BAN CƠ YẾU CHÍNH PHỦ
HỌC VIỆN KỸ THUẬT MẬT MÃ

CHUYÊN ĐỀ KỸ NGHỆ AN TOÀN MẠNG

NGHIÊN CỨU VÀ XÂY DỰNG HỆ THỐNG GIÁM SÁT SỰ KIỆN VÀ CẢNH BÁO BẤT THƯỜNG CỦA DỮ LIỆU CHO DOANH NGHIỆP

Ngành: An toàn thông tin
Mã số:

Sinh viên thực hiện:
Nguyễn Việt Dũng – MSSV: AT170613

Người hướng dẫn:
ThS.Nguyễn Thị Thu Thủy
Khoa An toàn thông tin – Học viện Kỹ thuật mật mã

Hà Nội, 2025

BAN CƠ YẾU CHÍNH PHỦ
HỌC VIỆN KỸ THUẬT MẬT MÃ

CHUYÊN ĐỀ KỸ NGHỆ AN TOÀN MẠNG

NGHIÊN CỨU VÀ XÂY DỰNG HỆ THỐNG GIÁM SÁT SỰ KIỆN VÀ CẢNH BÁO BẤT THƯỜNG CỦA DỮ LIỆU CHO DOANH NGHIỆP

Ngành: An toàn thông tin
Mã số:

Sinh viên thực hiện:
Nguyễn Việt Dũng – MSSV: AT170613

Người hướng dẫn:
ThS.Nguyễn Thị Thu Thủy
Khoa An toàn thông tin – Học viện Kỹ thuật mật mã

Hà Nội, 2025

MỤC LỤC

DANH MỤC KÝ HIỆU VÀ VIẾT TẮT

SIEM	Security Information and Event Management – Hệ thống quản lý sự kiện và thông tin bảo mật
SIM	Security Information Management – Hệ thống quản lý thông tin bảo mật
SEM	Security Event Management – Hệ thống quản lý sự kiện
FIM	File Intergrity Monitor – Giải pháp giám sát tính toàn vẹn
CIS	Center of Internet Security – Tiêu chuẩn đánh giá bảo mật
VLAN	Virtual Local Area Network – Phân vùng mạng LAN ảo
SOC	Security Operation Center – Trung tâm vận hành an ninh mạng
API	Application Programing Interface – Giao diện lập trình ứng dụng

DANH MỤC HÌNH ẢNH

DANH MỤC BẢNG

LỜI CẢM ƠN

Để hoàn thành Chuyên đề Kỹ nghệ An toàn mạng với đề tài "Nghiên cứu và xây dựng hệ thống giám sát và cảnh báo bất thường của dữ liệu cho doanh nghiệp" em xin bày tỏ lòng biết ơn sâu sắc đến ThS.Nguyễn Thị Thu Thủy – Giảng viên Khoa An toàn thông tin, Học viện Kỹ thuật mật mã, người đã tận tâm hướng dẫn và hỗ trợ em trong suốt quá trình thực hiện Chuyên đề. Cô không chỉ cung cấp những định hướng quý giá mà còn dành thời gian trao đổi, giải đáp những thắc mắc, giúp em vượt qua những khó khăn và thách thức. Những lời khuyên thiết thực cùng sự tận tình của cô đã giúp em hiểu sâu hơn về vấn đề nghiên cứu, cũng như hoàn thiện tốt đề tài của mình.

Em cũng xin gửi lời cảm ơn chân thành đến các thầy cô, cán bộ Hệ quản lý sinh viên, những người đã không ngừng nỗ lực tạo ra môi trường học tập thuận lợi cho sinh viên. Nhờ sự tận tụy và hướng dẫn của các thầy cô, em đã có cơ hội tiếp cận nhiều kiến thức chuyên sâu, được truyền đạt những kinh nghiệm quý báu từ thực tiễn, giúp em tự tin hơn khi thực hiện Chuyên đề. Bên cạnh đó, sự hỗ trợ của các cán bộ trong công tác quản lý sinh viên cũng đã giúp đỡ em rất nhiều, từ việc cung cấp thông tin, tài liệu đến việc tổ chức các buổi hướng dẫn, chia sẻ kinh nghiệm. Chính nhờ sự phối hợp nhịp nhàng giữa các thầy cô và cán bộ quản lý mà em có thể hoàn thành tốt chuyên đề này, không chỉ về mặt lý thuyết mà còn trong ứng dụng thực tiễn.

Bên cạnh đó, em không thể không nhắc đến sự động viên và ủng hộ từ gia đình và bạn bè, những người luôn đứng sau hỗ trợ tinh thần cho em trong suốt quá trình học tập và nghiên cứu. Sự khích lệ và tin tưởng của họ là nguồn động lực to lớn giúp em vượt qua mọi thử thách để đi đến thành công.

Cuối cùng, xin cảm ơn tất cả mọi người đã tạo những điều kiện tốt nhất để em có thể hoàn thành Chuyên đề này.

SINH VIÊN THỰC HIỆN

Nguyễn Việt Dũng

LỜI NÓI ĐẦU

Trong thời đại của công nghệ số, an ninh mạng đã trở thành một yếu tố then chốt trong việc đảm bảo sự phát triển bền vững và thành công của các tổ chức và doanh nghiệp. Khi lượng dữ liệu và số lượng các mối đe dọa bảo mật ngày càng gia tăng, việc quản lý và giám sát hệ thống mạng một cách hiệu quả trở nên cấp thiết hơn bao giờ hết. Trong bối cảnh đó, các hệ thống giám sát và quản lý sự kiện tập trung, như Splunk và Wazuh, đã nổi lên như những giải pháp tiên tiến nhằm giải quyết bài toán này.

Splunk mang đến khả năng tập hợp, phân tích và hiển thị dữ liệu từ nhiều nguồn khác nhau, cung cấp một cái nhìn toàn diện và trực quan về tình trạng hoạt động của hệ thống. Trong khi đó, Wazuh với chức năng giám sát tính toàn vẹn tệp tin và phát hiện bất thường đã đóng vai trò quan trọng trong việc bổ sung khả năng bảo mật chi tiết cho các tổ chức. Sự kết hợp giữa Splunk và Wazuh không chỉ tăng cường khả năng phát hiện các mối đe dọa, mà còn giúp cải thiện tính chính xác và độ tin cậy của các cảnh báo.

Chuyên đề Kỹ nghệ An toàn mạng với chủ đề "Xây dựng hệ thống giám sát sự kiện và cảnh báo bất thường của dữ liệu cho doanh nghiệp" nhằm nghiên cứu và triển khai một hệ thống tích hợp giữa Splunk và Wazuh. Mục tiêu chính của đồ án là đánh giá hiệu quả của sự kết hợp này trong việc nâng cao năng lực quản lý và giám sát hệ thống mạng, đồng thời thử nghiệm các giải pháp nhằm tối ưu hóa hiệu suất hoạt động. Quá trình thực hiện đồ án bao gồm việc phân tích các thành phần của từng hệ thống, triển khai mô hình thực nghiệm và đánh giá hiệu quả thông qua các thử nghiệm thực tế.

Bằng cách tận dụng sức mạnh của Splunk và Wazuh, Chuyên đề không chỉ tập trung vào việc xây dựng một hệ thống giám sát hiệu quả mà còn nhấn mạnh vào tính ứng dụng thực tiễn trong môi trường doanh nghiệp. Hy vọng rằng kết quả từ chuyên đề sẽ mang lại những đóng góp giá trị cho lĩnh vực an ninh mạng, cũng như mở ra các hướng nghiên cứu mới trong việc tích hợp và tối ưu hóa các công cụ giám sát và quản lý sự kiện.

SINH VIÊN THỰC HIỆN

Nguyễn Việt Dũng

CHƯƠNG 1. TỔNG QUAN VỀ HỆ THỐNG GIÁM SÁT SỰ KIỆN VÀ TÌNH HÌNH AN TOÀN THÔNG TIN TRONG CÁC DOANH NGHIỆP HIỆN NAY

1.1. Tổng quan về tình hình An toàn thông tin trong doanh nghiệp hiện nay

1.1.1. Các mối nguy hại trong An toàn thông tin

Trong tình hình xã hội ngày càng phát triển như hiện nay, lưu lượng sử dụng mạng ngày càng tăng cao, chuyển đổi số diễn ra liên tục trong cuộc sống con người và đặc biệt là trong quá trình phát triển của các doanh nghiệp. Môi trường mạng là môi trường vô cùng phức tạp và tiềm ẩn nhiều nguy cơ. Nhìn từ quan điểm an toàn thông tin, có vô số cách để tấn công, lấy cắp thông tin của một hệ thống. lỗ hổng của ứng dụng, lỗ hổng của dịch vụ trực tuyến (Web, mail,…), lỗ hổng của hệ điều hành,… Vì thế, việc thiết lập và duy trì an toàn thông tin là vô cùng khó khăn.

1.1.1.1. Lừa đảo và lấy cắp thông tin

Trong một tổ chức, với hàng nghìn công nhân viên, sẽ có thể tồn tại một số nhân lực tham gia tổ chức với mục đích xấu, điển hình như lấy cắp những thông tin quan trọng của công ty. Chuyện này có thể xảy ra trong những môi trường mang tính bí mật cao mà chỉ có thể khai thác từ bên trong như những công ty thuộc nhà nước, cơ quan quân sự,… Việc lấy cắp thông tin có thể được thực hiện dưới nhiều hình thức: Lấy cắp văn bản, lấy cắp thông tin số, cung cấp thông tin nội bộ cho bên ngoài.

Cách tốt nhất để phòng tránh nguy cơ này là phải có những chính sách bảo mật được thiết kế tốt. Những chính sách này có thể giúp người quản lý bảo mật thông tin thu thập thông tin, từ đó điều tra và đưa ra các kết luận chính xác, nhanh chóng. Khi đã có một chính sách tốt, người quản trị có thể sử dụng các kỹ thuật điều tra số để truy vết các hành động tấn công.

1.1.1.2. Hacker (Tin tặc)

Theo thống kê, trên thế giới hàng giờ có hàng nghìn cuộc tấn công từ tin tặc vào các hệ thống lớn nhỏ trên thế giới. Mỗi kẻ tin tắc đều có những thủ thuật, công cụ, kiến thức về hệ thống. Và những thông tin chúng kiếm được có thể được rao bán trên các "chợ đen" gây ra rò rỉ và thất thoát thông tin cho tổ chức bị khai thác.

Quy trình tấn công của tin tặc có thể bắt đầu từ việc thu thập thông tin, hắn sẽ thu thập thông tin nhiều nhất có thể về hệ thống mục tiêu. Những thông tin đó có thể là danh sách công nhân viên, loại công nghệ mà sản phẩm của công ty đó sử dụng, hệ điều hành,…. Sau khi đã thu thập đủ thông tin ở mức bề mặt, kẻ tấn công sẽ tiến

hành quét hệ thống sâu hơn để tìm các lỗ hổng. Các lỗ hổng này có thể tồn tại do hệ thống chưa được cập nhật bản vá, hoặc sai sót trong quá trình xử lý thông tin,… Từ đó tin tặc sẽ sử lợi dụng các lỗ hổng để xâm nhập hệ thống và tiến xa hơn tới chiếm các quyền quan trọng.

1.1.1.3. Lây lan mã độc

Lây lan mã độc có thể được kể đến như là một phương án tấn công của tin tặc. Trên thế giới hiện nay tồn tại rất nhiều chủng loại mã độc như Virus, Worm, Trojan Horse, Ransomeware,… Các loại mã độc này được tạo nên với nhiều mục đích, đa số sẽ là tạo ra một backdoor để tin tặc có thể xâm nhập hoặc mã hóa dữ liệu và đòi tiền chuộc.

Một ví dụ điển hình nổi tiếng nhất cho mã độc có thể kể đến Ransomeware WannaCry được phát hiện vào năm 2017, mã độc này dựa trên lỗ hổng Enternal Blue trong giao thức SMB được gọi tắt với mã MS17-010. Khi mã độc này xâm nhập được vào máy nạn nhân, nó sẽ chiếm quyền quản trị, sau đó mã hóa hoàn toàn hệ thống và hiện cảnh báo đòi tiền chuộc, sau thời gian đếm ngược người dùng không thực hiện chuyển tiền, mã độc sẽ tiến hành xóa hoàn toàn dữ liệu trên máy.

Tấn công dưới dạng mã độc đến nay vẫn luôn là một mối nguy hại tiềm tàng, vì mã độc có thể lây lan với tốc độ vô cùng lớn trong môi trường mạng, chỉ cần một node trong hệ thống bị nhiễm thì khả năng mã độc lan ra toàn bộ mô hình rất cao.

1.1.2. Tình trạng bảo mật trong doanh nghiệp hiện nay

Trên các nguồn tin tức An toàn thông tin, ta luôn thấy các lỗ hổng mới được phát hiện và các tổ chức đang bị tổn hại vì những lỗ hổng này. Người dùng thường nghĩ rằng hầu hết các vi phạm an ninh sẽ chỉ xảy ra với những tập đoàn lớn, công ty lớn. Nhưng trong thực tế, kẻ tấn công không chỉ quan tâm đến các doanh nghiệp lớn, mà các doanh nghiệp vừa và nhỏ cũng bị nhắm vào. Một môi trường doanh nghiệp nhỏ, với mức đầu tư vào công nghệ hạn chế sẽ là mục tiêu tốt, dễ dàng cho kẻ tấn công.

Theo báo cáo mới nhất của Securonix – Công ty chuyên về các giải pháp bảo mật thông tin (Texas, Mỹ) trong 12 tháng qua, số lượng lỗ hổng bảo mật mới được phát hiện lớn gần gấp đôi so với năm ngoái, trong khi số lượng mối đe dọa an ninh mạng đã tăng 482% so với cùng kỳ năm trước.

Cụ thể, Securonix cho biết, có 867 mối đe dọa và 35.776 báo cáo về sự xâm phạm (IOC), tăng tương ứng 482% và 380% so với năm trước. Tổng cộng có 582 mối đe dọa đã được phát hiện, phân tích và báo cáo trong khoảng thời gian này.

Có thể thấy rằng trong khoảng thời gian gần đây, giới doanh nghiệp lao đao vì sự xuất hiện của các hình thức tấn công mới, tinh vi hơn, khó đoán hơn. Mặt khác, trong khi những kẻ tấn công ngày càng được trang bị tốt hơn, phát triển hơn về công cụ thì các nhà chức trách dường như đang bị tụt lại phía sau và không kịp phản ứng với các mối đe dọa. Sự thiếu hụt chuyên gia đang cản trở nỗ lực cải thiện an ninh mạng. Mặc dù số lượng chuyên gia về an ninh mạng thống kế trên toàn thế giới đã tăng 350% trong 8 năm qua, khoảng 3,5 triệu người. Song, nhiều công ty vẫn đang gặp khó khăn trong vấn đề bảo mật thông tin

1.1.3. Mô hình kinh doanh và bảo mật

Khi một tập đoàn đã xây dựng thành công môi trường vận hành của mình, điều tiếp theo cần được xây dựng chính là tính bảo mật. Với một xuất phát điểm tốt, các chủ quản có thể muốn theo dõi hoạt động trên mạng cục bộ của mình và các thiết bị đang hoạt động trên đó. Vì vậy việc giám sát có thể tạo ra một lượng lớn dữ liệu đáng kể dưới dạng sự kiện nhật ký từ các hệ thống, để theo dõi những dữ liệu này những tập đoàn hiện nay thường triển khai sử dụng giải pháp quản lý nhật ký tập trung SIEM.

Giải pháp quản lý sự kiện và an toàn thông tin (SIEM) có thể giúp cho việc bảo mật tốt hơn cho doanh nghiệp. Đây là một giải pháp mạnh mẽ và có thể thực hiện nhiều công việc giúp quản trị viên bảo mật có thể theo dõi mạng của mình tốt hơn, và có thể dễ dàng truy vết khi xảy ra sự cố. Bên cạnh việc sử dụng SIEM, quản trị viên mạng có thể tích hợp thêm một số giải pháp theo dõi để thu thập thêm thông tin và giúp cho công tác bảo mật được toàn diện hơn.

1.2. Giới thiệu về SIEM

1.2.1. Khái niệm và sự ra đời của SIEM

1.2.1.1. Khái niệm về SIEM

Hệ thống SIEM được viết tắt của cụm từ Security Information and Event Management là một lĩnh vực trong bảo mật máy tính nơi các sản phẩm và dịch vụ phần mềm kết hợp quản lý thông tin bảo mật (SIM) và quản lý sự kiện bảo mật (SEM). Các giải pháp SIEM có khả năng phát hiện các mối đe dọa và lỗ hổng bảo mật tiềm ẩn bằng cách phân tích dữ liệu nhật ký và sự kiện trong thời gian thực từ nhiều nguồn khác nhau bao gồm cả mạng, bảo mật, máy chủ, cơ sở dữ liệu và ứng dụng. SIEM cung cấp các cảnh báo và thông báo cho các nhóm bảo mật để điều tra thêm. Các công cụ SIEM đã đã phát hiện đáng kể các dấu hiệu được cho là đáng

nghi trong những năm qua và hiện được coi là một công cụ thiết yếu để phát hiện, điều tra, và ứng phó với các mối đe dọa an ninh mạng nâng cao.

SIEM hoạt động như một kho lưu trữ trung tâm do nhật ký hệ thống tạo ra, thông qua các quy tắc logic mà quản trị viên thiết lập, SIEM sẽ chọn ra các sự kiện để thu thập. Với SIEM, quản trị viên có thể xem được nhiều thông tin, dấu vết khi có một cuộc tấn công xảy ra.

Tóm lại, SIEM cung cấp cho các tổ chức khả năng quan sát, theo dõi hoạt động trong mạng của họ để họ có thể ứng phó nhanh chóng với các cuộc tấn công mạng và đáp ứng các yêu cầu tuân thủ an toàn thông tin.

1.2.1.2. Sự ra đời của SIEM

Hệ thống giám sát nhật ký đã trở nên phổ biến hơn khi các cuộc tấn công mạng phức tạp khiến cho các doanh nghiệp buộc phải đưa ra nhiều chính sách buộc tuân thủ các cơ chế quy định để kiểm soát bảo mật theo Khung quản lý rủi ro. Các cấp độ ghi nhật ký của một hệ thống bắt đầu với chcuws năng chính là khắc phục sự cố lỗi hệ thống hoặc mã gỡ lỗi được biên dịch và chạy. Khi các hệ điều hành và mạng ngày càng phức tạp, việc tạo sự kiện và nhật ký trên các hệ thống này cũng tăng theo. Trong khi đó, việc ghi nhật ký hệ thống, bảo mật và ứng dụng không phải là cách duy nhất để thực hiện ứng phó sự cố. Chúng cung cấp khả năng theo dõi các hoạt động của hầu hết mọi hệ thống hoặc chuyển động liên quan đến người dùng trong một khoảng thời gian nhất định. Từ cuối những nhăm 1970, các nhóm làm việc để giúp thiết lập các tiêu chí cho việc quản lý các chương trình kiểm tra và giám sát đã được thành lập, nhiệm vụ của những nhóm này cũng tương tự với cách thức ghi nhật ký hệ thống và ứng phó khi có sự cố. Chính những ý tưởng này đã tạo nên nền móng để phát triển giải pháp SIEM như hiện nay.

1.2.2. Các thành phần của SIEM

SIEM được so sánh như một cỗ máy phức tạp, trong đó mỗi bộ phận đảm nhiệm một nhiệm vụ cụ thể. Các bộ phận này cần phối hợp đồng bộ với nhau; nếu không, hệ thống sẽ không thể hoạt động hiệu quả. Một hệ thống SIEM cơ bản có thể được chia thành 6 thành phần chính, bao gồm: **Source Device, Log Collection, Parsing/Normalization of Logs, Rule Engine, Log Storage, và Event Monitoring and Retrieval**. Mỗi thành phần hoạt động độc lập nhưng có sự tương tác với các thành phần khác. Nếu bất kỳ thành phần nào ngừng hoạt động, toàn bộ hệ thống SIEM sẽ bị ảnh hưởng và không thể vận hành như bình thường.

Source Device

Phần đầu tiên của SIEM là thiết bị nguồn cung cấp thông tin vào SIEM. Thiết bị nguồn có thể là một thiết bị thực tế trên mạng của bạn, chẳng hạn như bộ định tuyến, bộ chuyển mạch hoặc một số loại máy chủ, nhưng nó cũng có thể là nhật ký từ một ứng dụng hoặc bất cứ dữ liệu nào khác mà bạn đang có. Thiết bị nguồn không được xem là một phần thực của SIEM, khi nhìn vào SIEM ta có thể xem nó đóng vai trò quan trọng trong SIEM. Việc hiểu rõ những nguồn muốn lấy các bản ghi nhật ký là rất quan trọng trong việc triển khai SIEM, nó giúp tiết kiệm công sức, số tiền đáng kể và giảm sự phức tạp trong triển khai.

Hệ điều hành: Microsoft Windows và các biến thể của Linux và UNIX, AIX, Mac OS là những hệ điều hành phổ biến. Hầu hết các hệ điều hành về cơ bản công nghệ không giống nhau và thực hiện chuyên một nhiệm vụ nào đó nhưng tất cả đều có điểm chung là chúng tạo ra các bản ghi Log. Các bản ghi Log sẽ cho thấy hệ thống đã làm gì: Ai là người đăng nhập, làm những gì trên hệ thống?... Các bản ghi Log được tạo ra bởi một hệ điều hành về hệ thống và người sử dụng hoạt động sẽ rất hữu ích khi tiến hành ứng phó sự cố an ninh hoặc chuẩn đoán vấn đề hay chỉ là việc cấu hình sai.

Thiết bị: Muốn tương tác giữa quyền administrator của hệ thống với các thiết bị hầu hết các quản trị hệ thống không có quyền truy cập từ xa vào hệ thống để thực hiện một số việc quản lý cơ bản. Nhưng họ có thể quản lý các thiết bị thông qua một cổng giao diện đặc biệt. Giao diện này có thể dựa trên web, dòng lệnh... hoặc chạy qua một ứng dụng được tải về máy trạm của quản trị viên. Hệ điều hành các thiết bị mạng, chẳng hạn như Microsoft Windows hoặc các phiên bản Linux, đều là hệ điều hành thông thường nhưng có thể được cấu hình theo cách mà hệ điều hành thông thường không làm. Một ví dụ minh họa là router hoặc switch. Những thiết bị này không phụ thuộc vào nhà cung cấp, nhưng chúng không cho phép truy cập trực tiếp vào hệ thống điều hành cơ bản mà phải thông qua giao diện quản lý, như dòng lệnh hoặc giao diện web. Các thiết bị này lưu trữ các bản ghi Log, thường được cấu hình để gửi bản ghi thông qua SysLog hoặc FTP.

Ứng dụng chạy trên các hệ điều hành là nguồn cung cấp thông tin từ các chức năng khác nhau, chẳng hạn như DNS, DHCP, máy chủ web, hệ thống thư điện tử, và nhiều ứng dụng khác. Những bản ghi này thường cung cấp thông tin về tình trạng ứng dụng, chẳng hạn như thống kê, sai sót, hoặc thông tin tin nhắn.

Xác định bản ghi Log cần thiết: Sau khi xác định các thiết bị nguồn trong hệ thống, quản trị viên có thể xem xét việc thu thập bản ghi Log từ các thiết bị nào là cần thiết và có giá trị trong hệ thống SIEM.

Các thông tin trên là cơ sở quan trọng để xác định nguồn thiết bị cần thiết cho SIEM. Mặc dù có rất nhiều nguồn khác nhau, cần chọn lọc các thiết bị có giá trị cao nhất với SIEM, nhằm tối ưu hóa tài nguyên và đảm bảo hiệu suất của hệ thống.

Log Collection

Bước tiếp theo trong ghi nhật ký dịch vụ và thiết bị là khi có tất cả các bản ghi khác nhau từ các thiết bị gốc của chúng gửi đến SIEM. Cơ chế thực tế của nhật ký được truy xuất khác nhau tùy thuộc vào SIEM mà quản trị viên đang sử dụng. Ở mức cơ bản nhất, quy trình thu thập có thể được chia thành 02 phương thức cơ bản:

- Push Log: Thiết bị nguồn sẽ gửi nhật ký tới SIEM
    + **Ưu điểm:** Để thiết lập và cấu hình tại SIEM. Thông thường, bạn chỉ cần thiết lập một bộ thu và sau đó đưa nó vào thiết bị hệ thống. Một số ví dụ phổ biến có thể điều này sẽ là syslog. Khi cấu hình nguồn thiết bị sử dụng nhật ký hệ thống, thiết lập địa chỉ IP hoặc tên DNS của máy chủ nhật ký trên hệ thống mạng và thiết bị này sẽ tự động bắt đầu gửi nhật ký của nó qua hệ thống tới máy thu nhật ký hệ thống. Trong ví dụ này, máy chủ nhật ký hệ thống sẽ là máy thu trên SIEM.
    + **Nhược điểm:** Ví dụ, sử dụng UDP syslog trong môi trường của bạn sẽ có một số lỗ hổng bảo mật mà bạn sẽ muốn xem xét khi thiết kế triển khai SIEM của bạn. Vì bản chất của UDP là không cung cấp sự tin cậy và thứ tự truyền nhận mà TCP làm, các gói dữ liệu có thể không đến được thứ tự hoặc bị mất mà không có thông báo, nghĩa là bạn không có thể đảm bảo rằng các gói đến đích của chúng. Nếu một tình trạng xảy ra trên mạng, chẳng hạn như một rủi ro lan truyền mạnh mẽ trên toàn mạng, bạn có thể không nhận được các gói nhật ký hệ thống đến SIEM của bạn. Một số vấn đề bảo mật khác có thể phát sinh là nếu bạn không đạt quyền truy cập thích hợp trên bộ thu SIEM, một hệ thống được cấu hình sai hoặc người dùng độc hại có thể tràn vào SIEM của bạn với thông tin sai lệch, khiến các sự kiện thực tế được tin tưởng. Nếu như đây là một cuộc tấn công có chủ ý chống lại SIEM của bạn, có thể sẽ làm sai lệch các gói tin và đưa dữ liệu vào các trường hợp giả mạo. Vì lý do này, việc hiểu thiết bị nào đang gửi nhật ký tới SIEM là điều cần thiết.
- **Pull Log:** Không giống như phương pháp Push Log, trong đó thiết bị nguồn gửi nhật ký tới SIEM mà không có bất kỳ tương tác nào từ SIEM. Phương pháp này yêu cầu SIEM bắt đầu kết nối với thiết bị nguồn, yêu cầu truy xuất nhật ký thiết bị nguồn. Một số ví dụ điển hình, như nhật ký được

truy xuất từ máy chủ hoặc từ cơ sở dữ liệu. SIEM thiết lập kết nối đến mỗi nguồn thông qua hệ thống đăng nhập được lưu trữ và có thể truy vấn nhật ký. Một điều cần lưu ý khi sử dụng Pull Log đó là nhật ký có thể không được đưa vào SIEM theo thời gian thực, vì khi thực hiện Pull Log SIEM sẽ phải chủ động liên hệ tới thiết bị, điều này sẽ phù hợp với trường hợp quản trị viên hệ thống cần truy xuất định kỳ hoặc thủ công tới nhật ký của hệ thống.

Parsing/Normalization of Logs

Vô số các bản ghi log được gửi từ các thiết bị và ứng dụng trong hệ thống đến SIEM. Trong trường hợp ban đầu, tất cả log đi tới sẽ đều dưới dạng định dạng gốc ban đầu (Raw Log), do đó để truy xuất được các thông tin cần thiết thành các mục riêng biệt sẽ là không thể đối với người quản trị.

Để giải quyết được vấn đề trên, SIEM cần định dạng lại nhật ký sang một định dạng chuẩn duy nhất. Việc thay đổi tất cả các loại bản ghi log khác nhau thành một định dạng chung duy nhất được gọi là chuẩn hóa. Nếu các thiết bị không hỗ trợ các giao thức chuẩn hóa log thì sẽ cần dùng đến các phần mềm được gọi là Agent. Đây là việc cần được thực hiện để lấy các bản ghi có định dạng mà SIEM có thể hiểu được. Việc cài đặt các Agent có thể kéo dài quá trình triển khai SIEM nhưng người quản trị sẽ có được các bản ghi log theo định dạng mong muốn

Rule Engine/Correlation Engine

Sau thành phần Parsing/Normalization of Logs, sẽ đến phần kỹ thuật tương quan sự kiện, trong thành phần này được chia ra làm hai khái niệm:

- The Rule Engine: Tính năng cho phép mở rộng chuẩn hóa các sự kiện từ các nguồn khác nhau để kích hoạt cảnh báo trong SIEM dựa trên các điều kiện cụ thể được trích ra từ nhật ký. Các phương pháp viết các quy tắc SIEM thường khá đơn giản, nhưng cũng có thể trở nên phức tạp tùy vào loại SIEM cũng như yêu cầu của tổ chức. Có thể viết quy tắc bằng cách sử dụng logic Boolean (đúng/sai) để xác định xem điều kiện cụ thể có được đáp ứng hay không và kiểm tra sự khớp mẫu trong các trường dữ liệu.
- Correlation Engine: Là một tập hợp con của The Rule Engine. Thành phần này giúp hợp nhất nhiều sự kiện tiêu chuẩn từ các nguồn khác nhau thành một sự kiện liên quan duy nhất. Sự tương quan của các sự kiện tiêu chuẩn thành một sự kiện được thể hiện để đơn giản hóa quy trình ứng phó sự cố

cho hệ thống, bằng cách hiển thị một tín hiệu sự kiện được kích hoạt bởi nhiều sự kiện đến từ nhiều nguồn thiết bị khác nhau

Log Storage

Để làm việc với khối lượng nhật ký đi vào SIEM, người quản trị viên cần một nơi để lưu trữ dữ liệu để tuân thủ các tiêu chuẩn như PCI DSS hay để truy vấn trong tương lai. Thông thường sẽ có ba cách SIEM có thể lưu trữ nhật ký của nó: Trong Database, Flat text file, Binary File

- **Database:** Đối với lưu trữ nhật ký trong Database là cách hầu hết SIEM lưu trữ nhật ký. Cơ sở dữ liệu thường là một nền tảng như Oracle, MySQL,… Phương pháp này cho phép người quản trị tương tác và truy xuất log dễ dàng vì việc gọi cơ sở dữ liệu sẽ la một phần của Hệ quản trị cơ sở dữ liệu được tích hợp sẵn với giải pháp SIEM.
- **Flat text file:** Lưu trữ nhật ký bằng flat text file chỉ là lưu trữ dưới một dạng chuẩn tệp văn bản lưu trữ thông tin ở định dạng dễ đọc, thường là file có đuôi ".log". Tệp là tệp được phân tách, có một vài ký tự phục vụ cho việc truy xuất thông tin của SIEM. Lưu trữ bằng phương pháp này không được sử dụng thường xuyên vì nó không được thiết kế để mở rộng quy mô cho các hệ thống lớn. Vấn đề khác hệ thống dễ gặp phải khi sử dụng phương pháp này là hiệu suất vì việc đọc từ tệp sẽ chậm hơn so với các phương thức khác.
- **Binary file:** Định dạng tệp nhị phân là tệp sử dụng định dạng tùy chỉnh để lưu trữ thông tin nhị phân, chỉ được SIEM sử dụng để lưu trữ thông tin. SIEM là ứng dụng duy nhất biết cách đọc và ghi vào tệp có tính độc quyền cao này

Monitoring

Phần cuối cùng trong SIEM là phương pháp tương tác với nhật ký được lưu trữ trong SIEM. Sau khi có được tất cả dữ liệu về log thu thập được và qua xử lý, người quản trị cần một cách để có thể tương tác với thông tin. Thay vì các logs chỉ nằm trong SIEM cho mục đích lưu trữ. SIEM sẽ cung cấp cho người dùng giao diện quản trị, có thể là ứng dụng web hoặc phần mềm. Giao diện sẽ cho phép người sử dụng tương tác với dữ liệu được lưu trữ trong SIEM.

1.2.3. Cách hoạt động của SIEM

Giải pháp SIEM hoạt động bằng cách thu thập dữ liệu từ nhiều nguồn khác nhau như máy tính, thiết bị mạng, máy chủ,… Dữ liệu đó sẽ được chuẩn hóa và tổng

hợp. Tiếp theo, các chuyên gia bảo mật sẽ phân tích và khám phá dữ liệu, từ đó phân tích các mối đe dọa. Sau khi đã hoàn thành phân tích mối đe dọa và đưa ra được các dấu hiệu nhận biết, các chuyên gia sẽ có thể đưa dữ liệu dấu hiệu đó vào Rule Engine để tạo thành các cảnh báo nhằm giúp SIEM tự động phát hiện mối đe dọa và cảnh báo cho tổ chức trong tương lai

1.2.4. Lợi ích và hạn chế của SIEM

1.2.4.1. Lợi ích của SIEM

Cải thiện hiệu suất mạng

SIEM có thể cung cấp cho tổ chức một trong những tài nguyên quan trọng nhất mà tổ chức cần khi đề cập đến các cuộc tấn công mạng đó là thời gian. Việc triển khai SIEM đúng cách sẽ rút ngắn thời gian phát hiện và xác định mối đe dọa, cho phép các bên bảo mật của tổ chức có thể ứng phó sự cố nhanh hơn. Từ đó, tổ chức có thể giảm thiểu thiệt hại hoặc ngăn chặn hoàn toàn.

SIEM cũng có thể giúp tổ chức nắm bắt các mối đe dọa Zero-day. SIEM có thể được cấu hình để phát hiện các hoạt động liên quan đến một cuộc tấn công. Điều đó giúp xác định sớm các mối hiểm họa có thể được coi như Zero-day chưa được xác định, và nó có thể vượt qua sự kiểm duyệt của tường lửa, phần mềm chống virus.

Cung cấp phân tích chi tiết

SIEM cung cấp những thông tin chi tiết về vị trí và cách thức vi phạm xảy ra, giúp đội ngũ bảo mật của tổ chức kịp thời ứng cứu, tạo ra các bản sửa lỗi hoặc bản vá. Ngoài ra, SIEM cũng thu thập và cung cấp các bằng chứng để sử dụng vào những hoạt động pháp lý

Quản lý tập trung thời gian thực

Sử dụng SIEM sẽ giúp tổ chức nói chung và đội ngũ bảo mật nói riêng tiết kiệm thời gian truy xuất nhật ký trên từng thiết bị. Thay vì phải tới từng thiết bị và sử dụng một số chức năng như Event Viewer để đọc nhật ký. Với SIEM, người quản trị có thể đọc ngay trên giao diện ứng dụng/web của SIEM, bên cạnh đó nhưng nhật ký từ các nguồn khác nhau có thể được xem cùng lúc, tạo ra sự so sánh tương quan chính xác nhất. Ngoài ra, tính năng theo dõi, thu thập log thời gian thực cũng là điểm mạnh của SIEM, tính năng này giúp việc theo dõi và bảo vệ hệ thông được diễn ra liên tục.

1.2.4.2. Hạn chế của SIEM

Mất thời gian để triển khai

Tùy thuộc vào quy mô của hệ thống mạng, SIEM có thể mất nhiều thời gian để triển khai. Công việc triển khai sẽ bắt đầu từ việc xây dựng máy chủ SIEM, cài đặt Agent lên các máy trạm, vì thế quy mô càng lớn thời gian triển khai càng dài. Sau quá trình triển khai sẽ là quá trình cấu hình, theo dõi và lọc những thông tin không cần thiết. Công việc theo dõi và lọc thông tin sẽ cần diễn ra trong suốt quá trình vận hành SIEM ở tương lai.

Yêu cầu chuyên môn kỹ thuật

Hiệu quả mang lại của SIEM hoàn toàn dựa trên cách nó được thiết lập, cấu hình và giám sát. Mặc dù việc chuyển tất cả dữ liệu hoạt động trong mạng là cần thiết, tuy nhiên nếu lấy hết tất cả thông tin, kể cả thông tin nghiệp vụ hệ thống sẽ dẫn tới tình trạng quá tải và thừa dữ liệu. Trên thực tế nếu không có chuyên môn kỹ thuật để thiết lập SIEM, điều đó sẽ khiến hệ thống gặp nhiều vấn đề. Vì vậy việc thiết lập SIEM đòi hỏi kinh nghiệm và hiểu biết về những loại dữ liệu cần thu thập và cách giảm tải cho hệ thống.

Giá thành cao

Hiện nay trên thế giới có rất nhiều loại SIEM thuộc các hãng khác nhau, nhưng đa số đều là bản trả phí: Qradar, Splunk,… Trong hệ thống lớn việc triển khai những SIEM mà không sử dụng giấy phép (license) sẽ khiến cho SIEM không đủ chức năng theo dõi, gây nên những rủi ro bảo mật. Tuy nhiên, trên thị trường, các giấy phép của SIEM không hề rẻ, thậm chí có thể lên tới 5000$/năm, điều này cũng một phần gây nên áp lực về tài chính cho các doanh nghiệp, đặc biệt là các doanh nghiệp vừa và nhỏ.

1.3. Kết luận chương I

Trong chương 1, báo cáo đã cung cấp một cái nhìn tổng quan về thực trạng an toàn thông tin trong doanh nghiệp hiện nay, phân tích các mối nguy hại và thách thức mà tổ chức phải đối mặt trong bối cảnh môi trường mạng ngày càng phức tạp. Các mối đe dọa như lừa đảo, tấn công bởi tin tặc và sự lây lan mã độc đều cho thấy mức độ nguy hiểm của các rủi ro an ninh mạng đối với hệ thống doanh nghiệp.

Đồng thời, chương này cũng nêu bật tình trạng bảo mật trong các tổ chức, từ các doanh nghiệp lớn đến các doanh nghiệp vừa và nhỏ, đều phải đối mặt với các thách thức khác nhau, bao gồm sự thiếu hụt chuyên gia và sự chậm trễ trong việc ứng phó với các lỗ hổng mới. Ngoài ra, việc kết hợp mô hình kinh doanh với bảo mật được khẳng định là yếu tố quan trọng để nâng cao khả năng phòng thủ trước các mối đe dọa ngày càng tinh vi.

Bên cạnh đó, phần giới thiệu về hệ thống SIEM (Security Information and Event Management) đã cho thấy tầm quan trọng của một giải pháp tập trung trong việc quản lý và giám sát an ninh thông tin. Hệ thống SIEM không chỉ giúp phát hiện sớm các dấu hiệu tấn công mà còn hỗ trợ doanh nghiệp trong việc đáp ứng các yêu cầu tuân thủ về an toàn thông tin.

Kết luận lại, việc xây dựng một môi trường mạng an toàn đòi hỏi các doanh nghiệp không chỉ nâng cao nhận thức về các rủi ro, mà còn cần áp dụng những công cụ và chiến lược hiệu quả như SIEM để bảo vệ toàn diện hệ thống của mình. Những nội dung này sẽ làm nền tảng để chuyển sang các chương tiếp theo, nơi các giải pháp và ứng dụng cụ thể của hệ thống giám sát an ninh sẽ được trình bày chi tiết.

CHƯƠNG 2. NGHIÊN CỨU HỆ THỐNG GIÁM SÁT VÀ CẢNH BÁO BẤT THƯỜNG CỦA DỮ LIỆU CHO DOANH NGHIỆP

Chương 2 trình bày về việc khảo sát kiến trúc mạng tại một doanh nghiệp cụ thể, từ đó làm căn cứ để đưa ra mô hình mạng mô phỏng. Tình trạng hiện tại mô hình mạng như nào, số lượng thiết bị. Từ đó đưa ra cái nhìn khái quát nhất về mô hình mạng.

2.1. Cơ sở của phương pháp

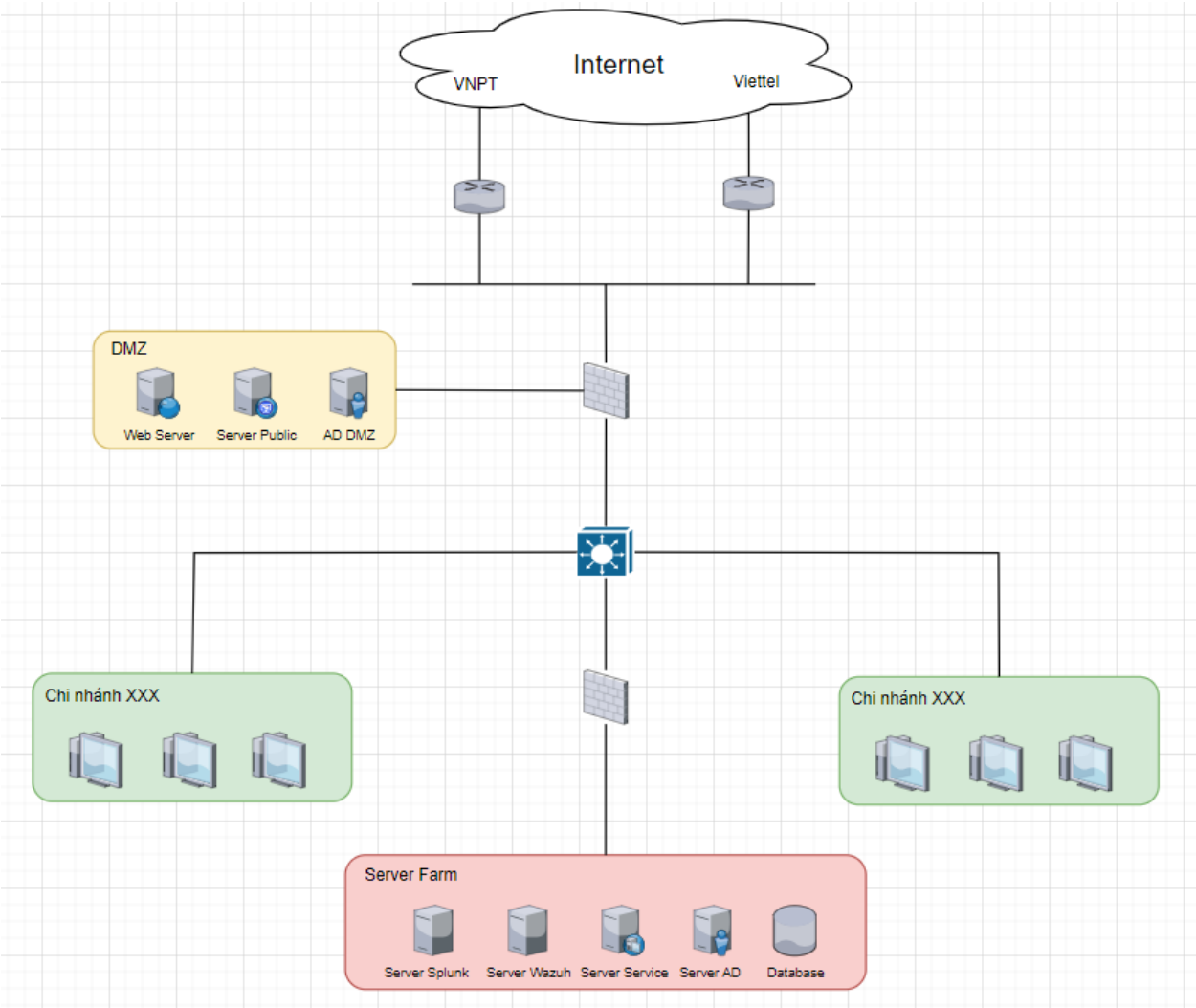
Trong bối cảnh công nghệ ngày càng phát triển, các doanh nghiệp phải đối mặt với nhiều mối đe dọa an ninh mạng phức tạp và tinh vi. Việc xây dựng một hệ thống giám sát sự kiện và cảnh báo bất thường đóng vai trò quan trọng trong việc đảm bảo tính toàn vẹn, bảo mật và ổn định của hệ thống công nghệ thông tin. Sự kết hợp giữa Splunk và Wazuh mang lại một giải pháp toàn diện, đáp ứng được các yêu cầu thực tiễn về bảo mật thông tin nhờ các khả năng vượt trội trong thu thập, phân tích và giám sát dữ liệu.

Splunk là một nền tảng mạnh mẽ trong việc thu thập, phân tích và hiển thị dữ liệu sự kiện từ nhiều nguồn khác nhau. Khả năng xử lý dữ liệu lớn theo thời gian thực giúp Splunk trở thành công cụ hữu hiệu trong việc phát hiện các bất thường, hỗ trợ điều tra và ứng phó với các mối đe dọa. Trong khi đó, Wazuh là một giải pháp nguồn mở cung cấp các chức năng giám sát an toàn thông tin, nổi bật với khả năng giám sát tính toàn vẹn tệp tin (FIM), phát hiện xâm nhập (IDS), và tuân thủ chính sách an ninh.

Phương pháp kết hợp Splunk và Wazuh tận dụng được ưu điểm của cả hai công cụ, giúp doanh nghiệp không chỉ quản lý tập trung các sự kiện an ninh mà còn giám sát chi tiết ở cấp độ hệ thống và dữ liệu. Splunk đảm nhiệm vai trò thu thập và phân tích dữ liệu từ các nguồn log, trong khi Wazuh cung cấp khả năng giám sát và cảnh báo chi tiết về các thay đổi không mong muốn trong hệ thống. Nhờ sự tích hợp này, doanh nghiệp có thể xây dựng một hệ thống giám sát mạnh mẽ, đáp ứng được yêu cầu thực tiễn trong việc bảo vệ tài sản số, giảm thiểu rủi ro và tăng cường khả năng ứng phó với các mối đe dọa.

Phương pháp này không chỉ mang lại hiệu quả cao trong giám sát và phát hiện bất thường mà còn đảm bảo tính linh hoạt, dễ mở rộng và tiết kiệm chi phí nhờ vào việc tận dụng các giải pháp hiện đại, phù hợp với mọi quy mô doanh nghiệp.

Điển hình, qua quá trình khảo sát tại một Công ty X, mô hình dưới đây là mô tả về hệ thống giám sát ATTT của tập đoàn này theo mô hình phân tán.



Hình 2.1. Mô hình hệ thống giám sát tại Công ty X

Như đã thấy trên hình 2.1, một hệ thống tại Công ty X có nhiều phân vùng và được phân bổ khá cụ thể, theo khảo sát thì hiện tại đội ngũ SOC thực hiện nhiệm vụ giám sát thông qua SIEM Splunk, kết hợp Wazuh để giám sát tính toàn vẹn tệp tin.

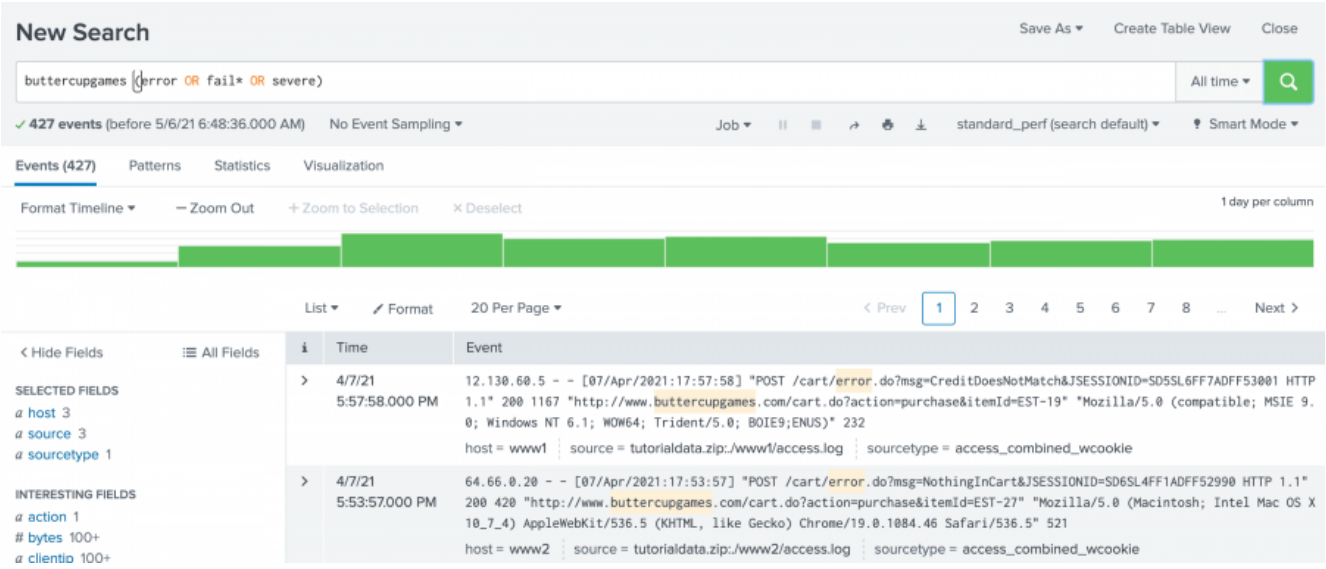
2.2. Hệ thống giám sát và quản lý sự kiện tập trung Splunk

2.2.1. Tổng quan về Splunk

Splunk là phần mềm cho phép tìm kiếm và duyệt logs cùng các dữ liệu trong thời gian thực (real-time). Người dùng có thể ngay lập tức phát hiện ra sự cố ở bất kì ứng dụng nào hoặc ở các máy chủ hay thiết bị, cảnh báo các nguy cơ tiềm ẩn và báo cáo các hoạt động của các dịch vụ và thành phần khác nhau trong mạng

Splunk là một công cụ dữ liệu rất linh hoạt và khả năng mở rộng cho các dữ liệu máy tính được tạo ra bởi cơ sở hạ tầng Công Nghệ Thông Tin. Nó thu thập, lập chỉ mục và khai thác những dữ liệu được tạo ra từ bất kì nguồn nào, định dạng hoặc

vị trí bao gồm cả đóng gói và các ứng dụng tùy chỉnh, máy chủ ứng dụng, máy chủ web, cơ sở dữ liệu, mạng, máy ảo, hypervisors, hệ điều hành và nhiều hơn nữa mà không cần phải phân tích cú pháp tùy chỉnh, bộ điều hợp hoặc một cơ sở dữ liệu trên các phụ trợ.



Hình 2.2. Hình ảnh minh họa về Splunk

Splunk có một số đặc điểm nổi bật như sau:

- Giải quyết tốt vấn đề hạ tầng mạng: Splunk cho phép ta tìm kiếm, cảnh báo và báo cáo trên mạng trong thời gian thực hiện các sự kiện mạng và các giao dịch hoàn chỉnh. Bắt đầu từ hiện tượng bất thường rồi tìm ra các nguyên nhân một cách nhanh chóng với các thông tin từ syslogs, SNMP traps, các cấu hình và dữ liệu netflow. Việc tích hợp Splunk với hệ thống giám sát cho phép phát hiện sớm các vấn đề và đi sâu vào tìm hiểu cặn kẽ nguyên nhân của vấn đề đó
- Giải quyết tốt vấn đề ảo hóa: Ảo hóa mang đến nhiều lợi ích không thể phủ nhận, nhưng nó cũng khiến cho hệ thống trở nên phức tạp hơn. Các vấn đề liên quan đến tài nguyên trên các máy chủ vật lí, máy ảo, năng suất hệ thống trở nên quá phức tạp khiến cho các hệ thống cũ không thể theo kịp. Từ đó, Splunk xuất hiện cung cấp một cái nhìn rõ ràng xuyên suốt các hệ thống ảo hóa. Nó thu thập tất cả các dữ liệu từ các hệ thống ảo hóa, hệ thống vật lý, các phiên giao dịch rồi tổng hợp, liên kết chúng lại với nhau để phục vụ công tác phân tích, tìm kiếm, tối ưu hệ thống
- Giám sát hệ thống Cloud (Đám mây): Sự phát triển của các cơ sở hạ tầng lai đã tạo ra một thách thức với các bộ phận CNTT phải làm thế nào để các đối tượng không nằm trong quyền kiểm soát trực tiếp của họ. Với khả năng

phân tích, lập chỉ mục cho mọi loại dữ liệu, Splunk cung cấp một cái nhìn chi tiết, chính xác về môi trường đám mây của bất kì nhà cung cấp nào

- Giám sát hệ thống email: Bản chất hệ thống tin nhắn là cực kì phực tạp. Từ truy tìm các tin nhắn email để quản lí tuân thủ và phân tích thư rác và tấn công lừa đảo Phising, cơ sở hạ tầng khá phức tạp và chuyên sâu. Splunk giúp ta tìm kiếm các giao tiếp tin nhắn trong thời gian thực trên cơ sở hạ tầng của bản thân
- Giám sát ứng dụng: Hệ thống phân bố ứng dụng phức tạp có thể ẩn chứa rất nhiều lỗi. Nhưng để tìm và sửa các lỗi này thì lại là một vấn đề không hề đơn giản, tiêu tốn nhiều thời gian cũng như tiền bạc. Những nhóm phát triển cũng như người quản trị không thể truy cập vào các dữ liệu mà họ cần để làm việc. Splunk cho phép điều tra phân tích các vấn đề một cách nhanh chóng từ khu vực quản lí trung tâm. Splunk cũng cho phép nhóm phát triển truy cập các dữ liệu họ cần để nhanh chóng giải quyết và khắc phục vấn đề

Lợi thế của Splunk so với các giải pháp SIEM khác

Linh hoạt mềm dẻo khi sử dụng: Linh hoạt, khả năng mở rộng và đủ linh hoạt từ bất kỳ nguồn dữ liệu, các ứng dụng tùy chỉnh và cơ sở dữ liệu. Splunk tự động cung cấp một cái nhìn chi tiết theo dòng thời gian của tất cả các dữ liệu thu thập được.

Điều tra theo thời gian thực:

- Splunk cho phép bạn xem thông tin thời gian thực từ an ninh và thiết bị mạng, hệ điều hành, cơ sở dữ liệu và các ứng dụng, trên một thời gian cho phép các đội an ninh để nhanh chóng phát hiện và hiểu được ý nghĩa end-to -end của một sự kiện an ninh.
- Splunk sẽ giải quyết được khó khăn của các hệ thống bảo mật hiện tại khi tìm kiếm và phát hiện các hành vi nguy hiểm đang hoạt động trong hệ thống. Với khả năng phát hiện từng hành vi bất hợp phát nhỏ nhất, Splunk sẽ giúp phát hiện những cuộc tấn công tinh vi nhằm vào hệ thống một cách nhanh chóng và hiệu quả nhất.
- Liên kết thông tin theo thời gian thực và cảnh báo: Tương quan của thông tin từ bộ dữ liệu khác nhau có thể cung cấp cái nhìn sâu sắc thêm và bối cảnh. Splunk có thể liên kết với tất cả các thông tin dữ liệu từ mọi nguồn trên hệ thống một cách nhanh chóng và chính xác theo thời gian thực.

- Splunk là phần mềm mã nguồn mở, có bản không tính phí nên không tốn kém khi triển khai.
- Giải quyết được hầu hết các bài toán trong giám sát hệ thống mạng: giám sát hạ tầng, giám sát dịch vụ, giám sát an ninh, giám sát người dùng... Đây là đặc điểm chính giúp cho Splunk trong tương lai sẽ được các tổ chức, doanh nghiệp sử dụng để triển khai hệ thống giám sát tập trung bên trong hệ thống mạng của họ

2.2.2. Tính năng của giải pháp Splunk

2.2.2.1. Quản lý ứng dụng của Splunk

Khắc phục sự cố nhanh hơn: Splunk giúp giảm sự phức tạp bằng cách cung cấp cho các nhà phát triển được truy cập vào log của ứng dụng thông qua một vị trí trung tâm mà không cần quyền truy cập vào hệ thống đó, khắc phcuj sự cố vấn đề một cách nhanh chóng, giảm chi phí và giảm thời gian điều tra và khắc phục sự cố tới 70%. Đồng thời, giám sát toàn bộ môi trường ứng dụng trong thời gian thực để ngăn chặn các vấn đề ảnh hưởng tới người dùng, giữ lại log từ các sự kiện định kì để ngăn ngừa mất mát

Nắm được hoạt động của toàn bộ ứng dụng:

- Cho phép truy vết và giám sát các giao tiếp của ứng dụng thông qua các tầng của kiến trúc phân tán và từ nhiều nguồn dữ liệu
- Phát hiện các bất thường hoặc vấn đề trong hoạt động, thời gian đáp ứng và chủ động giải quyết chúng trước khi nó ảnh hưởng tới người dùng, ứng dụng
- Theo dõi số liệu hoạt động quan trọng như thời gian đáp ứng end-to-end, độ dài thông điệp và hàng đợi và đếm số lần giao tiếp thất bại để đảm bảo ứng dụng đáp ứng được nhu cầu cần thiết
- Nắm được toàn bộ hoạt động của ứng dụng trong thời gian thực trên toàn bộ cơ sở hạ tầng ứng dụng
- Đạt được cái nhìn toàn diện về cách mà người dùng sử dụng dịch vụ, từ đó có thể cung cấp dịch vụ tốt hơn
- Làm phong phú hệ thống bằng cách thêm các nguồn CNTT như thông tin khách hàng, thông tin vị trí,...
- Không giống các công cụ quản lý truyền thống, splunk có thể index, phân tích, khai thác dữ liệu từ bất kì tầng ứng dụng nào. Nó cung cấp một góc nhìn toàn diện về toàn bộ hệ thống cơ sở hạ tầng

- Ngôn ngữ tìm kiếm trong Splunk giúp người dùng so sánh các sự kiện, các chỉ số hoạt động quan trọng khác
- Quyền điều khiển được trao cho nhiều nhóm trong một tổ chức. Những hiểu biết về dữ liệu ứng dụng có thể kết hợp với thông tin có cấu trúc như thông tin user hoặc giá cả thông tin để doanh nghiệp quyết định tốt hơn

2.2.2.2. Quản lý các hoạt động Công nghệ thông tin

Splunk cung cấp một cách tiếp cận tốt hơn, nó thu thập và lập indexes chứa tất cả các dữ liệu được tạo ra bởi hệ thống CNTT (Hệ thống mạng, server, OS, ảo hóa,...). Splunk hoạt dộng với bất kì dữ liệu mà máy tạo ra, bao gồm log, file cấu hình, số liệu hiệu suất, SNMP trap và các ứng dụng log tùy chỉnh

Giúp nắm bắt được hoạt động ảo hóa, hệ thống cloud private và public từ một giao diện trung tâm, dễ dàng tìm được nguồn gốc của vấn đề nhanh hơn 70% mà không cần phải tìm kiếm trong hệ thống, server hay máy ảo. Quản lý hệ thống trong thời gian thực, ngăn ngừa vấn đề xảy ra trước khi nó ảnh hưởng tới người dùng và có thêm dữ liệu để xử lí các sự kiện xảy ra định kì để tránh mất mát

Tương quan các sự kiện ở tất cả các tầng layer của hệ thống:

Tìm các liên kết giữa người sử dụng, hiệu suất các sự kiện liên quan tới cơ sở hạ tầng được cung cấp bởi Splunk kết hợp phân tích dữ liệu trong thời gian thực tương quan, so sánh với hàng triệu Terabytes

Quản lý môi trường để nhận biết được sự thay đổi, so sánh ngay lập tức để biết độ thiếu hụt hiệu năng của hệ thống, những vấn đề có sẵn hoặc vấn đề bảo mật an ninh

Giảm chi phí cung cấp các dịch vụ CNTT: Sử dụng sức mạnh và khả năng mở rộng của Splunk không chỉ cho hoạt động quản lý CNTT mà còn dùng để hỗ trợ kiểm tra, an ninh. Giảm số lượng các công cụ và kĩ năng cần để duy trì quản lý cơ sở hạ tầng phức tạp

Phân tích hoạt động:

- Splunk phân tích hoạt động toàn diện theo nhiều tầng giúp cho định hướng của doanh nghiệp tốt hơn tùy theo từng trường hợp cụ thể
- Chủ động trong việc nhận diện và khắc phục lỗi dịch vụ để đảm bảo sự hài lòng của khách hàng và giúp tăng số lượng khách hàng sử dụng
- Nắm bắt được những nguy hiểm tiềm tàng trong quá trình hoạt động kinh doanh, giúp đạt được các mục tiêu kinh doanh bằng cách cung cấp tầm nhìn toàn diện trên toàn hệ thống công nghệ không đồng nhất, các dịch vụ,

cách quản lý, lên kế hoạch về dung lượng, phân tích mức sử dụng của người dùng và nhiều hơn nữa.

Giám sát cơ sở hạ tầng:

- Máy chủ: Splunk cho phép chúng ta có thể chủ động giám sát các máy chủ và hiểu biết sâu hơn về hiệu suất, cấu hình, truy cập và các lỗi phát sinh. Tương quan hiệu suất máy chủ, các lỗi và dữ liệu sự kiện với người dùng, ảo hóa và ứng dụng thành phần để ngăn ngừa và khắc phục lỗi. Phân tích và tối ưu hóa chi phí cho việc theo dõi dung lượng máy chủ, báo cáo an ninh trong thời gian thực.
- Hệ thống lưu trữ: Splunk có thể cho ta tương quan log, số liệu hiệu suất và các sự kiện từ hệ thống lưu trữ với máy chủ, mạng và dữ liệu từ các ứng dụng để giải quyết các vấn đề và làm tăng sự hài lòng của khách hàng. Sử dụng công cụ phân tích mạnh mẽ để khắc phục sự cố trong thời gian thực và phân tích hiệu suất hệ thống lưu trữ. Giảm thời gian phát triển và cắt giảm chi phí bằng việc dễ dàng tích hợp với các nhà cung cấp dịch vụ lưu trữ, như NetApp và EMC
- Hệ thống mạng: Splunk cho phép ta có thể giám sát và theo dõi dữ liệu mạng từ các thiết bị không dây, switch, router, firewall và trên những thiết bị khác bằng cách sử dụng SNMP, Netflow, syslog, PCAP,…
- Chủ động nhận diện các vấn đề an ninh mạng và thực hiện phân tích vấn đề. Tương quan dữ liệu mạng với các ứng dụng, hệ thống lưu trữ và phân tích máy chủ để giữ cho mạng an toàn và hoạt động mọi lúc.

Splunk cho hệ điều hành:

- Tương quan số liệu hệ thống và dữ liệu sự kiện với các dữ liệu ở các tầng công nghệ khác một cách dễ dàng. Tìm liên kết giữa vấn đề hiệu suất ứng dụng và hệ điều hành, ảo hóa, hệ thống lưu trữ, mạng, và cơ sở hạ tầng máy chủ.
- Nắm được toàn bộ hoạt động hệ thống bằng cách cung cấp bảng điều khiển trung tâm môi trường không đồng bộ.
- Theo dõi những thay đổi và đảm bảo an ninh cho môi trường bằng cách giám sát môi trường để phát hiện những hoạt động bất ngờ, thay đổi vai trò của người sử dụng, truy cập trái phép,…

Quản lý ảo hóa:

- Cơ sở hạ tầng ảo hóa tạo ra môi trường năng động, nơi mà tài nguyên máy tính như máy chủ, storage, phần cứng mang được ảo hóa từ các ứng dụng, hệ điều hành và người sử dụng. Môi trường ảo hóa phức tạp đòi hỏi cách tiếp cận mới với các dịch vụ CNTT truyền thống như xử lý sự cố hiệu suất, quản lý và phân tích rủi ro.
- Ứng dụng ảo hóa của Splunk kết hợp sức mạnh và tính năng của Splunk Enterprise được thiết kế dành riêng cho công nghệ ảo hóa. Kết hợp dữ liệu hạ tầng ảo hóa với dữ liệu tầng công nghệ khác sẽ cho một góc nhìn bao quát hơn về hệ thống trung tâm dữ liệu.
- Splunk App cho ảo hóa có thể tương thích và thu thập dữ liệu ảo hóa từ các công nghệ ảo hóa như WMware vSphere, Citrix XenServer và Microsoft Hyper-V và công nghệ ảo hóa máy tính bàn như Citrix XenApp và Citrix XenDesktop.
- Nó tạo các báo cáo đa dạng, đồng nhất về các công nghệ ảo hóa từ tất cả các lớp ứng dụng và cơ sở hạ tầng
- Giúp chủ động ngăn chặn, quản lý vấn đề hiệu suất, tắc nghẽn cổ chai, những sự kiện bất ngờ, những thay đổi và lỗi an ninh bảo mật nguy hiểm. Nó phân tích và báo cáo chính xác giúp cho người dùng có trải nghiệm tối ưu
- Tương quan dữ liệu ảo hóa, giúp việc tìm ra các sự kiện có liên quan một cách dễ dàng hơn, tương quan các vấn đề về hiệu năng, mạng và kiến trúc hệ thống máy chủ.
- Giữ lại số liệu về hiệu suất hoạt động của máy để theo dõi và phân tích. Thu thập dữ liệu có chiều sâu từ máy chủ, máy ảo, hệ thống máy tính. Cung cấp khả năng hiển thị hoạt động và phân tích hoàn chỉnh bằng cách xác định khả năng của máy chủ, các máy ảo nhàn rỗi, các máy chủ sử dụng đúng mức, sức chứa dữ liệu, theo dõi thống kê hiệu suất để tìm mô hình sử dụng và tránh khả năng tắc nghẽn có thể.
- Theo dõi chi tiết sự thay đổi mà người dùng thực hiện, tự động hóa các tác vụ của vSphere cũng như báo cáo tình trạng các thành phần ảo. Cải thiện an ninh bằng cách giám sát môi trường để tìm các hoạt động đáng ngờ, vai trò của người sử dụng bị thay đổi, truy cập trái phép và nhiều hơn nữa

2.2.3. Thành phần của Splunk

2.2.3.1. Thành phần thu thập log của Splunk

Đối với bộ công cụ splunk thì thành phần thu thập log được chia làm ba loại là: universal forwarder, heavy forwarder và light forwarder.

Universal forwarder: là một streamlined Nó là phiên bản chuyên dụng của Splunk mà chỉ chứa các thành phần thiết yếu để chuyển dữ liệu từ máy trạm đến máy server. Đây là phiên bản khá gọn nhẹ để tích hợp trên các nguồn sinh log chính vì thế mà nó không bao gồm các tính năng như lập chỉ mục cho dữ liệu và tìm kiếm dữ liệu.

Heavy forwarder: Có kích thước nhỏ hơn một Splunk indexer nhưng vẫn giữ được hầu hết các tính năng ngoại trừ việc tìm kiếm các kết quả phân phối. Và một số thành phần ví dụ Web splunk nếu cần thiết có thể bị vô hiệu hóa để giảm bớt kích thước. Một heavy forwarder phân tích dữ liệu trước khi chuyển và có thể định tuyến dữ liệu dựa trên các đặc điểm như nguồn và các loại sự kiện. Nó cũng có thể đánh chỉ mục cho dữ liệu trên máy cục bộ cũng như chuyển dữ liệu đến một splunk đặc biệt khác.

Light forwarder: Những loại dữ liệu được chuyển đó là dữ liệu thô, dữ liệu chưa được phân tích và dữ liệu đã được phân tích. Mỗi công cụ chuyển tiếp dữ liệu cho phép chuyển các loại dữ liệu khác nhau. Với universal và light forwarder làm việc với dữ liệu thô và dữ liệu chưa được phân tích. Còn heavy forwarder làm việc với dữ liệu thô hoặc dữ liệu đã được phân tích.

Với dữ liệu thô thì luồng dữ liệu được chuyển tiếp như dữ liệu TCP đơn thuần. Dữ liệu không được chuyển đổi sang định dạng của splunk. Thiết bị chuyển tiếp chỉ lựa chọn dữ liệu và đẩy chúng đi. Với việc chuyển dữ liệu như thế này hữu ích cho việc chuyển dữ liệu sang hệ thống không phải là splunk.

Với dữ liệu không được phân tích thì universal forwarder thực hiện các tiến trình tối thiểu. Mặc dù dữ liệu không được phân tích nhưng nó vẫn được định dạng các thẻ để xác định nguồn, loại nguồn, host. Nó cũng chia các luồng dữ liệu thành các block có kích thước 64K. Thực hiện việc gắn nhãn thời gian lên luồng dữ liệu để bộ phận tiếp nhận có thể phân biệt được dữ liệu khi mà nó không có một mốc thời gian cụ thể. Universal forwarder không xác định, kiểm tra, và gán thẻ lên các sự kiện cá nhân.

Đối với dữ liệu đã được phân tích công cụ heavy forwarder chuyển đổi dữ liệu thành các dạng dữ liệu riêng biệt. Nó sẽ gán thẻ và chuyển dữ liệu đến splunk

indexer. Nó cũng có thể kiểm tra các sự kiện. Bởi vì dữ liệu đã được phân tích, sau đó bộ phận forwarder có thể thực hiện việc định tuyến dữ liệu dựa trên sự kiện dữ liệu, chẳng hạn như giá trị của các trường.

Định nghĩa Index:

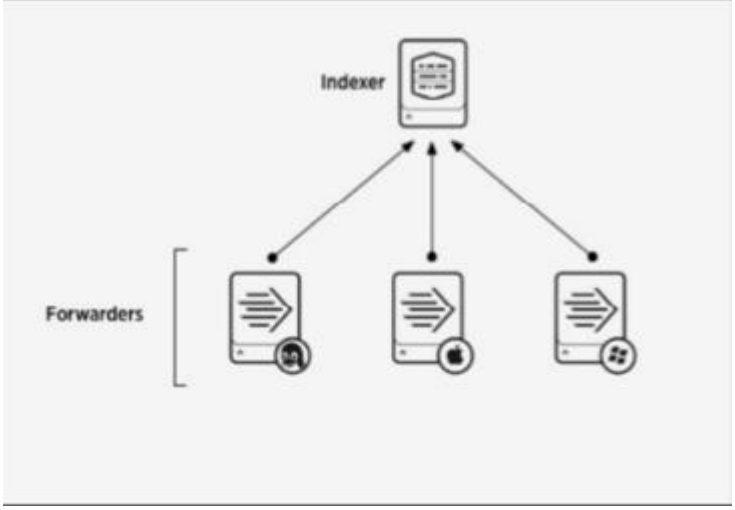
Splunk lưu trữ tất cả dữ liệu mà nó xử lí dưới dạng các index. Một index là một tập hợp các cơ sở dữ liệu với các thư mục con nằm ở SLPUNK_HOME/var/lib/splunk. Các index chứa hai file là dữ liệu thô và file index. Các loại index mặc định là:

- Main: tất cả dữ liệu xử lí được lưu trữ tại đây trừ nếu chúng không áp dụng các quy tắc khác.
- _ internal: lưu trữ log của splunk và các số liệu xử lí.
- _ audit: chứa các sự kiện liên quan đến sự giám sát thay đổi hệ thống, kiểm toán, và tất cả các lịch sử tìm kiếm của người dùng.

Một người quản trị splunk có quyền tạo một index, sửa đổi hay xóa bỏ hoặc thay thế một index đã tồn tại. Việc quản lí index được làm qua Web, CLI, và file cấu hình như index.conf.

Các mô hình dữ liệu

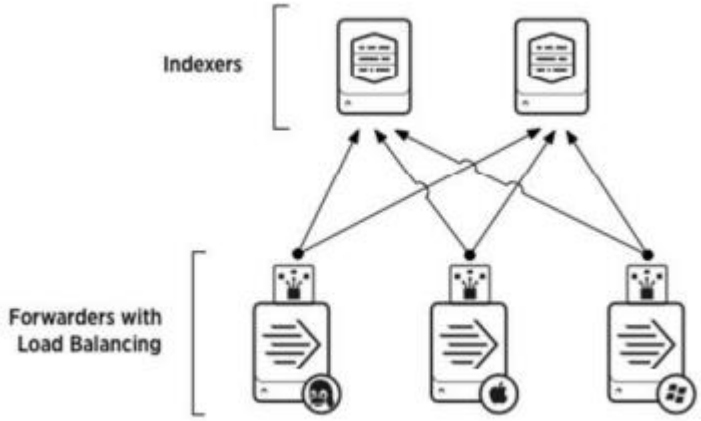
Mô hình dữ liệu tập trung: Là một trong những mô hình phổ biến với nhiều thiết bị forwarder từ các nguồn khác nhau gửi đến một splunk server. Mô hình này thường là các universal forwarder chuyển tiếp dữ liệu chưa phân tích từ máy trạm hoặc các thiết bị không phải là splunk server tới máy chủ splunk trung tâm để tổng hợp và đánh chỉ mục cho dữ liệu.



Hình 2.3. Mô hình thu thập dữ liệu log tập trung

Mô hình cân bằng tải: Mô hình định tuyến và lọc dữ liệu thì công cụ chuyển tiếp sẽ định tuyến dữ liệu đến một hệ thống splunk riêng hoặc một hệ thống thứ ba

dựa trên thông tin về nguồn, loại nguồn hoặc các mẫu có sẵn trong bản thân các sự kiện. Tất nhiên hệ thống này yêu cầu bộ công cụ heavy forwarder do cần phải phân tích dữ liệu để lấy thông tin. Ta có thể lọc dữ liệu gửi đi theo yêu cầu ví dụ như chỉ gửi đi những log chứa xâu kí tự error.



Hình 2.4. Mô hình thu thập dữ liệu log cân bằng tải

2.2.3.2. Thành phần xử lý dữ liệu đầu vào

Những loại dữ liệu được đưa vào Splunk bao gồm: các sự kiện của mạng, nguồn dữ liệu từ hệ điều hành windows, các nguồn khác.

Các sự kiện của mạng: Splunk có thể đánh chỉ mục dữ liệu từ bất kì cổng mạng nào. Ví dụ, Splunk có thể đánh chỉ mục cho dữ liệu từ syslog-nd hoặc bất kì ứng dụng khác có chức năng chuyển dữ liệu theo định dạng TCP. Nó cũng có thể đánh chỉ mục cho dữ liệu ở dạng UDP nhưng ta nên sử dụng TCP để nâng cao độ tin cậy của dữ liệu. Nó cũng có thể nhận và xử lý các sự kiện SNMP, các cảnh báo được đưa ra từ các thiết bị từ xa.

Nguồn dữ liệu từ windows: Các phiên bản splunk cho windows định nghĩa một loạt các input (đầu vào) riêng biệt cho windows. Nó cũng cung cấp việc đánh số trang trong hệ thống splunk để định nghĩa việc xác định các loại đầu vào riêng biệt đặc trưng cho windows như: Dữ liệu Windows event log, dữ liệu Windows Registry, dữ liệu WMI, dữ liệu Active Directory, dữ liệu từ các tiện ích được thiết lập giám sát

Nguồn dữ liệu khác: Splunk cũng hỗ trợ các loại nguồn dữ liệu như: Hàng đợi First-in, First-out, đầu vào từ các script: lấy dữ liệu từ các API và các giao diện từ các dữ liệu từ xa khác, các tin nhắn hàng đợi, module đầu vào: xác định khả năng đầu vào để mở rộng các khung Splunk. Ngoài những dữ liệu được đưa vào từ các công cụ forwarder người dùng có thể cấu hình để thêm dữ liệu mà ta mong muốn.

Sau khi đã nhận được dữ liệu Splunk tiến hành xử lí các sự kiện được chuyển vào. Các sự kiện là các bản ghi của hành động được lưu trữ lại trong tập tin nhật ký, lưu trữ trong các index. Các sự kiện cung cấp thông tin về các hệ thống, tạo ra các file nhật ký. Dưới đây là một sự kiện của hành động đăng xuất được ghi lại:

172.26.34.223-[01/oct/2020:12:05:27-0700]"GET/trade/app?action=logout HTTP/1.1" 200 2953

Quá trình xử lý dữ liệu bao gồm:

- Định dạng bộ kí tự cho dữ liệu đầu vào để phù hợp với định dạng mà Splunk có thể xử lý
- Quá trình phân mảnh các sự kiện
- Gán nhãn thời gian cho các sự kiện
- Trích xuất dữ liệu để tạo các trường đánh chỉ mục

Cấu hình nhãn thời gian (Timestamps):

Timestamps là rất quan trọng đối với Splunk. Nó sử dụng nhãn thời gian để tương quan các sự kiện theo thời gian, để tạo các biểu đồ thời gian trong Web splunk, và thiết lập các khoảng thời gian cho việc tìm kiếm.

Splunk gán các timestamp một cách tự động trong thời gian nó đánh chỉ mục sử dụng thông tin từ dữ liệu sự kiện thô. Nếu một sự kiện mà không chứa thời gian cụ thể, nó sẽ gán dựa theo các cách thức khác. Một vài dữ liệu có thể cần định nghĩa ra cách đánh timestamp.

Trích xuất các trường được đánh index trong Splunk:

Khi Splunk đánh chỉ mục dữ liệu, nó phân tích dữ liệu trong một chuỗi các sự kiện. Một phần của tiến trình này, nó thêm một số trường vào sự kiện dữ liệu. Các trường đó bao gồm những trường mặc định được tự động thêm vào và bất kì trường nào do người dùng định nghĩa

Quá trình thêm các trường tới sự kiện được gọi là trích xuất trường (field extraction). Có hai loại field extraction là:

- Indexed field extraction, Splunk lưu trữ các trường này trong index và coi nó như là một phần của dữ liệu sự kiện.
- Search-time field extraction, các trường chỉ được thêm vào trong quá trình tìm kiếm mà không được lưu trữ trong index.

Bảng 2.1. Các trường trong Index

Loại trường	Danh sách các trường	Mô tả
Trường nội bộ	_raw, _time, _indextime _cd	Trường này chứa thông tin mà Splunk sử dụng cho các tiến trình nội bộ
Trường mặc định cơ bản	Host, index, linecount, punct, source, sourcetype, splunk_server, timestamp	Trường này cung cấp các thông tin cơ bản về một sự kiện, chẳng hạn như nơi sinh ra sự kiện, loại dữ liệu mà nó chứa, vị trí index, có bao nhiêu dòng mà nó chứa và nó được sinh ra khi nào
Các trường thời gian mặc định	Date_hour, date_mday, Date_minute, Date_mounth, Data_second, date_wday, date_year, date_zone	Các trường này cung cấp thêm thông tin tìm kiếm các sự kiện theo các timestamp.

Khi Splunk đánh chỉ mục dữ liệu, nó gán thẻ mỗi sự kiện với một số trường. Những trường này sẽ trở thành một phần của sự kiện đã được đánh chỉ mục. Các trường này được tự động thêm vào và được xem như các trường mặc định.

Các trường mặc định phục vụ cho một số mục đích. Ví dụ như, trường index định nghĩa index mà sự kiện được lưu trữ, trường linecount mô tả số dòng mà sự kiện đó chứa, và timestamp định nghĩa thời gian sự kiện đó xảy ra. Splunk sử dụng giá trị trong một vài trường đặc biệt là sourcetype, khi đánh chỉ mục dữ liệu giúp tạo ra các sự kiện đúng. Khi một dữ liệu đã được đánh chỉ mục, có thể sử dụng các trường mặc định đó để thực hiện việc tìm kiếm

Định nghĩa về host, source, và sourcetype host

Một máy chủ của sự kiện thường là tên máy, địa chỉ IP, hoặc tên miền đầy đủ của các máy chủ mạng mà sự kiện đó sinh ra. Giá trị máy chủ cho phép dễ dàng xác định vị trí dữ liệu sinh ra từ các thiết bị cụ thể.

Source: nguồn của sự kiện là tên của tệp tin, luồng hoặc các đầu vào khác nơi mà các sự kiện được sinh ra. Các dữ liệu được giám sát từ các tệp tin và thư mục, giá trị của nguồn là các đường dẫn cụ thể như "/archive/server1/var/log/messages.0" hoặc "/var/log/". Giá trị của nguồn cho dữ liệu từ mạng là giao thức, cổng chẳng hạn như UDP:514.

Sourcetype: các loại nguồn của một sự kiện là định dạng của dữ liệu đầu vào mà dữ liệu sinh ra, chẳng hạn như "access_combined" hoặc "cisco_syslog". Loại nguồn xác định các Splunk định dạng dữ liệu.

So sánh giữa source và sourcetype

- Source là tên của tệp tin, luồng và các đầu vào khác xuất phát từ nguồn gốc các sự kiện đặc biệt.

- Sourcetype chỉ rõ định dạng của sự kiện. Splunk sử dụng trường này để xác định làm thế nào để định dạng luồng dữ liệu đi vào thành các sự kiện khác nhau.

Các sự kiện với cùng một loại nguồn có thể đến từ các nguồn khác nhau. Ví dụ người quản trị đang giám sát "source=/var/log/messages" và nhận đầu vào syslog trực tiếp từ udp: 514. Nếu tìm kiếm "sourcetype=linux_syslog", Splunk sẽ trả lại các sự kiện từ cả hai nguồn này

2.2.3.3. Thành phần đánh chỉ mục và lưu trữ

Với một lượng dữ liệu lớn được truyền từ các máy chủ về máy chủ tập trung thì việc lưu trữ và tìm kiếm sẽ rất khó khăn. Bởi vậy việc đầu tiên sau khi thu thập được log về sẽ phải lập chỉ mục cho dữ liệu và lưu trữ chúng phục vụ cho việc tìm kiếm. Có nhiều công cụ thực hiện công việc này ví dụ như elastic trong bộ công cụ logstash, hay splunk indexer trong bộ công cụ splunk.

Đối với công cụ splunk thì việc đầu tiên là splunk sẽ phải cung cấp dữ liệu, khi đã nhận được dữ liệu nó sẽ đánh chỉ số và làm cho chúng sẵn sàng để tìm kiếm. Với universal indexer được tích hợp thì splunk sẽ biến đổi dữ liệu thành một loạt các sự kiện liên quan đến từng lĩnh vực tìm kiếm. Ta có thể xử lí dữ liệu trước và sau khi splunk đánh chỉ số cho nó, nhưng điều này thường là không cần thiết.

Sau khi đánh index có thể bắt đầu tìm kiếm dữ liệu, hoặc sử dụng nó để tạo báo cáo, biểu đồ, cảnh báo hoặc nhiều công việc khácNhững loại dữ liệu mà splunk có thể đánh chỉ mục thường là bất kì một loại dữ liệu nào như windows event logs, webserver log, log từ các ứng dụng đang chạy, log từ hệ thống mạng, log giám sát, tin nhắn hàng đợi, tệp tin archive, hoặc bất kì nguồn nào có thể hữu ích.

Khi nguồn dữ liệu chuyển dữ liệu đầu vào splunk ngay lập tức đánh chỉ mục và đưa chúng đến các chuỗi dữ liệu để người dùng có thể tìm kiếm chúng ngay lập tức. Nếu như kết quả tìm kiếm không thỏa mãn yêu cầu, người quản trị có thể cấu hình lại cách đánh chỉ mục sao cho phù hợp.

Quá trình đánh index

- Event processing xảy ra qua hai giai đoạn là phân tích và đánh chỉ mục. Tất cả dữ liệu khi được đưa đến splunk đều được đẩy vào đường ống phân tích như các khối lớn khoản 10,000 byte. Trong quá trình phân tích splunk chuyển đổi các khối này thành các sự kiện phù hợp, nơi mà kết thúc tiến trình đánh chỉ mục cho các sự kiện.

- Trong quá trình phân tích, splunk thực hiện một vài hành động bao gồm: Trích xuất dữ liệu đặc trưng của mỗi dự kiện bao gồm host, source, sourcetype.
- Cấu hình các kí tự để thiết lập mã.
- Xác định việc chấm dứt dòng sử dụng quy tắc linebreaking. Có nhiều sự kiện ngắn chỉ một đến hai dòng nhưng cũng có những sự kiện chiếm rất nhiều dòng.
- Xác định "tem" thời gian hoặc tạo chúng nếu chúng không tồn tại. Trong cùng thời gian xử lý "tem" thời gian, splunk xác định ranh giới các sự kiện.
- Splunk có thể thiết lập để che dấu các sự kiện nhạy cảm như số thẻ tín dụng, số an sinh xã hội. Nó cũng có thể được cấu hình để áp dụng siêu dữ liệu trong các sự kiện sắp tới.

Trong quá trình đánh chỉ mục, Splunk thực hiện các tiến trình:

- Chia nhỏ các sự kiện thành các phân đoạn từ đó phục vụ cho việc tìm kiếm. Ta có thể tự định nghĩa kích thước của các phân đoạn tùy theo nhu cầu về tốc độ đánh chỉ mục và tìm kiếm, cũng như chất lượng tìm kiếm và hiệu năng của đĩa.
- Xây dựng cấu trúc dữ liệu chỉ mục. Ghi dữ liệu thô và các file index ra đĩa.

Quản lý Index:

Khi dữ liệu được thêm vào indexer xử lí và lưu trữ chúng dưới dạng index. Theo mặc định những dữ liệu đó được lưu trữ trong main index. Tuy nhiên người quản trị có thể cấu hình các index riêng cho các loại dữ liệu khác nhau.

Một index là một tập hợp các thư mục và tệp tin. Các thư mục có trong index còn được gọi là các bucket.

Để xem danh sách các index trong giao diện quản trị web ta truy nhập vào settings sau đó chọn Indexes. Có 3 loại index sẵn có là main, _internal, _audit.

Theo mặc định thì index main sẽ chứa tất cả các sự kiện. Indexer cũng có một vài index để sử dụng cho sự hoạt động trong bản thân hệ thống, cũng như cho các hoạt động khác như lập chỉ mục hay ghi lại các sự kiện.

Ta có thể tạo các index không hạn chế trong splunk. Tất cả các sự kiện mà không thuộc một index nào do người dùng định nghĩa thì sẽ được đẩy vào index main và kết quả tìm kiếm của ta nếu không chỉ ra tên index cụ thể đặt ra thì sẽ hiển thị các sự kiện trong main index

Việc tạo ra nhiều loại index đại diện cho nhiều loại dữ liệu có thể giúp:

- Kiểm soát được người dùng truy cập: khi phân quyền cho người dùng theo các role, ta có thể hạn chế người dùng tìm kiếm các thông tin nhạy cảm.
- Nếu có các chính sách khác nhau cho các dữ liệu khác nhau ta có thể chuyển dữ liệu từ index này sang index khác tùy theo nhu cầu sử dụng của ta. Một lí do khác để tạo ra nhiều index là nó sẽ rất hữu ích cho việc tìm kiếm. Giả sử ta có nhiều nguồn dữ liệu khác nhau như các sự kiện gửi từ windows và các sự kiện của một web server trên hệ điều hành linux. Tất cả dữ liệu này đều được lưu trữ trong cùng một index thì khi tìm kiếm các sự kiện trên windows phải tìm qua tất cả dữ liệu của hai nguồn, lúc này tốc độ chắc chắn sẽ chậm hơn rất nhiều.

Trong quá trình tìm kiếm mặc định các sự kiện ta tìm sẽ nằm trong main index, nếu muốn tìm kiếm trên các index riêng phải chỉ rõ tên index muốn tìm. Ví dụ lệnh tìm kiếm sau để tìm kiếm dữ liệu có trong index tên là fw_checkpoint và có user_name là dungnv:

index=“fw_checkpoint” user_name=“dungnv”

Khi dữ liệu trong một index đã đạt tới một thời gian nhất định hoặc khi kích thước index phát triển tới mức giới hạn, nó sẽ được đưa vào trạng thái đóng bang (Frozen). Nơi mà các indexer sẽ xóa nó từ các index mà nó được lưu trữ. Trước khi xóa dữ liệu indexer có thể di chuyển nó đến một nơi lưu trữ. Tất cả những việc trên đều phụ thuộc vào cách ta định nghĩa chính sách hết hạn của mình trong file indexes.conf.

Định nghĩa tuổi của dữ liệu:

Mỗi một thư mục index được xem như là một bucket (bộ chứa).

Một bucket được chuyển qua các giai đoạn như là các độ tuổi sau: Hot; Warm; Cold; Frozen; Thawed.

Một tuổi của bucket là việc chuyển từ giai đoạn hiện tại sang giai đoạn kết tiếp. Đối với dữ liệu vừa được đánh index thì nó sẽ đi tới giai đoạn hot bucket. Trong giai đoạn này dữ liệu có thể được tìm kiếm và ghi vào. Một index có thể có một vài hot bucket mở trong cùng một thời gian.

Khi một điều kiện nhất định xảy ra (ví dụ như hot bucket đạt đến một kích thước giới hạn hay splunkd được khởi động lại) thì hot bucket sẽ chuyển sang giai đoạn warm bucket và một hot bucket sẽ được tạo ra tại vị trí của nó. Warm bucket sẵn sàng cho việc tìm kiếm nhưng không cho phép ghi tiếp dữ liệu vào. Trong một index thì có rất nhiều warm bucket.

Khi một điều kiện tiếp tục được thỏa mãn (như index đạt đến số lượng tối đa các warm bucket). Indexer bắt đầu cuộn từ giai đoạn warm bucket sang cold, dựa trên tuổi của chúng. Nó luôn luôn lựa chọn những warm bucket lâu nhất để chuyển sang giai đoạn cold. Sau một thời gian quy định, các cold bucket sẽ chuyển sang trạng thái frozen. Tại thời điểm này chúng sẽ được lưu trữ hoặc được xóa đi. Để định nghĩa các chính sách quá hạn ta cần chỉnh sửa các thuộc tính trong file inputs.conf.

Nếu như dữ liệu frozen được lưu trữ nó có thể được khôi phục lại, đó là giai đoạn thawed. Giai đoạn này cho phép dữ liệu được phép tìm kiếm.

```
[windows]
homePath = $SPLUNK_DB\windows\db  #lưu event vào SSD
coldPath   = Y:\Splunk-Cold\windows\colddb #lưu event vào HDD
thawedPath = Y:\Splunk-thaweddb\windows\thaweddb #Thư mục recovery
sau khi nén event file
maxHotSpanSecs = 172800  #thời gian tồn tại event hot 2 day
maxWarmDBCount = 50   #số bucket warm 50
maxHotBuckets = 5   #số bucket hot 5
frozenTimePeriodInSecs = 10368000  #120 day tồn tại trong cold trước
khi move sang frozen
maxTotalDataSizeMB = 512000
enableDataIntegrityControl = 0
enableTsidxReduction = 0
maxDataSizeMB  = auto
enableDataIntegrityControl = 1
enableTsidxReduction = 1
compressRawdata = true #frozen nén lại khi bucket cold đạt 120 day
journalCompression = lz4 #thuật toán lz4
coldPath.maxDataSize = 200 #size của thư mục cold 200GB
coldToFrozenDir = Y:\Splunk-Archive\windows  #đường dẫn của thư mục
nén lưu trữ Frozen
```

Hình 2.5. Cấu hình indexes.conf

Mỗi một index sẽ chiếm giữ một thư mục cho riêng chúng ở trong "$SPLUNK_HOME/var/lib/splunk". Tên của thư mục là tên của index. Trong mỗi thư mục này chứa một chuỗi các thư mục con được phân chia theo các giai đoạn của bucket. Mặc định là thư mục db (hot/warm), cold, thawed. Ta có thể cấu hình vị trí lưu của các thư mục này sang ổ khác trong indexes.conf

Bảng 2.2. Vị trí của các thư mục lưu dữ liệu index

Giai đoạn	Vị trí mặc định	Chú ý
Hot	$SPLUNK_HOME/var/lib/splunk / defaultdb/db/*	Có thể có nhiều thư mục con cho mỗi hot bucket.
Warm	$SPLUNK_HOME/var/lib/splunk / defaultdb/db/*	Có các thư mục con riêng biệt cho warm bucket
Cold	$SPLUNK_HOME/var/lib/splunk / defaultdb/colddb/* 18001091	Có nhiều thư mục con cho cold bucket. Khi các warm bucket được chuyển qua cold, chúng thực hiện di chuyển các thư mục nhưng không đổi tên
Frozen	Dữ liệu của frozen sẽ bị xóa hoặc lưu trữ trong thư mục mà ta định nghĩa	Việc xóa là mặc định. Nếu ta cấu hình thì dữ liệu sẽ được lưu trữ
Thawed	$SPLUNK_HOME/var/lib/splunk / defaultdb/thaweddb/*	Vị trí của dữ liệu đã được lưu trữ và sau đó được khôi phục lại.

2.2.3.1. Thành phần cảnh báo (Alert)

Một cảnh báo là một hành động được kích hoạt dựa trên các kết quả của tìm kiếm. Khi tạo một cảnh báo, cần định nghĩa một điều kiện mà kích hoạt cảnh báo đó. Hành động điển hình là gửi email dựa trên các kết quả tìm kiếm. Ngoài ra cũng có thể chọn các hành động khác như chạy một đoạn mã script hoặc đưa chúng vào trong danh sách các cảnh báo. Với cùng một điều kiện cảnh báo có thể đưa chúng vào nhiều lựa chọn khác nhau như vừa gửi mail vừa chạy script. Để tránh việc gửi cảnh báo quá thường xuyên, ta cũng có thể giới hạn điều kiện cho một cảnh báo. Splunk định nghĩa ba loại cảnh báo là:

- Per result alert: Dựa trên việc tìm kiếm thời gian thực. Điều kiện kích hoạt là bất cứ khi nào việc tìm kiếm trả về một kết quả
- Scheduled alert. Chạy tìm kiếm theo lịch trình được chỉ định khi tạo cảnh báo. Ta định nghĩa các kết quả của việc tìm kiếm để kích hoạt cảnh báo đó.
- Rolling-window alert. Dựa trên việc tìm kiếm thời gian thực. Điều kiện kích hoạt là tập hợp các kết quả phù hợp của việc tìm kiếm trong một khung thời gian quy định.

Tiếp theo sẽ giới thiệu chi tiết các kịch bản của mỗi loại cảnh báo. Đối với per result alert: cảnh báo khi việc tìm kiếm thời gian thực trả về một kết quả phù hợp với điều kiện. Thông thường, ta định nghĩa một giới hạn điều kiện vì vậy cảnh báo được kích hoạt chỉ trong một khoảng thời gian quy định.

Các ví dụ về một kết quả trả về bao gồm các điều kiện dưới đây:

- Kích hoạt cảnh báo cho mỗi lần đăng nhập lỗi.
- Kích hoạt cảnh báo khi xảy ra những loại lỗi lựa chọn cho bất kì host nào.
- Cảnh báo xảy ra khi CPU trên host lên đến giá trị 100% trong một khoảng thời gian dài.

Cần chú ý khi triển khai per result alert trong một hệ thông yêu cầu độ sẵn sàng cao. Nếu một mạng ngang hàng không sẵn sàng, việc tìm kiếm thời gian thực có thể không được đảm bảo. Trong trường hợp này nên sử dụng scheduled alert.

Sử dụng một scheduled alert để đưa ra cảnh báo khi một lịch trình tìm kiếm trả về các kết quả phù hợp với điều kiện được định nghĩa. Một scheduled alert là hữu ích khi việc cảnh báo không cần thiết phải thực hiện luôn.

Một số ví dụ về scheduled alert:

- Kích hoạt cảnh báo chạy hàng ngày, cảnh báo xảy ra khi số lượng kết quả của ngày đó ít hơn 500.
- Kích hoạt cảnh báo theo giờ, giả sử như khi lỗi 404 trong mỗi giờ lớn hơn 100.

Rolling-window alert được sử dụng để giám sát các kết quả của việc tìm kiếm thời gian trong một khoảng thời gian được định nghĩa.

Ví dụ, giám sát các kết quả trong khoảng 10 phút hoặc mỗi giờ.

Ví dụ như: Một hành động cảnh báo sẽ xảy ra khi người dùng đăng nhập lỗi 3 lần trong vòng 10 phút. Ta có thể thiết lập điều chỉnh điều kiện để giới hạn việc gửi cảnh báo chỉ một lần trong vòng một giờ

Kích hoạt cảnh báo khi một máy chủ không thể chuyển một tệp tin đến máy chủ khác trong vòng một giờ. Có thể thiết lập điều chỉnh điều kiện để thực hiện cảnh báo cho một giờ cho mỗi máy

Sử dụng bộ điều chỉnh để giới hạn các cảnh báo

Một cảnh báo có thể được kích hoạt thường xuyên dựa trên các kết quả mà việc tìm kiếm trả về. Lịch trình chạy một cảnh báo cũng có thể kích hoạt các cảnh báo thường xuyên. Để giảm bớt hành động này theo yêu cầu của người dùng có hai cách sau:

- Giới hạn thời gian chạy cảnh báo.
- Xác định giá trị các trường mà kết quả tìm kiếm trả về.

Ví dụ, muốn tạo cảnh báo khi một lỗi hệ thống xảy ra, có khoảng 20 hoặc hơn 20 lỗi xảy ra mỗi phút nhưng người quản trị chỉ muốn gửi cảnh báo một lần mỗi giờ.

Để giảm bớt số lần cảnh báo trong trường hợp này cần phải cấu hình bộ điều chỉnh cho cảnh báo này như sau:

Bước 1: Từ trang tìm kiếm nhập vào thông tin sau index=_internal log_level=ERROR

Bước 2: Chọn Save As > Alert

Bước 3: Trong Result Type, chọn Real Time để cấu hình loại per result alert.

Bước 4: Chọn Next

Bước 5: Chọn các hành động muốn kích hoạt

Bước 6: Chọn Throttle

Bước 7: Chập log_level để giới hạn cảnh báo cho trường log_level. Ta có thể cấu hình bộ điều chỉnh để giới hạn nhiều hơn một trường.

Bước 8: Nhập 1 hour là thời gian giới hạn việc kích hoạt cảnh báo.

Bước 9: Nhấp vào Save Đối với việc tìm kiếm theo lịch trình được chạy thường xuyên, ta không muốn thông báo xảy ra cho mỗi lần chạy, có thể cấu hình bộ điều chỉnh để kiểm soát các cảnh báo trong một khung thời gian dài hơn.

Đối với việc tìm kiếm theo thời gian thực, nếu cấu hình chỉ cảnh báo một lần cho mỗi điều kiện kích hoạt, thì ta không cần phải cấu hình bộ điều chỉnh.

Khi ta cấu hình bộ điều chỉnh cho việc tìm kiếm theo thời gian thực, khi bắt đầu có thể đặt khoảng thời gian đưa ra cảnh báo phù hợp, sau đó có thể mở rộng khoảng thời gian nếu cần thiết. Việc này sẽ giúp ngăn chặn nhiều thông báo cho một hành động.

Ngoài ra ta có thể sử dụng một số Add-on của Splunk để tăng cường khả năng giám sát và bảo mật, một số ví dụ điển hình như:

- Telegram Alert Action: Cung cấp khả năng gửi các thông báo alert về Telegram thông qua Bot Telegram
- Webhook: Sử dụng Webhook sẽ cung cấp cho Splunk khả năng tự động phản ứng khi có một alert xảy ra, qua đây sẽ khiến Splunk có một phần khả năng tự động của SOAR
- MISP42: Cung cấp khả năng kết nối với server MISP – Một giải pháp Threat Intelligence Opensource, qua đây ta có thể sử dụng những dữ liệu từ MISP để triển khai một số alert khi người dùng nội bộ kết nối tới một IP độc hại hoặc mở một file nghi ngờ là mã độc.

2.2.4. Cách thức hoạt động của Splunk

Mức thấp nhất của kiến trúc Splunk mô tả các phương thức nhập liệu khác nhau được hỗ trợ bởi Splunk. Những phương thức nhập này có thể được cấu hình để gửi dữ liệu trên các bộ phân loại Splunk.

Trước khi dữ liệu đến được các bộ phân loại Splunk, nó có thể được phân tích cú pháp hoặc thao tác, có nghĩa là làm sạch dữ liệu có thể được thực hiện nếu cần. - Một khi dữ liệu được lập chỉ mục trên Splunk, nó sẽ tiến hành đi vào cụ thể để phân tích dữ liệu.

Splunk hỗ trợ hai loại triển khai: triển khai độc lập và triển khai phân tán. Tùy thuộc vào loại triển khai, tìm kiếm tương ứng được thực hiện. Công cụ Splunk có các thành phần bổ sung khác của quản lý dữ liệu, báo cáo và lên kế hoạch và cảnh báo. Toàn bộ công cụ Splunk được tiếp xúc với người dùng thông qua Splunk CLI, Splunk Web Interface, và Splunk SDK, được hỗ trợ bởi hầu hết các ngôn ngữ.

Splunk cài đặt một quy trình máy chủ phân tán trên máy chủ được gọi là splunkd. Quá trình này có trách nhiệm lập chỉ mục và xử lý một số lượng lớn dữ liệu thông qua các nguồn khác nhau. Splunkd có khả năng xử lý số lượng lớn dữ liệu phát trực tuyến và lập chỉ mục cho phân tích thời gian thực trên một hoặc nhiều đường ống (Pipeline)

Hình 2.6. Cơ chế hoạt động của Splunk

Danh sách các khối kiến trúc Splunk:

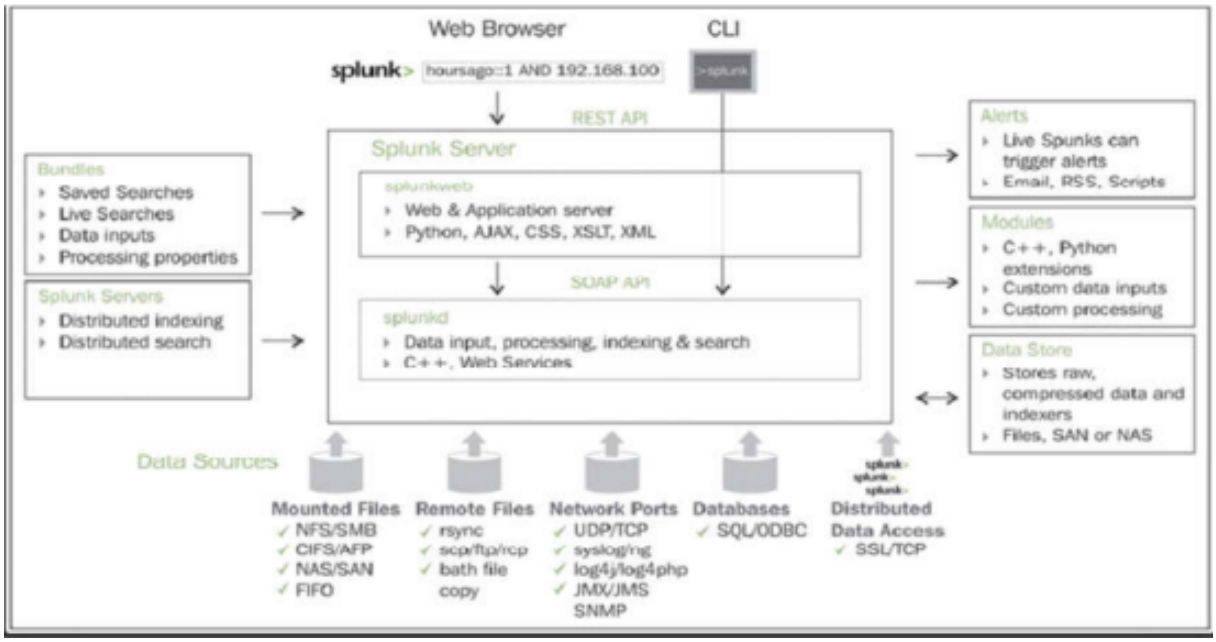
- **Pipeline:** Đây là một quá trình cấu hình đơn luồng duy nhất nằm trong splunk.

- **Bộ vi xử lý:** Chúng là những hàm số có thể tái sử dụng cá nhân hoạt động trên dữ liệu đi qua một đường ống. Đường ống trao đổi dữ liệu giữa họ thông qua một hàng đợi

Splunk cho phép người dùng tìm kiếm, điều hướng và quản lý dữ liệu trên Splunk Enterprise thông qua giao diện web được gọi là Splunk Web.

Một trong những thành phần quan trọng của kiến trúc của Splunk là kho dữ liệu. Nó có trách nhiệm nén và lưu trữ dữ liệu ban đầu (nguyên vẹn). Dữ liệu được lưu trữ trong các tệp Time Series Index (T SIDX).

Các triển khai của Splunk Enterprise có thể bao gồm từ việc triển khai các máy chủ đơn (có chỉ số vài gigabyte dữ liệu mỗi ngày và được truy cập bởi một vài người dùng đang tìm kiếm, phân tích và hình dung dữ liệu) tới các triển khai lớn của doanh nghiệp ở nhiều trung tâm dữ liệu, lập chỉ mục hàng trăm terabytes dữ liệu và tìm kiếm được thực hiện bởi hàng trăm người dùng. Splunk hỗ trợ giao thức truyền thông TCP chuyển tiếp dữ liệu từ một máy chủ Splunk sang một máy khác để lưu trữ dữ liệu và các yêu cầu phân phối và phân phối dữ liệu khác thông qua giao tiếp TCP Splunk-to-Splunk.



Hình 2.7: Sơ đồ hoạt động của Splunk

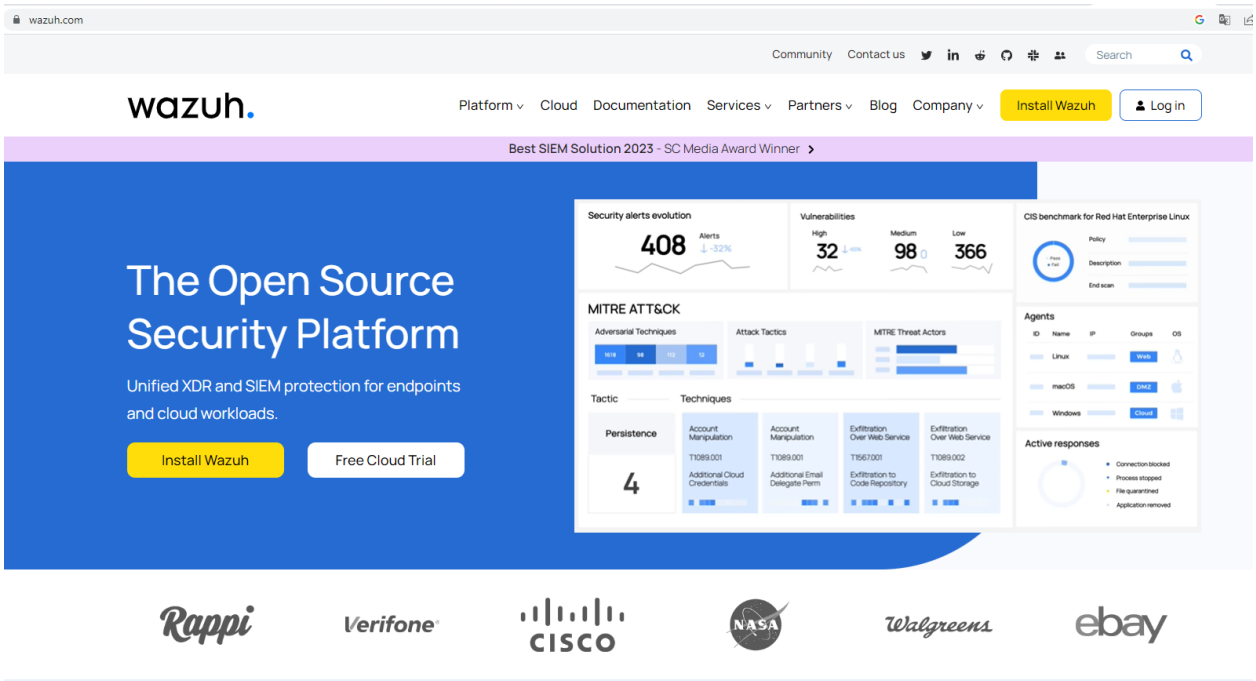
Bundles là các thành phần của kiến trúc Splunk lưu trữ cấu hình dữ liệu đầu vào, tài khoản người dùng, ứng dụng Splunk, tiện ích và môi trường khác

Các Modul là những thành phần của kiến trúc Splunk được sử dụng để thêm các tính năng mới bằng cách sửa đổi hoặc tạo bộ xử lý và đường ống (pipeline). Các Modul chỉ là các kịch bản tùy chỉnh và các phương pháp nhập dữ liệu hoặc phần mở rộng có thể thêm một tính năng mới hoặc sửa đổi các tính năng hiện có của Splunk.

2.3. Chức năng giám sát tính toàn vẹn tệp tin dựa trên Wazuh

2.3.1. Tổng quan về Wazuh

Wazuh là một nền tảng bảo mật mã nguồn mở và miễn phí. Khi mới được ra mắt với tên gọi ban đầu là OSSEC, nền tảng này đã được các tổ chức tin dùng như một giải pháp Host-IDS rất tin cậy với nhiều ưu điểm như dễ sử dụng, dễ dàng triển khai, mở rộng hệ thống khi cần thiết, miễn phí, có cộng đồng sử dụng lớn đi kèm với đó là khả năng hỗ trợ cao. Trong một số phiên bản phát hành gần đây, nhà phát triển của Wazuh đã định hướng lại cho nền tảng này phát triển không chỉ là một công cụ Host-IDS thông thường mà có tích hợp các khả năng của XDR cũng như SIEM vào trong giải pháp này. Từ đó, Wazuh có tiềm năng trở thành một nền tảng SIEM mạnh và có nhiều ưu điểm hơn các giải pháp SIEM hiện nay như Splunk, ELK Stack, Security Onion do được kết hợp các tính năng cũng như ưu điểm của SIEM, XDR, Host-IDS vào một giải pháp duy nhất.

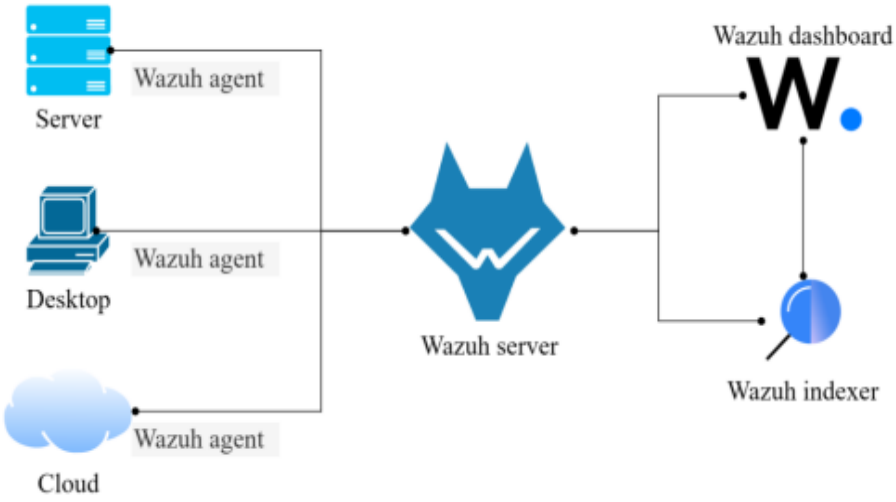


Hình 2.8. Trang chủ Wazuh

2.3.2. Thành phần và cách thức hoạt động của Wazuh

Giải pháp Wazuh hoạt động dựa trên tác nhân Wazuh (Wazuh agent) được triển khai trên các điểm cuối cần giám sát và ba thành phần trung tâm bao gồm: máy chủ Wazuh (Wazuh server), bộ lập chỉ mục Wazuh (Wazuh indexer) và bảng điều

khiển Wazuh (Wazuh dashboard). Hình ảnh dưới đây minh họa cho các thành phần của nền tảng bảo mật Wazuh.



Hình 2.9. Các thành phần trong Wazuh

2.3.2.1. Bộ xử lý trung tâm Wazuh

Bộ xử lý trung tâm Wazuh đóng vai trò trung tâm quản lý và xử lý dữ liệu từ Wazuh Agents. Bộ xử lý này là nơi nhận dữ liệu từ các Agents và thực hiện các quy tắc phân tích, giúp phát hiện các mối đe dọa an ninh, cung cấp các cảnh báo (alerts) và lưu trữ phục vụ cho quá trình điều tra số. Bộ xử lý trung tâm này bao gồm 3 thành phần chính:

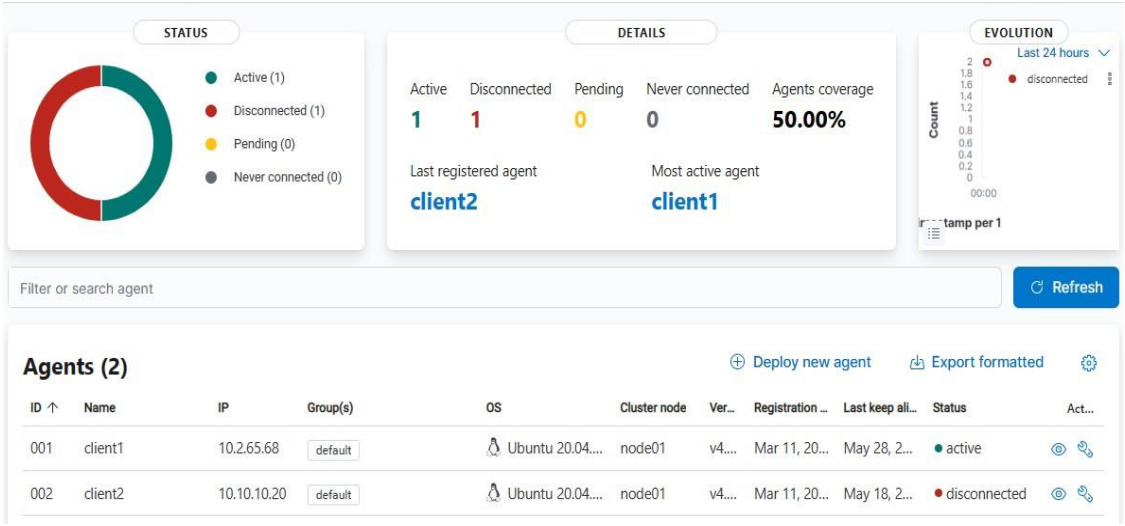
- Wazuh Indexer
- Wazuh Dashboard
- Wazuh Server

Wazuh Indexer là một thành phần quan trọng trong hệ thống giám sát an ninh mạng Wazuh đóng vai trò như một hệ quản trị cơ sở dữ liệu phân tán, được phát triển dựa trên OpenSearch. Opensearch là một phiên bản nâng cấp hơn của ElasticSearch - một hệ thống tìm kiếm và phân tích dữ liệu phân tán mạnh mẽ, cho phép tìm kiếm, truy vấn và phân tích dữ liệu một cách hiệu quả. Vì thế thành phần Indexer được sử dụng để lưu trữ và chỉ mục (index) dữ liệu, cụ thể là alert hoặc raw log nhận từ Wazuh Server sau quá trình phân tích, ngoài ra còn có thể nhận các thông tin nhận dạng người dùng và dữ liệu khác.

Bằng cách sử dụng Wazuh Indexer, người giám sát có thể thực hiện tìm kiếm, phân tích và theo dõi các sự kiện, alert và thông tin bảo mật quan trọng trong một giao diện trực quan. Điều này giúp người giám sát nắm bắt được tình hình bảo mật và đưa ra các biện pháp phòng ngừa và phản ứng kịp thời đối với các mối đe dọa

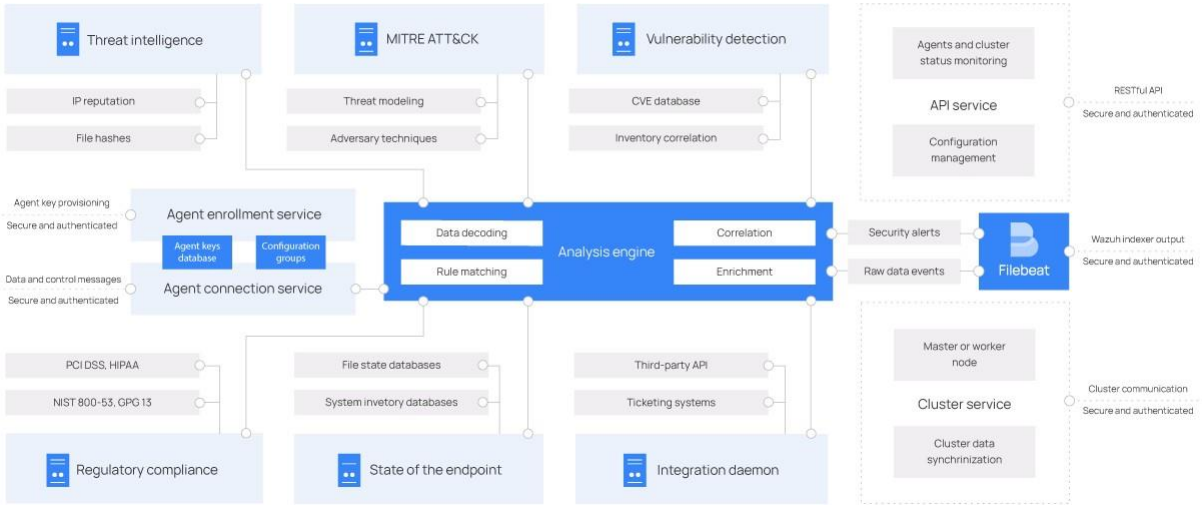
bảo mật. Wazuh Indexer là một phần không thể thiếu trong cấu trúc của Wazuh, đóng vai trò quan trọng trong việc xử lý và quản lý dữ liệu bảo mật, từ đó nâng cao khả năng giám sát và phân tích trong một hệ thống an ninh mạnh mẽ.

Tiếp đến, Wazuh Dashboard là thành phần cung cấp giao diện tổng quan về trạng thái và hoạt động của hệ thống giám sát Wazuh, được phát triển dựa trên Kibana. Điểm mạnh của Wazuh Dashboard là khả năng theo dõi và lọc dữ liệu từ các Agent trên một giao diện thống nhất. Wazuh Dashboard cung cấp các biểu đồ, bảng, và các thành phần khác để hiển thị thông tin quan trọng về alert, sự kiện, hành vi bất thường và nhiều khía cạnh khác liên quan đến bảo mật. Ngoài ra, Wazuh Dashboard sử dụng các API của Wazuh Server để quản lý toàn bộ hệ thống giám sát, bao gồm cả việc cấu hình Agent, thay đổi bộ decoder, bộ rule, CDB list và nhiều chức năng khác.



Hình 2.10. Giao diện giám sát Wazuh Agent

Cuối cùng, là trái tim của cả hệ thống Wazuh Server chính là trung tâm của Bộ xử lý, là nơi nhận dữ liệu từ các agents và đưa ra alert khi có một sự kiện khớp với tập luật được định sẵn (chẳng hạn phát hiện xâm nhập, file bị thay đổi, cấu hình không tuân thủ chính sách, rootkit,...). Ngoài ra, Wazuh Server còn cho phép đưa ra các phản ứng active response trước sự tấn công của các mối đe dọa. Việc phản ứng này được cấu hình tuỳ vào từng tổ chức cụ thể, có thể đưa ra các biện pháp khác nhau. Trong đồ án này, các phản ứng được xây dựng để khi hệ thống phát hiện mối đe doạ ngay lập tức đưa ra phản ứng như dừng truy cập tại địa chỉ IP, xóa file,... Tuy nhiên tùy từng tổ chức sẽ có các chính sách khác nhau, vậy nên các phản ứng được xây dựng trong đồ án chỉ nhằm mục đích tham chiếu tổng quan.



Hình 2.11. Kiến trúc thành phần trong Wazuh Server

Wazuh Server được cài đặt trên máy chủ vật lý, máy ảo hoặc máy trên cloud và gồm các thành phần chức năng được trình bày ngay sau đây. Dịch vụ đăng ký Agent (Agents enrollment service): Được sử dụng để đăng ký các Agent mới bằng cách cung cấp và phân phối khóa xác thực (authentication key pair). Quá trình này hỗ trợ xác thực qua chứng chỉ TLS/SSL hoặc xác thực bằng mật khẩu.

Dịch vụ kết nối với Agent (Agents connection service): Nhận dữ liệu từ các Agent, xác thực danh tính và mã hóa thông tin liên lạc giữa Agent và Wazuh Server. Ngoài ra, dịch vụ này cung cấp quản lý cấu hình tập trung và cho phép cài đặt các cấu hình cho Agent từ xa.

Bộ phân tích (Analysis engine): Xác định loại thông tin đang xử lý (ví dụ: Windows event log, SSH log hay log máy chủ web) và trích xuất các thành phần dữ liệu liên quan từ log (ví dụ: địa chỉ IP nguồn, ID sự kiện). Sau quá trình phân tích, nếu dữ liệu log trùng với bộ quy tắc được quy định trước, bộ phân tích kích hoạt lên alert và có thể thực hiện biện pháp đối phó một tự động (ví dụ: lệnh timeout request đến từ địa chỉ IP trên tường lửa).

Các events này được chia thành nhiều cấp độ (severity level) khác nhau từ 0 đến 15. Wazuh cho phép tuỳ chỉnh khi log nhận về match với rule định nghĩa sẵn ở level nhất định sẽ phát ra alert, mặc định là level 3. Level càng cao thì khả năng tổ chức bị tấn công mạng càng lớn. Bảng 2.1 mô tả 5 level cao nhất (10-15) trong bảng phân loại cấp độ rule của Wazuh.

Bảng 2.3: Bảng phân loại các cấp độ cao nhất của cảnh báo Wazuh

Level	Tiêu đề	Mô tả
10	Lỗi do người dùng	Ở level này, alert có thể chỉ ra một cuộc tấn công hoặc có thể chỉ là người dùng vô tình thực hiện

		các thao tác độc hại, ví dụ như nhập sai mật khẩu nhiều lần.
11	Cảnh báo kiểm tra tính toàn vẹn	Bao gồm các thông báo liên quan đến việc sửa đổi các file binary hoặc 1 file được giám sát thay đổi hoặc sự hiện diện của rootkit (Thực hiện bởi module Rootcheck). Và ở level này, cảnh báo cũng có thể chỉ ra một cuộc tấn công thành công.
12	Sự kiện quan trọng	Bao gồm các thông báo lỗi hoặc cảnh báo từ hệ thống, kernel, v.v. Alert level này có thể chỉ ra một cuộc tấn công chống lại một ứng dụng cụ thể.
13	Lỗi bất thường (Mức độ nguy hiểm cao)	Log thu được match với một cách thức tấn công mạng phổ biến như XSS, SQL, XXE, Command Injection, v.v
14	Sự kiện an ninh quan trọng	Liên kết với các mối tương quan (như từ các event khác), Wazuh đánh giá log này chỉ ra một cuộc tấn công đang nhắm vào hệ thống. Tuy nhiên, ở level này vẫn có tỉ lệ nhỏ rơi vào trường hợp dương tính giả (false positive)
15	Các cuộc tấn công nghiêm trọng	Không phải trường hợp dương tính giả nữa, đây đích thị là một cuộc tấn công mạng nhắm vào hệ thống. Yêu cầu hành động ngay lập tức.

Điểm mạnh của Wazuh so với các hệ thống quản lý log tập trung nằm ở khả năng tùy chỉnh và nâng cấp bộ phân tích. Đây cũng chính là thành phần mà đồ án này tập trung nâng cấp và tùy chỉnh.

Giao diện lập trình ứng dụng RESTful (Wazuh RESTful API): Cung cấp API tương tác với cơ sở hạ tầng Wazuh. Các API cung cấp các chức năng cho phép quản lý Agent, cấu hình máy chủ, theo dõi trạng thái toàn bộ cơ sở hạ tầng (Bao gồm trạng thái Agent, trạng thái kết nối đến các thành phần khác trong Bộ xử lý trung tâm,...), và giúp quản lý và chỉnh sửa các bộ rule. Các API này được sử dụng bởi thành phần Wazuh Dashboard được phát triển dựa trên ứng dụng Kibana, sẽ được giới thiệu kỹ hơn bên dưới.

Dịch vụ cụm Wazuh (Wazuh cluster daemon): Wazuh cho phép mở rộng theo chiều ngang bằng cách triển khai dưới dạng một cụm cluster. Với cấu hình theo cụm cluster này, kết hợp sử dụng với hệ thống cân bằng tải (load balancer), giúp cung cấp tính khả dụng cao cho Wazuh Server.

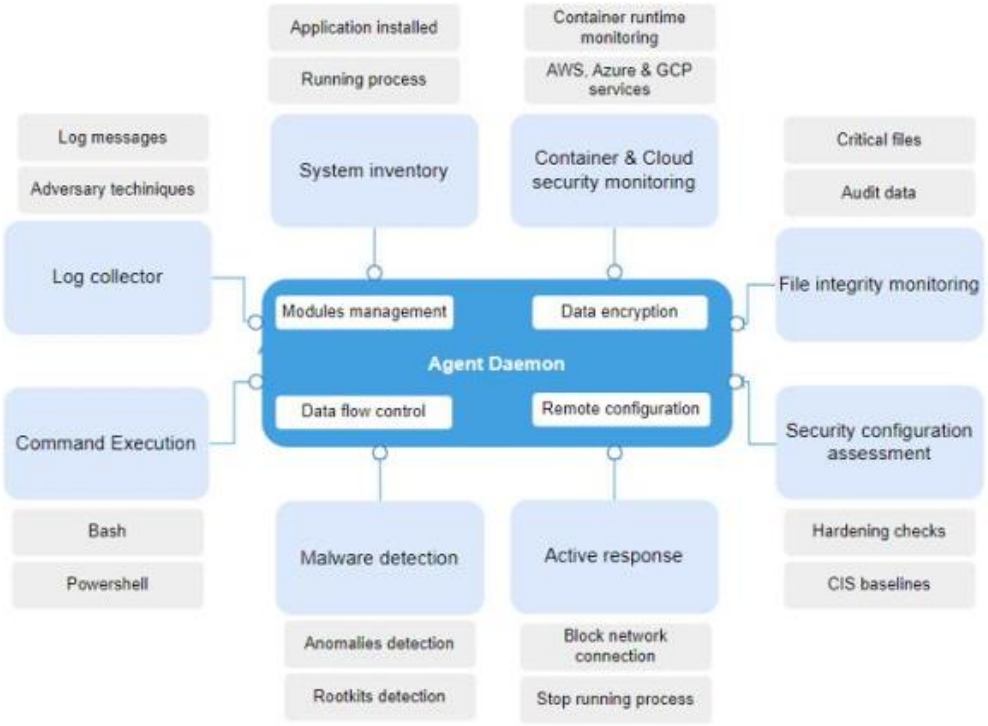
Filebeat: Gửi các alert và raw log tới các máy chủ Wazuh Indexer - được phát triển dựa trên OpenSearch (phiên bản nâng cấp của ElasticSearch), sẽ được giới thiệu kỹ hơn bên dưới. Filebeat đọc kết quả từ công cụ phân tích (Analysis engine) và gửi các events trong thời gian thực đến Wazuh Indexer. Cung cấp khả năng cân bằng tải khi kết nối với một cụm Elasticsearch nhiều nút.

2.3.2.2. Wazuh Agent

Một thành phần quan trọng nữa của hệ thống giám sát được gọi là agent. Các agent được cài đặt trên các máy chủ, máy trạm để thu thập thông tin log và gửi về thành phần Wazuh Server để phân tích và giám sát.

Công việc chính của các agent là thu thập dữ liệu từ các nguồn khác nhau trên hệ thống, bao gồm các log hệ thống, log truy cập với máy chủ chạy Web Server, cấu hình hệ thống và nhiều nguồn dữ liệu khác.

Wazuh Agent hoạt động hiệu quả trên nhiều nền tảng, bao gồm Windows, Linux, macOS và nhiều hệ điều hành khác. Điều này thể hiện sự linh hoạt, đa dạng của Wazuh. Ngay cả khi không hỗ trợ giám sát các thiết bị mạng như Switch, firewall với các hệ điều hành chuyên biệt, các agent cũng có thể nhận log thông qua một syslog server trung gian. Trong đồ án này tập trung cấu hình trên các server làm trọng tâm. Trên các hệ điều hành Unix-based, Wazuh agent chạy nhiều tiến trình riêng và giao tiếp với nhau thông qua local Unix domain socket. Một trong số các tiến trình này sẽ chịu trách nhiệm giao tiếp và gửi dữ liệu đến Wazuh server. Các tác vụ hoặc tiến trình khác nhau của agent sẽ chịu trách nhiệm giám sát khác nhau (chẳng hạn file kiểm tra tính toàn vẹn của file, đọc log, quét các cấu hình hệ thống). Đối với các hệ điều hành Windows, chỉ có một tiến trình agent duy nhất sử dụng mutex để chạy đa nhiệm các tác vụ. Chính cách xây dựng này giúp cho khả năng tùy chỉnh (bật hoặc tắt module trên hệ thống giám sát) của Wazuh trở nên linh hoạt và hiệu quả hơn.



Hình 2.12. Kiến trúc Wazuh Agent

Dưới đây là mô tả về mục đích của các module của Agent:

Bộ thu thập log (Log collector): Đây là module chính được xây dựng trên các Agent. Module này đọc các file log như sự kiện Windows, syslog hoặc log ứng dụng, log truy cập,v.v để gửi về Wazuh Server.

Bộ thực thi lệnh (Command execution): Agent thực thi các lệnh đã được ủy quyền định kỳ, thu thập kết quả và gửi thông tin báo cáo tới Wazuh Server để phân tích tiếp. Module này cho phép thực thi lệnh từ xa từ Wazuh Server, có nghĩa rằng người vận hành hệ thống giám sát có thể thực thi command nhất định trên máy chủ được giám sát, thông qua các Agent được cài đặt. Module này không được bật mặc định. Module này còn cho phép thực thi một số lệnh cho phép block IP dùng tường lửa cài sẵn trên server, cho phép xóa các file mã độc và một số lệnh khác sẽ được đề cập cụ thể ở chương sau của đồ án này, chương triển khai.

Bộ giám sát toàn vẹn tệp tin (File integrity monitoring): Module thực hiện giám sát hệ thống tệp tin, báo cáo khi có sự thay đổi như tạo, xóa hoặc sửa đổi tệp tin. Module này theo dõi các thay đổi trong thuộc tính, quyền truy cập, sở hữu và cả nội dung của tệp tin. Khi một sự kiện xảy ra, module này ghi lại thông tin về ai, cái gì và khi nào diễn ra sự kiện. Để sử dụng module FIM có hiệu quả, người vận hành chỉ nên chỉ định giám sát trên một số tệp tin hoặc thư mục nhất định.

Bộ đánh giá cấu hình bảo mật (Security configuration assessment): Module này cung cấp đánh giá cấu hình liên tục, sử dụng các kiểm tra tích hợp dựa trên các tiêu chuẩn CIS (Center of Internet Security). Người dùng cũng có thể tạo các kiểm tra SCA riêng để giám sát và áp dụng chính sách bảo mật của mình. Module này được sử dụng như một hoạt động đánh giá cấu hình theo tiêu chuẩn bảo mật (Vulnerability Assessment – VA).

Bộ quản lý danh mục hệ thống (System inventory): Module này thực hiện quét định kỳ, thu thập dữ liệu danh mục như phiên bản hệ điều hành, các cổng giao tiếp mạng, tiến trình đang chạy, ứng dụng đã cài đặt và danh sách cổng đang mở. Kết quả quét được lưu trữ trong cơ sở dữ liệu SQLite cục bộ có thể được truy vấn từ xa.

Bộ phát hiện mã độc (Malware detection): Module này sử dụng phương pháp không dựa trên chữ ký (non-signature-based) để phát hiện những sự bất thường và sự hiện diện có thể của rootkit. Bộ phát hiện mã độc cũng tìm kiếm các tiến trình, tệp tin và cổng ẩn, trong khi giám sát các lời gọi hệ thống.

Bộ phản ứng (Active response): Module này thực hiện các hành động tự động khi phát hiện mối đe dọa, kích hoạt các phản ứng như chặn kết nối mạng, dừng tiến trình đang chạy hoặc xóa tệp tin độc hại. Người dùng cũng có thể tạo các phản ứng tùy chỉnh khi cần thiết, ví dụ như chạy một chương trình nhỏ trong môi trường sandbox, bắt gói tin mạng hoặc quét tệp tin với phần mềm diệt virus.

Ngoài ra, Wazuh hỗ trợ giám sát các công nghệ mới, hiện đại như container hoá và đám mây. Tuy nhiên các module này chưa thực sự phát huy hiệu quả bằng những giải pháp giám sát chuyên dụng cho từng công nghệ. Vì thế đồ án này không thực hiện cấu hình các chức năng này cho hệ thống.

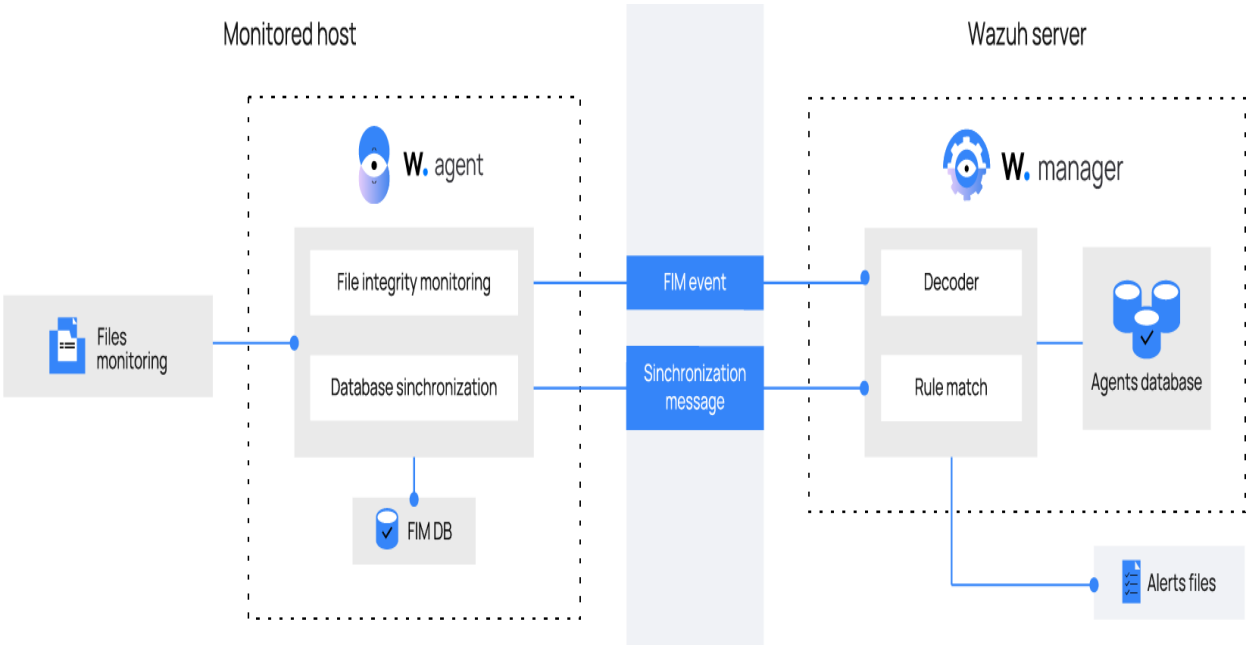
Giám sát bảo mật container (Container security monitoring): Module giám sát bảo mật container tích hợp với Docker Engine API để giám sát các thay đổi trong môi trường container. Ví dụ, module này phát hiện các thay đổi trong các image container, cấu hình mạng. Ngoài ra, module này cũng cảnh báo về các container đang chạy ở privileged mode, điều này là một nguy cơ về cấu hình nghiêm trọng mà nhiều tổ chức vẫn mắc phải.

Giám sát bảo mật đám mây (Cloud security monitoring): Module này giám sát các nhà cung cấp dịch vụ đám mây như Amazon AWS, Microsoft Azure hoặc Google GCP bằng cách tương tác trực tiếp với các API của nhà cung cấp. Module này có khả năng phát hiện các thay đổi trong cơ sở hạ tầng đám mây (ví dụ: tạo người dùng mới, sửa đổi security group, dừng một instance, v.v.) và thu thập log

của dịch vụ đám mây (ví dụ: AWS CloudTrail, AWS Macie, AWS Guard Duty, Azure Active Directory, v.v.)

2.3.3. Chức năng giám sát tính toàn vẹn tệp tin dựa trên Wazuh

Hệ thống giám sát toàn vẹn tệp (File Integrity Monitoring - FIM) của Wazuh theo dõi các tệp đã chọn và kích hoạt cảnh báo khi các tệp này thay đổi. Thành phần chịu trách nhiệm cho nhiệm vụ này được gọi là **Syscheck**. Thành phần này lưu trữ giá trị checksum, các thuộc tính khác của tệp và thường xuyên so sánh chúng với các tệp hiện tại trên hệ thống nhằm phát hiện sự thay đổi.



Hình 2.13. Luồng hoạt động của Module FIM

2.4. Kết luận chương II

Trong chương này, chúng ta đã tiến hành nghiên cứu chi tiết về hệ thống giám sát sự kiện và cảnh báo bất thường dựa trên việc kết hợp Splunk và Wazuh. Những điểm chính có thể được tóm tắt như sau:

Splunk được phân tích như một giải pháp quản lý và giám sát sự kiện tập trung mạnh mẽ. Các tính năng vượt trội của Splunk, như thu thập, tìm kiếm, phân tích và trực quan hóa dữ liệu thời gian thực, đã được trình bày rõ ràng. Hơn nữa, cấu trúc thành phần và cách thức hoạt động của Splunk cũng được giải thích để làm sáng tỏ vai trò của nó trong việc giúp doanh nghiệp xử lý dữ liệu log một cách hiệu quả.

Wazuh được giới thiệu như một công cụ mạnh mẽ để giám sát tính toàn vẹn của tệp tin và cảnh báo an ninh. Chức năng giám sát này đã được phân tích kỹ lưỡng thông qua các thành phần, cơ chế hoạt động và khả năng tích hợp với Splunk, từ đó tối ưu hóa việc bảo vệ dữ liệu trong hệ thống doanh nghiệp.

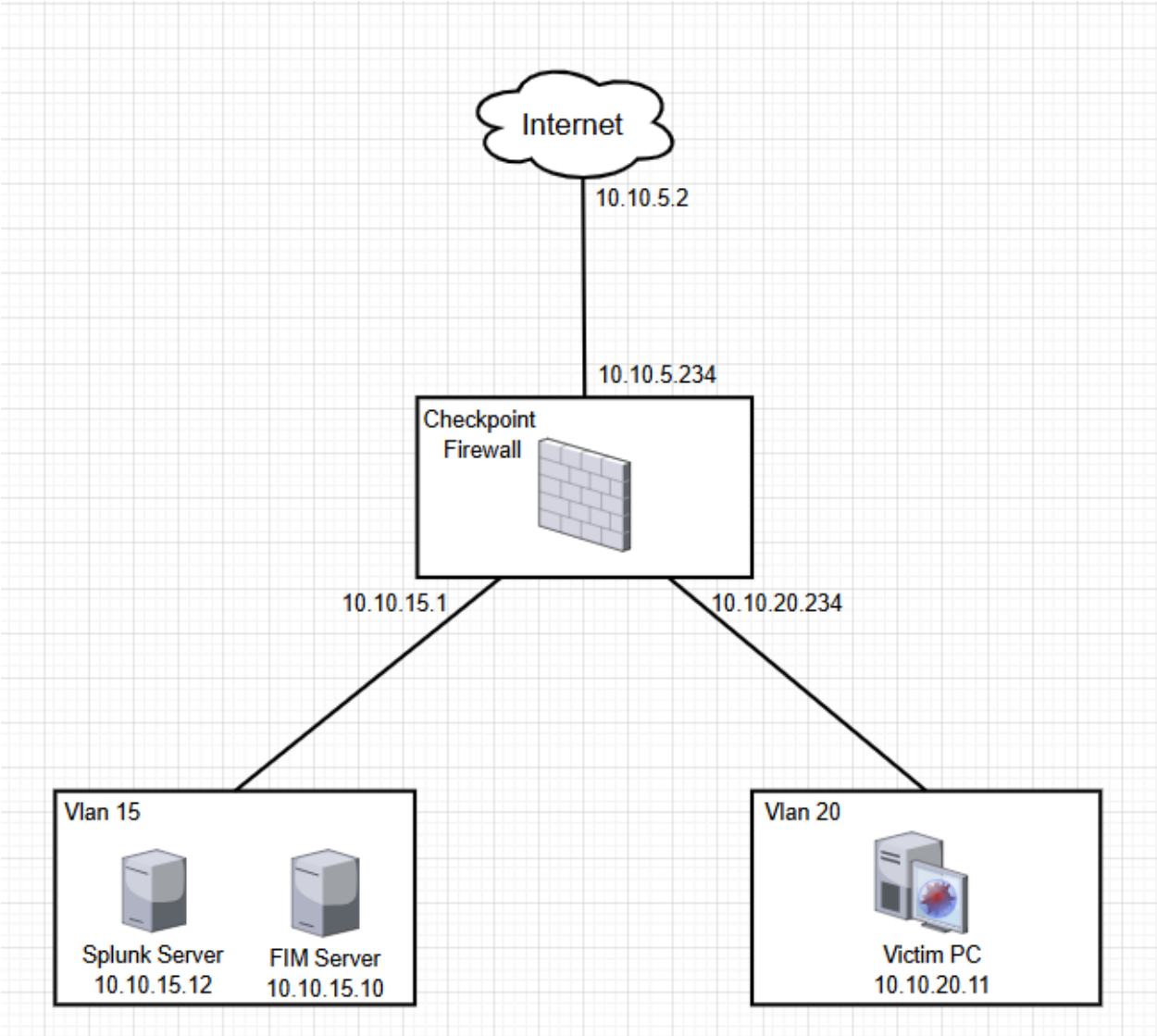
Sự kết hợp giữa Splunk và Wazuh tạo ra một hệ thống giám sát toàn diện, vừa cung cấp khả năng phân tích log mạnh mẽ, vừa hỗ trợ giám sát tính toàn vẹn dữ liệu. Điều này mang lại cho doanh nghiệp không chỉ khả năng phát hiện sự kiện bất thường mà còn cải thiện tính minh bạch và độ tin cậy của hệ thống.

Tóm lại, chương II đã cung cấp nền tảng lý thuyết và phân tích chi tiết, tạo tiền đề cho việc xây dựng và triển khai hệ thống giám sát sự kiện và cảnh báo bất thường trong thực tế. Những kiến thức này sẽ được áp dụng vào các phần tiếp theo để thiết kế và đánh giá hệ thống cụ thể.

CHƯƠNG 3. THỰC HIỆN XÂY DỰNG HỆ THỐNG GIÁM SÁT SỰ KIỆN VÀ CẢNH BÁO BẤT THƯỜNG CỦA DỮ LIỆU CHO DOANH NGHIỆP

Từ việc đáp ứng tốt các yếu tố về hệ thống, Chương 3 sẽ triển khai thực tế qua từng giai đoạn như tạo mô hình mạng, phân chia phân vùng mạng, cài đặt và cấu hình hệ thống. Hệ thống được đưa vào thực nghiệm nhằm có được góc nhìn tốt nhất để nhận xét về hệ thống mới sau khi tích hợp. Từ đó rút ra kết luận qua quá trình thực nghiệm.

3.1. Mô hình thực nghiệm



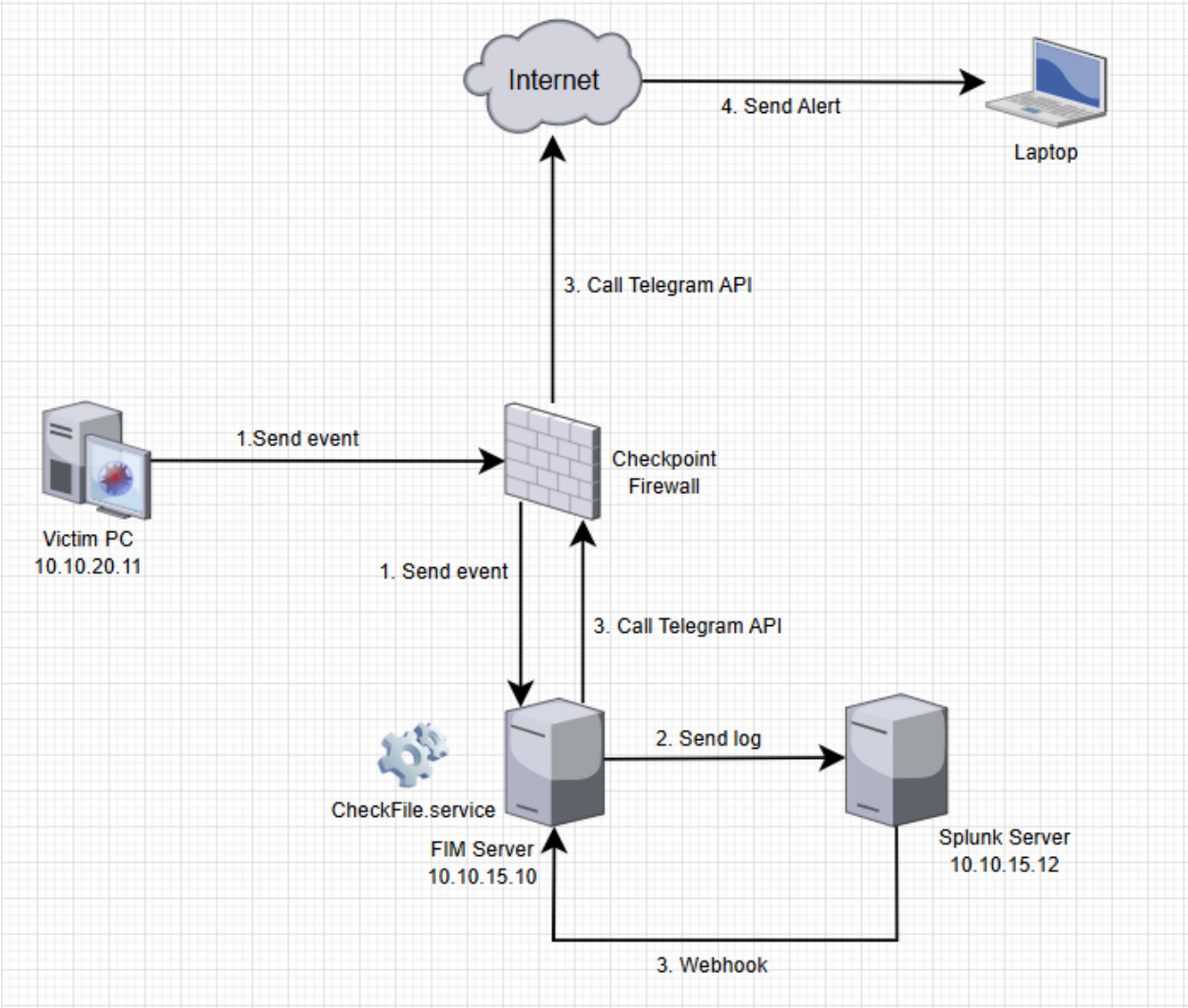
Hình 3.1. Mô hình hệ thống thực nghiệm mô phỏng

Trong mô hình trên, các máy chủ được chia về các phân vùng cũng như thể hiện chức năng của riêng mình:

- Vlan 15: Đây là Vlan chứa các máy chủ phục vụ cho quá trình thu thập, giám sát log
- Vlan 20: Đây là Vlan làm việc của nhân viên công ty, thể hiện trên hình bằng một máy đại diện.
- Máy chủ Splunk: Được đặt trong phân vùng Vlan 15, thực hiện nhiệm vụ giám sát hệ thống toàn cục, thu thập log OS qua cổng 9997/TCP, và thu thập log FIM từ Wazuh qua cổng 1024/TCP
- Máy chủ Wazuh: Đặt trong phân vùng Vlan 15, với mục tiêu giám sát tính toàn vẹn tệp tin của các máy Client. Máy chủ sẽ thực hiện quản lý các agent qua cổng 1515/TCP, nhận log từ agent gửi lên qua cổng 1514/TCP
- Tường lửa Checkpoint: Có nhiệm vụ chia phân vùng Vlan, mở kết nối giữa các phân vùng với nhau.

3.2. Triển khai SIEM Splunk kết hợp Wazuh FIM thử nghiệm cảnh báo bất thường dữ liệu

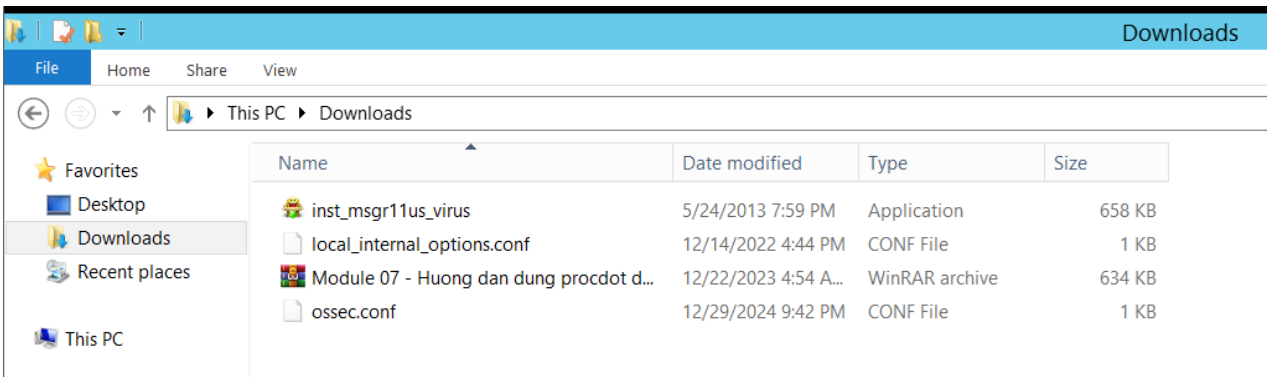
Giả sử đội SOC đang giám sát cho hệ thống Công ty X trên 2 server 10.10.15.10, 10.10.15.20. Đối với công việc giám sát trên SIEM sẽ tốn rất nhiều thời gian và có thể gây ra bỏ sót các luồng nhật ký. Dựa trên những khó khăn đó, đội SOC cần tạo ra một kênh truyền với nhiệm vụ cảnh báo các sự kiện cần theo dõi theo điều kiện đưa ra.



Hình 3.2. Sơ đồ logic luồng dữ liệu

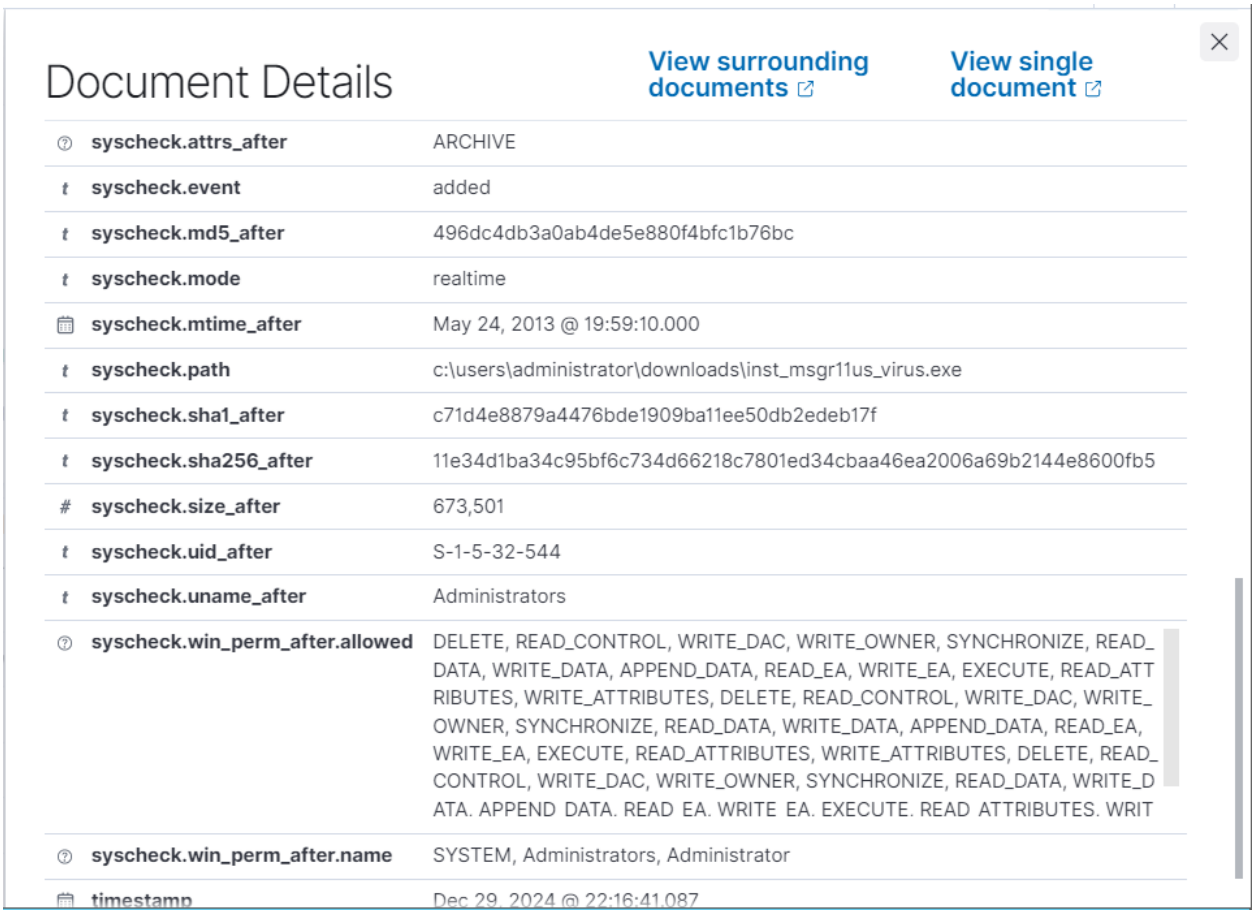
Mục tiêu của thực nghiệm: Mục tiêu của thực nghiệm này là xây dựng một hệ thống tự động cảnh báo khi có một người dùng thực hiện tải hoặc một tệp tin nào đó tự động sinh ra các tệp tin con tại các vị trí trong hệ thống mà nghi ngờ là mã độc. Qua đó cảnh báo ngay khi có sự kiện xảy ra, giúp đội SOC có thể có phản ứng sớm với tình huống tránh xảy ra sự cố. Kết quả cần đạt được sẽ là khả năng tự động phát hiện, cảnh báo mã độc về kênh truyền Telegram

Bước 1: Người dùng tải một file mã độc về, sau đó tiến hành giải nén file mã độc đó ra để lấy file thực thi bên trong



Hình 3.3. File thực thi mã độc được sinh ra sau khi giải nén

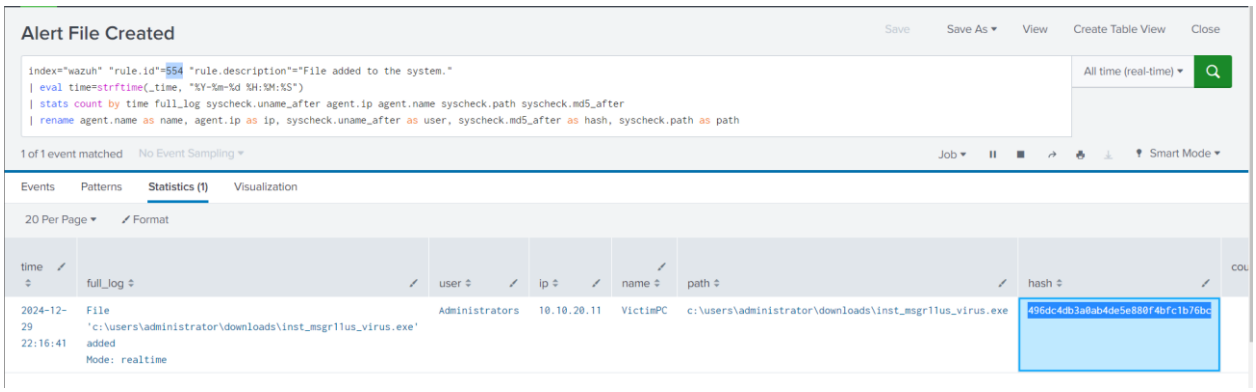
Bước 2: Sự kiện tạo file được gửi lên Wazuh theo cấu hinh giám sát đã được đặt ra từ trước và sau đó gửi từ Wazuh về Splunk



Hình 3.4. Log ghi nhận trên Wazuh

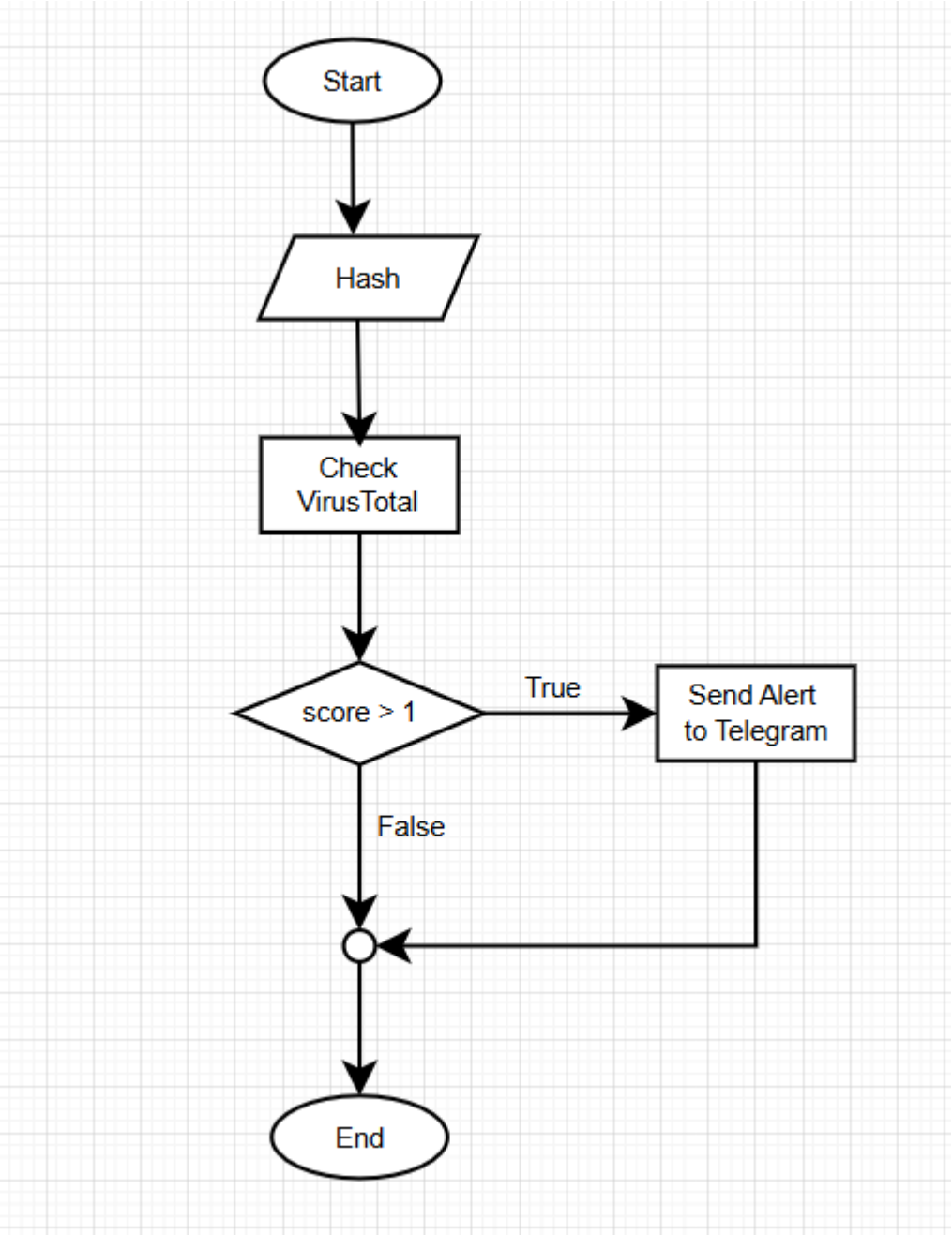
```
<!-- Directories to check  (perform all possible verifications) -->
<directories realtime="yes" check_all="yes" report_changes="yes">C:\Users\*\3D Objects</directories>
<directories realtime="yes" check_all="yes" report_changes="yes">C:\Users\*\Contacts</directories>
<directories realtime="yes" check_all="yes" report_changes="yes">C:\Users\*\Desktop</directories>
<directories realtime="yes" check_all="yes" report_changes="yes">C:\Users\*\Documents</directories>
<directories realtime="yes" check_all="yes" report_changes="yes">C:\Users\*\Downloads</directories>
<directories realtime="yes" check_all="yes" report_changes="yes">C:\Users\*\Favorites</directories>
<directories realtime="yes" check_all="yes" report_changes="yes">C:\Users\*\Links</directories>
<directories realtime="yes" check_all="yes" report_changes="yes">C:\Users\*\Music</directories>
<directories realtime="yes" check_all="yes" report_changes="yes">C:\Users\*\Pictures</directories>
<directories realtime="yes" check_all="yes" report_changes="yes">C:\Users\*\Saved Games</directories>
<directories realtime="yes" check_all="yes" report_changes="yes">C:\Users\*\Searches</directories>
<directories realtime="yes" check_all="yes" report_changes="yes">C:\Users\*\Videos</directories>
<directories realtime="yes" check_all="yes" report_changes="yes" check_sum="yes">C:\Users\*\Downloads</directories>
<!-- Directories to check schedule -->
```

Hình 3.5. Cấu hình giám sát File



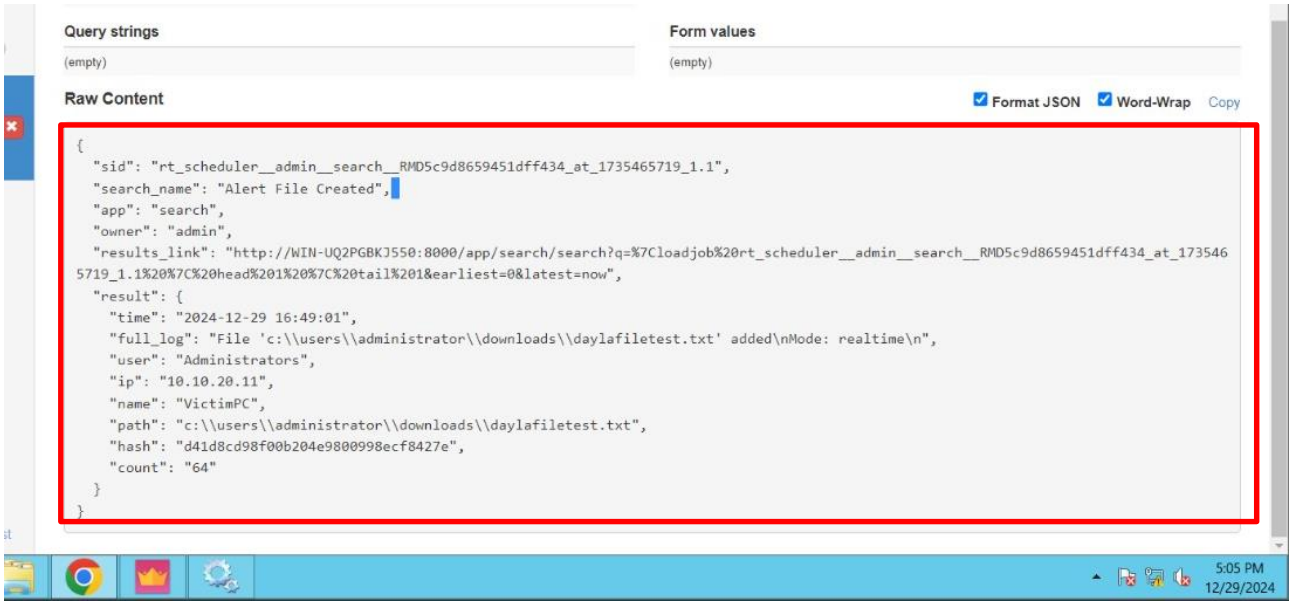
Hình 3.6. Log ghi nhận từ Splunk

Bước 3: Nhận thấy rằng dữ liệu gửi về Splunk có chứa mã hash SHA256, MD5,… từ đó có thể là cơ sở để đội SOC tạo ra một service với nhiệm vụ kiểm tra những mã này trên API của VirusTotal và với những tệp tin có đánh giá là mã độc thì sẽ gửi cảnh báo về Telegram. Sau đó, thông qua chức năng sử dụng Webhook khi có sự kiện xảy ra của Splunk ta sẽ gọi tới service đó để thực thi công việc.



Hình 3.7. Lưu đồ thuật toán của service CheckFile

Như có thể thấy trên lưu đồ, chương trình sẽ thực hiện lấy mã hash từ SIEM gửi về sau đó sẽ kiểm tra với VirusTotal thông qua API, sau đó nếu có đánh giá (có điểm) thì sẽ lập tức gửi cảnh báo về. Tuy nhiên trước khi thực hiện viết code, ta cần xem xét mẫu dữ liệu gửi từ Webhook sau khi có sự kiện xảy ra là gì.



Hình 3.8. Mẫu dữ liệu gửi từ SIEM nếu có sự kiện

Với mẫu dữ liệu trên, đội SOC thực hiện viết lên một chương trình thu thập dữ liệu theo mẫu gửi từ SIEM và thực hiện so sánh.

```
@RestController
public class FileCheckController {
    @Async
    @PostMapping("/Check-File")
    public void checkFile(@RequestBody String json) throws Exception {
        //TODO: process POST request
        ObjectMapper mapper = new ObjectMapper();
        JsonNode rootNode = mapper.readTree(json);
        JsonNode resultNode = rootNode.path(fieldName:"result");
        JsonNode hash = resultNode.path(fieldName:"hash");
        JsonNode path = resultNode.path(fieldName:"path");
        JsonNode user = resultNode.path(fieldName:"user");

        FileCheckService check = new FileCheckService();
        check.checkFileByVirusTotal(hash.asText(), path.asText(), user.asText());

    }

}
```

Hình 3.9. Thực hiện tạo Controller để lấy dữ liệu khi SIEM gọi tới

Ngoài việc lấy dữ liệu từ SIEM để so sánh, đội SOC cũng cần biết được quy tắc gửi dữ liệu tới API của VirusTotal, cũng như mẫu dữ liệu trả về từ API để có thể phục vụ việc trích xuất dữ liệu và gửi cảnh báo lên Telegram

```java
@Service
public class FileCheckService {
    private static final String apiUrl = "https://www.virustotal.com/api/v3/files/";
    private static final String apiKey = "da4b2918878e460916ff948f3fcd4a9cab86477dd600621950568d24f1c0764d";

    public void checkFileByVirusTotal(String hashFromSiem, String pathFromSiem, String userFromSiem) throws Exception {
        //Tạo rest template để gửi request
        RestTemplate restTemplate = new RestTemplate();
        String url = apiUrl + hashFromSiem;

        //Tạo header
        HttpHeaders headers = new HttpHeaders();
        headers.setContentType(MediaType.APPLICATION_JSON);
        headers.add(headerName:"Accept", headerValue:"application/json");
        headers.add(headerName:"X-Apikey", headerValue:"da4b2918878e460916ff948f3fcd4a9cab86477dd600621950568d24f1c0764d");

        //Gửi request và nhận phản hồi
        HttpEntity<String> entity = new HttpEntity<>(headers);
        try {
            ResponseEntity<String> response = restTemplate.exchange(url, HttpMethod.GET, entity, responseType:String.class);
```

Hình 3.10. Thực hiện gán headers và gửi request lên API của VirusTotal

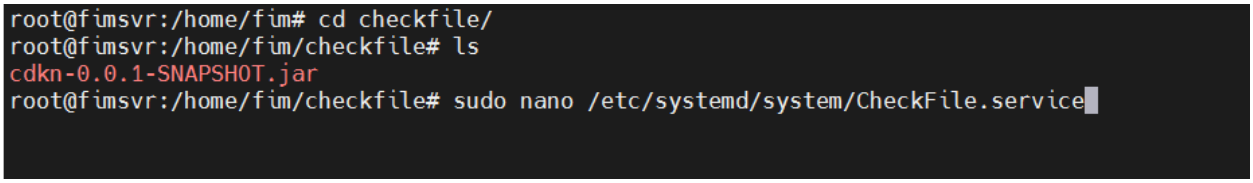
```java
if (response.getStatusCode() == HttpStatus.OK) {
    //Parse response JSON
    ObjectMapper mapper = new ObjectMapper();
    JsonNode rootNode = mapper.readTree(response.getBody());
    JsonNode lastAnalysisStats = rootNode.path(fieldName:"data")
                                         .path(fieldName:"attributes")
                                         .path(fieldName:"last_analysis_stats");
    int malicious = lastAnalysisStats.path(fieldName:"malicious").asInt();
    int score = malicious;

    //Cảnh báo nếu score lớn hơn 1, tức là có trên 1 bên đánh giá của virus total đánh giá đây là độc hại
    if (score > 1) {
        String message = "<b>☢ SPLUNK ALERT MALICIOUS FILE ☢</b>\n\n"
        + "Detect a malicious file has been created" + "\n\n"
        + "<b>PATH:</b> " + pathFromSiem + "\n"
        + "<b>HASH:</b> " + hashFromSiem + "\n"
        + "<b>USER:</b> " + userFromSiem +"\n"
        + "<b>SCORE:</b> " + score + "\n\n"
        + "Please check!";
        sendMessage(message);
    }
```

Hình 3.11. Điều kiện xảy ra nếu có kết quả trả về

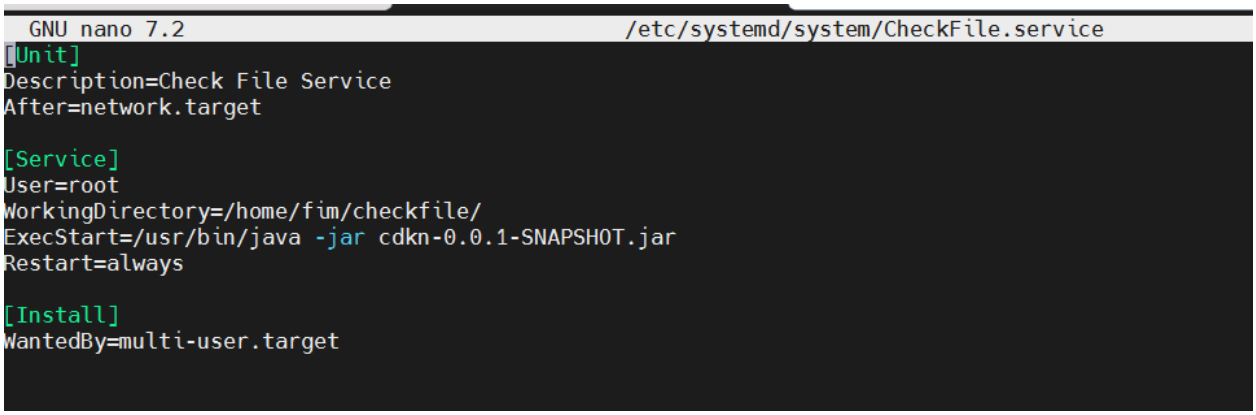
```java
public static void sendMessage(String message) throws Exception {
    String botToken = "7344529813:AAF03-IYaC1nnwwbnOFUx1BY94DiABrygbw";
    String chatID = "-4526958985";
    String teleApiUrl = "https://api.telegram.org/bot" + botToken + "/sendMessage";

    RestTemplate restTemplate = new RestTemplate();

    // Tạo body request dưới dạng JSON
    Map<String, String> requestBody = new HashMap<>();
    requestBody.put(key:"chat_id", chatID);
    requestBody.put(key:"text", message);
    requestBody.put(key:"parse_mode", value:"HTML"); // cho mark down để viết đậm
    // Gửi request POST lên api của tele
    restTemplate.postForObject(teleApiUrl, requestBody, responseType:String.class);


}
```

Hình 3.12. Hàm gửi tin nhắn về Telegram

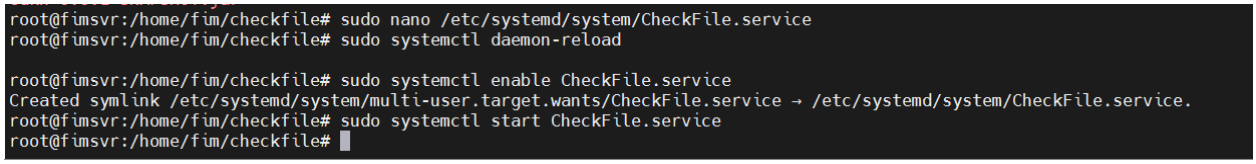
Bước 4: Thực hiện triển khai mã code thành một service trên máy chủ Wazuh để máy chủ Splunk có thể gọi tới khi có sự kiện.



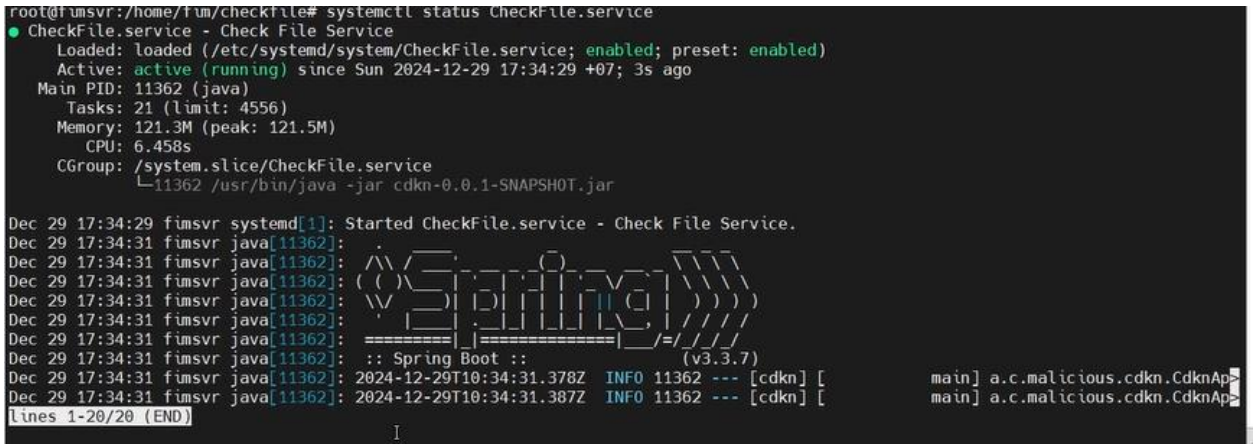
Hình 3.13. Thực hiện chuyển code thành định dạng theo ngôn ngữ



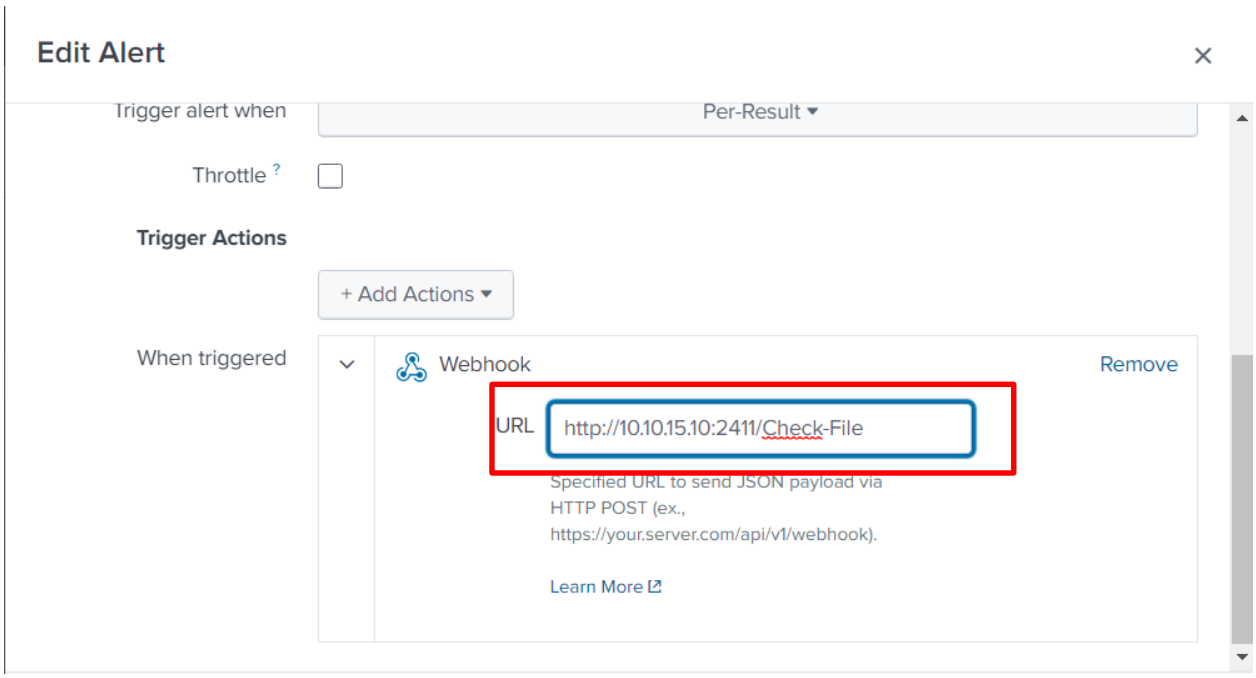
Hình 3.14. Tạo file có đuôi service và gọi tới file code



Hình 3.15. Thực hiện chạy service

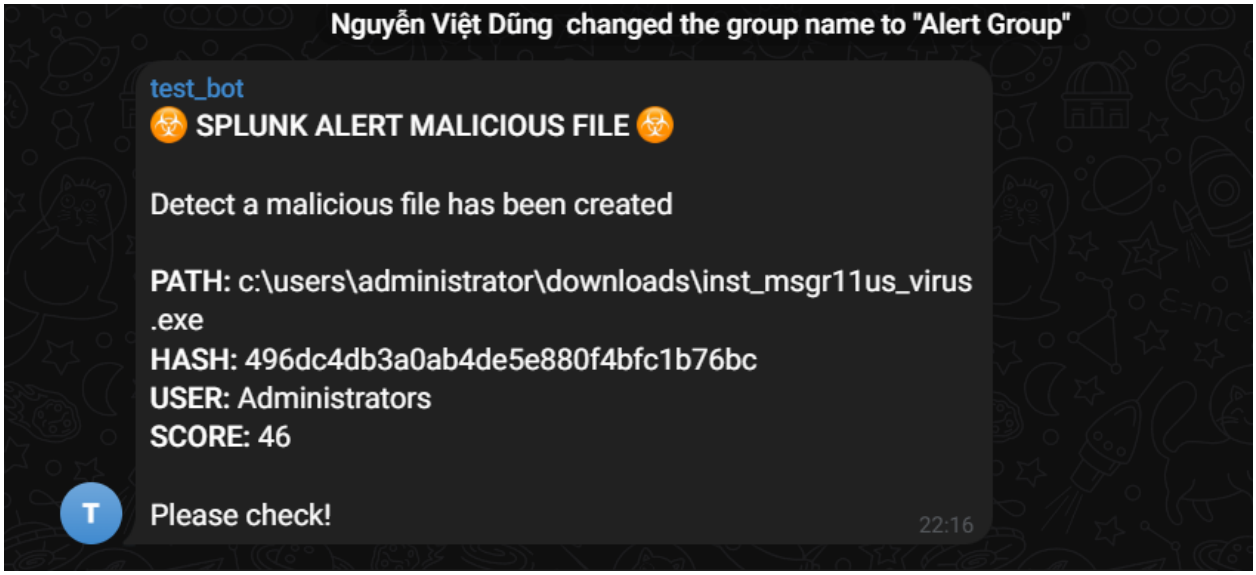


Hình 3.16. Service chạy thành công

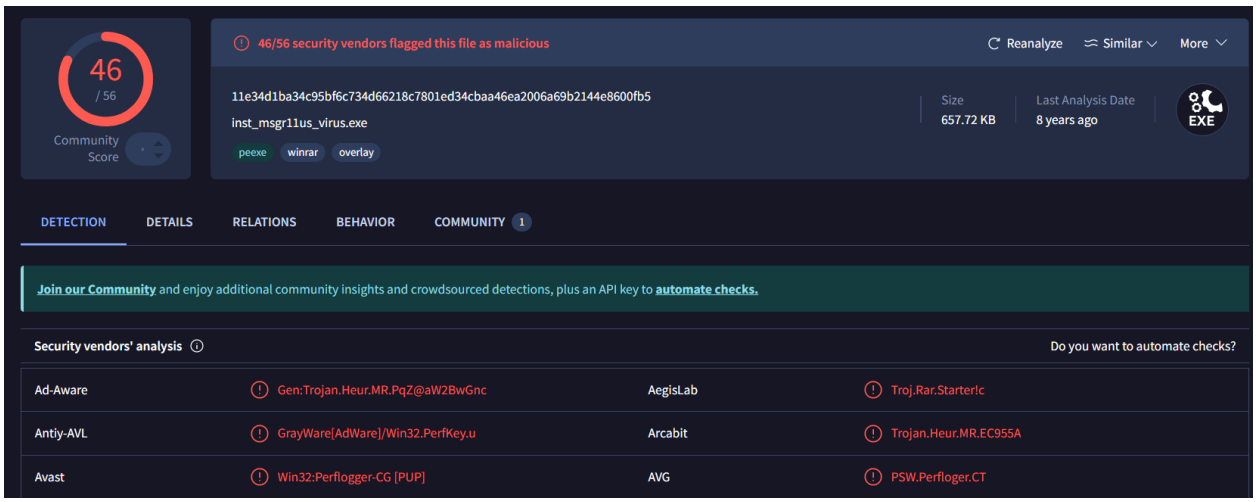


Hình 3.17. Trỏ trigger Webhook tới Server Wazuh trên cổng của Service

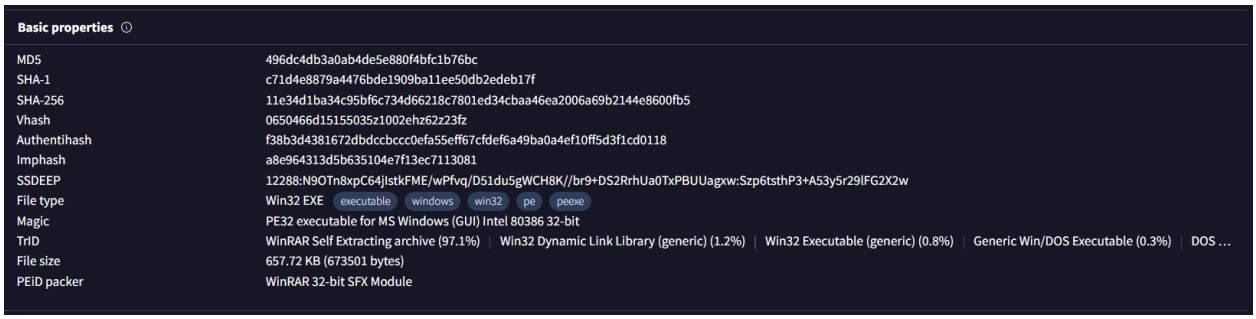
Bước 5: Kiểm tra kết quả khi đã triển khai cảnh báo. Nhận thấy khi có một file sinh ra, nó đã được kiểm tra qua VirusTotal và nếu có số điểm độc hại cao nó sẽ được cảnh báo về Telegram



Hình 3.18. Cảnh báo được gửi về Telegram



Hình 3.19. Dữ liệu khi kiểm tra trên VirusTotal



Hình 3.20. Dữ liệu khi kiểm tra trên VirusTotal

3.3. Kết luận chương III

Trong chương này, quá trình xây dựng và triển khai hệ thống giám sát sự kiện và cảnh báo bất thường cho doanh nghiệp đã được thực hiện thông qua mô hình thử nghiệm cụ thể. Các nội dung chính có thể được tóm tắt như sau:

Mô hình thực nghiệm: Đã được thiết kế để mô phỏng một môi trường thực tế, bao gồm cả hạ tầng Splunk và Wazuh. Mô hình này không chỉ đáp ứng được các yêu cầu giám sát sự kiện mà còn đảm bảo khả năng phát hiện và cảnh báo bất thường về dữ liệu.

Triển khai Splunk và Wazuh: Việc tích hợp hai hệ thống đã được thực hiện thành công, với các bước cấu hình và triển khai cụ thể. Splunk đảm nhận vai trò thu thập và phân tích log, trong khi Wazuh cung cấp chức năng giám sát tính toàn vẹn tệp tin (FIM). Ngoài ra, chuyên đề cũng cho thấy khả năng kết hợp giữa Splunk và các Service tự thiết kế. Sự kết hợp này đã minh chứng khả năng xử lý linh hoạt và hiệu quả trong việc phát hiện các sự kiện an ninh.

Kết quả thử nghiệm: Hệ thống đã phát hiện chính xác khi có một tệp tin độc hại được thêm vào hệ thống. Kết quả thực nghiệm khẳng định rằng mô hình này

không chỉ hoạt động ổn định mà còn đáp ứng được các yêu cầu bảo mật cho doanh nghiệp.

Kết thúc chương này, chuyên đề đã chứng minh được tính khả thi và hiệu quả của việc triển khai hệ thống giám sát sự kiện và cảnh báo bất thường dựa trên Splunk và Wazuh. Đây là bước tiến quan trọng để ứng dụng vào thực tế, góp phần nâng cao khả năng bảo mật thông tin cho doanh nghiệp. Các kết quả và bài học từ quá trình thực nghiệm sẽ được sử dụng làm cơ sở để hoàn thiện các nghiên cứu tiếp theo.

KẾT LUẬN

Đồ án "Nghiên cứu và xây dựng hệ thống giám sát sự kiện và cảnh báo bất thường của dữ liệu cho doanh nghiệp" đã hoàn thành mục tiêu nghiên cứu và triển khai hệ thống giám sát an toàn thông tin trong môi trường doanh nghiệp, với việc áp dụng các công nghệ tiên tiến như SIEM và Wazuh để giám sát và bảo vệ dữ liệu.

Chương 1 đã cung cấp cái nhìn tổng quan về tình hình an toàn thông tin trong các doanh nghiệp hiện nay, làm nổi bật các mối nguy hại và tình trạng bảo mật, đồng thời giới thiệu về khái niệm và sự phát triển của SIEM, cũng như các thành phần và lợi ích của hệ thống này trong việc bảo vệ an toàn thông tin.

Chương 2 đã nghiên cứu sâu về các giải pháp giám sát sự kiện và cảnh báo bất thường, với trọng tâm là hệ thống Splunk và Wazuh. Các tính năng và cách thức hoạt động của Splunk đã được làm rõ, cùng với chức năng giám sát tính toàn vẹn tệp tin của Wazuh. Phân tích này đã chỉ ra rằng sự kết hợp giữa Splunk và Wazuh mang lại một hệ thống giám sát mạnh mẽ, có thể phát hiện và cảnh báo các sự cố bất thường trong dữ liệu, góp phần nâng cao khả năng bảo vệ an toàn thông tin cho doanh nghiệp.

Chương 3 trình bày quá trình thực hiện và triển khai hệ thống giám sát, từ mô hình thực nghiệm đến kết quả triển khai các giải pháp. Các thử nghiệm kết hợp SIEM Splunk với Wazuh đã thành công trong việc phát hiện các bất thường dữ liệu và cung cấp các cảnh báo kịp thời. Những kết quả này đã khẳng định tính hiệu quả và khả năng ứng dụng của hệ thống trong việc nâng cao bảo mật thông tin cho doanh nghiệp.

Tuy nhiên, đồ án cũng nhận thấy rằng vẫn còn nhiều hướng nghiên cứu tiềm năng để phát triển hệ thống giám sát này. Việc tích hợp các công nghệ mới như trí tuệ nhân tạo (AI) và học máy (ML) vào hệ thống giám sát có thể giúp nâng cao khả năng dự đoán và phát hiện các mối đe dọa, đồng thời tối ưu hóa hiệu suất của hệ thống. Hướng nghiên cứu này sẽ tiếp tục được mở rộng trong tương lai, nhằm đáp ứng nhu cầu bảo mật ngày càng cao của các doanh nghiệp trong kỷ nguyên số.

TÀI LIỆU THAM KHẢO

[1] Eskelinen, T. (2022). Development of open-source siem and security operation centre in a company.

[2] Detken, K. O., Rix, T., Kleiner, C., Hellmann, B., & Renners, L. (2015, September). SIEM approach for a higher level of IT security in enterprise networks. In 2015 IEEE 8th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS) (Vol. 1, pp. 322-327). IEEE.

[3] Brochard, M. Defining and measuring the maintainability of Splunk apps.

[4] Nisa, A. K. (2023). *Implementasi File Integrity Monitoring System Menggunakan Wazuh Open Security Platform* (Doctoral dissertation, Universitas Gadjah Mada).

[5] Kurniawan, K., Ekelhart, A., & Kiesling, E. (2021). An att&ck-kg for linking cybersecurity attacks to adversary tactics and techniques.

PHỤ LỤC

FileCheckController.java

```
package alert.checkfile.malicious.cdkn.controller;

import org.springframework.scheduling.annotation.Async;
import org.springframework.web.bind.annotation.RestController;
import com.fasterxml.jackson.databind.JsonNode;
import com.fasterxml.jackson.databind.ObjectMapper;
import alert.checkfile.malicious.cdkn.service.FileCheckService;
import org.springframework.web.bind.annotation.PostMapping;
import org.springframework.web.bind.annotation.RequestBody;

@RestController
public class FileCheckController {
   @Async
   @PostMapping("/Check-File")
   public void checkFile(@RequestBody String json) throws Exception {
      //TODO: process POST request
      ObjectMapper mapper = new ObjectMapper();
      JsonNode rootNode = mapper.readTree(json);
      JsonNode resultNode = rootNode.path("result");
      JsonNode hash = resultNode.path("hash");
      JsonNode path = resultNode.path("path");
      JsonNode user = resultNode.path("user");

      FileCheckService check = new FileCheckService();
      check.checkFileByVirusTotal(hash.asText(), path.asText(), user.asText());

   }

}
```

FileCheckService.java

```
package alert.checkfile.malicious.cdkn.service;

import java.util.HashMap;
import java.util.Map;
import org.springframework.http.HttpEntity;
import org.springframework.http.HttpHeaders;
import org.springframework.http.HttpMethod;
import org.springframework.http.HttpStatus;
import org.springframework.http.MediaType;
import org.springframework.http.ResponseEntity;
import org.springframework.stereotype.Service;
import org.springframework.web.client.RestTemplate;
import com.fasterxml.jackson.databind.JsonNode;
import com.fasterxml.jackson.databind.ObjectMapper;

@Service
public class FileCheckService {
   private static final String apiUrl = "https://www.virustotal.com/api/v3/files/";
   private static final String apiKey =
"da4b2918878e460916ff948f3fcd4a9cab86477dd600621950568d24f1c0764d";

   public void checkFileByVirusTotal(String hashFromSiem, String
pathFromSiem, String userFromSiem) throws Exception {
      //Tạo rest template để gửi request
      RestTemplate restTemplate = new RestTemplate();
      String url = apiUrl + hashFromSiem;

      //Tạo header
      HttpHeaders headers = new HttpHeaders();
      headers.setContentType(MediaType.APPLICATION_JSON);
      headers.add("Accept", "application/json");
      headers.add("X-Apikey", apiKey);
```

```
        //Gửi request và nhận phản hồi
        HttpEntity<String> entity = new HttpEntity<>(headers);
        try {
            ResponseEntity<String> response = restTemplate.exchange(url,
HttpMethod.GET, entity, String.class);

            //Kiểm tra xem kết quả trả về có thành công không
            if (response.getStatusCode() == HttpStatus.OK) {
                //Parse response JSON
                ObjectMapper mapper = new ObjectMapper();
                JsonNode rootNode = mapper.readTree(response.getBody());
                JsonNode lastAnalysisStats = rootNode.path("data")
                                                .path("attributes")
                                                .path("last_analysis_stats");
                int malicious = lastAnalysisStats.path("malicious").asInt();
                int score = malicious;

                //Cảnh báo nếu score lớn hơn 1, tức là có trên 1 bên đánh giá của virus
total đánh giá đây là độc hại
                if (score > 1) {
                    String message = "<b>☣ SPLUNK ALERT MALICIOUS FILE
☣</b>\n\n"
                        + "Detect a malicious file has been created" + "\n\n"
                        + "<b>PATH:</b> " + pathFromSiem + "\n"
                        + "<b>HASH:</b> " + hashFromSiem + "\n"
                        + "<b>USER:</b> " + userFromSiem +"\n"
                        + "<b>SCORE:</b> " + score + "\n\n"
                        + "Please check!";
                    sendMessage(message);
                }
            } else {
                System.out.println("Failed to fetch data from VirusTotal. Status: " +
response.getStatusCode());
            }
```

```
        } catch (Exception e) {
            // TODO: handle exception
            throw new Exception("Error while checking file with VirusTotal: " +
e.getMessage());
        }
    }

    public static void sendMessage(String message) throws Exception {
        String botToken = "7344529813:AAF03-
IYaC1nnwwbnOFUx1BY94DiABrygbw";
        String chatID = "-4526958985";
        String teleApiUrl = "https://api.telegram.org/bot" + botToken +
"/sendMessage";

        RestTemplate restTemplate = new RestTemplate();

        // Tạo body request dưới dạng JSON
        Map<String, String> requestBody = new HashMap<>();
        requestBody.put("chat_id", chatID);
        requestBody.put("text", message);
        requestBody.put("parse_mode", "HTML"); // cho mark down để viết đậm
        // Gửi request POST lên api của tele
        restTemplate.postForObject(teleApiUrl, requestBody, String.class);

    }
}
```