Event Management*), giúp thu thập, phân tích và quản lý dữ liệu bảo mật từ nhiều nguồn khác nhau, bao gồm máy chủ, thiết bị mạng, ứng dụng, và các dịch vụ đám mây.

Một số vai trò chính của Wazuh trong quản lý sự kiện bảo mật bao gồm:

- *Thu thập dữ liệu bảo mật* từ nhiều nguồn thông qua các agent hoặc tích hợp với các hệ thống khác như tường lửa, hệ điều hành, cơ sở dữ liệu, và ứng dụng.
- *Phân tích và tương quan sự kiện* nhằm phát hiện các hành vi bất thường hoặc các mối đe dọa tiềm ẩn dựa trên các quy tắc đã được thiết lập trước.
- *Cảnh báo theo thời gian thực* khi phát hiện các mối đe dọa hoặc hành vi bất thường, từ đó cung cấp thông tin chi tiết cho đội ngũ bảo mật để đưa ra phản ứng kịp thời.
- *Tạo báo cáo tuân thủ* cho các tiêu chuẩn bảo mật như GDPR, HIPAA, PCI-DSS, giúp các tổ chức duy trì và chứng minh việc tuân thủ các quy định bảo mật quan trọng.

1.2.2 Phát hiện xâm nhập (HIDS/NIDS)

Wazuh tích hợp khả năng *HIDS* (*Host-based Intrusion Detection System*), cung cấp khả năng phát hiện xâm nhập ở cấp độ máy chủ thông qua việc giám sát các thay đổi trên hệ thống và phát hiện các hành vi bất thường.

Nó giúp theo dõi:

- *Nhật ký hệ thống*: Wazuh theo dõi và phân tích các nhật ký của hệ điều hành, thiết bị mạng, và ứng dụng để phát hiện các hoạt động đáng ngờ như đăng nhập thất bại, cố gắng truy cập trái phép, và các thay đổi không hợp lệ trong cấu hình.
- *Tệp tin và cấu hình hệ thống*: Wazuh có thể giám sát tính toàn vẹn của tệp tin và hệ thống (FIM - File Integrity Monitoring), cảnh báo khi có bất kỳ thay đổi trái phép nào xảy ra.
- *Phát hiện phần mềm độc hại*: Wazuh có thể so sánh các tệp tin hệ thống với cơ sở dữ liệu về phần mềm độc hại để phát hiện và cảnh báo về các tệp tin hoặc quy trình có dấu hiệu nhiễm mã độc.

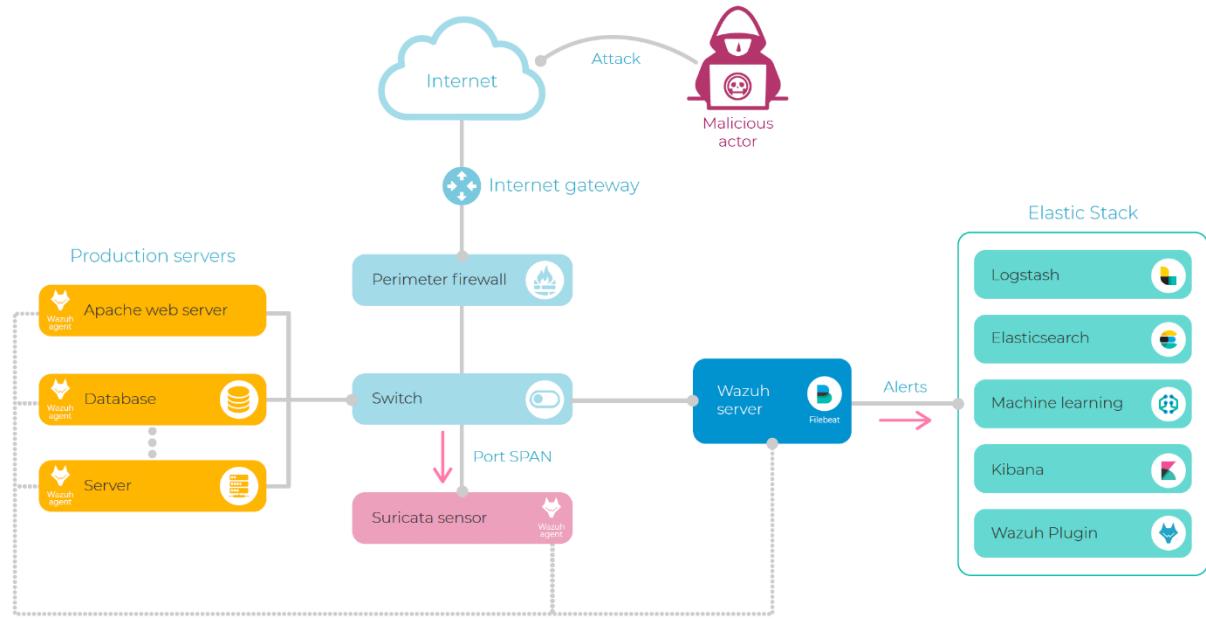
1.2.3 Giám sát bảo mật liên tục

Wazuh đóng vai trò như một nền tảng *giám sát bảo mật liên tục* bằng cách thu thập và phân tích thông tin bảo mật trong thời gian thực từ các máy chủ, ứng dụng và mạng lưới. Nó giúp:

- *Phát hiện mối đe dọa kịp thời*: Với khả năng phân tích dựa trên các mẫu tấn công và hành vi bất thường, Wazuh giúp phát hiện sớm các cuộc tấn công hoặc hành vi xâm phạm an ninh, từ đó giảm thiểu thiệt hại.
- *Tăng cường khả năng phản ứng*: Wazuh cung cấp khả năng tích hợp với các hệ thống khác để tự động hóa việc phản hồi khi phát hiện mối đe dọa, giảm thời gian phản ứng.
- *Tối ưu hóa hiệu suất bảo mật*: Wazuh cung cấp các bảng điều khiển (dashboard) trực quan giúp quản trị viên dễ dàng theo dõi và đánh giá trạng thái an ninh của hệ thống một cách nhanh chóng và toàn diện.

1.2.4 Tích hợp với Elastic Stack để phân tích dữ liệu bảo mật

Wazuh tích hợp chặt chẽ với *Elastic Stack* (*Elasticsearch, Logstash, và Kibana*) để cung cấp khả năng lưu trữ, phân tích, và trực quan hóa dữ liệu bảo mật.



Hình 5: Nâng cao phân tích bảo mật tích hợp Wazuh với Elastic Stack

Sự kết hợp này mang lại lợi ích vượt trội trong việc:

- *Lưu trữ dữ liệu an toàn và hiệu quả*: Elasticsearch giúp lưu trữ khối lượng lớn dữ liệu bảo mật, đồng thời cung cấp các công cụ tìm kiếm và phân tích mạnh mẽ.
- *Trực quan hóa dữ liệu bảo mật*: Kibana cung cấp các giao diện đồ họa trực quan để hiển thị dữ liệu bảo mật, giúp quản trị viên dễ dàng theo dõi và phân tích tình hình bảo mật trong thời gian thực.
- *Tự động hóa quy trình phân tích*: Logstash xử lý và định dạng dữ liệu từ nhiều nguồn khác nhau, đảm bảo tính toàn vẹn và đầy đủ của thông tin được gửi tới Wazuh để phân tích.

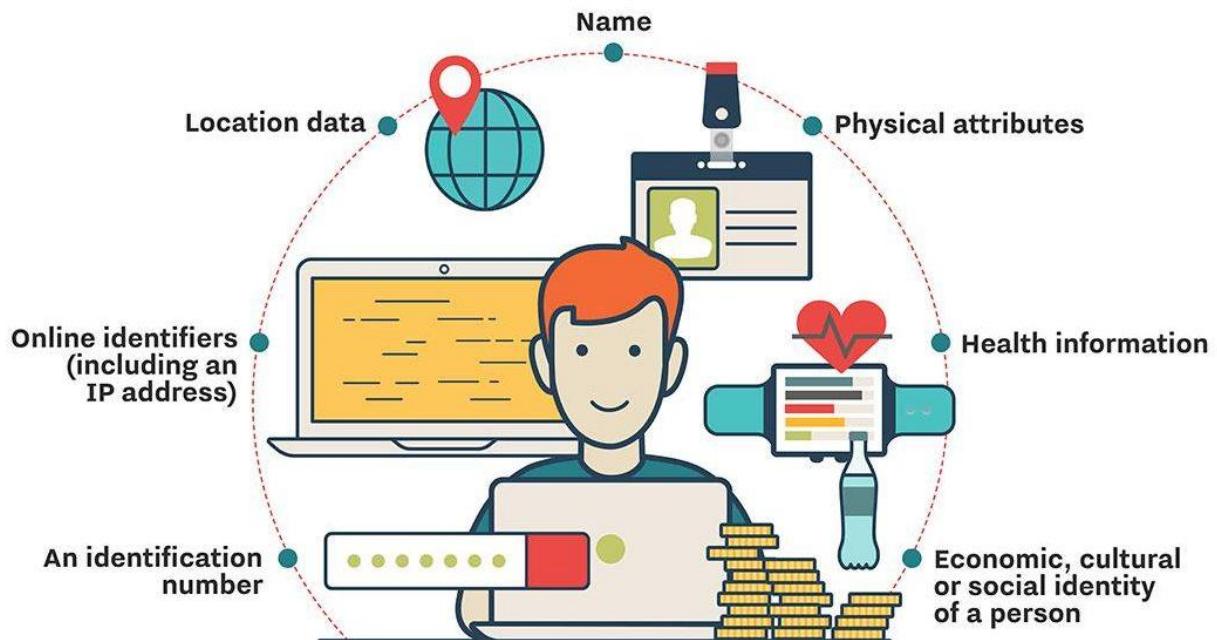
1.2.5 Đáp ứng các yêu cầu tuân thủ bảo mật

Wazuh đóng vai trò quan trọng trong việc giúp các tổ chức *tuân thủ các tiêu chuẩn và quy định bảo mật*. Nó cung cấp các báo cáo chi tiết và tự động về các sự kiện bảo mật, giúp tổ chức chứng minh việc tuân thủ các quy định về bảo mật thông tin như:

- *GDPR*: Đảm bảo rằng tổ chức xử lý và bảo vệ dữ liệu cá nhân theo quy định của Liên minh Châu Âu.

GDPR PERSONAL DATA

The EU's General Data Protection Regulation defines personal data as any information related to a person that can be used to directly or indirectly identify them, including:



Hình 6: Tiêu chuẩn GDPR bảo vệ dữ liệu cá nhân

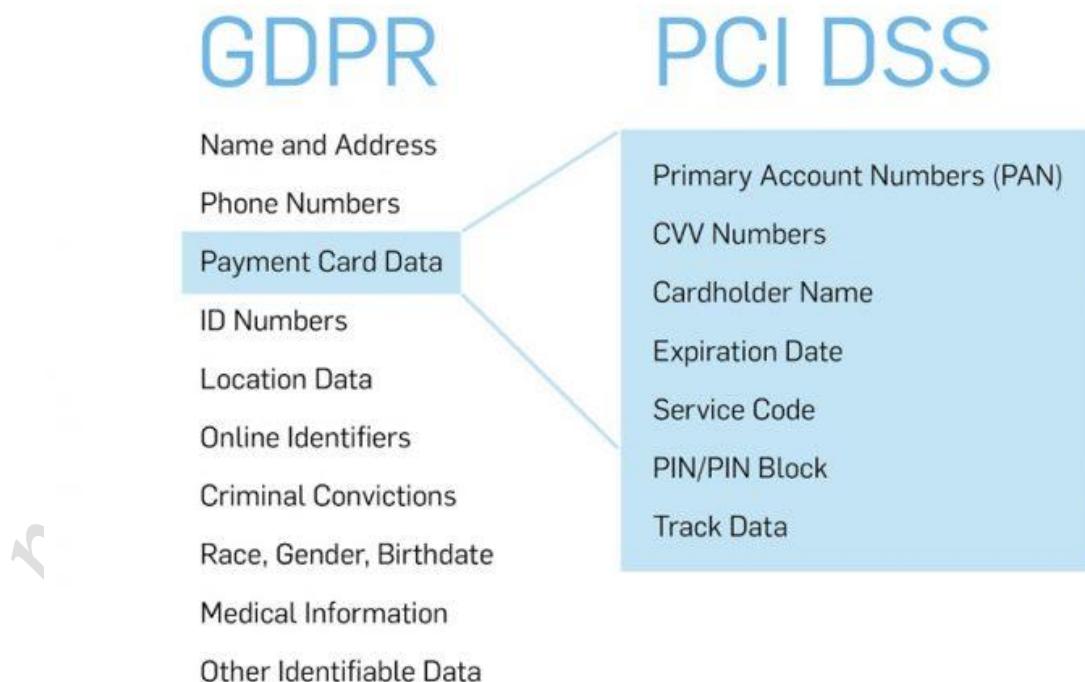
- HIPAA: Hỗ trợ tổ chức y tế tuân thủ các tiêu chuẩn bảo mật dữ liệu sức khỏe.

ĐỐI TƯỢNG CẦN TUÂN THỦ LUẬT HIPAA



Hình 7: Các đối tượng cần tuân thủ HIPAA

- *PCI-DSS*: Giúp các tổ chức quản lý thông tin thẻ tín dụng theo yêu cầu của Hội đồng Chuẩn thanh toán.



Hình 8: Các mục tiêu giám sát PCI-DSS

1.2.6 Phát hiện và phản ứng mở rộng (XDR)

Wazuh cung cấp khả năng *mở rộng phát hiện và phản ứng* (*XDR*), cho phép theo dõi và xử lý sự cố trên nhiều lớp bảo mật khác nhau, bao gồm mạng, thiết bị đầu cuối, và các ứng dụng. Điều này giúp tổ chức có tầm nhìn toàn diện và khả năng phản ứng nhanh chóng với các mối đe dọa phức tạp trong môi trường bảo mật hiện đại.

1.3 Mục tiêu và phạm vi dự án

1.3.1 Mục tiêu của dự án

Dự án “*Giải pháp SIEM & XDR sử dụng mã nguồn mở Wazuh*” nhằm mục tiêu xây dựng một hệ thống bảo mật toàn diện cho tổ chức, giúp phát hiện và phản ứng nhanh chóng với các mối đe dọa bảo mật. Các mục tiêu chính của dự án bao gồm:

- 1. Xây dựng giải pháp SIEM hiệu quả:** Tận dụng Wazuh để triển khai một hệ thống SIEM, có khả năng thu thập và phân tích dữ liệu bảo mật từ nhiều nguồn khác nhau trong thời gian thực, giúp giám sát các sự kiện và phát hiện mối đe dọa một cách hiệu quả.
- 2. Tích hợp khả năng XDR (Extended Detection and Response):** Mở rộng khả năng phát hiện và phản ứng với các mối đe dọa trên toàn hệ thống, từ endpoint (thiết bị đầu cuối), ứng dụng, đến mạng lưới và các dịch vụ đám mây. Điều này giúp cải thiện khả năng bảo mật tổng thể, tăng cường tầm nhìn và khả năng phản ứng với các cuộc tấn công mạng phức tạp.
- 3. Tối ưu hóa quá trình giám sát bảo mật:** Tự động hóa quá trình phát hiện và phản ứng với sự cố bảo mật, giảm thiểu thời gian phát hiện và phản ứng trước các mối đe dọa tiềm ẩn.
- 4. Đáp ứng yêu cầu tuân thủ bảo mật:** Đảm bảo rằng hệ thống tuân thủ các tiêu chuẩn bảo mật và quy định quốc tế như GDPR, HIPAA, PCI-DSS, giúp bảo vệ dữ liệu và ngăn ngừa vi phạm pháp lý.
- 5. Giảm chi phí và tối ưu hóa tài nguyên:** Sử dụng nền tảng mã nguồn mở Wazuh để triển khai một giải pháp bảo mật toàn diện với chi phí thấp hơn so với các giải pháp thương mại, đồng thời giảm sự phụ thuộc vào các nhà cung cấp độc quyền.

1.3.2 Phạm vi của dự án

Phạm vi của dự án bao gồm các hoạt động cụ thể dưới đây, nhằm đảm bảo xây dựng và triển khai thành công giải pháp *SIEM && XDR* dựa trên nền tảng Wazuh:

1. Cài đặt và cấu hình Wazuh:

- Triển khai Wazuh trên hạ tầng mạng của tổ chức, bao gồm việc cài đặt máy chủ Wazuh và các agent trên các thiết bị đầu cuối, máy chủ và thiết bị mạng.
- Tích hợp Wazuh với các nguồn nhật ký bảo mật chính (firewall, máy chủ, thiết bị mạng, ứng dụng, dịch vụ đám mây, v.v.).

2. Thu thập và phân tích dữ liệu bảo mật:

- Xác định và thu thập dữ liệu từ các hệ thống cần thiết để đảm bảo phát hiện và giám sát toàn diện.
- Cấu hình các quy tắc giám sát và phát hiện sự kiện bảo mật, phân tích dữ liệu để nhận diện các mối đe dọa tiềm ẩn.

3. Tích hợp với các công cụ phân tích và hiển thị dữ liệu:

- Tích hợp Wazuh với *Elastic Stack* (Elasticsearch, Logstash, Kibana) để lưu trữ, xử lý và trực quan hóa dữ liệu bảo mật.
- Xây dựng các dashboard tùy chỉnh để giám sát thời gian thực và báo cáo các sự kiện bảo mật.

4. Xây dựng quy trình phản ứng sự cố:

- Thiết lập quy trình phản hồi tự động khi phát hiện sự cố bảo mật, từ việc cảnh báo cho đội ngũ bảo mật đến thực hiện các hành động phản ứng (chặn IP, cách ly thiết bị, gửi cảnh báo qua email).
- Đào tạo đội ngũ IT và bảo mật về cách xử lý sự cố dựa trên cảnh báo và báo cáo từ hệ thống Wazuh.

5. Đảm bảo tuân thủ và báo cáo bảo mật:

- Xây dựng các báo cáo tuân thủ dựa trên dữ liệu thu thập từ Wazuh, đảm bảo hệ thống tuân thủ các tiêu chuẩn như GDPR, HIPAA, và PCI-DSS.
- Tạo ra các báo cáo định kỳ về tình trạng bảo mật của tổ chức.

6. Kiểm tra và đánh giá hệ thống:

- Thực hiện kiểm thử hệ thống SIEM & XDR để đánh giá hiệu quả phát hiện và phản ứng với các mối đe dọa.

- Đánh giá hiệu suất và độ ổn định của hệ thống, từ đó tối ưu hóa cấu hình và khả năng mở rộng.

7. Bảo trì và cải tiến hệ thống:

- Thiết lập các quy trình bảo trì định kỳ cho hệ thống Wazuh, đảm bảo tính khả dụng và bảo mật cao nhất.
- Cập nhật hệ thống thường xuyên để tích hợp các tính năng mới và vá các lỗ hổng bảo mật.

CHƯƠNG 2: CƠ SỞ LÝ THUYẾT

2.1 Khái niệm SIEM (Security Information and Event Management)

SIEM (Security Information and Event Management) là một giải pháp bảo mật tổng hợp, kết hợp các chức năng của quản lý thông tin bảo mật (*Security Information Management - SIM*) và quản lý sự kiện bảo mật (*Security Event Management - SEM*) nhằm cung cấp khả năng giám sát, phát hiện và phản ứng với các mối đe dọa an ninh mạng trong thời gian thực.

SIEM có khả năng thu thập và phân tích dữ liệu bảo mật từ nhiều nguồn khác nhau như máy chủ, thiết bị mạng, ứng dụng, cơ sở dữ liệu và hệ thống đầu cuối. Nó giúp tổ chức có một tầm nhìn toàn diện về các sự kiện và hoạt động bảo mật xảy ra trong hệ thống, đồng thời cung cấp các công cụ và tính năng cần thiết để phát hiện các cuộc tấn công, vi phạm và các mối đe dọa tiềm ẩn.

Các chức năng chính của SIEM:

1. *Thu thập dữ liệu bảo mật*: SIEM thu thập dữ liệu từ nhiều nguồn khác nhau trong hệ thống như tường lửa, hệ điều hành, ứng dụng, cơ sở dữ liệu và thiết bị mạng. Những dữ liệu này bao gồm các bản ghi sự kiện (event logs), nhật ký hệ thống (system logs), và các nhật ký bảo mật (security logs).
2. *Lưu trữ và quản lý nhật ký*: Dữ liệu được thu thập bởi SIEM sẽ được lưu trữ và quản lý một cách tập trung, đảm bảo tính toàn vẹn và sẵn sàng cho việc phân tích sau này. SIEM có thể lưu trữ dữ liệu trong thời gian dài, đáp ứng các yêu cầu về tuân thủ pháp luật và quy định bảo mật.

3. Phân tích và tương quan sự kiện: SIEM sử dụng các quy tắc, thuật toán và máy học để phân tích dữ liệu bảo mật và phát hiện các sự kiện đáng ngờ hoặc bất thường. Tính năng tương quan cho phép SIEM kết hợp các sự kiện từ nhiều nguồn khác nhau để phát hiện các mẫu hành vi có thể là dấu hiệu của một cuộc tấn công.

4. Cảnh báo và phản ứng tự động: Khi phát hiện một sự kiện bất thường hoặc mối đe dọa, SIEM có thể tự động phát ra cảnh báo cho quản trị viên bảo mật. Ngoài ra, hệ thống còn có thể tự động thực hiện các hành động phòng thủ như chặn IP, cách ly thiết bị bị xâm nhập, hoặc gửi cảnh báo qua email.

5. Báo cáo và tuân thủ: SIEM cung cấp các báo cáo chi tiết về các sự kiện bảo mật và các hoạt động liên quan đến bảo mật. Các báo cáo này có thể được sử dụng để chứng minh tính tuân thủ với các tiêu chuẩn và quy định bảo mật như *PCI-DSS*, *HIPAA*, *GDPR*, và nhiều tiêu chuẩn khác.

Lợi ích của SIEM:

- *Giám sát bảo mật tập trung:* SIEM cung cấp một nền tảng duy nhất để giám sát và quản lý tất cả các sự kiện bảo mật trong hệ thống. Điều này giúp các tổ chức dễ dàng quản lý và theo dõi tình trạng an ninh của mình.

- *Phát hiện và phản ứng với các mối đe dọa nhanh chóng:* Với khả năng phân tích thời gian thực, SIEM giúp tổ chức phát hiện và phản ứng với các cuộc tấn công mạng nhanh chóng, giảm thiểu thiệt hại và ngăn chặn sự lan rộng của mối đe dọa.

- *Hỗ trợ tuân thủ quy định:* SIEM giúp tổ chức thu thập và lưu trữ dữ liệu bảo mật theo các tiêu chuẩn pháp luật và quy định bảo mật, từ đó hỗ trợ trong việc tuân thủ các yêu cầu về bảo mật.

- *Phân tích sâu và điều tra sự cố:* SIEM cung cấp các công cụ để phân tích chi tiết sự kiện bảo mật và điều tra nguyên nhân gốc rễ của các cuộc tấn công. Điều này giúp tổ chức không chỉ phản ứng mà còn học hỏi và cải thiện chiến lược phòng thủ an ninh của mình.

2.2 Khái niệm XDR (Extended Detection and Response)

XDR là một giải pháp an ninh mạng tiên tiến, được thiết kế để mở rộng khả năng phát hiện và phản ứng với các mối đe dọa trên nhiều lớp khác nhau của hệ thống, bao gồm thiết bị đầu cuối (endpoints), mạng lưới, máy chủ, ứng dụng và đám mây. Không giống như các hệ thống bảo mật truyền thống chỉ tập trung vào một khía cạnh hoặc

lớp của hệ thống, XDR kết hợp và tương quan dữ liệu từ nhiều nguồn để cung cấp một tầm nhìn toàn diện về các sự kiện và mối đe dọa bảo mật.

XDR được xem là sự phát triển của các giải pháp như *EDR* (*Endpoint Detection and Response*), giúp mở rộng khả năng phát hiện và phản ứng từ chỉ các thiết bị đầu cuối sang toàn bộ môi trường bảo mật của tổ chức. Điều này giúp tổ chức bảo vệ toàn diện và phản ứng nhanh chóng hơn trước các cuộc tấn công phức tạp và có tổ chức.

Các tính năng chính của XDR:

1. *Phát hiện và phản ứng trên nhiều lớp bảo mật*: XDR có khả năng thu thập dữ liệu và phát hiện các mối đe dọa trên nhiều lớp của hệ thống, bao gồm:

- Endpoint (thiết bị đầu cuối): XDR có khả năng giám sát và phát hiện các sự kiện bảo mật xảy ra trên máy tính cá nhân, điện thoại di động và các thiết bị kết nối khác.
- Mạng lưới: XDR thu thập dữ liệu từ các thiết bị mạng như tường lửa, bộ định tuyến (routers), bộ chuyển mạch (switches), và các hệ thống phát hiện xâm nhập (IDS/IPS).
- Máy chủ và ứng dụng: Giám sát hoạt động của máy chủ và các ứng dụng quan trọng, bao gồm cả hệ thống quản lý cơ sở dữ liệu và các dịch vụ web.
- Dịch vụ đám mây: XDR theo dõi các hoạt động trên các dịch vụ đám mây, như các nền tảng điện toán đám mây (AWS, Azure, Google Cloud) và các ứng dụng SaaS.

2. *Tích hợp và tương quan dữ liệu*: XDR thu thập và tích hợp dữ liệu từ nhiều nguồn bảo mật khác nhau và sử dụng các thuật toán phân tích nâng cao để tương quan các sự kiện từ nhiều lớp bảo mật. Điều này giúp phát hiện các mẫu hành vi phức tạp và các mối đe dọa có tổ chức mà các hệ thống bảo mật riêng lẻ có thể bỏ sót.

3. *Tự động hóa phản ứng*: Khi phát hiện các sự kiện bất thường hoặc các mối đe dọa, XDR có thể tự động thực hiện các hành động phản ứng như cách ly thiết bị bị xâm nhập, chặn IP, hoặc triển khai các biện pháp bảo mật khác nhằm giảm thiểu rủi ro.

4. *Cung cấp tầm nhìn tổng thể*: XDR cung cấp một tầm nhìn tổng thể và dễ hiểu về các sự kiện bảo mật trong toàn bộ hệ thống của tổ chức. Điều này cho phép đội ngũ bảo mật có cái nhìn bao quát và nhanh chóng hiểu được tình trạng an ninh hiện tại, từ đó thực hiện các biện pháp phòng ngừa và phản ứng kịp thời.

5. Phát hiện mối đe dọa tiên tiến: XDR tích hợp các công cụ phân tích hành vi và trí tuệ nhân tạo (AI) để phát hiện các mối đe dọa tiên tiến như các cuộc tấn công zero-day hoặc các cuộc tấn công đã qua mặt được các hệ thống bảo mật truyền thống.

Lợi ích của XDR:

- *Tích hợp bảo mật toàn diện:* XDR giúp tổ chức bảo vệ toàn bộ hệ sinh thái IT, từ thiết bị đầu cuối, hệ thống mạng đến các ứng dụng và dịch vụ đám mây. Điều này giúp cải thiện hiệu quả bảo mật tổng thể và giảm thiểu khoảng trống trong việc phát hiện mối đe dọa.
- *Phát hiện mối đe dọa sớm hơn:* Với khả năng thu thập và phân tích dữ liệu từ nhiều lớp bảo mật, XDR có thể phát hiện các cuộc tấn công phức tạp trước khi chúng gây ra thiệt hại lớn. Hệ thống cảnh báo sớm giúp tổ chức phản ứng nhanh chóng và hiệu quả.
- *Phản ứng tự động và nhanh chóng:* Khả năng tự động phản ứng với các sự cố bảo mật giúp giảm thiểu thiệt hại do các cuộc tấn công gây ra. XDR có thể ngay lập tức thực hiện các biện pháp bảo vệ cần thiết, mà không cần sự can thiệp của con người trong nhiều trường hợp.
- *Giảm thiểu khối lượng công việc và chi phí:* XDR giúp tối ưu hóa quy trình bảo mật bằng cách tự động hóa các nhiệm vụ giám sát, phát hiện và phản ứng với sự cố bảo mật. Điều này giúp giảm khối lượng công việc cho đội ngũ bảo mật, đồng thời tiết kiệm chi phí vận hành.
- *Cải thiện khả năng điều tra và ứng phó sự cố:* XDR cung cấp các công cụ mạnh mẽ để điều tra sâu hơn về các cuộc tấn công, giúp tổ chức nhanh chóng xác định nguyên nhân gốc rễ của sự cố và khôi phục hệ thống một cách an toàn.

XDR so với SIEM và EDR:

- *So với SIEM:* SIEM chủ yếu thu thập và phân tích dữ liệu từ nhiều nguồn, cung cấp khả năng giám sát và phát hiện các mối đe dọa trên toàn hệ thống. Tuy nhiên, SIEM không tự động hóa phản ứng với sự cố, và khả năng tích hợp dữ liệu từ các lớp bảo mật khác nhau thường phụ thuộc vào cách triển khai và tích hợp các giải pháp riêng lẻ. Trong khi đó, XDR cung cấp khả năng tích hợp mạnh mẽ hơn và phản ứng tự động đối với các mối đe dọa trên nhiều lớp.

- *So với EDR:* EDR tập trung chủ yếu vào việc bảo vệ và phản ứng với các sự cố bảo mật trên thiết bị đầu cuối. XDR mở rộng khả năng này ra toàn bộ hệ thống, bao gồm cả mạng lưới, máy chủ và đám mây, từ đó mang lại khả năng bảo vệ toàn diện hơn.

2.3 Giới thiệu về Wazuh

Wazuh là một nền tảng mã nguồn mở mạnh mẽ, được thiết kế để cung cấp khả năng giám sát an ninh, phát hiện mối đe dọa, và tuân thủ chính sách bảo mật cho các tổ chức. Wazuh kết hợp các chức năng của *SIEM (Security Information and Event Management)* và *XDR (Extended Detection and Response)*, giúp tổ chức có khả năng giám sát bảo mật toàn diện trên toàn bộ hệ thống IT, từ thiết bị đầu cuối đến ứng dụng và dịch vụ đám mây.

Wazuh giúp doanh nghiệp thu thập và phân tích dữ liệu bảo mật từ nhiều nguồn khác nhau, bao gồm hệ thống mạng, thiết bị đầu cuối, máy chủ, ứng dụng, và môi trường đám mây. Nhờ khả năng phát hiện và phản ứng với các mối đe dọa nhanh chóng, Wazuh trở thành một trong những giải pháp bảo mật hàng đầu được sử dụng trong nhiều tổ chức trên thế giới.

Các tính năng chính của Wazuh:

1. Phát hiện mối đe dọa (Threat Detection):

Wazuh cung cấp khả năng phát hiện các mối đe dọa bảo mật bằng cách phân tích các nhật ký (logs), sự kiện, và dữ liệu thu thập từ nhiều hệ thống khác nhau. Nó sử dụng các quy tắc bảo mật tùy chỉnh và phân tích hành vi để xác định các hoạt động đáng ngờ, vi phạm bảo mật, hoặc cuộc tấn công mạng.

2. Quản lý nhật ký (Log Data Analysis):

Wazuh thu thập và phân tích dữ liệu nhật ký từ hệ thống, ứng dụng, và các thiết bị mạng khác nhau. Dữ liệu này được lưu trữ và phân tích để phát hiện các sự kiện bảo mật, lỗi hệ thống, hoặc hành vi bất thường. Wazuh hỗ trợ phân tích nhật ký theo thời gian thực, cung cấp các cảnh báo tức thì khi phát hiện các mối đe dọa tiềm ẩn.

3. Quản lý tuân thủ (Compliance Management):

Wazuh giúp các tổ chức tuân thủ các tiêu chuẩn và quy định bảo mật quan trọng như *GDPR*, *PCI-DSS*, *HIPAA*, *NIST*, và nhiều tiêu chuẩn khác. Nó tự động kiểm tra và

đánh giá các hệ thống để đảm bảo chúng tuân thủ các chính sách bảo mật, và cung cấp báo cáo chi tiết về tình trạng tuân thủ.

4. *Giám sát bảo mật đám mây (Cloud Security Monitoring):*

Wazuh cung cấp giải pháp giám sát bảo mật cho các môi trường đám mây như *Amazon Web Services (AWS)*, *Microsoft Azure*, và *Google Cloud Platform (GCP)*. Nó theo dõi các sự kiện và hoạt động trên đám mây để phát hiện các mối đe dọa và vi phạm bảo mật, đồng thời giúp các tổ chức đảm bảo tuân thủ các quy định về bảo mật trên đám mây.

5. *Quản lý tính toàn vẹn tệp (File Integrity Monitoring – FIM):*

Wazuh giám sát và kiểm tra sự thay đổi của các tệp tin và thư mục quan trọng trong hệ thống. Bất kỳ sự thay đổi nào đều được ghi lại và phân tích để đảm bảo rằng không có sự can thiệp bất thường vào hệ thống, giúp phát hiện sớm các hành vi xâm nhập hoặc tấn công.

6. *Phân tích lỗ hổng bảo mật (Vulnerability Detection):*

Wazuh có khả năng quét hệ thống và phát hiện các lỗ hổng bảo mật trong phần mềm và hệ điều hành. Nó so sánh cấu hình hệ thống với các cơ sở dữ liệu lỗ hổng bảo mật công cộng và cung cấp các cảnh báo khi phát hiện các lỗ hổng có thể bị khai thác.

7. *Tích hợp với các công cụ bảo mật khác:*

Wazuh có thể tích hợp với nhiều công cụ bảo mật khác như *Elasticsearch*, *Kibana*, *Suricata*, và các hệ thống *IDS/IPS* khác để cung cấp giải pháp bảo mật toàn diện. Sự tích hợp này cho phép các tổ chức tận dụng tối đa các dữ liệu bảo mật từ nhiều nguồn khác nhau.

Kiến trúc của Wazuh:

Wazuh bao gồm các thành phần chính như sau:

- *Wazuh Manager*: Đây là thành phần quản lý trung tâm, chịu trách nhiệm phân tích dữ liệu bảo mật, áp dụng các quy tắc phát hiện và tạo cảnh báo dựa trên dữ liệu thu thập được từ các agent.

- *Wazuh Agent*: Là phần mềm được cài đặt trên các thiết bị đầu cuối, máy chủ, và hệ thống để thu thập dữ liệu bảo mật, bao gồm nhật ký hệ thống, thông tin cấu hình, và các sự kiện bảo mật khác. Agent gửi dữ liệu này đến Wazuh Manager để phân tích.

- *Elasticsearch*: Là nơi lưu trữ và lập chỉ mục các dữ liệu bảo mật được thu thập và phân tích bởi Wazuh. Dữ liệu này sau đó được sử dụng để tạo báo cáo và hiển thị trong giao diện người dùng Kibana.
- *Kibana*: Là công cụ trực quan hóa dữ liệu, được sử dụng để tạo bảng điều khiển (dashboard) và báo cáo về tình trạng bảo mật của hệ thống.

Lợi ích của Wazuh:

- *Mã nguồn mở và miễn phí*: Là một nền tảng mã nguồn mở, Wazuh có thể được sử dụng miễn phí, giúp giảm thiểu chi phí cho các tổ chức mà vẫn đảm bảo tính bảo mật toàn diện.
- *Khả năng mở rộng cao*: Wazuh có khả năng mở rộng linh hoạt, phù hợp cho cả các doanh nghiệp nhỏ và các tổ chức lớn, với khả năng giám sát hàng ngàn thiết bị và hệ thống.
- *Phát hiện mối đe dọa và quản lý tuân thủ hiệu quả*: Wazuh cung cấp các tính năng phát hiện mối đe dọa và quản lý tuân thủ mạnh mẽ, giúp các tổ chức không chỉ bảo vệ hệ thống của mình mà còn đảm bảo tuân thủ các tiêu chuẩn và quy định bảo mật.

Tóm lại, Wazuh là một giải pháp bảo mật toàn diện và hiệu quả, phù hợp cho mọi loại hình doanh nghiệp. Với khả năng phát hiện mối đe dọa, quản lý nhật ký, giám sát tuân thủ và tích hợp mạnh mẽ, Wazuh là một công cụ lý tưởng cho các tổ chức cần bảo vệ hệ thống IT và môi trường đám mây của mình trong bối cảnh các mối đe dọa an ninh mạng ngày càng phức tạp.

2.4 So sánh giữa các giải pháp SIEM/XDR khác và Wazuh

Tiêu chí	Wazuh	Giải pháp SIEM/XDR khác
Mã nguồn	Mã nguồn mở, miễn phí	Phần lớn là mã nguồn đóng, có phí cao
Chi phí	Không có chi phí bản quyền, chỉ tốn chi phí triển khai	Thường có phí bản quyền cao, bao gồm cả phí bảo trì
Khả năng tích hợp	Tích hợp tốt với Elasticsearch, Kibana, Suricata, AWS, Azure, GCP	Tùy thuộc vào nhà cung cấp, thường bị hạn chế bởi công nghệ độc quyền
Khả năng mở rộng	Linh hoạt, có thể giám sát hàng ngàn thiết bị	Tùy thuộc vào giải pháp, thường yêu cầu nâng cấp bản quyền để mở rộng
Phát hiện mối đe dọa	Cung cấp khả năng phát hiện mối đe dọa theo thời gian thực dựa trên quy tắc và hành vi	Tương tự, nhưng một số giải pháp có khả năng AI và machine learning nâng cao hơn
Quản lý nhật ký	Phân tích nhật ký từ nhiều nguồn khác nhau	Tương tự, nhưng một số giải pháp cung cấp thêm tính năng giám sát nâng cao
Tuân thủ bảo mật	Hỗ trợ tuân thủ các tiêu chuẩn như PCI-DSS, HIPAA, GDPR	Tương tự, thường có hỗ trợ tuân thủ tốt hơn nhưng phụ thuộc vào gói dịch vụ
Tự động hóa phản ứng	Có khả năng tự động hóa phản ứng sự cố	Nhiều giải pháp khác có tự động hóa tốt hơn, đặc biệt là XDR hiện đại
Hỗ trợ giám sát đám mây	Hỗ trợ giám sát trên AWS, Azure, GCP	Tương tự, nhưng các giải pháp thương mại có thể hỗ trợ nhiều dịch vụ đám mây hơn
Tính dễ sử dụng	Yêu cầu kiến thức về thiết lập và tích hợp hệ thống	Thường có giao diện người dùng trực quan hơn và hỗ trợ tốt hơn
Tùy biến	Có khả năng tùy biến cao với các quy tắc và cấu hình	Tùy biến nhưng bị hạn chế bởi cấu trúc hệ thống độc quyền
Hỗ trợ cộng đồng	Cộng đồng mã nguồn mở mạnh, tài liệu phong phú	Thường có hỗ trợ chuyên nghiệp, nhưng cộng đồng người dùng nhỏ hơn
Bảo mật hệ thống	Phát hiện và ngăn chặn mối đe dọa trên nhiều lớp, từ thiết bị đầu cuối đến đám mây	Một số giải pháp XDR hiện đại có khả năng bảo mật tiên tiến hơn trên các nền tảng đa dạng
Thời gian triển khai	Triển khai nhanh chóng nhưng cần thời gian để tinh chỉnh	Thường triển khai nhanh hơn nhờ có các mô hình được thiết lập trước

Hình 9: So sánh giữa các giải pháp SIEM/XDR khác và Wazuh

CHƯƠNG 3: PHÂN TÍCH YÊU CẦU

3.1 Yêu cầu kỹ thuật

Để triển khai giải pháp **SIEM/XDR** dựa trên nền tảng mã nguồn mở *Wazuh*, cần phải đáp ứng một số yêu cầu kỹ thuật cơ bản nhằm đảm bảo hệ thống hoạt động ổn định và hiệu quả. Dưới đây là các yêu cầu kỹ thuật chi tiết:

1. Yêu cầu phần cứng:

- Máy chủ Wazuh Manager:
 - CPU: Tối thiểu 4 lõi, khuyến nghị 8 lõi để xử lý các tác vụ phân tích và giám sát bảo mật.
 - RAM: Tối thiểu 8 GB, khuyến nghị 16 GB hoặc nhiều hơn đối với các hệ thống lớn.
 - Ổ cứng: Tối thiểu 100 GB dung lượng lưu trữ cho dữ liệu nhật ký, khuyến nghị sử dụng SSD cho hiệu suất đọc/ghi tốt hơn.
 - Băng thông mạng: Kết nối mạng băng thông rộng, ổn định để quản lý và thu thập dữ liệu từ các nguồn khác nhau.
- Máy chủ Elasticsearch (nếu sử dụng riêng biệt):
 - CPU: Tối thiểu 8 lõi, khuyến nghị 16 lõi để đáp ứng yêu cầu truy vấn và lập chỉ mục dữ liệu.
 - RAM: Tối thiểu 16 GB, khuyến nghị 32 GB đối với khối lượng dữ liệu lớn.
 - Ổ cứng: Tối thiểu 500 GB dung lượng lưu trữ, khuyến nghị 1 TB hoặc hơn cho hệ thống giám sát lớn
- Máy chủ Agent (các thiết bị đầu cuối):
 - CPU: Tối thiểu 2 lõi.
 - RAM: Tối thiểu 4 GB.
 - Ổ cứng: Dung lượng lưu trữ tối thiểu 10 GB để lưu các log tạm thời.
 - Kết nối mạng: Đảm bảo các Agent có thể gửi dữ liệu về máy chủ Wazuh Manager một cách liên tục.

2. Yêu cầu phần mềm:

- Hệ điều hành:

- Wazuh Manager: Hỗ trợ các hệ điều hành Linux (Ubuntu, CentOS, Debian, Red Hat) và Windows Server.

- Wazuh Agent: Tương thích với Linux, Windows, macOS, và nhiều hệ điều hành khác (bao gồm cả Unix và hệ thống nhúng).

- Phần mềm cần thiết:

- Elasticsearch: Phiên bản Elasticsearch 7.x trở lên để lưu trữ và tìm kiếm dữ liệu bảo mật.

- Kibana: Đèn trực quan hóa dữ liệu và tạo bảng điều khiển (dashboard).

- Wazuh Manager: Phần mềm quản lý trung tâm cho hệ thống Wazuh.

- Wazuh Agent: Cài đặt trên các máy chủ hoặc thiết bị đầu cuối để thu thập dữ liệu bảo mật.

- Node.js và Nginx (hoặc Apache): Dùng cho giao diện quản lý và API của Wazuh.

3. Yêu cầu kết nối mạng:

- Giao thức và cổng mạng:

- Wazuh Agent giao tiếp với Wazuh Manager qua giao thức TCP, sử dụng các cổng 1514 và 1515.

- Elasticsearch và Kibana sử dụng cổng mặc định 9200 và 5601 (có thể tùy chỉnh theo yêu cầu).

- Tường lửa:

- Mở các cổng cần thiết trên tường lửa để các thành phần của hệ thống có thể giao tiếp với nhau.

- Đảm bảo hệ thống mạng có đủ băng thông và độ trễ thấp để tránh mất dữ liệu khi thu thập nhật ký từ nhiều nguồn khác nhau.

4. Yêu cầu bảo mật:

- Xác thực và phân quyền:

- Cấu hình xác thực và phân quyền người dùng để đảm bảo rằng chỉ những người có quyền mới có thể truy cập vào hệ thống quản lý Wazuh và Kibana.

- Sử dụng TLS/SSL để mã hóa các kết nối giữa Wazuh Agent và Wazuh Manager, cũng như giữa các thành phần khác như Elasticsearch và Kibana.

- Sao lưu và khôi phục dữ liệu:

- Thiết lập cơ chế sao lưu dữ liệu định kỳ cho Elasticsearch để đảm bảo rằng dữ liệu nhật ký không bị mất mát.

- Cấu hình khôi phục hệ thống nhanh chóng trong trường hợp hệ thống bị lỗi hoặc gặp sự cố an ninh.

5. Yêu cầu mở rộng và bảo trì:

- **Khả năng mở rộng:** Hệ thống phải dễ dàng mở rộng khi số lượng thiết bị hoặc dữ liệu giám sát tăng lên, có thể triển khai cluster cho Elasticsearch để đảm bảo hiệu suất và khả năng chịu lỗi.
- **Công cụ giám sát hiệu suất:** Sử dụng các công cụ giám sát hiệu suất như Prometheus hoặc Grafana để theo dõi hoạt động của các thành phần hệ thống như CPU, RAM, I/O và dung lượng lưu trữ.

6. Yêu cầu khác:

- **Kỹ thuật viên và quản trị hệ thống:** Đội ngũ quản trị viên cần có kiến thức về Linux, mạng, bảo mật và các công cụ liên quan như Elasticsearch, Kibana, và các hệ thống SIEM khác.
- **Hỗ trợ cộng đồng và tài liệu:** Wazuh có cộng đồng mạnh mẽ và tài liệu phong phú, tuy nhiên cần dành thời gian tìm hiểu và tinh chỉnh hệ thống phù hợp với quy mô và yêu cầu cụ thể của tổ chức.

Việc đáp ứng đầy đủ các yêu cầu kỹ thuật nêu trên sẽ giúp triển khai thành công giải pháp Wazuh cho SIEM/XDR, mang lại khả năng giám sát an ninh toàn diện và hiệu quả cho hệ thống doanh nghiệp.

3.2 Yêu cầu bảo mật

Trong quá trình triển khai và vận hành giải pháp SIEM/XDR dựa trên Wazuh, yêu cầu bảo mật là yếu tố then chốt nhằm đảm bảo rằng hệ thống không chỉ giúp phát hiện và phản ứng với các mối đe dọa mà bản thân nó cũng được bảo vệ an toàn trước các rủi ro bảo mật. Dưới đây là các yêu cầu bảo mật cần được áp dụng trong quá trình triển khai và vận hành hệ thống.

1. Mã hóa dữ liệu truyền tải

- TLS/SSL:

- Tất cả dữ liệu truyền tải giữa các thành phần của Wazuh như Wazuh Agent, Wazuh Manager, Elasticsearch, và Kibana phải được mã hóa để đảm bảo an toàn. Sử dụng Transport Layer Security (TLS) hoặc Secure Sockets Layer (SSL) để bảo vệ các kết nối mạng nhằm ngăn chặn việc nghe lén hoặc đánh cắp dữ liệu.

- Cấu hình chứng chỉ SSL cho Wazuh Manager và các Agent để bảo vệ việc trao đổi dữ liệu nhạy cảm.

2. Xác thực và phân quyền

- Xác thực người dùng:

- Hệ thống phải được thiết lập xác thực mạnh mẽ (như OAuth, SAML, hoặc LDAP) để đảm bảo chỉ những người dùng hợp lệ mới có thể truy cập vào các thành phần của Wazuh và dữ liệu bảo mật.

- Hỗ trợ Multi-Factor Authentication (MFA) để tăng cường bảo mật cho các tài khoản quản trị.

- Phân quyền truy cập:

- Cần triển khai các chính sách phân quyền cụ thể dựa trên vai trò (Role-Based Access Control - RBAC). Quyền truy cập vào dữ liệu và hệ thống quản lý phải được phân chia cẩn thận dựa trên vai trò của từng cá nhân để ngăn ngừa các hành vi truy cập không hợp lệ.

- Hạn chế quyền truy cập chỉ cho những người dùng cần thiết để thực hiện công việc của họ.

3. Bảo mật hệ thống và cơ sở hạ tầng

- Bảo mật hệ điều hành:

- Máy chủ chạy Wazuh Manager và Elasticsearch phải được bảo mật kỹ lưỡng, bao gồm cập nhật đầy đủ các bản vá bảo mật, cấu hình tường lửa, và vô hiệu hóa các dịch vụ không cần thiết.

- Sử dụng các công cụ kiểm tra lỗ hổng bảo mật để định kỳ kiểm tra máy chủ và hệ thống xem có lỗ hổng nào tồn tại không.

- Tường lửa và kiểm soát truy cập mạng:

- Cấu hình tường lửa để giới hạn các kết nối đến và từ Wazuh Manager, Elasticsearch, và Kibana. Chỉ cho phép các IP tin cậy truy cập vào các thành phần này.

- Sử dụng VPN hoặc mạng riêng ảo để quản lý và truy cập vào hệ thống từ xa một cách an toàn.

4. Giám sát bảo mật liên tục

- Giám sát và ghi nhật ký:

- Tất cả các hoạt động quan trọng, bao gồm truy cập hệ thống, thay đổi cấu hình, và các sự kiện bảo mật, phải được ghi lại trong nhật ký (logs) và giám sát liên tục để phát hiện các hành vi bất thường hoặc nguy cơ bảo mật.

- Định kỳ kiểm tra và phân tích các nhật ký để phát hiện các dấu hiệu của cuộc tấn công tiềm tàng.

- Cảnh báo tự động: Cấu hình các cảnh báo tự động dựa trên các quy tắc phát hiện mối đe dọa của Wazuh. Các cảnh báo này sẽ giúp phát hiện sớm các hành vi đáng ngờ hoặc các cuộc tấn công tiềm tàng như brute-force, xâm nhập trái phép hoặc hành vi giả mạo dữ liệu.

5. Sao lưu và khôi phục hệ thống

- Sao lưu dữ liệu:

- Cần thiết lập một cơ chế sao lưu định kỳ cho toàn bộ dữ liệu, bao gồm dữ liệu nhật ký trong Elasticsearch, cấu hình Wazuh, và các dữ liệu liên quan khác.

- Dữ liệu sao lưu cần được mã hóa và lưu trữ tại các vị trí an toàn để tránh bị đánh cắp hoặc mất mát.

- Kế hoạch khôi phục sau sự cố (Disaster Recovery): Thiết lập kế hoạch khôi phục hệ thống sau sự cố, đảm bảo rằng hệ thống có thể phục hồi nhanh chóng

từ bản sao lưu trong trường hợp bị tấn công hoặc sự cố hệ thống. Kế hoạch này nên được thử nghiệm định kỳ.

6. Bảo vệ dữ liệu và quyền riêng tư

- Bảo vệ dữ liệu nhạy cảm:

- Dữ liệu nhạy cảm trong quá trình phân tích, giám sát, và thu thập phải được bảo vệ thông qua các biện pháp mã hóa dữ liệu ở trạng thái nghỉ (data at rest) và khi đang truyền tải (data in transit).

- Các thông tin như mật khẩu, khóa mã hóa, và các thông tin bảo mật khác phải được lưu trữ an toàn và không được ghi lại dưới dạng văn bản thuần túy.

- Tuân thủ quy định bảo mật: Wazuh phải được cấu hình để tuân thủ các tiêu chuẩn và quy định bảo mật quan trọng như GDPR, HIPAA, PCI-DSS, ISO 27001, và các quy định bảo mật khác liên quan đến bảo vệ dữ liệu và quyền riêng tư.

7. Đánh giá bảo mật định kỳ

- Kiểm tra bảo mật:

- Định kỳ thực hiện các bài kiểm tra bảo mật (penetration testing) để đánh giá khả năng bảo vệ của hệ thống trước các cuộc tấn công mạng.

- Thực hiện kiểm tra định kỳ đối với cấu hình bảo mật của hệ thống và cập nhật các bản vá bảo mật ngay khi có.

- Kiểm toán bảo mật: Hệ thống phải được kiểm toán định kỳ để đảm bảo rằng tất cả các yêu cầu bảo mật đang được thực thi đúng cách. Báo cáo kiểm toán phải được lưu trữ để đánh giá và cải thiện bảo mật khi cần thiết.

8. Tự động hóa và phản ứng sự cố

- Phản ứng sự cố tự động: Thiết lập cơ chế phản ứng tự động đối với các sự cố an ninh, như cách ly hệ thống bị xâm nhập, chặn IP đáng ngờ hoặc tắt các dịch vụ bị tấn công.
- Kịch bản phản ứng sự cố: Phát triển và thực hiện các kịch bản phản ứng sự cố bảo mật, đảm bảo rằng đội ngũ quản trị có thể phản ứng kịp thời và chính xác với các cuộc tấn công mạng hoặc sự cố an ninh.

Yêu cầu bảo mật này nhằm bảo vệ hệ thống SIEM/XDR dựa trên nền tảng Wazuh khỏi các rủi ro và mối đe dọa tiềm ẩn, đảm bảo rằng hệ thống hoạt động ổn định và an toàn trong môi trường mạng phức tạp.

3.3 Yêu cầu hệ thống và triển khai

Để triển khai giải pháp SIEM/XDR dựa trên nền tảng mã nguồn mở Wazuh, các yêu cầu hệ thống và quy trình triển khai đóng vai trò quan trọng trong việc đảm bảo hệ thống vận hành hiệu quả và mở rộng khi cần thiết. Dưới đây là các yêu cầu và hướng dẫn cụ thể để thiết lập và triển khai giải pháp này.

1. Yêu cầu hệ thống

- Wazuh Manager:Là thành phần trung tâm chịu trách nhiệm quản lý và phân tích dữ liệu bảo mật từ các agent. Để hệ thống hoạt động hiệu quả, cần đáp ứng các yêu cầu về phần cứng và phần mềm như sau:

- CPU: Tối thiểu 4 lõi, khuyến nghị 8 lõi hoặc nhiều hơn cho môi trường lớn.
- RAM: Tối thiểu 8 GB, khuyến nghị 16 GB trở lên.
- Disk: Tối thiểu 100 GB, nên sử dụng ổ SSD cho hiệu suất tốt.

- Hệ điều hành: Hỗ trợ các hệ điều hành phổ biến như Ubuntu, Debian, CentOS, Red Hat, và Windows Server.

- Elasticsearch:Là công cụ lưu trữ và tìm kiếm dữ liệu bảo mật. Các yêu cầu cho Elasticsearch phụ thuộc vào số lượng nhật ký và dữ liệu cần phân tích:

- CPU: Tối thiểu 8 lõi, khuyến nghị 16 lõi.
- RAM: Tối thiểu 16 GB, khuyến nghị 32 GB cho các hệ thống lớn.
- Disk: Tối thiểu 500 GB, khuyến nghị từ 1 TB trở lên với hệ thống lớn.

- Hệ điều hành: Linux hoặc Windows đều được hỗ trợ.

- Wazuh Agent: Được cài đặt trên các thiết bị đầu cuối để thu thập và gửi dữ liệu bảo mật về cho Wazuh Manager. Các yêu cầu hệ thống cho Wazuh Agent tùy thuộc vào thiết bị cụ thể, thường không yêu cầu cao về phần cứng:

- CPU: Tối thiểu 2 lõi.
- RAM: Tối thiểu 4 GB.

- Disk: Dung lượng lưu trữ tối thiểu 10 GB.
- Hệ điều hành: Hỗ trợ Linux, Windows, macOS, Unix, và các hệ điều hành nhúng.

2. Quy trình triển khai hệ thống

Triển khai giải pháp Wazuh bao gồm các bước cơ bản như sau:

Bước 1: Chuẩn bị môi trường

- Lập kế hoạch triển khai: Xác định quy mô hệ thống, số lượng agent, yêu cầu lưu trữ dữ liệu và hiệu suất cần thiết.
- Chuẩn bị phần cứng và phần mềm: Đảm bảo các máy chủ được cấu hình theo các yêu cầu phần cứng và phần mềm đã nêu ở trên.
- Cài đặt hệ điều hành: Lựa chọn hệ điều hành cho Wazuh Manager, Elasticsearch, và Kibana (Linux được khuyến nghị).

Bước 2: Cài đặt Wazuh Manager

- Cài đặt Wazuh Manager: Tải và cài đặt Wazuh Manager từ trang chủ Wazuh hoặc thông qua các kho phần mềm của hệ điều hành (APT, YUM).
- Cấu hình Wazuh Manager: Cấu hình các thông số như địa chỉ IP, cổng kết nối, và các thiết lập bảo mật (chứng chỉ SSL/TLS).
- Kết nối với Elasticsearch: Cấu hình để Wazuh Manager gửi dữ liệu bảo mật đến Elasticsearch để lưu trữ và phân tích.

Bước 3: Cài đặt và cấu hình Elasticsearch và Kibana

- Cài đặt Elasticsearch: Elasticsearch phải được cài đặt và cấu hình trên một máy chủ riêng (hoặc cùng máy chủ tùy vào quy mô hệ thống). Sử dụng phiên bản 7.x trở lên.
- Cấu hình Cluster Elasticsearch (nếu cần): Với môi trường lớn, nên triển khai cụm (cluster) Elasticsearch để tăng khả năng mở rộng và hiệu suất.

- Cài đặt Kibana: Kibana sẽ cung cấp giao diện đồ họa cho việc giám sát và phân tích dữ liệu. Cấu hình Kibana để kết nối với Elasticsearch.

Bước 4: Cài đặt Wazuh Agent trên các máy chủ/thiết bị đầu cuối

- Tải và cài đặt Agent: Cài đặt Wazuh Agent trên các thiết bị đầu cuối từ trang chủ Wazuh hoặc thông qua các kho phần mềm.
- Cấu hình Agent: Thiết lập các thông số như địa chỉ của Wazuh Manager, cổng kết nối, và phương thức mã hóa dữ liệu truyền tải.
- Đăng ký Agent với Wazuh Manager: Sử dụng mã hóa và chứng chỉ để bảo mật kênh liên lạc giữa Wazuh Agent và Wazuh Manager.

Bước 5: Kiểm tra và giám sát hệ thống

- Kiểm tra kết nối: Đảm bảo rằng tất cả các agent có thể kết nối thành công đến Wazuh Manager và gửi dữ liệu.
- Cấu hình cảnh báo: Thiết lập các quy tắc cảnh báo trong Wazuh để phát hiện các sự kiện bảo mật quan trọng.
- Tạo Dashboard trên Kibana: Sử dụng Kibana để tạo các bảng điều khiển (dashboard) nhằm trực quan hóa dữ liệu bảo mật.

3. Quản lý và bảo trì hệ thống

Sau khi triển khai thành công hệ thống Wazuh, cần có kế hoạch quản lý và bảo trì định kỳ để đảm bảo hệ thống hoạt động ổn định và hiệu quả:

- Cập nhật phần mềm: Đảm bảo các thành phần của Wazuh, Elasticsearch, và Kibana luôn được cập nhật với các phiên bản mới nhất để bảo mật và cải thiện hiệu năng.
- Sao lưu dữ liệu: Cần có kế hoạch sao lưu định kỳ dữ liệu trong Elasticsearch cũng như các cấu hình quan trọng của hệ thống.
- Kiểm tra bảo mật định kỳ: Thực hiện kiểm tra bảo mật hệ thống để phát hiện và khắc phục kịp thời các lỗ hổng bảo mật.

4. Khả năng mở rộng

- Mở rộng Wazuh Manager: Khi hệ thống lớn dần, có thể triển khai nhiều Wazuh Manager để phân tán tải và tăng hiệu suất.
- Mở rộng Elasticsearch: Nếu khối lượng dữ liệu lớn, có thể thêm nhiều nút (node) Elasticsearch để tăng khả năng lưu trữ và xử lý dữ liệu.
- Cân bằng tải: Sử dụng Load Balancer để phân phối tải giữa các máy chủ Wazuh Manager hoặc Elasticsearch nhằm tăng cường hiệu suất và khả năng chịu lỗi.

Việc đáp ứng các yêu cầu hệ thống và tuân thủ quy trình triển khai chặt chẽ sẽ giúp đảm bảo hệ thống SIEM/XDR dựa trên Wazuh hoạt động ổn định, có thể mở rộng, và cung cấp hiệu suất giám sát an ninh tốt nhất cho tổ chức.

3.4 Yêu cầu tích hợp với các hệ thống khác

1. Tích hợp với hệ thống quản lý danh tính và truy cập (IAM)

- Tích hợp Active Directory (AD) / LDAP:

- Để theo dõi và ghi nhật ký hoạt động của người dùng trong môi trường mạng doanh nghiệp, Wazuh có thể tích hợp với Active Directory hoặc LDAP. Việc này giúp ghi lại các sự kiện đăng nhập, thay đổi chính sách quyền hạn, hoặc các hành vi nghi ngờ liên quan đến tài khoản người dùng.

- Các thông tin từ AD/LDAP có thể được sử dụng để tạo ra các cảnh báo bảo mật dựa trên hành vi người dùng (User Behavior Analytics - UBA).

- Hỗ trợ Single Sign-On (SSO): Hệ thống Wazuh có thể tích hợp với các dịch vụ SSO như SAML, OAuth, hoặc OpenID Connect để đơn giản hóa quá trình quản lý tài khoản người dùng và đảm bảo rằng chỉ người dùng đã được xác thực mới có quyền truy cập vào dữ liệu bảo mật.

2. Tích hợp với hệ thống quản lý sự cố bảo mật (SOAR)

- Automation và Orchestration:

- Wazuh có thể được tích hợp với các công cụ quản lý và tự động hóa phản ứng sự cố (SOAR), như TheHive, Cortex, hoặc Phantom, để tự động thực hiện các hành động phản ứng với sự cố an ninh.

- Ví dụ: Khi một sự kiện bất thường được phát hiện, hệ thống có thể kích hoạt phản ứng tự động như khóa tài khoản, cách ly máy chủ hoặc chặn kết nối mạng từ địa chỉ IP đáng ngờ.

- Tích hợp quy trình điều tra: Wazuh có thể gửi dữ liệu đến các hệ thống SOAR để tự động hóa việc phân tích, điều tra sự cố và tạo ra các báo cáo bảo mật chi tiết. SOAR giúp giảm thiểu thời gian phản ứng và tăng cường khả năng xử lý sự cố hiệu quả.

3. Tích hợp với hệ thống tường lửa và IDS/IPS

- Tích hợp với Firewall/UTM:

- Wazuh hỗ trợ thu thập và phân tích dữ liệu từ các tường lửa như Cisco ASA, Palo Alto Networks, Fortinet, và pfSense. Việc tích hợp này giúp theo dõi luồng truy cập mạng, phát hiện các hành vi đáng ngờ như tấn công DDoS, dò tìm lỗ hổng, và các hành vi truy cập trái phép.

- Dữ liệu từ tường lửa được gửi đến Wazuh để phân tích và phát hiện các hành vi tấn công tiềm tàng, đồng thời có thể tự động thực hiện các hành động phản ứng như chặn IP hoặc khóa kết nối.

- Tích hợp với hệ thống phát hiện xâm nhập (IDS/IPS):

- Wazuh có khả năng tích hợp với các hệ thống IDS/IPS như Suricata, Snor, hoặc Zeek để nhận các sự kiện an ninh liên quan đến các cuộc tấn công mạng. Những dữ liệu này được phân tích để tạo ra các cảnh báo sớm và giúp phát hiện các hành vi bất thường trong mạng lưới.

- Kết hợp thông tin từ IDS/IPS với dữ liệu từ Wazuh giúp tăng cường khả năng phát hiện và phản ứng với các mối đe dọa bảo mật.

4. Tích hợp với hệ thống giám sát hạ tầng CNTT (IT Infrastructure Monitoring)

- Tích hợp với Prometheus/Grafana:

- Wazuh có thể được tích hợp với các hệ thống giám sát hiệu suất và tài nguyên hệ thống như Prometheus hoặc Grafana. Điều này cho phép theo dõi và cảnh báo dựa

trên hiệu suất hạ tầng, tài nguyên sử dụng, và các thông số hoạt động khác của hệ thống máy chủ.

- Bằng cách tích hợp với Prometheus/Grafana, dữ liệu từ Wazuh có thể được kết hợp với thông tin về tài nguyên hệ thống để đưa ra các cảnh báo bảo mật dựa trên sự thay đổi hiệu suất.

- Tích hợp với Nagios/Zabbix:

- Wazuh có thể nhận và phân tích dữ liệu từ các công cụ giám sát hệ thống như Nagios hoặc Zabbix, giúp theo dõi tình trạng hệ thống và cảnh báo khi có dấu hiệu của sự cố như lỗi phần cứng, hiệu suất giảm, hoặc các vấn đề về kết nối mạng.

5. Tích hợp với công nghệ đám mây

- Tích hợp với AWS, Azure, GCP:

- Wazuh cung cấp khả năng tích hợp với các dịch vụ đám mây như Amazon Web Services (AWS), Microsoft Azure, và Google Cloud Platform (GCP). Điều này cho phép thu thập và giám sát dữ liệu bảo mật từ các dịch vụ đám mây như CloudTrail, CloudWatch, Azure Security Center, và Google Security Command Center.

- Các sự kiện an ninh trên nền tảng đám mây có thể được thu thập và phân tích để phát hiện các lỗ hổng hoặc mối đe dọa liên quan đến tài nguyên đám mây, quyền truy cập, hoặc lưu trữ dữ liệu.

- Giám sát container và microservices: Wazuh hỗ trợ tích hợp với các công nghệ container như Docker và Kubernetes để giám sát và ghi lại các hoạt động của container và dịch vụ microservices. Điều này giúp phát hiện sớm các sự kiện bảo mật liên quan đến container và hạ tầng phân tán.

6. Tích hợp với hệ thống quản lý log và SIEM khác

- Tích hợp với Splunk, Elastic Stack, Graylog:

- Wazuh có thể tích hợp và gửi dữ liệu bảo mật đến các hệ thống quản lý log và SIEM phổ biến khác như Splunk, Elastic Stack, hoặc Graylog để tăng cường khả năng phân tích và tổng hợp dữ liệu.

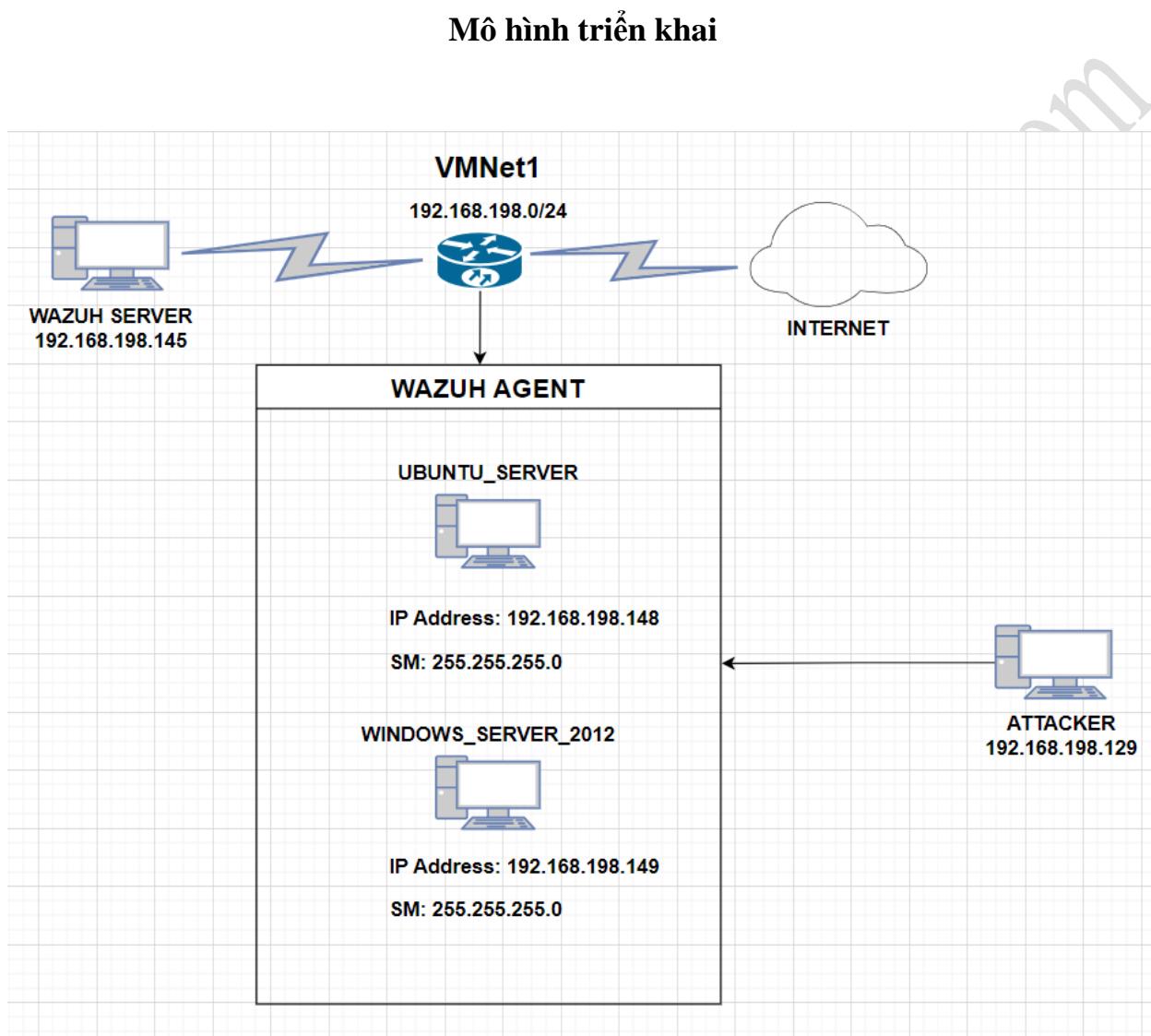
- Các hệ thống này cung cấp khả năng phân tích chuyên sâu, dashboard trực quan và tích hợp dữ liệu từ nhiều nguồn để hỗ trợ trong việc điều tra và xử lý sự cố bảo mật.

7. Tích hợp với hệ thống quản lý rủi ro và tuân thủ (GRC)

Tích hợp với OpenGRC hoặc các hệ thống GRC khác: Wazuh có thể tích hợp với các hệ thống Governance, Risk, and Compliance (GRC) để hỗ trợ quản lý các yêu cầu tuân thủ và báo cáo rủi ro. Dữ liệu từ Wazuh có thể được chuyển đến các hệ thống này để đánh giá rủi ro bảo mật và đảm bảo tuân thủ các quy định bảo mật như GDPR, HIPAA, và PCI-DSS.

CHƯƠNG 4: TRIỂN KHAI VÀ THỰC NGHIỆM

4.1 Triển khai Wazuh Server và Agents



Hình 10: Mô hình triển khai thực nghiệm

4.1.1 Cài đặt Wazuh

Bước 1: Cập nhật hệ thống

Trước khi bắt đầu, hãy đảm bảo rằng hệ thống được cập nhật với các bản vá và phần mềm mới nhất.

```
sudo apt update
```

```
sudo apt upgrade -y
```

Bước 2: Tải các gói cần thiết cho máy

```
apt install apt-transport-https zip unzip lsb-release curl gnupg net-tools (cả 2 máy  
Wazuh Server và Agent Ubuntu)
```

Bước 3: Cài đặt khóa GPG (GNU Privacy Guard: là một loại khóa mã hóa được sử dụng trong hệ thống mã hóa **GPG**, một phần mềm mã nguồn mở dùng để mã hóa và xác thực dữ liệu. GPG hỗ trợ cả mã hóa **symmetric** (mã hóa đối xứng) và **asymmetric** (mã hóa bất đối xứng), trong đó mã hóa bất đối xứng là cơ chế thường dùng cho các khóa GPG.)

```
curl -s https://artifacts.elastic.co/GPG-KEY-elasticsearch | gpg --no-default-keyring --keyring gnupg-ring:/usr/share/keyrings/elasticsearch.gpg --import && chmod 644 /usr/share/keyrings/elasticsearch.gpg
```

```
root@ubuntu-server:~# curl -s https://artifacts.elastic.co/GPG-KEY-elasticsearch | gpg --no-default-keyring --keyring gnupg-ring:/usr/share/keyrings/elasticsearch.gpg --import && chmod 644 /usr/share/keyrings/elasticsearch.gpg
gpg: keyring '/usr/share/keyrings/elasticsearch.gpg' created
gpg: directory '/root/.gnupg' created
gpg: /root/.gnupg/trustdb.gpg: trustdb created
gpg: key D27D666CD88E42B4: public key "Elasticsearch (Elasticsearch Signing Key) <dev_ops@elasticsearch.org>" imported
gpg: Total number processed: 1
gpg:           imported: 1
```

Bước 4: Thêm kho lưu trữ Wazuh

```
echo "deb [signed-by=/usr/share/keyrings/elasticsearch.gpg]  
https://artifacts.elastic.co/packages/7.x/apt stable main" | tee  
/etc/apt/sources.list.d/elastic-7.x.list
```

```
root@ubuntu-server:~# echo "deb [signed-by=/usr/share/keyrings/elasticsearch.gpg] http://artifacts.elastic.co/packages/7.x/apt stable main" | tee /etc/apt/sources.list.d/elastic-7.x.list
deb [signed-by=/usr/share/keyrings/elasticsearch.gpg] https://artifacts.elastic.co/packages/7.x/apt stable main
```

Bước 5: Cài đặt gói ElasticSearch

Cập nhật lại danh sách gói và cài đặt gói **ElasticSearch**:

```
sudo apt update
```

```
sudo apt install elasticsearch=7.17.9
```

Bước 6: Download file cấu hình /etc/elasticsearch/elasticsearch.yml

```
curl -so /etc/elasticsearch/elasticsearch.yml
```

```
https://packages.wazuh.com/4.4/tpl/elastic-basic/elasticsearch\_all\_in\_one.yml
```

Bước 7: Tạo các chứng chỉ (cert)

```
curl -so /usr/share/elasticsearch/instances.yml
```

```
https://packages.wazuh.com/4.4/tpl/elastic-basic/instances\_aio.yml
```

```
/usr/share/elasticsearch/bin/elasticsearch-certutil cert ca --pem --in instances.yml --keep-ca-key --out ~/certs.zip
```

Bước 8: Giải nén tệp certs.zip

```
unzip ~/certs.zip -d ~/certs
```

```
root@ubuntu-server:~# unzip ~/certs.zip -d ~/certs
Archive:  /root/certs.zip
  creating: /root/certs/ca/
  inflating: /root/certs/ca/ca.crt
  inflating: /root/certs/ca/ca.key
  creating: /root/certs/elasticsearch/
  inflating: /root/certs/elasticsearch/elasticsearch.crt
  inflating: /root/certs/elasticsearch/elasticsearch.key
root@ubuntu-server:~#
```

Bước 9: Tạo thư mục /etc/elasticsearch/certs

```
mkdir /etc/elasticsearch/certs/ca -p
```

```
cp -R ~/certs/ca/ ~/certs/elasticsearch/* /etc/elasticsearch/certs/
```

```
chown -R elasticsearch: /etc/elasticsearch/certs
```

```
chmod -R 500 /etc/elasticsearch/certs
```

```
chmod 400 /etc/elasticsearch/certs/ca/ca.* /etc/elasticsearch/certs/elasticsearch.*  
rm -rf ~/certs/ ~/certs.zip
```

Bước 10: Kích hoạt và bắt đầu dịch vụ Elasticsearch

```
systemctl daemon-reload  
systemctl enable elasticsearch.service  
systemctl start elasticsearch.service
```

```
root@ubuntu-server:~# systemctl daemon-reload
```

```
root@ubuntu-server:~# systemctl enable elasticsearch.service  
Synchronizing state of elasticsearch.service with SysV service script with /lib/systemd/systemd-sysv-install.  
Executing: /lib/systemd/systemd-sysv-install enable elasticsearch  
Created symlink /etc/systemd/system/multi-user.target.wants/elasticsearch.service → /lib/systemd/system/elasticsearch.service.  
root@ubuntu-server:~# systemctl start elasticsearch.service  
root@ubuntu-server:~#
```

Bước 11: Tạo thông tin xác thực ngẫu nhiên

```
/usr/share/elasticsearch/bin/elasticsearch-setup-passwords auto
```

```
root@ubuntu-server:~# /usr/share/elasticsearch/bin/elasticsearch-setup-passwords auto  
Initiating the setup of passwords for reserved users elastic,apm_system,kibana,kibana_system,logstash_system,beats_system,remote_monitoring_user.  
The passwords will be randomly generated and printed to the console.  
Please confirm that you would like to continue [y/N]  
  
Changed password for user apm_system  
PASSWORD apm_system = 2WXhg6qo2eWWj7uIvrLX  
  
Changed password for user kibana_system  
PASSWORD kibana_system = zuWMu43uauloNS0HnUlb  
  
Changed password for user kibana  
PASSWORD kibana = zuWMu43uauloNS0HnUlb  
  
Changed password for user logstash_system  
PASSWORD logstash_system = gCZjDK02ACG4Ucu7e3zf  
  
Changed password for user beats_system  
PASSWORD beats_system = 5vCtUrkP2sciLZHLwE3t  
  
Changed password for user remote_monitoring_user  
PASSWORD remote_monitoring_user = v1ZZYUbExYuXvQiozdRU  
  
Changed password for user elastic  
PASSWORD elastic = ybGj7NqxjETxQWklmpaV
```

Lưu lại password đã được tạo ngẫu nhiên ở trên

Bước 12: Kiểm tra quá trình cài đặt

```
curl -XGET https://localhost:9200 -u elastic:ybGj7NqxjETxQWklmpaV -k
```

ybGj7NqxjETxQWklmpaV: đây là mật khẩu của elastic

```
root@ubuntu-server:~# curl -XGET https://localhost:9200 -u elastic:ybGj7NqxjETxQWklmpaV -k
{
  "name" : "elasticsearch",
  "cluster_name" : "elasticsearch",
  "cluster_uuid" : "I0VN41S2RDa1Ad9t1VBSRA",
  "version" : {
    "number" : "7.17.9",
    "build_flavor" : "default",
    "build_type" : "deb",
    "build_hash" : "ef4822227ee6b9e70e502f0f0daa52435ee634d",
    "build_date" : "2023-01-31T05:34:43.305517834Z",
    "build_snapshot" : false,
    "lucene_version" : "8.11.1",
    "minimum_wire_compatibility_version" : "6.8.0",
    "minimum_index_compatibility_version" : "6.0.0-beta1"
  },
  "tagline" : "You Know, for Search"
}
```

Bước 13: Cài đặt khóa GPG

```
curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | gpg --no-default-keyring --keyring gnupg-ring:/usr/share/keyrings/wazuh.gpg --import && chmod 644 /usr/share/keyrings/wazuh.gpg
```

Bước 14: Thêm kho lưu trữ Wazuh

```
echo "deb [signed-by=/usr/share/keyrings/wazuh.gpg]
https://packages.wazuh.com/4.x/apt/ stable main" | tee -a
/etc/apt/sources.list.d/wazuh.list
```

Bước 15: Cập nhật thông tin gói

sudo apt update

Bước 16: Cài đặt gói quản lý Wazuh

apt install wazuh-manager

Bước 17: Kích hoạt và bắt đầu dịch vụ quản lý Wazuh

systemctl daemon-reload

systemctl enable wazuh-manager.service

systemctl start wazuh-manager.service

```
root@ubuntu-server:~# systemctl daemon-reload
root@ubuntu-server:~# systemctl enable wazuh-manager.service
Created symlink /etc/systemd/system/multi-user.target.wants/wazuh-manager.service → /lib/systemd/system/wazuh-manager.service.
root@ubuntu-server:~# systemctl start wazuh-manager.service
```

Bước 18: Kiểm tra trạng thái của Wazuh

systemctl status wazuh-manager

```
root@ubuntu-server:~# systemctl status wazuh-manager
● wazuh-manager.service - Wazuh manager
   Loaded: loaded (/lib/systemd/system/wazuh-manager.service; enabled; vendor>
   Active: active (running) since Thu 2024-09-12 23:56:00 +07; 28s ago
     Process: 93339 ExecStart=/usr/bin/env /var/ossec/bin/wazuh-control start (c>
       Tasks: 146 (limit: 2204)
      Memory: 750.4M
        CGroup: /system.slice/wazuh-manager.service
                  └─93413 /var/ossec/framework/python/bin/python3 /var/ossec/api/scr>
                    ├─93453 /var/ossec/bin/wazuh-authd
                    ├─93469 /var/ossec/bin/wazuh-db
                    ├─93484 /var/ossec/framework/python/bin/python3 /var/ossec/api/scr>
                    ├─93487 /var/ossec/framework/python/bin/python3 /var/ossec/api/scr>
                    ├─93490 /var/ossec/framework/python/bin/python3 /var/ossec/api/scr>
                    ├─93503 /var/ossec/bin/wazuh-execd
                    ├─93517 /var/ossec/bin/wazuh-analysisd
                    ├─93560 /var/ossec/bin/wazuh-syscheckd
                    ├─93575 /var/ossec/bin/wazuh-remoted
                    ├─93585 /var/ossec/bin/wazuh-logcollector
                    ├─93626 /var/ossec/bin/wazuh-monitord
                    ├─93636 /var/ossec/bin/wazuh-modulesd
```

Bước 19: Cài đặt gói Filebeat

apt install filebeat=7.17.9

Bước 20: Tải cấu hình Filebeat

curl -so /etc/filebeat/filebeat.yml https://packages.wazuh.com/4.4/tpl/elasticsearch-basic/filebeat_all_in_one.yml

Bước 21: Tải mẫu cảnh báo cho Elasticsearch và cấp quyền go+r cho /etc/filebeat/wazuh-template.json

curl -so /etc/filebeat/wazuh-template.json
<https://raw.githubusercontent.com/wazuh/wazuh/4.4/extensions/elasticsearch/7.x/wazuh-template.json>

chmod go+r /etc/filebeat/wazuh-template.json

```
root@ubuntu-server:~# curl -sO /etc/filebeat/wazuh-template.json https://raw.githubusercontent.com/wazuh/wazuh/4.4/extensions/elasticsearch/7.x/wazuh-template.json
root@ubuntu-server:~# chmod +o+r /etc/filebeat/wazuh-template.json
```

Bước 22: Tải modun Wazuh cho filebeat

```
curl -s https://packages.wazuh.com/4.x/filebeat/wazuh-filebeat-0.2.tar.gz | tar -xvz -C /usr/share/filebeat/module
```

Bước 23: Chỉnh sửa tệp /etc/filebeat/filebeat.yml

```
nano /etc/filebeat/filebeat.yml
```

```
GNU nano 4.8          /etc/filebeat/filebeat.yml
# Wazuh - Filebeat configuration file
output.elasticsearch.hosts: ["127.0.0.1:9200"]
output.elasticsearch.password: <elasticsearch_password>

filebeat.modules:
  - module: wazuh
    alerts:
      enabled: true
    archives:
      enabled: false

setup.template.json.enabled: true
setup.template.json.path: /etc/filebeat/wazuh-template.json
setup.template.json.name: wazuh
setup.template.overwrite: true
setup.ilm.enabled: false

output.elasticsearch.protocol: https
output.elasticsearch.ssl.certificate: /etc/elasticsearch/certs/elasticsearch.crt
output.elasticsearch.ssl.key: /etc/elasticsearch/certs/elasticsearch.key
[ Read 32 lines ]
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos
^X Exit  ^R Read File  ^L Replace  ^U Paste Text  ^T To Spell  ^F Go To Line
```

Thay thế <elasticsearch_password> bằng mật khẩu của elastic đã tạo và lưu trước đó

Bước 24: Sao chép các chứng chỉ vào /etc/filebeat/certs/

```
cp -r /etc/elasticsearch/certs/ca/ /etc/filebeat/certs/
cp /etc/elasticsearch/certs/elasticsearch.crt /etc/filebeat/certs/filebeat.crt
cp /etc/elasticsearch/certs/elasticsearch.key /etc/filebeat/certs/filebeat.key
```

Bước 25: Kích hoạt và bắt đầu dịch vụ filebeat

```
systemctl daemon-reload
```

```
systemctl enable filebeat.service
```

```
systemctl start filebeat.service
```

Để đảm bảo dịch vụ filebeat đã được cài đặt thành công chúng ta chạy lệnh sau:

```
filebeat test output
```

```
root@ubuntu-server:~# filebeat test output
elasticsearch: https://127.0.0.1:9200...
  parse url... OK
  connection...
    parse host... OK
    dns lookup... OK
    addresses: 127.0.0.1
    dial up... OK
  TLS...
    security: server's certificate chain verification is enabled
    handshake... OK
    TLS version: TLSv1.3
    dial up... OK
  talk to server... OK
  version: 7.17.9
```

Bước 26: Cài đặt gói Kibana

```
apt install kibana=7.17.9
```

Bước 27: Sao chép các chứng chỉ Elasticsearch vào thư mục cấu hình kibana

```
mkdir /etc/kibana/certs/ca -p
```

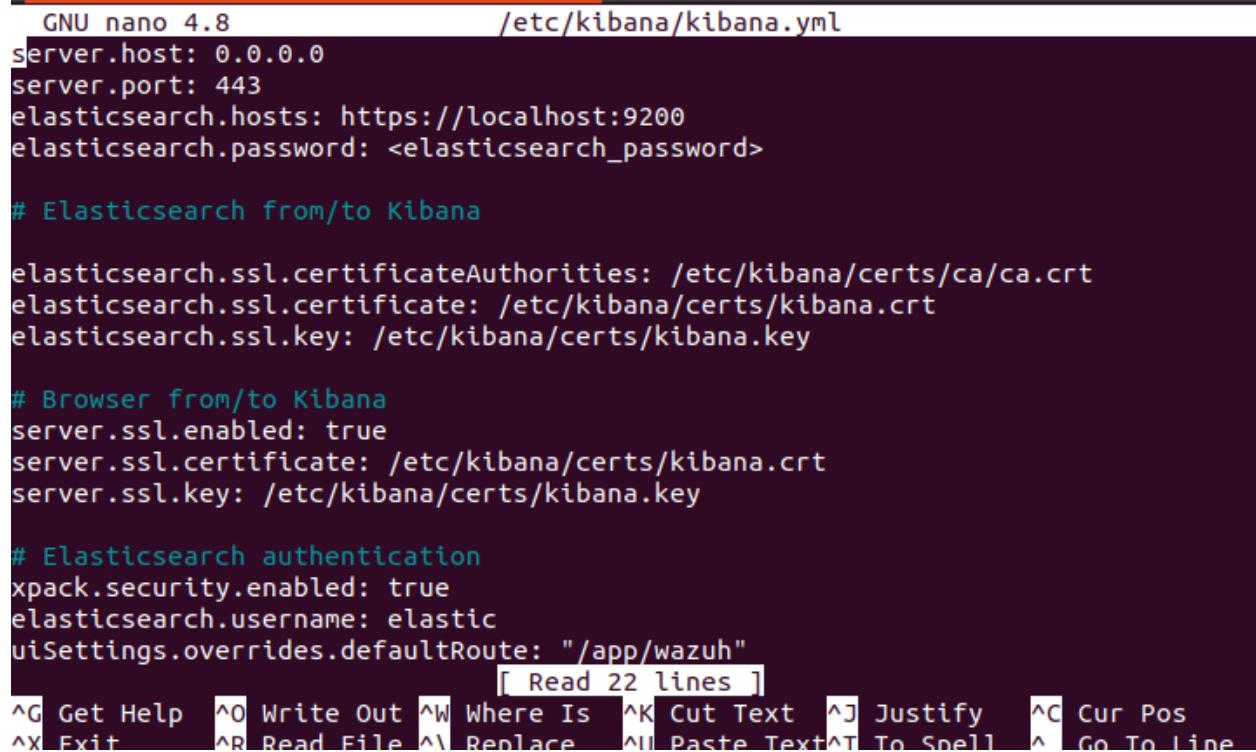
```
cp -R /etc/elasticsearch/certs/ca/ /etc/kibana/certs/
cp /etc/elasticsearch/certs/elasticsearch.key /etc/kibana/certs/kibana.key
cp /etc/elasticsearch/certs/elasticsearch.crt /etc/kibana/certs/kibana.crt
chown -R kibana:kibana /etc/kibana/
chmod -R 500 /etc/kibana/certs/
chmod 440 /etc/kibana/certs/ca/ca.* /etc/kibana/certs/kibana.*
```

Bước 28: Tải tệp cấu hình kibana

```
curl -so /etc/kibana/kibana.yml https://packages.wazuh.com/4.4/tpl/elastic-
basic/kibana_all_in_one.yml
```

Bước 29: Chỉnh sửa tệp /etc/kibana/kibana.yml

```
nano /etc/kibana/kibana.yml
```



```
GNU nano 4.8          /etc/kibana/kibana.yml
server.host: 0.0.0.0
server.port: 443
elasticsearch.hosts: https://localhost:9200
elasticsearch.password: <elasticsearch_password>

# Elasticsearch from/to Kibana

elasticsearch.ssl.certificateAuthorities: /etc/kibana/certs/ca/ca.crt
elasticsearch.ssl.certificate: /etc/kibana/certs/kibana.crt
elasticsearch.ssl.key: /etc/kibana/certs/kibana.key

# Browser from/to Kibana
server.ssl.enabled: true
server.ssl.certificate: /etc/kibana/certs/kibana.crt
server.ssl.key: /etc/kibana/certs/kibana.key

# Elasticsearch authentication
xpack.security.enabled: true
elasticsearch.username: elastic
uiSettings.overrides.defaultRoute: "/app/wazuh"
[ Read 22 lines ]
^G Get Help  ^O Write Out  ^W Where Is  ^K Cut Text  ^J Justify  ^C Cur Pos
^X Exit      ^R Read File  ^L Replace   ^V Paste Text  ^T To Spell  ^_ Go To Line
```

Tương tự, thay thế <elasticsearch_password> bằng mật khẩu của elastic đã tạo ngẫu nhiên và lưu trước đó

Bước 30: Tạo thư mục /usr/share/kibana/data

```
mkdir /usr/share/kibana/data
```

```
chown -R kibana:kibana /usr/share/kibana/
```

Bước 31: Cài đặt plugin Wazuh kibana

```
cd /usr/share/kibana/
```

```
sudo -u kibana /usr/share/kibana/bin/kibana-plugin install  
https://packages.wazuh.com/4.x/ui/kibana/wazuh_kibana-4.4.5_7.17.9-1.zip
```

```
root@ubuntu-server:~# cd /usr/share/kibana/  
root@ubuntu-server:/usr/share/kibana# sudo -u kibana /usr/share/kibana/bin/kiban  
a-plugin install https://packages.wazuh.com/4.x/ui/kibana/wazuh_kibana-4.4.5_7.1  
7.9-1.zip  
Attempting to transfer from https://packages.wazuh.com/4.x/ui/kibana/wazuh_kiban  
a-4.4.5_7.17.9-1.zip  
Transferring 36505918 bytes.....  
Transfer complete  
Retrieving metadata from plugin archive  
Extracting plugin archive  
Extraction complete  
Plugin installation complete
```

Bước 32: Liên kết socket của kibana vào cổng đặc quyền 443

```
setcap 'cap_net_bind_service=+ep' /usr/share/kibana/node/bin/node
```

```
root@ubuntu-server:/usr/share/kibana# setcap 'cap_net_bind_service=+ep' /usr/sha  
re/kibana/node/bin/node  
root@ubuntu-server:/usr/share/kibana#
```

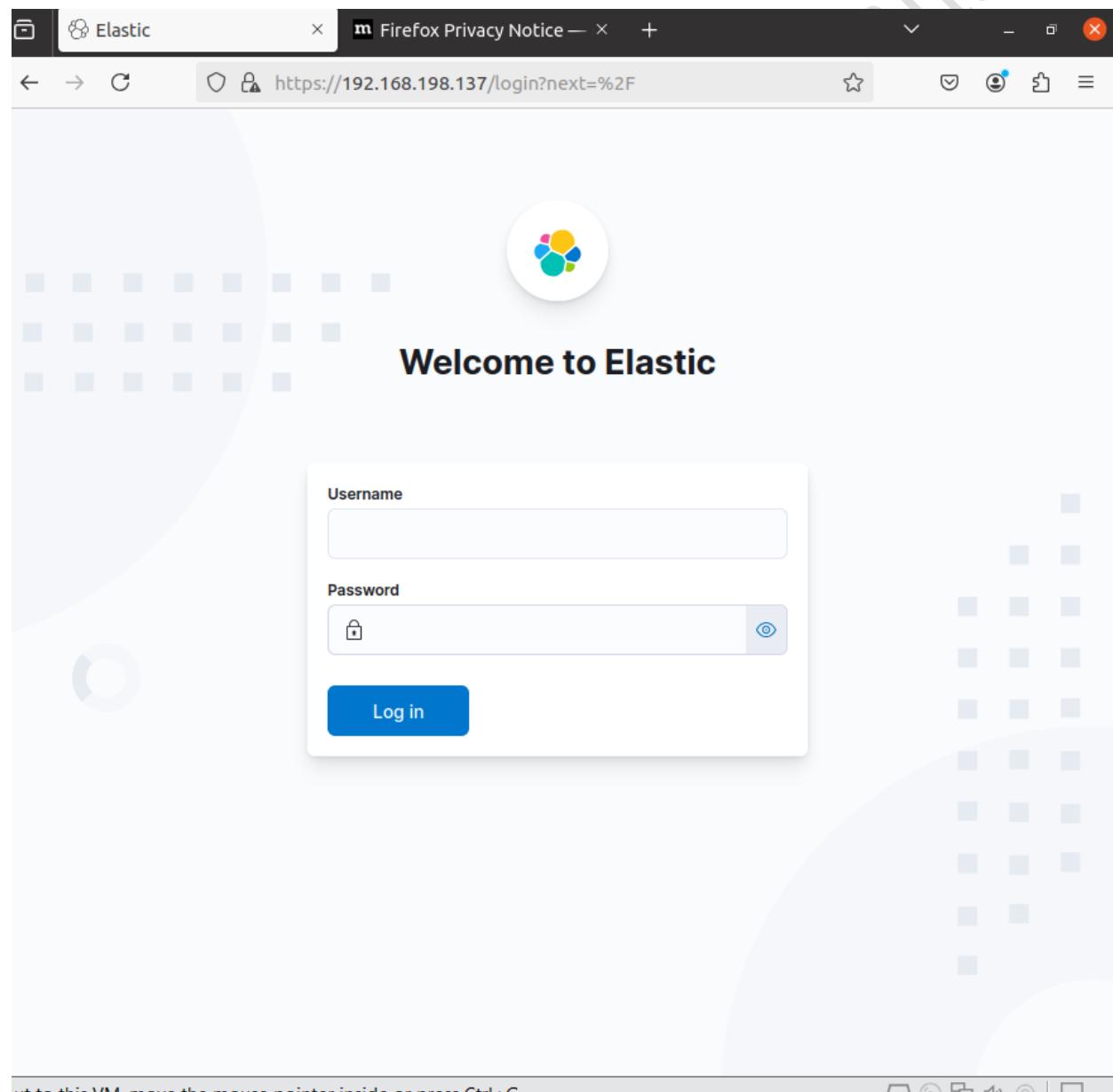
Bước 33: Kích hoạt và bắt đầu dịch vụ kibana

systemctl daemon-reload

systemctl enable kibana.service

systemctl start kibana.service

Bước 34: Truy cập vào giao diện web của Wazuh



Username: elastic

Password: *đã tạo và lưu trước đó*

4.1.2 Triển khai giám sát các Agent

- **Đầu tiên, đối với Ubuntu Agent:**

Bước 1: Import khóa GPG

```
curl -s https://packages.wazuh.com/key/GPG-KEY-WAZUH | gpg --no-default-keyring --keyring gnupg-ring:/usr/share/keyrings/wazuh.gpg --import && chmod 644 /usr/share/keyrings/wazuh.gpg
```

Bước 2: Thêm kho lưu trữ

```
echo "deb [signed-by=/usr/share/keyrings/wazuh.gpg]  
https://packages.wazuh.com/4.x/apt/ stable main" | tee -a  
/etc/apt/sources.list.d/wazuh.list
```

Bước 3: Cập nhật các thông tin gói

```
apt update
```

Bước 4: Triển khai wazuh agent

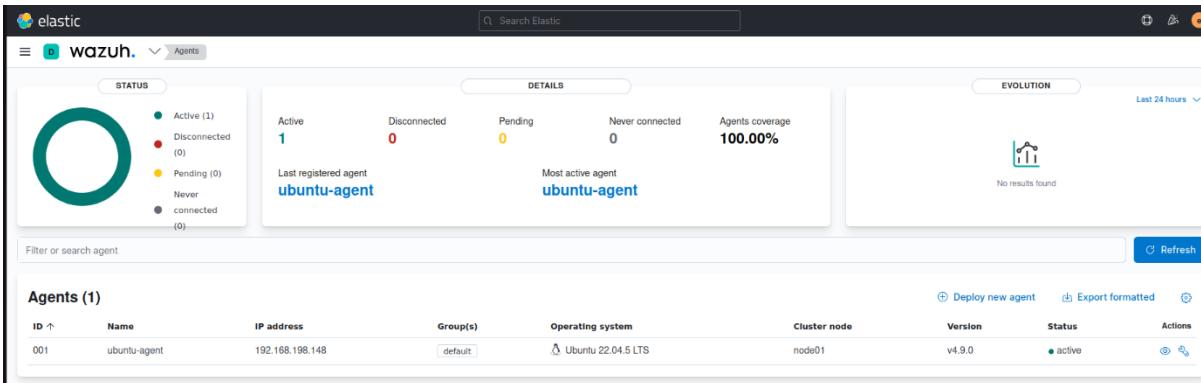
```
WAZUH_MANAGER=192.168.198.145 apt install wazuh-agent
```

192.168.198.145: Địa chỉ của wazuh server

Bước 5: Kích hoạt và bắt đầu dịch vụ wazuh agent

```
systemctl daemon-reload  
systemctl enable wazuh-agent.service  
systemctl start wazuh-agent.service
```

Kiểm tra: Ta đã thành công kết nối giám sát với máy ubuntu-agent

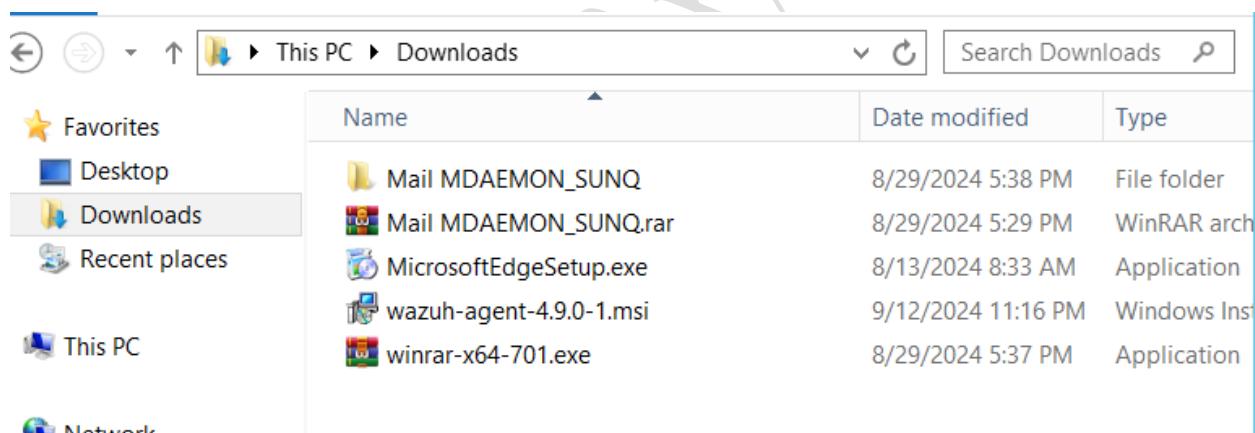


- **Tiếp theo, với máy Agent Windows Server 2012:**

Ta sẽ vào trang chủ của Wazuh để tải về phần mềm quản lý Wazuh Agent

Link: <https://documentation.wazuh.com/current/installation-guide/wazuh-agent/wazuh-agent-package-windows.html>

Bước 1: Cài đặt Wazuh Agent lưu ở thư mục download



Bước 2: Triển khai Wazuh Agent

Mở CMD

cd Downloads

wazuh-agent-4.9.0-1.msi /q WAZUH_MANAGER=192.168.198.145

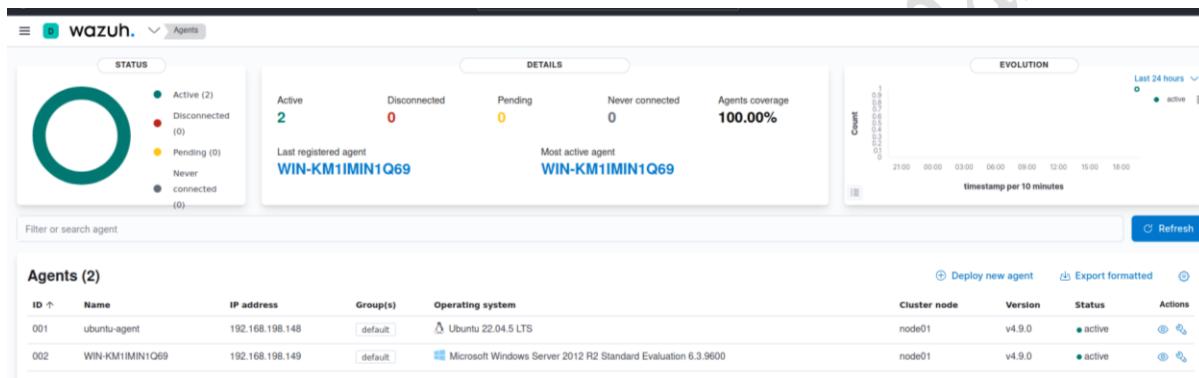
192.168.198.145: Địa chỉ của wazuh server

Bước 3: Bắt đầu chạy Wazuh Agent

NET START Wazuh

```
C:\Users\Administrator\Downloads>NET START Wazuh
The Wazuh service is starting.
The Wazuh service was started successfully.
```

Kiểm tra: thành công kết nối giám sát máy Windows Server Agent



4.2 Cấu hình Wazuh phát hiện cuộc tấn công Brute-Force

Tấn công **brute-force** (hay còn gọi là tấn công dò tìm mật khẩu bằng phương pháp vét cạn) là một phương pháp tấn công bảo mật, trong đó kẻ tấn công thử mọi tổ hợp có thể của các ký tự, số, hoặc ký hiệu cho đến khi tìm ra thông tin đăng nhập đúng hoặc khóa mã hóa.

Kịch bản: Sử dụng máy tấn công là kali linux với IP: 192.168.198.129 tấn công lên máy Agent Ubuntu bằng giao thức SSH và tấn công máy Agent Windows Server bằng giao thức RDP

- Máy tấn công ta sử dụng một công cụ có tên là Hydra

- Chạy Hydra trong vòng khoảng 2p khi cuộc tấn công diễn ra, Wazuh Server sẽ thu thập thông tin log và phân tích sau đó hiển thị cảnh báo đây là cuộc tấn công brute-force

- **Tấn công máy Agent Ubuntu**

Lưu ý: Hãy đảm bảo rằng máy Agent Ubuntu có cài đặt dịch vụ SSH và mở default port: 22

B1: Tạo file wordlist có tên là pass.txt với 10 mật khẩu khác nhau

nano pass.txt



```
GNU nano 8.1          pass.txt *
```

```
1
2
3
4
5
6
7
8
9
abc123
```

```
File Name to Write: pass.txt
^G Help      M-D DOS Format     M-A Append      M-B Backup File
^C Cancel    M-M Mac Format     M-P Prepend    ^T Browse
```

B2: Chạy hydra

hydra -l ubuntu -P pass.txt 192.168.198.148 ssh

```

└─(root㉿kali)-[~/Desktop]
# hydra -l abc -P pass.txt 192.168.198.148 ssh
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in
military or secret service organizations, or for illegal purposes (this is n
on-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-29 09:
40:52 [WARNING] Many SSH configurations limit the number of parallel tasks, it is r
ecommended to reduce the tasks: use -t 4
[DATA] max 10 tasks per 1 server, overall 10 tasks, 10 login tries (l:1/p:10)
, ~1 try per task
[DATA] attacking ssh://192.168.198.148:22/
1 of 1 target completed, 0 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-29 09:
40:57

```

Kết quả: Vào Wazuh Server để kiểm tra kết quả giám sát máy chủ Ubuntu

Với máy Agent Ubuntu có các cảnh báo sau:

- Rule ID 5710: là đang cố đăng nhập tài khoản người dùng không tồn tại. Quy tắc ở cấp độ 5. Chúng ta có thể thấy được địa chỉ nguồn của máy attacker là: 192.168.198.129.

> Sep 29, 2024 @ 20:40:57.050	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: Attempt to login using a non-existent user	5	5710
> Sep 29, 2024 @ 20:40:57.047	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: Attempt to login using a non-existent user	5	5710
> Sep 29, 2024 @ 20:40:57.047	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: Attempt to login using a non-existent user	5	5710
> Sep 29, 2024 @ 20:40:57.047	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: Attempt to login using a non-existent user	5	5710
> Sep 29, 2024 @ 20:40:57.045	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: Attempt to login using a non-existent user	5	5710
> Sep 29, 2024 @ 20:40:57.040	T1110.001 T1021.004	Credential Access, Lateral Movement	sshd: Attempt to login using a non-existent user	5	5710

Table JSON Rule

@timestamp	2024-09-29T13:40:57.050Z
_id	OxsDPpIBinS_hBkiWWFL
agent.id	001
agent.ip	192.168.198.148
agent.name	ubuntu-agent
data.srcip	192.168.198.129
data.srcuser	abc
decoder.name	sshd
decoder.parent	sshd
full_log	Sep 29 13:40:56 ubuntu-agent sshd[5612]: Failed password for invalid user abc from 192.168.198.129 port 59264 ssh2
id	1727617257.1691882
input.type	log
location	journald

- Rule ID 5503: người dùng đăng nhập thất bại. Cấp độ 5
- Rule ID 5551: nhiều lần đăng nhập không thành công trong 1 khoảng thời gian ngắn. Cấp độ 10

Sep 29, 2024 @ 20:40:55.0 93	T1110	Credential Access	PAM: Multiple failed logins in a small period of time.	10	5551
Sep 29, 2024 @ 20:40:55.0 83	T1110.001	Credential Access	PAM: User login failed.	5	5503

- **Tấn công máy Agent Windows Server**
Lưu ý: Hãy đảm bảo rằng đã cài Remote Desktop trên Agent Windows Server và cấu hình user kết nối RDP

Chạy hydra

hydra -l abc -P pass.txt rdp://192.168.198.149

```

53:18
└─(root㉿kali)-[~/Desktop]
# hydra -l abc -P pass.txt rdp://192.168.198.149
Hydra v9.5 (c) 2023 by van Hauser/THC & David Maciejak - Please do not use in
military or secret service organizations, or for illegal purposes (this is n
on-binding, these *** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2024-09-29 11:
32:15
[WARNING] rdp servers often don't like many connections, use -t 1 or -t 4 to
reduce the number of parallel connections and -W 1 or -W 3 to wait between co
nnection to allow the server to recover
[INFO] Reduced number of tasks to 4 (rdp does not like many parallel connecti
ons)
[WARNING] the rdp module is experimental. Please test, report - and if possib
le, fix.
[DATA] max 4 tasks per 1 server, overall 4 tasks, 10 login tries (l:1/p:10),
~3 tries per task
[DATA] attacking rdp://192.168.198.149:3389/
1 of 1 target completed, 0 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2024-09-29 11:
32:19

```

Vào Wazuh để kiểm tra kết quả giám sát Agent Windows Server

Với máy Agent Windows Server có cảnh báo sau:

Rule ID 60122: Lỗi người dùng đăng nhập hoặc mật khẩu sai. Cấp độ 5

T1531	Impact	Logon Failure - Unknown user or bad password	5	60122
-------	--------	--	---	-------

4.3 Cấu hình Wazuh phát hiện các cuộc tấn công SQL Injection

SQL Injection là một hình thức tấn công bảo mật trên ứng dụng web, trong đó kẻ tấn công chèn các câu lệnh SQL độc hại vào đầu vào (input) của ứng dụng để thực hiện các truy vấn không mong muốn tới cơ sở dữ liệu. Mục tiêu của cuộc tấn công này là truy cập, thay đổi hoặc xóa dữ liệu mà không được phép.

Kịch bản: Tấn công vào database. Kẻ tấn công sẽ sử dụng kỹ thuật SQL Injection để tấn công vào database của máy chủ Web:

- Wazuh Server: sẽ chịu trách nhiệm thu thập log phân tích và sau đó hiển thị cảnh báo
- Wazuh Agent: sẽ là máy victim bị tấn công SQL Injection

- Kali linux: là máy attacker

Lưu ý: Tường lửa có thể đang chặn cổng 80. Đảm bảo rằng tường lửa đã mở cổng 80 cho HTTP. Có thể sử dụng lệnh sau để kiểm tra cấu hình tường lửa:

- Trên Ubuntu với UFW (Uncomplicated Firewall):

```
sudo ufw status
```

- Nếu cổng 80 không được mở, có thể mở bằng cách:

```
sudo ufw allow 80/tcp
```

B1: Ở máy Agent Ubuntu cập nhật và cài đặt máy chủ Web Apache

```
apt install apache2
```

B2: Kiểm tra trạng thái dịch vụ Apache đã chạy hay chưa

```
systemctl status apache2.service
```

```
root@ubuntu-agent:~# systemctl status apache2.service
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.service; enabled; vendor prese>
   Active: active (running) since Mon 2024-09-30 00:12:41 +07; 1min 23s ago
     Docs: https://httpd.apache.org/docs/2.4/
 Main PID: 4543 (apache2)
    Tasks: 55 (limit: 2135)
   Memory: 5.5M
      CPU: 175ms
     CGroup: /system.slice/apache2.service
             └─4543 /usr/sbin/apache2 -k start
                 ├─4544 /usr/sbin/apache2 -k start
                 ├─4545 /usr/sbin/apache2 -k start
                 └─4546 /usr/sbin/apache2 -k start

Sep 30 00:12:41 ubuntu-agent systemd[1]: Starting The Apache HTTP Server...
Sep 30 00:12:41 ubuntu-agent apachectl[4542]: AH00558: apache2: Could not reli>
Sep 30 00:12:41 ubuntu-agent systemd[1]: Started The Apache HTTP Server.
```

B3: Cập nhật cấu hình tệp ossec.conf

Điều này cho phép máy Agent Ubuntu giám sát log truy cập của máy chủ Web Apache

Thêm đoạn script sau và lưu lại:

```
#sql_injection

<ossec_config>
    <localfile>
        <log_format>apache</log_format>
        <location>/var/log/apache2/access.log</location>
    </localfile>
</ossec_config>
```

```
GNU nano 6.2                               /var/ossec/etc/ossec.conf *

<localfile>
    <log_format>syslog</log_format>
    <location>/var/ossec/logs/active-responses.log</location>
</localfile>

<localfile>
    <log_format>syslog</log_format>
    <location>/var/log/dpkg.log</location>
</localfile>

</ossec_config>

#sql_injection
<ossec_config>
    <localfile>
        <log_format>apache</log_format>
        <location>/var/log/apache2/access.log</location>
    </localfile>
</ossec_config>

^G Help      ^O Write Out  ^W Where Is   ^K Cut          ^T Execute   ^C Location
^X Exit      ^R Read File  ^\ Replace    ^U Paste       ^J Justify   ^/ Go To Line
```

B4: Khởi động lại dịch vụ wazuh agent để áp dụng các thay đổi cấu hình

systemctl restart wazuh-agent.service

B5: Ở máy attacker nhập lệnh tấn công

```
curl -XGET "http://192.168.198.148/users/?id=SELECT+*+FROM+users";
```

```

└─(root㉿kali)-[~/Desktop]
# curl -XGET "http://192.168.198.148/users/?id=SELECT+*+FROM+users";
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<html><head>
<title>404 Not Found</title>
</head><body>
<h1>Not Found</h1>
<p>The requested URL was not found on this server.</p>
<hr>
<address>Apache/2.4.52 (Ubuntu) Server at 192.168.198.148 Port 80</address>
</body></html>

```

Kết quả: Đây là một cảnh báo về cuộc tấn công SQL Injection có:

- Rule ID là 31103 và level là 7

Sep 30,					
2024 @	T1190	Initial Access	SQL injection attempt.	7	31103
> 00:42:23.1					
42					

- Chúng ta có thể thấy được địa chỉ nguồn của cuộc tấn công này là: 192.168.198.129

data.protocol GET

data.srcip 192.168.198.129

data.url /users/?id=SELECT+*+FROM+users

decoder.name web-accesslog

- Đây là cuộc tấn công vào database nhằm lấy thông tin của người dùng

full_log 192.168.198.129 - - [30/Sep/2024:00:42:22 +0700] "GET /users/?
id=SELECT+*+FROM+users HTTP/1.1" 404 438 "-" "curl/8.8.0"

4.4 Cấu hình Wazuh chặn địa chỉ IP độc hại truy cập đến Web Server

- *Đối với máy Agent Ubuntu*

Bước 1: Cấu hình Wazuh Agent: cấu hình tệp ossec.conf: theo dõi nhật ký truy cập Apache

nano /var/ossec/etc/ossec.conf

Ctrl+W để search: gõ <localfile>, xong nhấn Enter và thêm đoạn script sau:

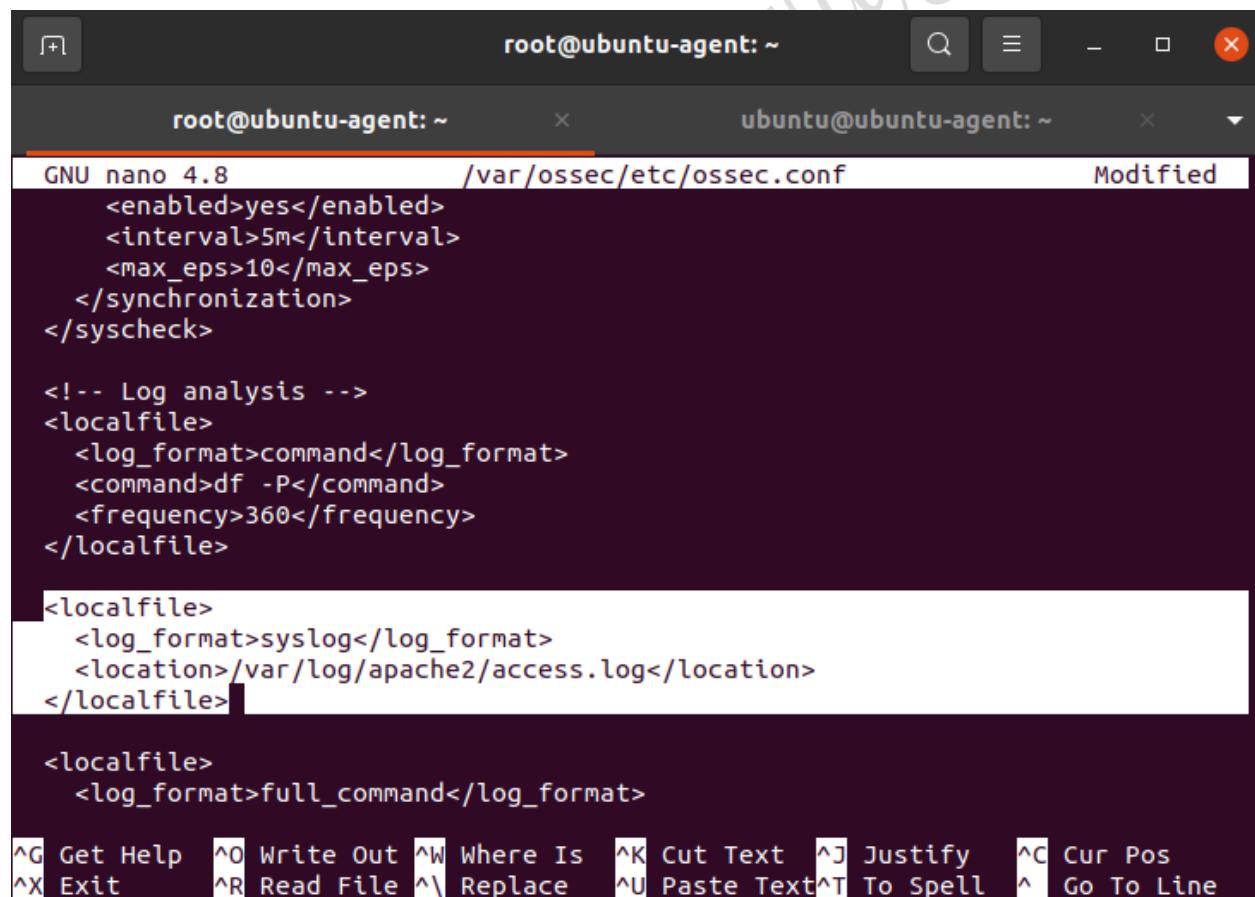
```
<localfile>

<log_format>syslog</log_format>

<location>/var/log/apache2/access.log</location>

</localfile>
```

Ctrl+X nhấn y, Enter để lưu



```
root@ubuntu-agent: ~          root@ubuntu-agent: ~          ubuntu@ubuntu-agent: ~          Modified
GNU nano 4.8                  /var/ossec/etc/ossec.conf
<enabled>yes</enabled>
<interval>5m</interval>
<max_eps>10</max_eps>
</synchronization>
</syscheck>

<!-- Log analysis -->
<localfile>
  <log_format>command</log_format>
  <command>df -P</command>
  <frequency>360</frequency>
</localfile>

<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/apache2/access.log</location>
</localfile>

<localfile>
  <log_format>full_command</log_format>
```

Hình 24. Cấu hình tệp ossec.conf

Script này là khôi lệnh theo dõi nhật ký truy cập Apache

Bước 5: Khởi động lại Wazuh Agent để áp dụng các thay đổi

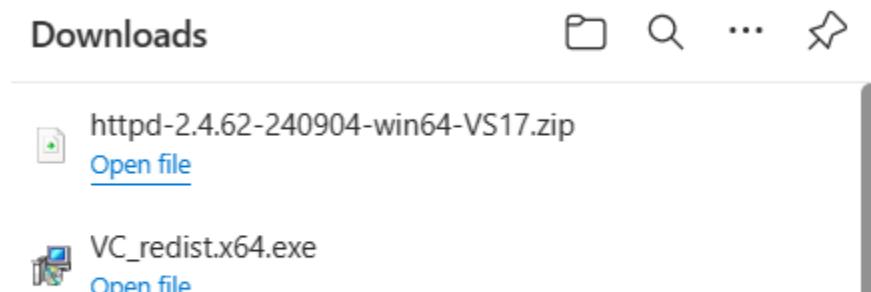
systemctl restart wazuh-agent.service

- **Đối với máy Agent Windows Server**

Bước 1: Cài đặt Web Apache

Cài Visual C++ Redistributable Visual Studio và file apache theo link sau:

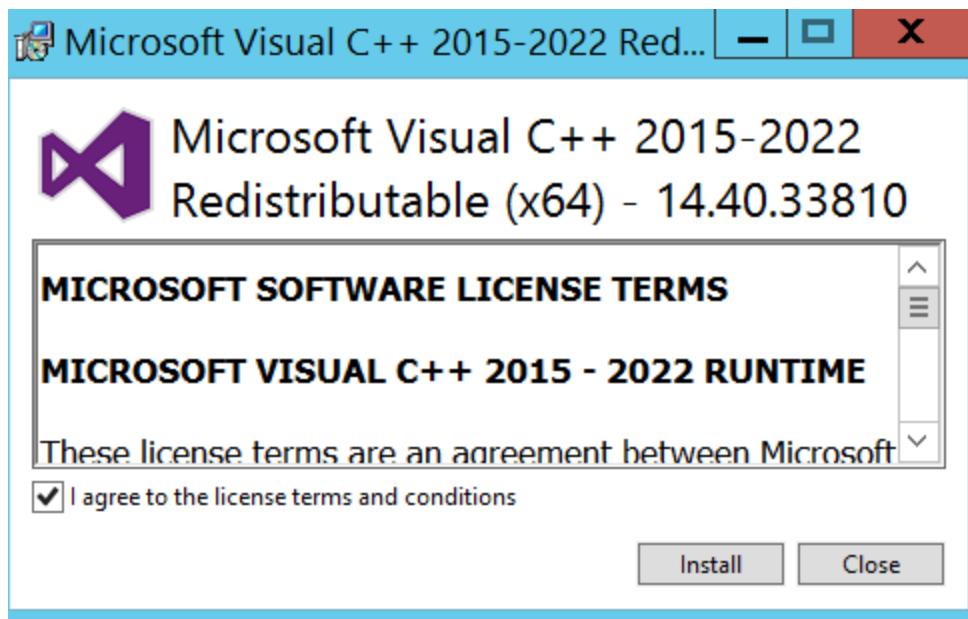
<https://www.apachelounge.com/download/>



[See more](#)

Hình 25. Download apache và visual code redistribut cho Windows Server 2012

Cài đặt vc_redist trước

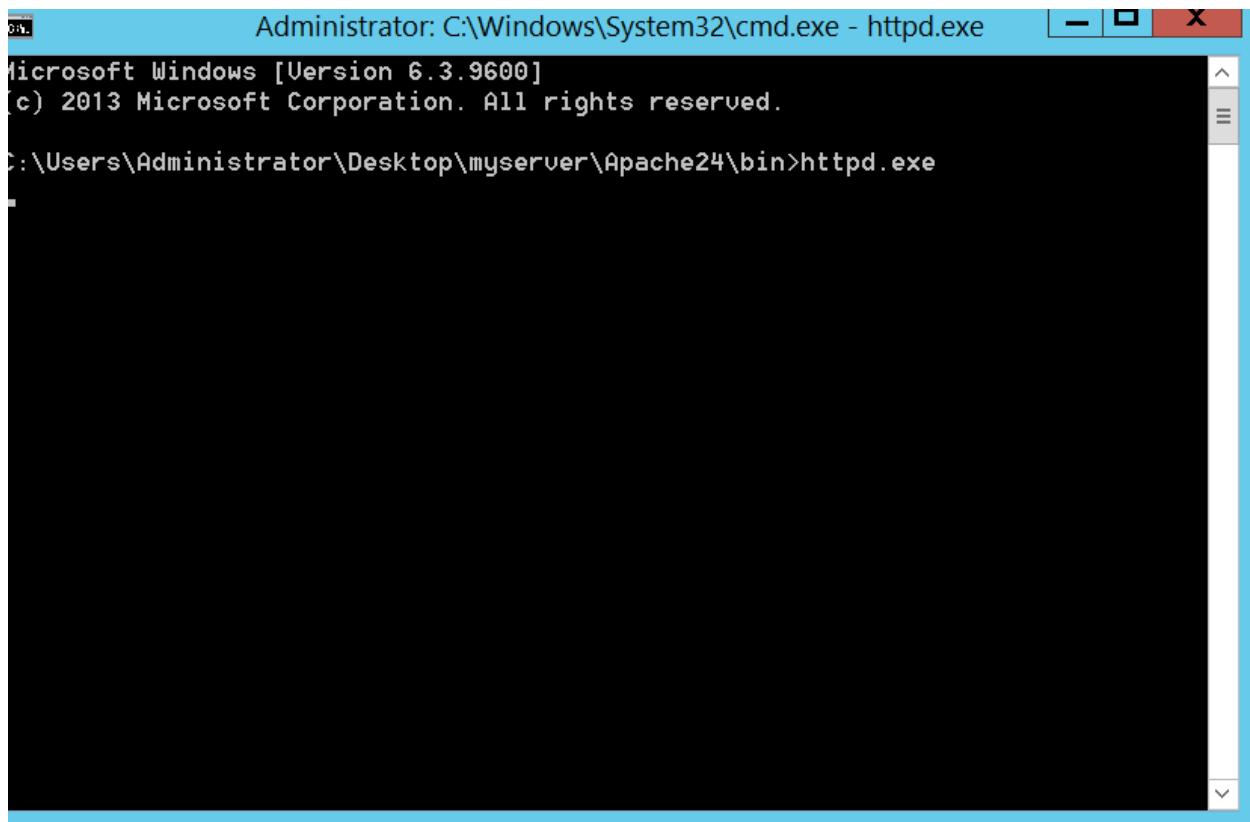


Hình 26. Cài đặt vc_redist

Tiếp theo cài đặt apache theo link hướng dẫn sau:

<https://www.bing.com/videos/riverview/relatedvideo?q=web+apache+download+for+windows+server+2012&mid=929BC09D85BB9ECB2E7C929BC09D85BB9ECB2E7C&FORM=VIRE>

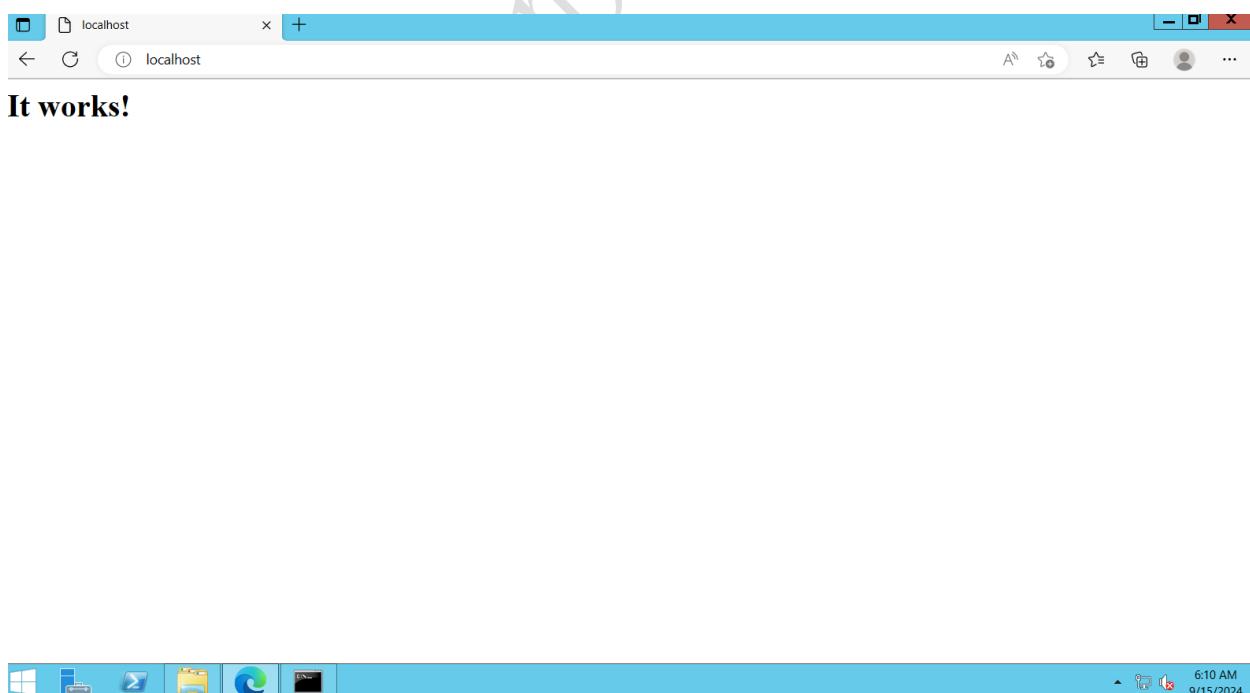
Bước 2: Chạy và kiểm tra trạng thái Apache



```
Administrator: C:\Windows\System32\cmd.exe - httpd.exe
Microsoft Windows [Version 6.3.9600]
(c) 2013 Microsoft Corporation. All rights reserved.

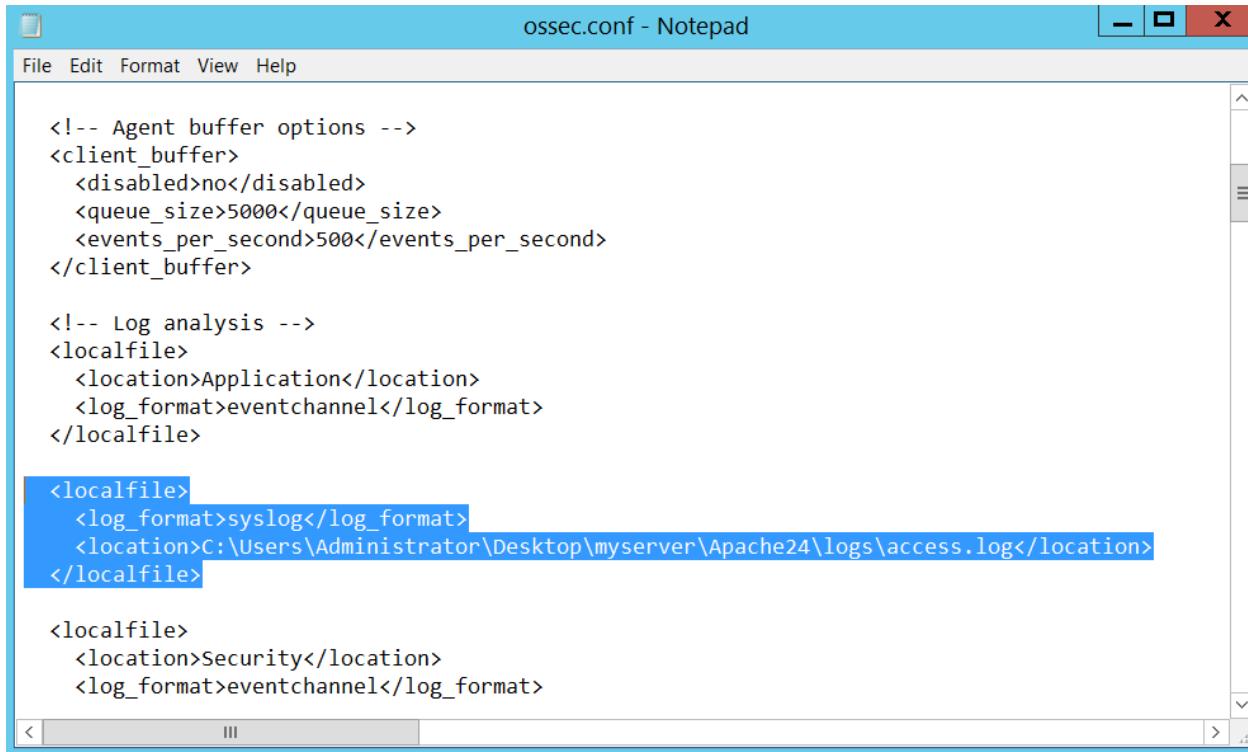
C:\Users\Administrator\Desktop\myserver\Apache24\bin>httpd.exe
```

Hình 27. Chạy httpd.exe



Hình 28. Web apache đang hoạt động trên Windows Server 2012

Bước 3: Cấu hình Wazuh Agent



```
<!-- Agent buffer options -->
<client_buffer>
  <disabled>no</disabled>
  <queue_size>5000</queue_size>
  <events_per_second>500</events_per_second>
</client_buffer>

<!-- Log analysis -->
<localfile>
  <location>Application</location>
  <log_format>eventchannel</log_format>
</localfile>

<localfile>
  <log_format>syslog</log_format>
  <location>C:\Users\Administrator\Desktop\myserver\Apache24\logs\access.log</location>
</localfile>

<localfile>
  <location>Security</location>
  <log_format>eventchannel</log_format>
```

Script này là khái lệnh theo dõi nhật ký truy cập Apache

Bước 4: Khởi động lại Wazuh agent để áp dụng các thay đổi

Mở PowerShell

Restart-Service -Name WazuhSvc

Hoặc

Restart-Service -Name wazuh

- *Đối với máy Wazuh Server*

Bước 1: Cài đặt tiện ích wget

apt update && apt install -y wget

Bước 2: Tải cơ sở dữ liệu của alienVault IP

```
wget https://raw.githubusercontent.com/firehol/blocklist-ipsets/master/alienVault_reputation.ipset -O /var/ossec/etc/lists/alienVault_reputation.ipset
```

Bước 3: Thêm địa chỉ IP attacker vào cơ sở dữ liệu alienVault IP

```
echo "192.168.198.129" >> /var/ossec/etc/lists/alienVault_reputation.ipset
```

192.168.198.129 là địa chỉ của attacker

Bước 4: Tải script để chuyển đổi định dạng tệp

```
wget https://wazuh.com/resources/iplist-to-cdblist.py -O /tmp/iplist-to-cdblist.py
```

Bước 5: Chuyển đổi định dạng alienVault .ipset sang định dạng .gpg

```
/var/ossec/framework/python/bin/python3 /tmp/iplist-to-cdblist.py  
/var/ossec/etc/lists/alienVault_reputation.ipset /var/ossec/etc/lists/blacklist-alienVault
```

```
root@ubuntu-server:~# /var/ossec/framework/python/bin/python3 /tmp/iplist-to-cdblist.py /var/ossec/etc/lists/alienVault_reputation.ipset /var/ossec/etc/lists/blacklist-alienVault  
[ /var/ossec/etc/lists/alienVault_reputation.ipset ] -> [ /var/ossec/etc/lists/blacklist-alienVault ]  
root@ubuntu-server:~#
```

Bước 6: Xóa tệp không cần thiết

```
rm -rf /var/ossec/etc/lists/alienVault_reputation.ipset
```

```
rm -rf /tmp/iplist-to-cdblist.py
```

Bước 7: Gán quyền cho tệp /var/ossec/etc/lists/blacklist-alienVault

```
chown wazuh:wazuh /var/ossec/etc/lists/blacklist-alienVault
```

Bước 8: Kích hoạt tập lệnh active response

```
nano /var/ossec/etc/rules/local_rules.xml
```

```

<group name="attack">

<rule id="100100" level="10">

<if_group>web|attack|attacks</if_group>

<list field="srcip" lookup="address_match_key">etc/lists/blacklist-
alienvault</list>

<description>IP address found in AlientVault reputation database</description>

</rule>

</group>

```

```

GNU nano 6.2                               /var/ossec/etc/rules/local_rules.xml *
<! --
Dec 10 01:02:02 host sshd[1234]: Failed none for root from 1.1.1.1 port 1066 ssh2
-->
<rule id="100001" level="5">
  <if_sid>5716</if_sid>
  <srcip>1.1.1.1</srcip>
  <description>sshd: authentication failed from IP 1.1.1.1.</description>
  <group>authentication_failed,pci_dss_10.2.4,pci_dss_10.2.5,</group>
</rule>

</group>

<group name="attack">
  <rule id="100100" level="10">
    <if_group>web|attack|attacks</if_group>
    <list field="srcip" lookup="address_match_key">etc/lists/blacklist-alienvault</list>
    <description>IP address found in AlientVault reputation database</description>
  </rule>
</group>

^G Help          ^O Write Out      ^W Where Is      ^K Cut           ^T Execute       ^C Location
^X Exit          ^R Read File       ^\ Replace        ^U Paste         ^J Justify       ^/ Go To Line

```

Bước 9: Cấu hình Wazuh Server

Thêm quy tắc để kích hoạt tập lệnh phản hồi vào tệp bộ quy tắc local rules

`nano /var/ossec/etc/ossec.conf`

Thêm:

`<list>etc/lists/blacklist-alienvault</list>`

```
<ruleset>
  <!-- Default ruleset -->
  <decoder_dir>ruleset/decoders</decoder_dir>
  <rule_dir>ruleset/rules</rule_dir>
  <rule_exclude>0215-policy_rules.xml</rule_exclude>
  <list>etc/lists/audit-keys</list>
  <list>etc/lists/amazon/aws-eventnames</list>
  <list>etc/lists/security-eventchannel</list>
  <list>etc/lists/blacklist-alienVault</list>

  <!-- User-defined ruleset -->
  <decoder_dir>etc/decoders</decoder_dir>
  <rule_dir>etc/rules</rule_dir>

  G Get Help      ^O Write Out    ^W Where Is      ^K Cut Text      ^J Justify      ^C Cur Pos
  X Exit          ^R Read File    ^\ Replace       ^U Paste Text    ^T To Spell      ^     Go To Line
```

#For Ubuntu endpoint

```
<ossec_config>
```

```
  <active-response>
```

```
    <command>firewall-drop</command>
```

```
    <location>local</location>
```

```
    <rules_id>100100</rules_id>
```

```
    <timeout>120</timeout>
```

```
  </active-response>
```

```
</ossec_config>
```

#Script này ngăn kết nối mạng đến từ điểm cuối của attacker Ubuntu trong 120s

#For Windows endpoint

```
<ossec_config>
```

```
  <active-response>
```

```
    <command>netsh</command>
```

```

<location>local</location>
<rules_id>100100</rules_id>
<timeout>120</timeout>
</active-response>
</ossec_config>

```

#Script này sẽ chặn IP attacker Windows trong 120s

```

GNU nano 6.2                               /var/ossec/etc/ossec.conf
<localfile>
  <log_format>syslog</log_format>
  <location>/var/log/dpkg.log</location>
</localfile>

</ossec_config>

#For Ubuntu endpoint
<ossec_config>
  <active-response>
    <command>firewall-drop</command>
    <location>local</location>
    <rules_id>100100</rules_id>
    <timeout>120</timeout>
  </active-response>
</ossec_config>

#For Windows endpoint
<ossec_config>
  <active-response>
    <command>netsh</command>
    <location>local</location>
    <rules_id>100100</rules_id>
    <timeout>120</timeout>
  </active-response>
</ossec_config>

```

Bước 10: Khởi động lại Wazuh Server để áp dụng các thay đổi

systemctl restart wazuh-manager.service

- Trên máy attacker truy cập vào máy chủ web apache agent ubuntu

Giả lập tấn công bằng lệnh sau:
curl http://192.168.198.148
192.168.198.148: địa chỉ ip web apache agent ubuntu

Kết quả: Vào wazuh server để xem cảnh báo:

- Phát hiện địa chỉ IP truy cập vào web server

Security Alerts					
Time	Technique(s)	Tactic(s)	Description	Level	Rule ID
Sep 30, 2024 @ 22:20:09.772			IP address found in AlienVault reputation database	10	100100
Table JSON Rule					
@timestamp	2024-09-30T15:20:09.772Z				
_id	jm-EQ5lB093sm6xshpgT				
agent.id	001				
agent.ip	192.168.198.148				
agent.name	ubuntu-agent				
data.id	200				
data.protocol	GET				
data.srcip	192.168.198.129				
data.url	/				
decoder.name	web-accesslog				
full_log	192.168.198.129 - - [30/Sep/2024:22:20:09 +0700] "GET / HTTP/1.1" 200 10926 "-" "curl/8.8.0"				
id	1727709609.4213246				
input.type	log				
location	/var/log/apache2/access.log				
manager.name	anh-att				
rule.description	IP address found in AlienVault reputation database				

- Sau đó wazuh đã block địa chỉ IP đó trong 120s

Sep 30, 2024 @ > <u>22:22:10.7</u> 80	Host Unblocked by firewall-drop Active Response	3	652
Sep 30, 2024 @ > <u>22:20:10.6</u> 37	Host Blocked by firewall-drop Active Response	3	651

- Trên máy attacker truy cập vào máy chủ web apache agent windows server

Giả lập tấn công bằng lệnh sau:

curl http://192.168.198.149

192.168.198.149: địa chỉ ip web apache agent windows server

```
(root㉿kali)-[~]
# curl http://192.168.198.149
<html><body><h1>It works!</h1></body></html>
```

Kết quả: Vào wazuh server để xem cảnh báo:

- Phát hiện địa chỉ IP truy cập vào web server

Sep 30, 2024 @ 23:02:45.796		IP address found in AlienVault reputation database	10	100100
Table	JSON	Rule		
			@timestamp	2024-09-30T16:02:45.796Z
			_id	JhyrQ5iBMfnejZT3irbr
			agent.id	002
			agent.ip	192.168.198.149
			agent.name	WIN-KM1IMIN1Q69
			data.id	200
			data.protocol	GET
			data.srcip	192.168.198.129
			data.url	/
			decoder.name	web-accesslog
			full_log	192.168.198.129 - - [30/Sep/2024:09:02:45 -0700] "GET / HTTP/1.1" 200 46

- Sau đó wazuh đã block địa chỉ IP đó trong 120s

Sep 30, 2024 @ 23:02:47.273	Active response: active-response/bin/netsh.exe - add	3	657
Table JSON Rule			
@timestamp	2024-09-30T16:02:47.273Z		
_id	JxyrQ5IBMfnejZT3jrbJ		
agent.id	002		
agent.ip	192.168.198.149		
agent.name	WIN-KM1IMIN1Q69		
data.command	add		
data.origin.module	wazuh-execd		
data.origin.name	node01		
data.parameters.alert.agent.id	002		
data.parameters.alert.agent.ip	192.168.198.149		
data.parameters.alert.agent.name	WIN-KM1IMIN1Q69		
data.parameters.alert.data.id	200		
🔍 🔗 📋 data.parameters.alert.data.id			
▼ Sep 30, 2024 @ 23:04:48.323	Active response: active-response/bin/netsh.exe - delete	3	657
Table JSON Rule			
@timestamp	2024-09-30T16:04:48.323Z		
_id	OxytQ5IBMfnejZT3d7bx		
agent.id	002		
agent.ip	192.168.198.149		
agent.name	WIN-KM1IMIN1Q69		
data.command	delete		
data.origin.module	wazuh-execd		
data.origin.name	node01		
data.parameters.alert.agent.id	002		
data.parameters.alert.agent.ip	192.168.198.149		
data.parameters.alert.agent.name	WIN-KM1IMIN1Q69		
data.parameters.alert.data.id	200		
data.parameters.alert.data.protocol	GET		
data.parameters.alert.data.srcip	192.168.198.129		
🔍 🔗 📋 data.parameters.alert.data.srcip			

4.5 Tích hợp VirusTotal để phát hiện và xóa các phần mềm độc hại

Kịch bản: Không sử dụng máy attacker. Trong trường hợp này chúng ta sẽ giám sát tính toàn vẹn của tệp và dùng API VirusTotal để quét các tệp đó. Sau đó, ta sẽ cấu hình Wazuh Server để kích hoạt lệnh phản hồi và xóa các tệp mà VirusTotal phát hiện là độc hại. Để sử dụng VirusTotal, chúng ta cần khóa API VirusTotal. Trong trường hợp sử dụng này để xác thực Wazuh Server với API VirusTotal.

1.Kịch bản tải malware ở máy Agent Ubuntu

- Ở máy Agent Ubuntu

B1: Cấu hình file ossec.conf. Cấu hình trong khối <syscheck> để thay đổi định dạng giám sát thư mục root theo thời gian thực.

nano /var/ossec/etc/ossec.conf

Thêm:

```
<directories realtime="yes">/root</directories>
```

```
<!-- File integrity monitoring -->
<syscheck>
  <disabled>no</disabled>
  <directories realtime="yes">/root</directories>

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

  <scan_on_start>yes</scan_on_start>

  <!-- Directories to check (perform all possible verifications) -->
  <directories>/etc,/usr/bin,/usr/sbin</directories>
  <directories>/bin,/sbin,/boot</directories>
```

B2: Để xử lý đầu vào json từ tập lệnh active response, ta cần cài đặt jq cho máy Agent

apt install jq

B3: Tạo file `/var/ossec/active-response/bin/remove-threat.sh`

Để kích hoạt phản hồi xóa file độc hại từ endpoint

Thêm đoạn script sau:

```
#!/bin/bash
```

```
LOCAL=`dirname $0`;
```

```
cd $LOCAL
```

```
cd ../
```

```
PWD=`pwd`
```

```
read INPUT_JSON
```

```
FILENAME=$(echo $INPUT_JSON | jq -r .parameters.alert.data.virustotal.source.file)
```

```
COMMAND=$(echo $INPUT_JSON | jq -r .command)
```

```
LOG_FILE="${PWD}/../logs/active-responses.log"
```

```
#----- Analyze command -----#
```

```
if [ ${COMMAND} = "add" ]
```

```
then
```

```
# Send control message to execd
```

```
printf '{"version":1,"origin":{ "name":"remove-threat","module":"active-response"},"command":"check_keys", "parameters":{"keys":[] }}\n'
```

```
read RESPONSE
```

```
COMMAND2=$(echo $RESPONSE | jq -r .command)
```

```
if [ ${COMMAND2} != "continue" ]
```

```
then
```

```
echo "date '+%Y/%m/%d %H:%M:%S' $0: $INPUT_JSON Remove threat active response aborted" >> ${LOGFILE}
```

```
exit 0;
```

```
fi
```

```
fi
```

```
# Removing file
rm -f $FILENAME
if [ $? -eq 0 ]; then
    echo "`date '+%Y/%m/%d %H:%M:%S'` $0: $INPUT_JSON Successfully removed
threat" >> ${LOG_FILE}
else
    echo "`date '+%Y/%m/%d %H:%M:%S'` $0: $INPUT_JSON Error removing threat"
>> ${LOG_FILE}
fi

exit 0;
```

```

root@wazuh-agent:~#
GNU nano 6.2          /var/ossec/active-response/bin/remove-threat.sh *
#!/bin/bash

LOCAL=`dirname $0`;
d $LOCAL
d ../

WD=`pwd` 

read INPUT_JSON
FILENAME=$(echo $INPUT_JSON | jq -r .parameters.alert.data.virustotal.source.file)
COMMAND=$(echo $INPUT_JSON | jq -r .command)
LOG_FILE="${PWD}/../logs/active-responses.log"

----- Analyze command -----
if [ ${COMMAND} = "add" ]
then
# Send control message to execd
printf '{"version":1,"origin":{"name":"remove-threat","module":"active-response"},"command":'
read RESPONSE
COMMAND2=$(echo $RESPONSE | jq -r .command)
if [ ${COMMAND2} != "continue" ]
then
echo "`date '+%Y/%m/%d %H:%M:%S'` $0: $INPUT_JSON Remove threat active response aborted" >> ${LOG_FILE}
exit 0;
fi
i

: Removing file
rm -f $FILENAME
if [ $? -eq 0 ]; then
echo "`date '+%Y/%m/%d %H:%M:%S'` $0: $INPUT_JSON Successfully removed threat" >> ${LOG_FILE}
else
echo "`date '+%Y/%m/%d %H:%M:%S'` $0: $INPUT_JSON Error removing threat" >> ${LOG_FILE}
fi

exit 0;else


```

B4: Thay đổi quyền sở hữu và quyền của tệp </var/ossec/active-response/bin/remove-threat.sh>

sudo chmod 750 /var/ossec/active-response/bin/remove-threat.sh

sudo chown root:wazuh /var/ossec/active-response/bin/remove-threat.sh

B5: Khởi động lại Wazuh Agent để áp dụng các thay đổi

sudo systemctl restart wazuh-agent

- Ở máy Wazuh Server

B1: Thêm các quy tắc sau vào tệp local_rules.xml

Các quy tắc này được thêm vào để cảnh báo về những thay đổi trong thư mục root được phát hiện khi FIM quét

```
nano /var/ossec/etc/rules/local_rules.xml
```

Thêm đoạn script:

```
<group name="syscheck,pci_dss_11.5,nist_800_53_SI.7,">  
    <!-- Rules for Linux systems -->  
    <rule id="100200" level="7">  
        <if_sid>550</if_sid>  
        <field name="file">/root</field>  
        <description>File modified in /root directory.</description>  
    </rule>  
    <rule id="100201" level="7">  
        <if_sid>554</if_sid>  
        <field name="file">/root</field>  
        <description>File added to /root directory.</description>  
    </rule>  
</group>
```

```

GNU nano 6.2                               /var/ossec/etc/rules/local_rules.xml *

<!-- Example -->
<group name="local,syslog,sshd,">

<!--
Dec 10 01:02:02 host sshd[1234]: Failed none for root from 1.1.1.1 port 1066 ssh2
-->
<rule id="100001" level="5">
  <if_sid>5716</if_sid>
  <srcip>1.1.1.1</srcip>
  <description>sshd: authentication failed from IP 1.1.1.1.</description>
  <group>authentication_failed,pci_dss_10.2.4,pci_dss_10.2.5,</group>
</rule>

</group>

<group name="attack">
  <rule id="100100" level="10">
    <if_group>web|attack|attacks</if_group>
    <list field="srcip" lookup="address_match_key">etc/lists/blacklist-alienVault</list>
    <description>IP address found in AlienVault reputation database</description>
  </rule>
</group>

<group name="syscheck,pci_dss_11.5,nist_800_53_SI.7,">
  <!-- Rules for Linux systems -->
  <rule id="100200" level="7">
    <if_sid>550</if_sid>
    <field name="file">/root</field>
    <description>File modified in /root directory.</description>
  </rule>
  <rule id="100201" level="7">
    <if_sid>554</if_sid>
    <field name="file">/root</field>
    <description>File added to /root directory.</description>
  </rule>
</group>

```

B2: Thêm script quy tắc vào file ossec.conf vào máy wazuh server

nano /var/ossec/etc/ossec.conf

Để kích hoạt tích hợp VirusTotal, ta cần thay thế VirusTotal API Key mặc định bằng khóa API Key VirusTotal của mình. Điều này kích hoạt truy vấn VirusTotal bất cứ khi nào.

Thêm :

#VirusTotal

<ossec_config>

```

<integration>
  <name>virustotal</name>
  <api_key>7948aa22354fe9532c6b2d1c13e46fa715376c1fba0</api_key> <!--
Replace with your VirusTotal API key ->
  <rule_id>100200,100201</rule_id>
  <alert_format>json</alert_format>
</integration>
</ossec_config>

```

```

GNU nano 6.2          /var/ossec/etc/ossec.conf *
#For Windows endpoint
<ossec_config>
  <active-response>
    <command>netsh</command>
    <location>local</location>
    <rules_id>100100</rules_id>
    <timeout>120</timeout>
  </active-response>
</ossec_config>

#VirusTotal
<ossec_config>
  <integration>
    <name>virustotal</name>
    <api_key>7948aa22354fe9532c6b2d185392c672a5749c13e46fa715376c1fba0</api_key>
    <rule_id>100200,100201</rule_id>
    <alert_format>json</alert_format>
  </integration>
</ossec_config>

^G Help      ^O Write Out   ^W Where Is   ^K Cut        ^T Execute
^X Exit      ^R Read File   ^\ Replace    ^U Paste      ^J Justify

```

Tiếp tục thêm:

```
<ossec_config>
<command>
  <name>remove-threat</name>
  <executable>remove-threat.sh</executable>
  <timeout_allowed>no</timeout_allowed>
</command>
<active-response>
  <disabled>no</disabled>
  <command>remove-threat</command>
  <location>local</location>
  <rules_id>87105</rules_id>
</active-response>
</ossec_config>
```

```

#VirusTotal
<ossec_config>
  <integration>
    <name>virustotal</name>
    <api_key>7948aa22354fe9532c6b2d185392c672a5</api_key>
    <rule_id>100200,100201</rule_id>
    <alert_format>json</alert_format>
  </integration>
</ossec_config>

<ossec_config>
  <command>
    <name>remove-threat</name>
    <executable>remove-threat.sh</executable>
    <timeout_allowed>no</timeout_allowed>
  </command>
  <active-response>
    <disabled>no</disabled>
    <command>remove-threat</command>
    <location>local</location>
    <rules_id>87105</rules_id>
  </active-response>
</ossec_config>

```

Điều này cho phép trực tiếp phản hồi và kích hoạt tập lệnh remove.sh khi VirusTotal gắn cờ một tệp là độc hại.

B3: Thêm quy tắc sau vào tệp local_rules.xml vào máy Wazuh server

sudo nano /var/ossec/etc/rules/local_rules.xml

```

<group name="virustotal">
  <rule id="100092" level="12">
    <if_sid>657</if_sid>
    <match>Successfully removed threat</match>

```

```

<description>$parameters.program removed threat located at
$(parameters.alert.data.virustotal.source.file)</description>

</rule>

<rule id="100093" level="12">
  <if_sid>657</if_sid>
  <match>Error removing threat</match>
  <description>Error removing threat located at
$(parameters.alert.data.virustotal.source.file)</description>
</rule>

</group>

<!-- Rules for Linux systems -->
<rule id="100200" level="7">
  <if_sid>550</if_sid>
  <field name="file">/root</field>
  <description>File modified in /root directory.</description>
</rule>
<rule id="100201" level="7">
  <if_sid>554</if_sid>
  <field name="file">/root</field>
  <description>File added to /root directory.</description>
</rule>
</group>

<group name="virustotal,">
  <rule id="100092" level="12">
    <if_sid>657</if_sid>
    <match>Successfully removed threat</match>
    <description>$parameters.program removed threat located at $(parameters.alert.data.vir
  </rule>
  <rule id="100093" level="12">
    <if_sid>657</if_sid>
    <match>Error removing threat</match>
    <description>Error removing threat located at $(parameters.alert.data.virustotal.source.
  </rule>
</group>

```

Khối lệnh này cảnh báo về kết quả phản hồi đang hoạt động

B4: Khởi động lại máy Wazuh server để áp dụng các thay đổi
 systemctl restart wazuh-manager.service

Tại máy Agent Ubuntu tải xuống một tệp độc hại thử nghiệm vào thư mục root

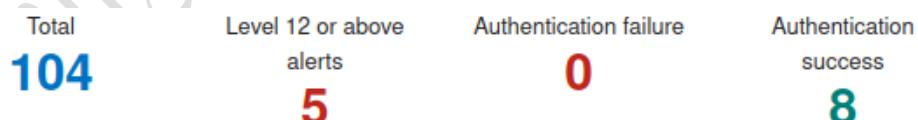
curl -LO <https://secure.eicar.org/eicar.com> && ls -lah eicar.com

```
root@ubuntu-agent:~# curl -LO https://secure.eicar.org/eicar.com && ls -lah eicar.com
% Total    % Received % Xferd  Average Speed   Time     Time      Current
                                         Dload  Upload   Total   Spent   Left  Speed
100     68  100     68    0      0   17      0  0:00:04  0:00:03  0:00:01   17
-rw-r--r-- 1 root root 68  14:46 eicar.com
root@ubuntu-agent:~#
```

Kết quả:

Security Alerts						
Time ↓	Technique(s)	Tactic(s)	Description	Level	Rule ID	
Oct 1, 2024 @ > 16:40:10.17 6			active-response/bin/remove-threat.sh removed threat located at /root/eicar.com	12	100092	
Oct 1, 2024 @ > 16:40:08.99 1	T1070.004 T1485	Defense Evasion, Impact	File deleted.	7	553	
Oct 1, 2024 @ > 16:40:08.71 0	T1203	Execution	VirusTotal: Alert - /root/eicar.com - 63 engines detected this file	12	87105	

- Phát hiện cảnh báo file độc hại với Rule ID là 87105 và level là 12. Cấp cảnh báo 12 này cho ta biết là event rất quan trọng



- Có 63 công cụ đã phát hiện ra file này là file độc hại

location	virustotal
manager.name	anh-attt
rule.description	VirusTotal: Alert - /root/eicar.com - 63 engines detected this file
rule.firedtimes	1

- Và Wazuh đã xóa file độc hại này ngay sau đó

Oct 1, 2024				
@	T1070.004	T1485	Defense Evasion, Impact	File deleted.
16:40:08.99				
1				

Table **JSON** **Rule**

@timestamp	2024-10-01T09:40:08.991Z
------------	--------------------------

_id	JupzR5IB9Dlp8o-apYEI
-----	----------------------

agent.id	001
----------	-----

agent.ip	192.168.198.148
----------	-----------------

agent.name	ubuntu-agent
------------	--------------

decoder.name	syscheck_deleted
--------------	------------------

full_log	File '/root/eicar.com' deleted Mode: realtime
----------	--

- Cuối cùng là hiển thị phản hồi xóa threat nằm ở /root

location	/var/ossec/logs/active-responses.log
----------	--------------------------------------

manager.name	anh-attt
--------------	----------

rule.description	active-response/bin/remove-threat.sh removed threat located at /root/eicar.com
------------------	--

rule.firedtimes	1
-----------------	---

2. Kịch bản tải malware ở máy Agent Windows Server

- Ở máy Agent Windows Server

B1: Vào tệp ossec.conf

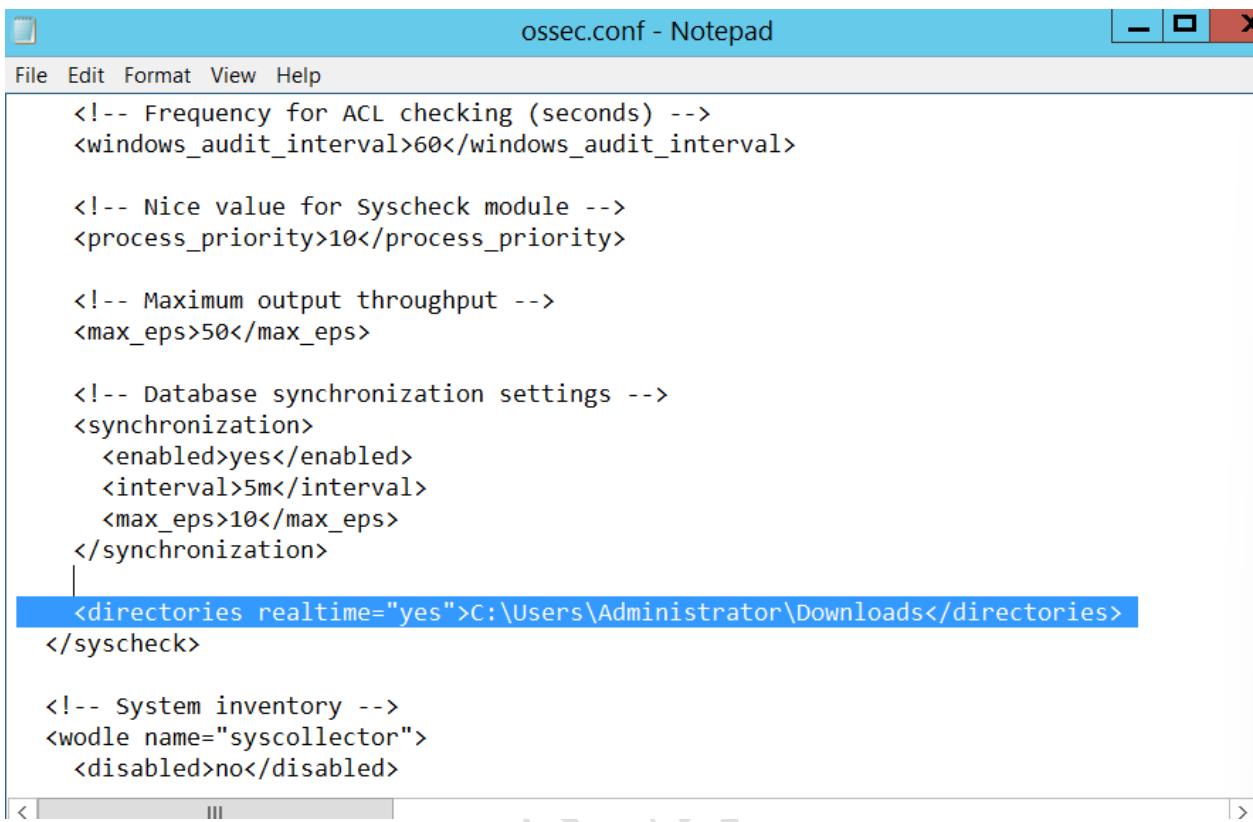
Tìm khôi <syscheck>: đảm bảo nó được đặt <disabled>no</disabled>

⇒ Điều này cho phép module Wazuh FIM giám sát các thay đổi thư mục

Thêm script trong khối <syscheck> để định cấu hình 1 thư mục để được theo dõi theo thời gian thực realtime.

Trong TH này, định cấu hình Wazuh để giám sát thư mục :

```
<directories realtime="yes">C:\Users\Administrator\Downloads</directories>
```



The screenshot shows a Notepad window titled "ossec.conf - Notepad". The file contains XML configuration code for the Ossec host agent. A specific line, "`<directories realtime="yes">C:\Users\Administrator\Downloads</directories>`", is highlighted with a blue selection bar.

```
<!-- Frequency for ACL checking (seconds) -->
<windows_audit_interval>60</windows_audit_interval>

<!-- Nice value for Syscheck module -->
<process_priority>10</process_priority>

<!-- Maximum output throughput -->
<max_eps>50</max_eps>

<!-- Database synchronization settings -->
<synchronization>
    <enabled>yes</enabled>
    <interval>5m</interval>
    <max_eps>10</max_eps>
</synchronization>

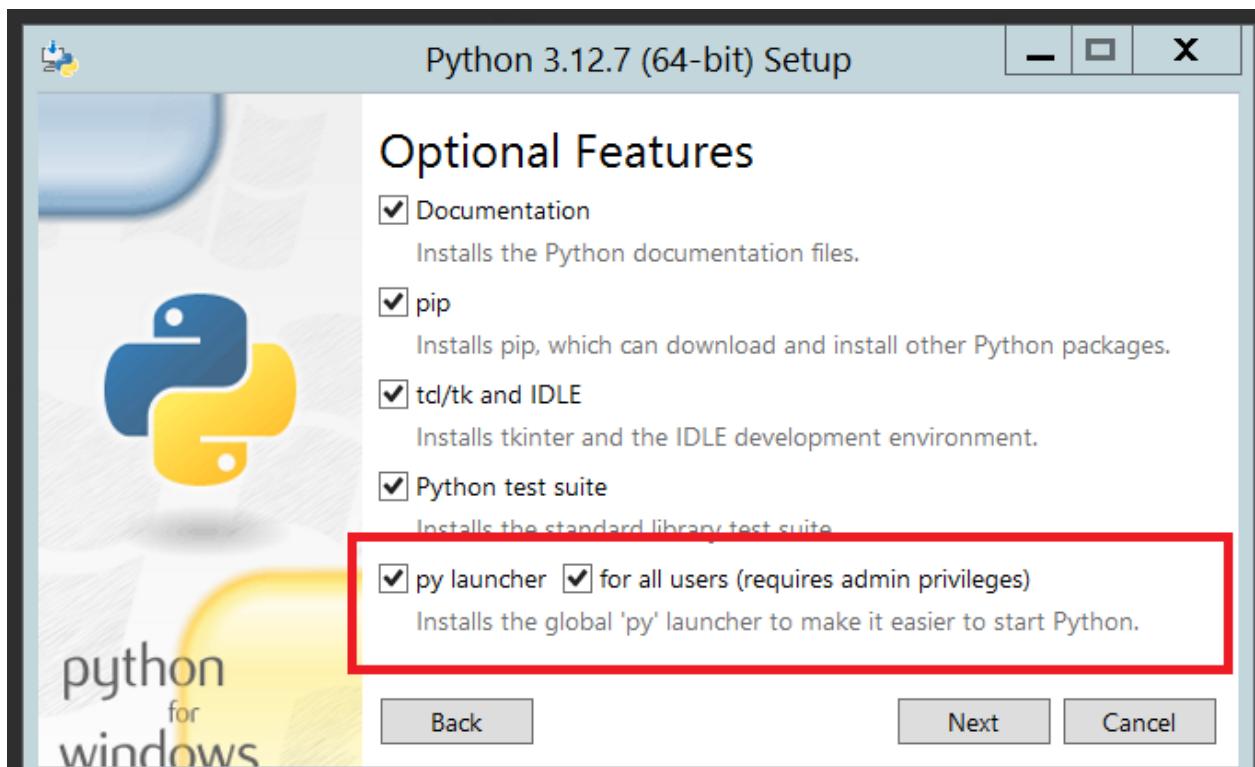
<directories realtime="yes">C:\Users\Administrator\Downloads</directories>
</syscheck>

<!-- System inventory -->
<wodle name="syscollector">
    <disabled>no</disabled>
```

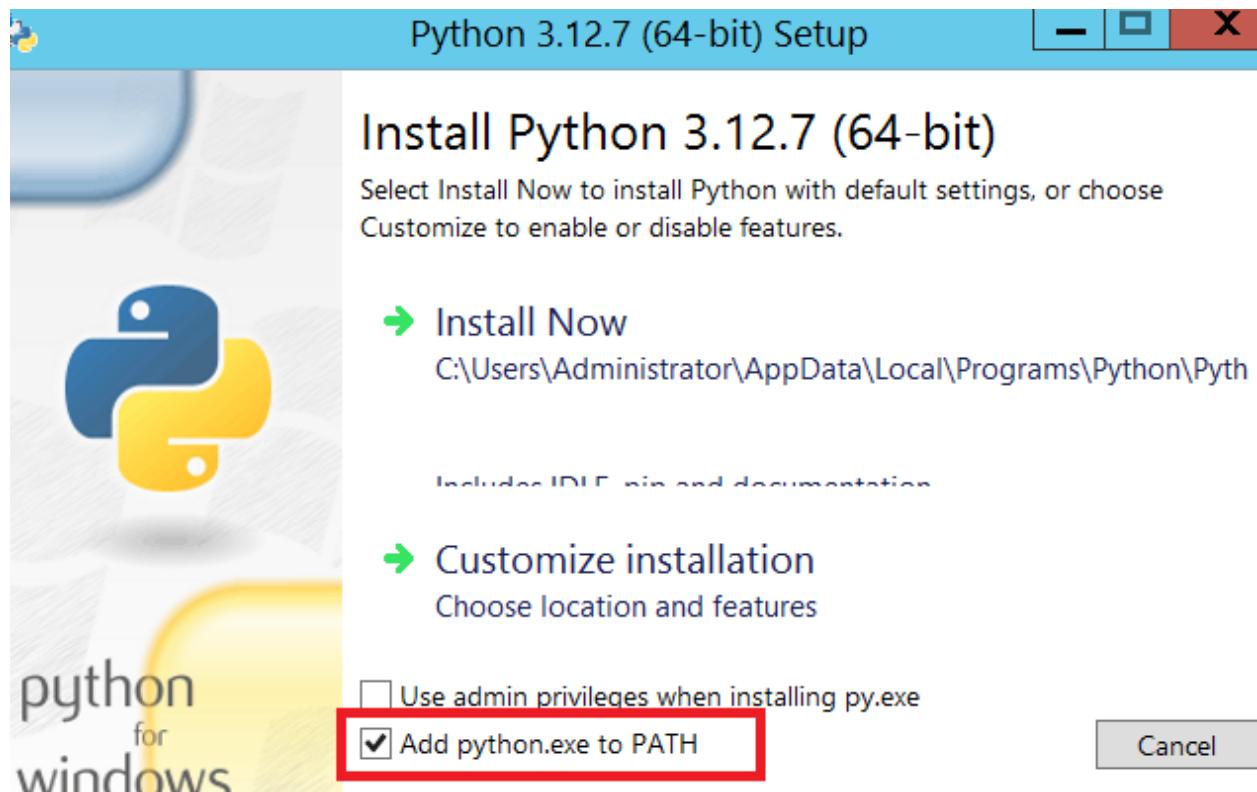
B2: Tải xuống trình cài đặt thực thi Python từ [trang web chính thức của Python](#).

B3: Chạy trình cài đặt Python sau khi tải xuống. Đảm bảo kiểm tra các box sau:

- `Install launcher for all users`



- [Add Python 3.X to PATH](#)(Điều này đặt trình thông dịch vào đường dẫn thực thi)



Kiểm tra xem python đã cài thành công chưa

```
Windows PowerShell
Copyright (C) 2013 Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> python
Python 3.12.7 (tags/v3.12.7:0b05ead, Oct 1 2024, 03:06:41) [MSC v.1941 64 bit
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

B4: Sau khi Python hoàn tất quá trình cài đặt, hãy mở terminal PowerShell dành cho quản trị viên và sử dụng **pip** để cài đặt PyInstaller:

pip install pyinstaller

pyinstaller --version

```

PS C:\Users\Administrator\Downloads> pip install pyinstaller
Collecting pyinstaller
  Downloading pyinstaller-6.10.0-py3-none-win_amd64.whl.metadata (8.3 kB)
Collecting setuptools>=42.0.0 (from pyinstaller)
  Downloading setuptools-75.1.0-py3-none-any.whl.metadata (6.9 kB)
Collecting altgraph (from pyinstaller)
  Downloading altgraph-0.17.4-py2.py3-none-any.whl.metadata (7.3 kB)
Collecting pyinstaller-hooks-contrib>=2024.8 (from pyinstaller)
  Downloading pyinstaller_hooks_contrib-2024.8-py3-none-any.whl.metadata (16 kB)
Collecting packaging>=22.0 (from pyinstaller)
  Downloading packaging-24.1-py3-none-any.whl.metadata (3.2 kB)
Collecting pefile>=2022.5.30 (from pyinstaller)
  Downloading pefile-2024.8.26-py3-none-any.whl.metadata (1.4 kB)
Collecting pywin32-ctypes>=0.2.1 (from pyinstaller)
  Downloading pywin32_ctypes-0.2.3-py3-none-any.whl.metadata (3.9 kB)
Downloading pyinstaller-6.10.0-py3-none-win_amd64.whl (1.3 MB)
----- 1.3/1.3 MB 7.6 MB/s eta 0:00:00
Downloading packaging-24.1-py3-none-any.whl (53 kB)
Downloading pefile-2024.8.26-py3-none-any.whl (74 kB)
Downloading pyinstaller_hooks_contrib-2024.8-py3-none-any.whl (322 kB)
Downloading pywin32_ctypes-0.2.3-py3-none-any.whl (30 kB)
Downloading setuptools-75.1.0-py3-none-any.whl (1.2 MB)
----- 1.2/1.2 MB 10.5 MB/s eta 0:00:00
Downloading altgraph-0.17.4-py2.py3-none-any.whl (21 kB)
Installing collected packages: altgraph, setuptools, pywin32-ctypes, pefile, pyinstaller
Successfully installed altgraph-0.17.4 packaging-24.1 pefile-2024.8.26 pyinstaller-6.10.0
PS C:\Users\Administrator\Downloads> pyinstaller --version

```

Ở đây, bạn sử dụng Pyinstaller để chuyển đổi tập lệnh Python phản hồi đang hoạt động thành ứng dụng thực thi có thể chạy trên điểm cuối endpoint Windows.

B5: Tạo một tập lệnh active response `remove-threat.py` để xóa tệp khỏi điểm cuối Windows

Thêm đoạn script này vào file `remove-threat.py`:

```

#!/usr/bin/python3

# Copyright (C) 2015-2022, Wazuh Inc.

# All rights reserved.

```

```

import os
import sys
import json
import datetime

```

```

if os.name == 'nt':
    LOG_FILE = "C:\\Program Files (x86)\\ossec-agent\\active-response\\active-
responses.log"
else:
    LOG_FILE = "/var/ossec/logs/active-responses.log"

ADD_COMMAND = 0
DELETE_COMMAND = 1
CONTINUE_COMMAND = 2
ABORT_COMMAND = 3

OS_SUCCESS = 0
OS_INVALID = -1

class message:
    def __init__(self):
        self.alert = ""
        self.command = 0

    def write_debug_file(ar_name, msg):
        with open(LOG_FILE, mode="a") as log_file:
            log_file.write(str(datetime.datetime.now().strftime('%Y/%m/%d %H:%M:%S'))
+ " " + ar_name + ": " + msg +"\n")

    def setup_and_check_message(argv):
        # get alert from stdin

```

```
input_str = ""

for line in sys.stdin:
    input_str += line
    break

try:
    data = json.loads(input_str)
except ValueError:
    write_debug_file(argv[0], 'Decoding JSON has failed, invalid input format')
    message.command = OS_INVALID
    return message

message.alert = data

command = data.get("command")

if command == "add":
    message.command = ADD_COMMAND
elif command == "delete":
    message.command = DELETE_COMMAND
else:
    message.command = OS_INVALID
    write_debug_file(argv[0], 'Not valid command: ' + command)

return message
```

```
def send_keys_and_check_message(argv, keys):

    # build and send message with keys
    keys_msg = json.dumps({ "version": 1,"origin":{ "name": argv[0],"module":"active-response"}, "command":"check_keys", "parameters":{ "keys":keys }})

    write_debug_file(argv[0], keys_msg)

    print(keys_msg)
    sys.stdout.flush()

    # read the response of previous message
    input_str = ""

    while True:

        line = sys.stdin.readline()

        if line:

            input_str = line
            break

    # write_debug_file(argv[0], input_str)

    try:

        data = json.loads(input_str)

    except ValueError:
```

```
    write_debug_file(argv[0], 'Decoding JSON has failed, invalid input format')
    return message

action = data.get("command")

if "continue" == action:
    ret = CONTINUE_COMMAND
elif "abort" == action:
    ret = ABORT_COMMAND
else:
    ret = OS_INVALID
    write_debug_file(argv[0], "Invalid value of 'command'")

return ret

def main(argv):
    write_debug_file(argv[0], "Started")

    # validate json and get command
    msg = setup_and_check_message(argv)

    if msg.command < 0:
        sys.exit(OS_INVALID)

    if msg.command == ADD_COMMAND:
```

```
alert = msg.alert["parameters"]["alert"]
keys = [alert["rule"]["id"]]
action = send_keys_and_check_message(argv, keys)

# if necessary, abort execution
if action != CONTINUE_COMMAND:

    if action == ABORT_COMMAND:
        write_debug_file(argv[0], "Aborted")
        sys.exit(OS_SUCCESS)

    else:
        write_debug_file(argv[0], "Invalid command")
        sys.exit(OS_INVALID)

try:
    file_path =
msg.alert["parameters"]["alert"]["data"]["virustotal"]["source"]["file"]
    if os.path.exists(file_path):
        os.remove(file_path)
        write_debug_file(argv[0], json.dumps(msg.alert) + " Successfully removed
threat")

    except OSError as error:
        write_debug_file(argv[0], json.dumps(msg.alert) + "Error removing threat")

else:
    write_debug_file(argv[0], "Invalid command")
```

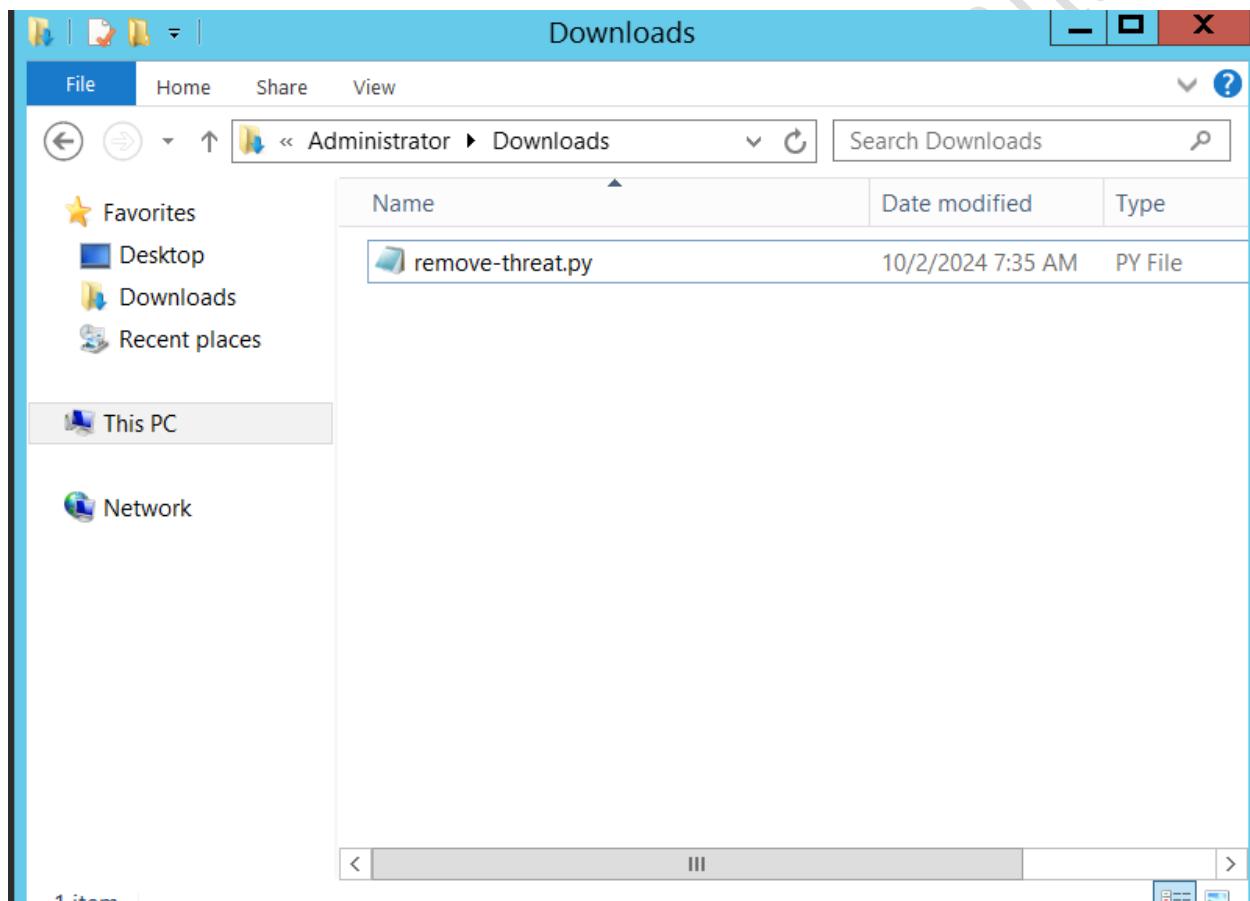
```

        write_debug_file(argv[0], "Ended")

        sys.exit(OS_SUCCESS)

if __name__ == "__main__":
    main(sys.argv)

```

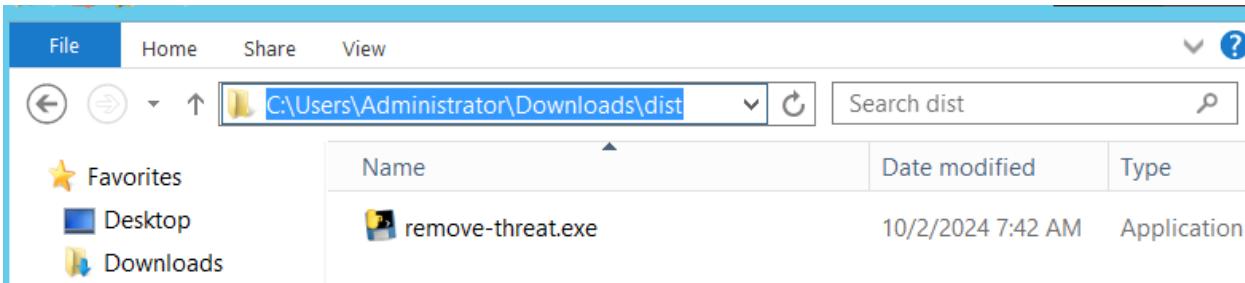


B6: Chuyển đổi tập lệnh Python phản hồi đang hoạt động **remove-threat.py** thành ứng dụng thực thi Windows. Chạy lệnh PowerShell sau với tư cách quản trị viên để tạo tệp thực thi:

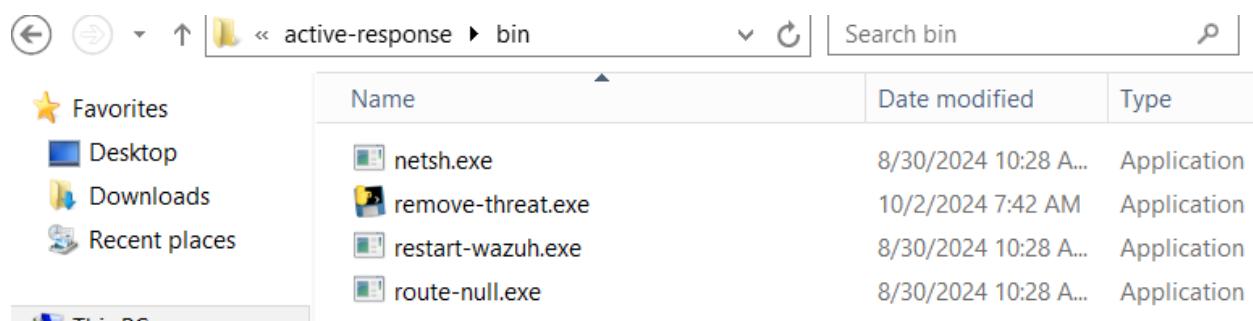
```
> pyinstaller -F \path_to_remove-threat.py
```

```
PS C:\Users\Administrator\Downloads> pyinstaller -F C:\Users\Administrator\Downloads\remove-threat.py
```

Lưu ý đường dẫn nơi **pyinstaller** tạo ra **remove-threat.exe**



B7: Di chuyển tập tin thực thi remove-threat.exe vào thư mục **C:\Program Files (x86)\ossec-agent\active-response\bin**



B8: Khởi động lại tác nhân Wazuh để áp dụng các thay đổi. Chạy lệnh PowerShell sau với tư cách quản trị viên:

```
> Restart-Service -Name wazuh
```

- Ở máy Wazuh Server

Thực hiện các bước sau trên Wazuh Server để cấu hình tích hợp VirusTotal. Các bước này cũng kích hoạt và kích hoạt tập lệnh phản hồi hoạt động bất cứ khi nào phát hiện tệp đáng ngờ.

B1: Thêm cấu hình sau vào **/var/ossec/etc/ossec.conf trên máy chủ Wazuh để kích hoạt tích hợp VirusTotal.**

Thay thế <YOUR_VIRUS_TOTAL_API_KEY>bằng [khóa API VirusTotal](#) của chính mình . Điều này cho phép kích hoạt truy vấn VirusTotal bất cứ khi nào bất kỳ quy tắc nào trong nhóm [syscheck](#) FIM được kích hoạt:

```
<ossec_config>
  <integration>
    <name>virustotal</name>
    <api_key><YOUR_VIRUS_TOTAL_API_KEY></api_key> <!-- Replace with your
VirusTotal API key -->
    <group>syscheck</group>
    <alert_format>json</alert_format>
  </integration>
</ossec_config>
```

B2: Thêm các khối sau vào tệp [/var/ossec/etc/ossec.conf](#) máy chủ Wazuh.

Điều này cho phép phản hồi chủ động và kích hoạt tệp [remove-threat.exe](#) thực thi khi truy vấn VirusTotal trả về kết quả khớp dương tính với các mối đe dọa:

```
<ossec_config>
  <command>
    <name>remove-threat</name>
    <executable>remove-threat.exe</executable>
    <timeout_allowed>no</timeout_allowed>
  </command>

  <active-response>
    <disabled>no</disabled>
    <command>remove-threat</command>
    <location>local</location>
    <rules_id>87105</rules_id>
  </active-response>
</ossec_config>
```

```

#VirusTotal for Agent Windows Server
<ossec_config>
  <integration>
    <name>virustotal</name>
    <api_key>7948aa22354fe9532c6b2d185392c672a5749c13e46fa715376c1fba02ee8938</api_key> <!-- Re>
    <group>syscheck</group>
    <alert_format>json</alert_format>
  </integration>
</ossec_config>

<ossec_config>
  <command>
    <name>remove-threat</name>
    <executable>remove-threat.exe</executable>
    <timeout_allowed>no</timeout_allowed>
  </command>

  <active-response>
    <disabled>no</disabled>
    <command>remove-threat</command>
    <location>local</location>
    <rules_id>87105</rules_id>
  </active-response>
</ossec_config>

```

B3: Thêm các quy tắc sau vào tệp `/var/ossec/etc/rules/local_rules.xml` máy chủ Wazuh để cảnh báo về kết quả active response.

```

<group name="virustotal,">
  <rule id="100092" level="12">
    <if_sid>657</if_sid>
    <match>Successfully removed threat</match>
    <description>$ (parameters.program) removed threat located at
$ (parameters.alert.data.virustotal.source.file)</description>
  </rule>

  <rule id="100093" level="12">
    <if_sid>657</if_sid>
    <match>Error removing threat</match>
    <description>Error removing threat located at
$ (parameters.alert.data.virustotal.source.file)</description>
  </rule>
</group>

```

```

#For Agent Windows Server
<group name="virustotal,">
  <rule id="100092" level="12">
    <if_sid>657</if_sid>
    <match>Successfully removed threat</match>
    <description>$(parameters.program) removed threat located at $(parameters.alert.data.virustotal.source.file)</description>
  </rule>

  <rule id="100093" level="12">
    <if_sid>657</if_sid>
    <match>Error removing threat</match>
    <description>Error removing threat located at $(parameters.alert.data.virustotal.source.file)</description>
  </rule>
</group>

```

B4: Khởi động lại trình quản lý Wazuh để áp dụng các thay đổi cấu hình:

systemctl restart wazuh-manager.service

- Mô phỏng tấn công:

B1: Thực hiện theo các bước sau để tạm thời tắt tính năng bảo vệ chống vi-rút Microsoft Defender theo thời gian thực trong Windows Security:

- Nhập vào menu **Start** và nhập để tìm kiếm ứng dụng đó. **Windows Security**
- Chọn **Windows Security app** từ kết quả, đi tới **Virus & threat protection** và trong phần **Virus & threat protection settings**, chọn **Manage settings**.
- Tắt tính năng **Real-time protection**.

B2: Tải tệp EICAR test xuống **C:\Users\<USER_NAME>\Downloads** thư mục trên điểm cuối Windows.

```

> Invoke-WebRequest -Uri https://secure.eicar.org/eicar.com.txt -OutFile eicar.txt
> cp .\eicar.txt C:\Users\<USER_NAME>\Downloads

```

```

PS C:\Users\Administrator> Invoke-WebRequest -Uri https://secure.eicar.org/eicar.com.txt -OutFile eicar.txt
PS C:\Users\Administrator> cp .\eicar.txt C:\Users\Administrator\Downloads
PS C:\Users\Administrator>

```

Nếu báo lỗi này:

```
PS C:\Users\Administrator> Invoke-WebRequest -Uri https://secure.eicar.org/eicar.com.txt -OutFile eicar.txt
Invoke-WebRequest : The request was aborted: Could not create SSL/TLS secure channel.
At line:1 char:1
+ Invoke-WebRequest -Uri https://secure.eicar.org/eicar.com.txt -OutFile eicar.txt
+ ~~~~~
+ CategoryInfo          : InvalidOperation: (System.Net.HttpWebRequest:HttpWebRequest) [Invoke-WebRequest], WebException
+ FullyQualifiedErrorId : WebCmdletWebResponseException,Microsoft.PowerShell.Commands.InvokeWebRequestCommand
```

Lỗi này là do PowerShell không thể thiết lập kết nối SSL/TLS an toàn với máy chủ. Điều này thường xảy ra khi phiên bản TLS mặc định của hệ thống không tương thích với yêu cầu bảo mật của máy chủ đích.

thì ta có thể fix như sau:

1. Mở PowerShell với quyền Administrator:

- Nhấn Start, gõ **PowerShell**.
- Nhấp chuột phải vào **Windows PowerShell** và chọn **Run as administrator**.

2. Chạy lệnh sau để thiết lập TLS 1.2:

```
[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
```

Hoặc

[Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12 -bor
[Net.SecurityProtocolType]::Tls11 -bor [Net.SecurityProtocolType]::Tls (Nếu muốn hỗ trợ cả
các phiên bản TLS khác (như TLS 1.1 hoặc TLS 1.0), có thể kết hợp chúng bằng cách sử dụng
toàn tử -bor)

```
PS C:\Users\Administrator> [Net.ServicePointManager]::SecurityProtocol = [Net.SecurityProtocolType]::Tls12
PS C:\Users\Administrator> Invoke-WebRequest -Uri https://secure.eicar.org/eicar.com.txt -OutFile eicar.txt
PS C:\Users\Administrator> cp ./eicar.txt C:\Users\Administrator\Downloads
PS C:\Users\Administrator>
```

Thao tác này kích hoạt truy vấn VirusTotal và tạo cảnh báo. Ngoài ra, tập lệnh active response sẽ tự động xóa tệp.

Kết quả: Vào Wazuh Server để xem cảnh báo

Time	rule.description	timestamp per 30 minutes	rule.level	rule.id
> May 1, 2024 @ 23:12:10.563	VirusTotal: Alert - c:\users\thecotilking\downloads\eicar.txt - 64 engines detected this file		12	87105
> May 1, 2024 @ 23:09:06.173	active-response/bin/remove-threat.exe removed threat located at c:\users\thecotilking\downloads\eicar.txt		12	100092
> May 1, 2024 @ 23:07:53.995	File deleted.		7	553
> May 1, 2024 @ 23:07:39.655	VirusTotal: Alert - c:\users\thecotilking\downloads\eicar.txt - 64 engines detected this file		12	87105
> May 1, 2024 @ 23:05:21.518	File added to the system.		5	554

- Có 64 công cụ đã phát hiện ra file này là độc hại.
- File đã bị xóa ngay sau khi được tải xuống.

CHƯƠNG 5: KẾT LUẬN VÀ HƯỚNG PHÁT TRIỂN

5.1 Tóm tắt kết quả đạt được

Sau khi triển khai và thực hiện dự án "*Giải pháp SIEM && XDR sử dụng mã nguồn mở Wazuh*", các kết quả sau đây đã được đạt được:

1. *Cải thiện khả năng giám sát và phát hiện mối đe dọa*
 - Phát hiện và cảnh báo sớm các mối đe dọa an ninh, bao gồm tấn công brute-force, SQL injection, và các tệp tin độc hại.
 - Phát hiện và chặn IP độc hại truy cập đến Web Server.
2. *Tích hợp thành công với các hệ thống khác*: Wazuh được tích hợp với các hệ thống hiện có như tường lửa, hệ thống phát hiện xâm nhập (IDS/IPS), và các ứng dụng quản lý bảo mật khác như VirusTotal để cung cấp một bức tranh bảo mật toàn diện.

5.2 Những khó khăn và thách thức trong quá trình triển khai

Trong quá trình triển khai dự án, cũng đã gặp phải một số khó khăn và thách thức, cụ thể như sau:

1. *Khả năng mở rộng hệ thống*

- Wazuh cần được triển khai trên một môi trường hạ tầng đủ mạnh để có thể xử lý các tác vụ phức tạp như phân tích log thời gian thực, phát hiện các mối đe dọa và phản ứng kịp thời. Tuy nhiên, việc mở rộng hệ thống có thể gặp khó khăn về mặt tài nguyên hạ tầng, như CPU, RAM, dung lượng lưu trữ.

- Việc đảm bảo hiệu suất hệ thống khi mở rộng số lượng agent và thiết bị giám sát đòi hỏi phải có kế hoạch chi tiết và cài đặt đúng cách các thành phần như Wazuh Cluster, Elasticsearch.

2. Thiếu tài liệu và hỗ trợ kỹ thuật

- Mặc dù Wazuh là một nền tảng mã nguồn mở mạnh mẽ, nhưng đôi khi tài liệu hỗ trợ cho các vấn đề cụ thể chưa đầy đủ, đặc biệt là khi gặp các tình huống cần cầu hình nâng cao.

- Ngoài ra, việc tìm kiếm sự hỗ trợ từ cộng đồng hoặc nhóm phát triển chính thức cũng có thể mất thời gian do không có sự hỗ trợ thương mại trực tiếp (trừ khi sử dụng dịch vụ trả phí).

5.3 Hướng phát triển và cải thiện trong tương lai

Dự án "*Giải pháp SIEM && XDR sử dụng mã nguồn mở Wazuh*" là một giải pháp bảo mật tiềm năng, nhưng để tối ưu hóa và mở rộng trong tương lai, cần có những kế hoạch phát triển và cải thiện. Dưới đây là các hướng phát triển và cải thiện chính:

1. Tối ưu hóa hiệu suất hệ thống

- Cải thiện khả năng xử lý log: Khối lượng dữ liệu log mà hệ thống thu thập sẽ tăng lên theo thời gian và số lượng thiết bị. Do đó, cần nghiên cứu và áp dụng các biện pháp tối ưu hóa khả năng xử lý log, giảm thiểu độ trễ trong việc phân tích log và cảnh báo. Điều này có thể bao gồm việc sử dụng Elasticsearch Cluster, tối ưu hóa cấu hình của Wazuh Manager và tăng cường tài nguyên phần cứng.

- Tối ưu bộ nhớ và tài nguyên hệ thống: Đảm bảo hệ thống có thể mở rộng để quản lý số lượng lớn agent mà không làm giảm hiệu suất, thông qua việc áp dụng các phương pháp phân tán, load balancing hoặc tăng cường phần cứng.

2. Mở rộng khả năng tích hợp

- Tích hợp với nhiều nền tảng khác: Wazuh có thể được tích hợp với các hệ thống khác như SIEM doanh nghiệp hoặc các dịch vụ đám mây lớn như AWS, Azure, và Google Cloud Platform để giám sát toàn diện hơn. Trong tương lai, việc mở rộng tích hợp với nhiều hệ thống hơn nữa sẽ giúp cung cấp một giải pháp bảo mật mạnh mẽ hơn.

- Hỗ trợ thêm các công cụ bảo mật khác: Tích hợp với các giải pháp tường lửa nâng cao, IPS/IDS, và các công cụ quản lý lỗ hổng sẽ giúp cải thiện khả năng phát hiện và phản ứng nhanh chóng đối với các mối đe dọa tiềm ẩn.

3. Phát triển tính năng phân tích và dự đoán

- **Ứng dụng trí tuệ nhân tạo và machine learning:** Sử dụng AI và ML để phân tích và dự đoán các mối đe dọa bảo mật dựa trên các mẫu hành vi và dữ liệu log thu thập được. Điều này có thể giúp Wazuh không chỉ phản ứng với các mối đe dọa hiện tại mà còn dự đoán trước các mối đe dọa trong tương lai.

- **Cải thiện hệ thống cảnh báo thông minh:** Tinh chỉnh các rules và thuật toán cảnh báo để giảm bớt cảnh báo sai (false positives) và tăng khả năng phát hiện đúng các mối đe dọa thực sự.

4. Tăng cường giám sát và quản lý tập trung

Phát triển hệ thống giám sát tập trung: Cải thiện hệ thống giám sát tập trung cho nhiều máy chủ và thiết bị, giúp quản lý dễ dàng hơn trong các môi trường doanh nghiệp lớn hoặc hệ thống phức tạp. Điều này giúp tiết kiệm thời gian và nâng cao hiệu quả trong việc phát hiện các mối đe dọa tiềm tàng.

TÀI LIỆU THAM KHẢO

Hiểu về SIEM && XDR:

https://www.researchgate.net/publication/372503637_Battle_of_Defenses_Understanding_SIEM_vs_XDR_in_Modern_Cybersecurity

Cài đặt Wazuh: <https://documentation.wazuh.com/current/installation-guide/wazuh-agent/wazuh-agent-package-linux.html>

Hướng dẫn thực nghiệm: <https://documentation.wazuh.com/current/proof-of-concept-guide/index.html>