BAN CƠ YẾU CHÍNH PHỦ
HỌC VIỆN KỸ THUẬT MẬT MÃ

CHUYÊN ĐỀ KỸ NGHỆ

NGHIÊN CỨU GIẢI PHÁP PHÁT HIỆN VÀ PHẢN HỒI MỞ RỘNG EXTENDED DETECTION AND RESPONSE (XDR)

Ngành: An toàn thông tin

Sinh viên thực hiện:

Nguyễn Trường Giang - MSSV: AT170414

Đỗ Hoài Nam - MSSV: AT170636

Người hướng dẫn:

TS. Hoàng Đức Thọ

Khoa An toàn thông tin – Học viện kỹ thuật mật mã

Hà Nội, 2023

BAN CƠ YẾU CHÍNH PHỦ

HỌC VIỆN KỸ THUẬT MẬT MÃ

CHUYÊN ĐỀ KỸ NGHỆ

NGHIÊN CỨU GIẢI PHÁP PHÁT HIỆN VÀ PHẢN HỒI MỞ RỘNG EXTENDED DETECTION AND RESPONSE (XDR)

Ngành: An toàn thông tin

Sinh viên thực hiện:

Nguyễn Trường Giang - MSSV: AT170414

Đỗ Hoài Nam - MSSV: AT170636

Người hướng dẫn:

TS. Hoàng Đức Thọ

Khoa An toàn thông tin – Học viện kỹ thuật mật mã

Hà Nội, 2023

MỤC LỤC

MỤC LỤCs..... 3

DANH MỤC HÌNH VẼ..... 5

LỜI CẢM ƠN..... 7

LỜI NÓI ĐẦU..... 8

CHƯƠNG 1: TỔNG QUAN VỀ AN TOÀN THÔNG TIN..... 9

1.1. Các khái niệm cơ bản trong an toàn thông tin..... 9

1.2. Thực trạng an toàn thông tin tại Việt Nam..... 12

1.3. Xu hướng an toàn thông tin của các doanh nghiệp tại Việt Nam..... 12

1.4. Các mối đe dọa và kỹ thuật tấn công mạng phổ biến hiện nay..... 14

1.5. Các giải pháp phòng chống tấn công mạng..... 16

1.5.1. Chống mã độc và thư rác (Antivirus, Antispam)..... 16

1.5.2. Tường lửa (Firewall)..... 18

1.5.3. Phát hiện và ngăn chặn xâm nhập (IDPS)..... 19

1.5.4. Hệ thống bẫy (Honeypot)..... 21

1.5.5. Mạng riêng ảo VPN..... 22

1.5.6. Chống rò rỉ dữ liệu (DLP)..... 23

Kết luận chương 1..... 24

CHƯƠNG 2. CÁC GIẢI PHÁP GIÁM SÁT VÀ ỨNG PHÓ SỰ CỐ TRONG HỆ THỐNG MẠNG MÁY TÍNH..... 25

2.1 Các giải pháp giám sát và ứng phó sự cố trong hệ tống mạng máy tính..... 25

2.1.1 Security orchestration, automation and response (SOAR)..... 26

2.1.2. Security information and event management (SIEM)..... 26

2.1.3. Endpoint Detection and Response (EDR)..... 27

2.1.4. Extended Dectecion and Response (XDR)..... 28

2.2. Các thành phần, kiến trúc và cách thức hoạt động của XDR..... 30

2.2.1. Thành phần của XDR 30

2.2.2. Cách thức hoạt động và kiến trúc của XDR 32

2.3. Bộ công cụ Opensource Wazuh, TheHive, Cortex XDR. 36

2.3.1. Wazuh – Nền tảng bảo mật mã nguồn mở 36

2.3.3. Cortex XDR – Giải pháp phát hiện và phản hồi mở rộng 46

Kết luận chương 2: 49

CHƯƠNG 3. TRIỂN KHAI VÀ THỰC NGHIỆM HỆ THỐNG 50

3.1. Mô hình hệ thống giám sát 50

3.2. Kịch bản thử nghiệm 51

3.3. Đánh giá kết quả 66

Kết luận chương 3 66

KẾT LUẬN CHUYÊN ĐỀ 67

TÀI LIỆU THAM KHẢO 68

PHỤ LỤC 70

DANH MỤC HÌNH VẼ

Hình 1.1. Các biện pháp bảo vệ an toàn thông tin 11

Hình 1.2. Minh họa việc chặn gói tin tộc hại của tưởng lửa 18

Hình 2.1. SIM, SEM và SIEM 27

Hình 2.2. Các chức năng chính của EDR..... 28

Hình 2.3. Cortex XDR..... 30

Hình 2.4. Cách thức NDR hoạt động 31

Hình 2.5. Cách CSPM hoạt động 32

Hình 2.6. Kiến trúc của XDR 33

Hình 2.7. Cortex Data Lake..... 35

Hình 2.8. Cách Wazuh được xây dựng..... 36

Hình 2.9. Sơ đồ liên kết các module của Wazuh Agent 37

Hình 2.10. Sơ đồ cấu hình các cluster trong Wazuh Indexer 40

Hình 2.11. Wazuh Architecture 41

Hình 2.12. Giao diện bảng thống kê của Wazuh dashboard 43

Hình 2.13. Giao diện trang thống kê của TheHive 44

Hình 2.14. Cortex XDR architecture..... 48

Hình 3.1. Sơ đồ tổng quát hệ thống..... 50

Bảng 3.1: Yêu cầu cầu hình của hệ thống giám sát 50

Quá trình cài đặt nằm trong phần phụ lục 50

Hình 3.2. Các thành phần của Silver C2 Framework 52

Hình 3.3. Mô hình thử kiệm Sliver C2 53

Hình 3.4. Khởi chạy Sliver C2 Framework 61

Hình 3.5. Tạo listener trong Sliver 61

Hình 3.6. Tạo payload để chạy trên máy nạn nhân..... 62

Hình 3.7. Kết quả sau khi thực thi file trên máy user 62

Hình 3.8. Liệt kê các sessions hiện có 62
Hình 3.9. Sử dụng phiên kết nối đến windows 10 62
Hình 3.10. Tạo reverse shell 63
Hình 3.11. PID trên Windows 10 63
Hình 3.12. Inject vào tiến trình TextInputHost.exe 63
Hình 3.13. Kết quả trên Wazuh siem 64
Hình 3.14. Alerts được đẩy về TheHive 64
Hình 3.15. Tạo incident từ những alert được đẩy về 65
Hình 3.16. Analyzer VirusTotal_GetReport_3_1 65
Hình 3.17. Giao diện note incident trên TheHive 65
Hình 3.18. Jobs history của Cortex XDR 66

LỜI CẢM ƠN

Trong quá trình thực hiện đề tài chuyên đề kỹ nghệ này này, nhóm em đã nhận được sự giúp đỡ tận tình của cán bộ hướng dẫn là TS. Hoàng Đức Thọ - Giảng viên Khoa An toàn thông tin - Học viện Kỹ thuật Mật mã đã tận tình hướng dẫn và hỗ trợ nhóm em trong việc lựa chọn đề tài, thực hiện triển khai và hoàn thiện báo cáo chuyên đề của mình.

Trong quá trình hoàn thiện đề tài, nhóm em đã cố gắng tìm những giải pháp thật tốt. Song với kiến thức và khả năng còn hạn chế, báo cáo của nhóm em không tránh khỏi những thiếu sót cần phải cải tiến thêm để hoàn thiện hơn. Nhóm em kính mong nhận được những góp ý của các thầy cô và những người quan tâm đến báo cáo này.

Nhóm em xin chân thành cảm ơn!.

SINH VIÊN THỰC HIỆN CHUYÊN ĐỀ

Nguyễn Trường Giang

Đỗ Hoài Nam

LỜI NÓI ĐẦU

Hiện nay, mạng máy tính đóng vai trò quan trọng trong cuộc sống cá nhân và công việc của chúng ta. Chúng tạo điều kiện cho việc giao tiếp, cho phép truy cập vào tài nguyên quý giá và hỗ trợ đa dạng các hoạt động kinh doanh. Tuy nhiên, sự phụ thuộc vào mạng máy tính cũng mang đến những rủi ro bảo mật đáng kể. Các mối đe dọa liên tục phát triển các phương pháp mới và tinh vi để khai thác các lỗ hổng và truy cập trái phép vào dữ liệu và hệ thống nhạy cảm. Do đó, các tổ chức và cá nhân phải sử dụng các giải pháp bảo vệ mạnh mẽ để bảo vệ tài sản số của mình và đảm bảo tính toàn vẹn, bảo mật và khả dụng của mạng.

Để đáp ứng nhu cầu bảo vệ an toàn thông tin cho các thiết bị trong hệ thống mạng máy tính, nhóm em xin được đề xuất giải pháp XDR. Giải pháp XDR – là giải pháp mở rộng khả năng của các giải pháp truyền thống như Endpoint Detection and Response (EDR) và Network Detection and Response (NDR) để cung cấp một tầm nhìn toàn diện hơn về các mối đe dọa.

Mục tiêu của việc nghiên cứu thực hiện chuyên đề kỹ nghệ với đề tài "Nghiên cứu giải pháp phát hiện và phản hồi mở rộng Extended Detection and Response (XDR)" là nhằm xây dựng giải pháp giám sát các sự kiện an toàn thông tin nhằm giám sát, bảo vệ hạ tầng của doanh nghiệp từ các phần mềm mã nguồn mở.

Đề tài chuyên đề có cấu trúc như sau:

Chương 1: Tổng quan về an toàn thông tin

Chương 2: Các giải pháp giám sát và ứng phó sự cố trong hệ thống mạng máy tính

Chương 3: Triển khai và thục nghiệm hệ thống

SINH VIÊN THỰC HIỆN CHUYÊN ĐỀ

Nguyễn Trường Giang

Đỗ Hoài Nam

CHƯƠNG 1: TỔNG QUAN VỀ AN TOÀN THÔNG TIN

1.1. Các khái niệm cơ bản trong an toàn thông tin

Trong suốt chiều dài lịch sử tồn tại và phát triển của mình, con người sử dụng thông tin như một công cụ để trao đổi và truyền đạt kiến thức. Trong kỉ nguyên bùng nổ thông tin hiện nay, thông tin dã trở thành một trong những nhu cầu, phương tiện sản xuất sống còn của con người. Tất cả các cá nhân hay doanh nghiệp đều sử dụng thông tin – Ví dụ, thông tin cá nhân, thông tin khách hàng,…Nếu thông tin đó bị xâm phạm một cách trái phép bằng một cách nào đấy có thể khiến cho hoạt động của doanh nghiệp bị ảnh hưởng, thập chí sụp đổ. Việc bảo vệ những thông tin ấy khỏi những sự truy cập bất hợp pháp được gọi là An toàn thông tin (Information Security).

An toàn thông tin – Information Security, được định nghĩa là việc bảo vệ thông tin và các hệ thống thông tin tránh khỏi việc bị truy nhập, sử dụng, tiết lộ, gián đoạn, sửa đổi hoặc phá hoại trái phép. Hiểu một cách nôm na an toàn thông tin là quá trình phát hiện và ngăn chặn mọi hành vi sử dụng trái phép máy tính máy tính/ điện thoại hoặc thiết bị chứa thông tin. Nó liên quan đến quá trình bảo vệ chống lại những kẻ xâm phạm sử dụng tài nguyên máy tính cá nhân hoặc văn phòng của bạn với mục đích xấu hoặc vì lợi ích riêng bất hợp pháp, hoặc thậm chí do vô tình. [1]

Tính toàn vẹn, tính bí mật và tính khả dụng là 3 tính chất cơ sở cốt lõi của an toàn thông tin. Trong đó:

- **Tính bí mật - Confidentiality**: là tính chất đảm bảo thông tin chỉ cung cấp cho những người có thẩm quyền. Điều này bao gồm việc đảm bảo những người không có thẩm quyền sẽ không có khả năng tiếp cận với các dữ liệu cá nhân hoặc bí mật (tính bí mật) và người sở hữu dữ liệu riêng tư có toàn quyền lưu trữ, sử dụng, và cấp phép cho người khác được tiếp cận với dữ liệu đó (tính riêng tư hoặc liên quan đến vấn đề bản quyền). Dữ liệu hoặc hệ thống bị mất tính bí mật khi nó bị tiếp cận hoặc tiết lộ trái phép.
- **Tính toàn vẹn - Integrity** là tính chất đảm bảo thông tin không bị thay đổi một cách trái phép hoặc thay đổi không như ý muốn. Bảo vệ thông tin chống lại việc sửa đổi hoặc phá hủy trái phép hoặc vô ý, bao gồm cả việc đảm bảo tính xác thực và chống chối bỏ của thông tin. Mất tính toàn vẹn xảy ra khi

thông tin bị sửa đổi hoặc phá hủy trái phép hoặc sai lệch do lỗi đường truyền. Điều này bao gồm việc đảm bảo dữ liệu không bị thay đổi và đảm bảo cả về nguồn gốc của thông tin. Tính toàn vẹn cũng bao gồm việc đảm bảo rằng một hệ thống thực hiện đúng và đầy đủ các chức năng được thiết kế mà không có sự cố ý hoặc vô tình thao túng trái phép hệ thống.

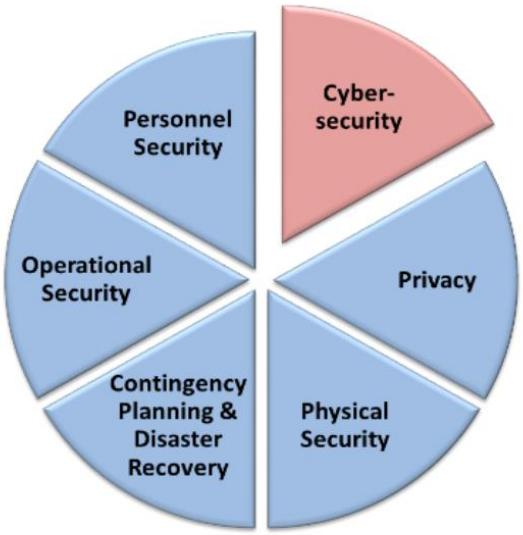
- **Tính khả dụng - Availability** đảm bảo khả năng truy cập thông tin, tính năng của hệ thống thông tin mỗi khi người dùng hợp lệ có nhu cầu.

Ba tính chất bí mật, toàn vẹn và khả dụng của thông tin được gọi là tam giác CIA (Confidentiality - Integrity - Availability) và là 3 tính chất cốt lõi quan trọng nhất của an toàn thông tin. Mọi vấn đề liên quan đến việc đảm bảo an toàn thông tin đều xoay quanh việc đảm bảo 3 tính chất này của thông tin.

Với việc càng ngày càng nhiều thông tin được số hóa, lưu trữ, xử lý và trảo đổi trực tuyến, khái niệm **An ninh mạng** (Cybersecurity) đã ra đời và trở thành một yếu tố chủ chốt của an toàn thông tin.

An ninh mạng – **Cybersecurity:** được định nghĩa là phương pháp bảo vệ an toàn cho máy tính, mạng, ứng dụng phần mềm, hệ thống quan trọng và dữ liệu khỏi các mối đe dọa kỹ thuật số tiềm ẩn. Các tổ chức chịu trách nhiệm bảo mật dữ liệu để duy trì lòng tin của khách hàng cũng như đáp ứng việc tuân thủ quy định. Họ sử dụng các biện pháp và công cụ an ninh mạng để bảo vệ dữ liệu nhạy cảm khỏi bị truy cập trái phép cũng như ngăn chặn gián đoạn trong hoạt động kinh doanh gây ra bởi hoạt động mạng ngoài ý muốn. Các tổ chức triển khai an ninh mạng bằng cách hợp lý hóa công tác phòng vệ kỹ thuật số giữa con người, quy trình và công nghệ. [2]

Là một phần của an toàn thông tin, an ninh mạng hoạt động kết hợp với nhiều kiểu bảo mật khác, một trong số đó được biểu diễn trong hình 1.1. Tổng thể, những thành phần giúp đảm bảo an toàn thông tin này cung cấp phương án phòng thủ trước nhiều mối đe dọa tới dữ liệu thông tin của tổ chức.



Hình 1.1. Các biện pháp bảo vệ an toàn thông tin

• **An ninh vật lý (Physical Security)** – Biện pháp bảo vệ vật lý của tài sản như ổ khóa, hàng rào,….

• **An ninh nhân sự (Personnel Security)** – Việc đánh giá hành vi, tính chính trực, phán đoán, lòng trung thành, độ tin cậy và sự ổn định của các cá nhân đối với các nhiệm vụ và trách nhiệm đòi hỏi sự đáng tin cậy.

• **Phương án dự phòng và phục hồi sau thảm họa (Contingency Planning and Disaster Recovery)** – phương án khôi phục hoạt động bình thường của doanh nghiệp sau khi sự cố hay còn được biết đến là kế hoạch kinh doanh liên tục (Business Continuity Planning)

• **An ninh vận hành (Operational Security)** – Bảo vệ các kế hoạch kinh doanh, quy trình vận hành hằng ngày.

• **Riêng tư (Privacy)** – Bảo vệ các thông tin cá nhân

Chỉ cần thiếu một trong những thành phần này sẽ làm giảm độ hiệu quả các thành phần còn lại. Ví dụ, phương án bảo vệ an ninh vật tốt nhưng cũng không có ý nghĩa nếu nhân sự mà bạn tuyển có ý định làm hại đến công việc của tổ chức (an ninh nhân sự kém).

1.2. Thực trạng an toàn thông tin tại Việt Nam

Trong kỷ nguyên bùng nổ thông tin hiện nay, những hành vi tấn công của tội phạm mạng là vô cùng tinh vi và phức tạp, cùng với sự phát triển của AI, việc tội phạm mạng tạo ra các phầm mềm độc hại cũng trở nên dễ dàng hơn. Việc bảo đảm an toàn thông tin của dữ liệu trên các nền tảng số, thông tin cá nhân của người dân trong bối cảnh ứng dụng trí tuệ nhân tạo (AI) như: Chat GPT, Deepfake làm tăng cường các hình thức tội phạm mạng, lừa đảo, tấn công mạng, đang là một thách thức lớn đối với cộng đồng. “Triển khai một công nghệ, dịch vụ mới trên không gian mạng, bên cạnh những hiệu quả và tiện ích, cần cảnh giác khả năng công nghệ bị khai thác để lừa đảo”.

Với sự trợ giúp của Chat GPT, tội phạm mạng có thể tạo ra một phần mềm đánh cắp dữ liệu tinh vi, phần mềm này là hoàn toàn mới và có thể vượt qua giám sát của các ứng dụng chống mã độc phổ biến hiện nay. Một điển hình khác là tạo đoạn hội thoại, đoạn phim giả mạo người thân để đánh lừa các nạn nhân. Thủ đoạn này không khó khi ứng dụng công nghệ mới và mang lại lợi nhuận kinh tế rất cao cho tội phạm mạng, trong khi nhiều người dùng chưa có nhận thức đúng và đủ về bảo mật an toàn thông tin.

Theo báo cáo, trong bảy tháng năm 2023, Trung tâm Giám sát an toàn không gian mạng quốc gia (Cục An toàn thông tin) ghi nhận, có 9.519 cuộc tấn công mạng tại Việt Nam gây ra sự cố vào các hệ thống thông tin. Trong đó, riêng tháng 7/2023 là 988 cuộc; ngăn chặn 926 website lừa đảo; trong đó, nhiều trang giả mạo các ngân hàng, tổ chức tài chính…

Trong tháng 7/2023, hệ thống kỹ thuật của Trung tâm Giám sát an toàn không gian mạng quốc gia ghi nhận 56.373 điểm yếu, lỗ hổng an toàn thông tin tại các hệ thống thông tin của các cơ quan, tổ chức nhà nước. Số lượng điểm yếu, lỗ hổng an toàn thông tin tại các hệ thống thông tin của các cơ quan, tổ chức là rất lớn, một vài lỗ hổng bị các nhóm APT (tấn công có chủ đích) khai thác.[3]

1.3. Xu hướng an toàn thông tin của các doanh nghiệp tại Việt Nam

Theo Cisco, chỉ 17% tổ chức ở Việt Nam có chỉ số sẵn sàng cần thiết ở mức "trưởng thành" để có thể chống lại các rủi ro an ninh mạng hiện đại ngày nay. Chỉ số đã được phát triển trong bối cảnh hậu COVID-19, thế giới hỗn hợp, nơi mà người dùng và dữ liệu phải được bảo mật ở bất cứ nơi đâu. Báo cáo nêu bật những lĩnh vực mà doanh nghiệp đang

hoạt động tốt và chỉ ra nguy cơ những lỗ hổng an ninh mạng sẽ ngày càng lớn nếu các nhà lãnh đạo về bảo mật và doanh nghiệp toàn cầu không hành động.

Các tổ chức đã chuyển từ mô hình hoạt động tĩnh (nơi mọi người vận hành từ các thiết bị đơn lẻ từ một địa điểm, kết nối với mạng tĩnh) sang một thế giới hỗn hợp trong đó họ hoạt động từ nhiều thiết bị ở nhiều địa điểm, kết nối với nhiều mạng, truy cập các ứng dụng trên đám mây khi đang di chuyển, đồng thời tạo ra lượng dữ liệu khổng lồ. Điều này đã vô tình tạo ra những thách thức an ninh mạng mới cho các doanh nghiệp.

Bản báo cáo với tiêu đề "Chỉ số sẵn sàng an ninh mạng của Cisco: Khả năng phục hồi trong một thế giới hỗn hợp" đo lường mức độ sẵn sàng của các doanh nghiệp trong việc duy trì khả năng phục hồi an ninh mạng trước các mối đe dọa hiện đại. Các biện pháp bao gồm 5 lĩnh vực cốt lõi tạo thành cơ sở cho các biện pháp phòng vệ bắt buộc: danh tính, thiết bị, mạng, khối lượng công việc ứng dụng và dữ liệu, đồng thời bao gồm 19 giải pháp khác nhau.

Cuộc khảo sát nghiên cứu giấu mặt được thực hiện bởi một bên thứ ba yêu cầu 6.700 nhà lãnh đạo an ninh mạng tư nhân trên 27 thị trường cho biết họ đã thực hiện giải pháp nào và giai đoạn triển khai ra sao. Các công ty sau đó được phân loại thành 4 giai đoạn sẵn sàng tăng dần: Beginner (mới bắt đầu), Formative (hình thành), Progressive (phát triển) và Mature (trưởng thành).

- Mới bắt đầu (Tổng điểm dưới 10): Giai đoạn đầu triển khai giải pháp
- Hình thành (Điểm từ 11 - 44): Mức độ triển khai không nhiều, hoạt động sẵn sàng cho an ninh mạng dưới mức trung bình
- Phát triển (Điểm từ 45 - 75): Mức độ triển khai dày dặn, hoạt động sẵn sàng cho an ninh mạng trên mức trung bình
- Trưởng thành (Điểm từ 76 trở lên): Đạt được các giai đoạn triển khai nâng cao và sẵn sàng để giải quyết các rủi ro bảo mật

Bên cạnh đó, nghiên cứu chỉ ra có 17% doanh nghiệp Việt Nam đang ở giai đoạn trưởng thành, hơn một nửa doanh nghiệp trong số đó ở giai đoạn mới bắt đầu (5%) hoặc giai đoạn hình thành (46%). Mặc dù các tổ chức ở Việt Nam đang hoạt động tốt hơn nhiều so với mức trung bình toàn cầu (15% doanh nghiệp ở giai đoạn trưởng thành), nhưng con số này vẫn còn thấp do có nhiều rủi ro.

Các nhà lãnh đạo doanh nghiệp phải thiết lập cơ sở sẵn sàng trên 05 lĩnh vực bảo mật để xây dựng tổ chức một cách an toàn và linh hoạt. Hành động này đặc biệt quan trọng bởi 93% người trả lời khảo sát có kế hoạch tăng ngân sách bảo mật của họ lên ít nhất 10% trong 12 tháng tới. Bằng cách này, các tổ chức có thể phát huy thế mạnh của mình và ưu tiên những lĩnh vực họ có thể phát triển hơn song song cải thiện khả năng phục hồi của mình. [4]

1.4. Các mối đe dọa và kỹ thuật tấn công mạng phổ biến hiện nay

Các cuộc tấn công mạng có thể nhắm mục tiêu vào nhiều nạn nhân từ người dung cá nhân đến doanh nghiệp hoặc thậm chí là chính phủ. Khi nhắm mục tiêu vào các doanh nghiệp hoặc tổ chức khác, mục tiêu của tin tặc thường là truy cập vào các tài nguyên nhạy cảm và có giá trị của công ty, chẳng hạn như dữ liệu khách hang hoặc chi tiết thanh toán. Dưới đây là 10 kỹ thuật tấn công mạng phổ biến hiện nay:

- **Mã độc – Malware:** Mã độc là những chương trình được tạo ra với mục đích gây hại cho máy tính, mạng hoặc là server. Mã độc là loại hình tấn công phổ biến nhất trong tấn công mạng, phần lớn bởi vì thuật ngữ này bao gồm nhiều tập hợp con như ransomware, trojan, spyware, viruses, worms, keyloggers, bots, cryptojacking, và bất kỳ loại tấn công phần mềm độc hại nào khác tận dụng phần mềm theo cách độc hại.
- **Tấn công từ chối dịch vụ - DoS:** Tấn công từ chối dịch vụ (DoS) là một cuộc tấn công độc hại, có chủ đích, làm tràn ngập mạng với các yêu cầu sai nhằm làm gián đoạn hoạt động của hệ thống. Trong một cuộc tấn công DoS, người dung không thể thực hiện các tác vụ thông thường và cần thiết, chẳng hạn như truy cập email, trang web, tài khoản trực tuyến hoặc các tài nguyên khác do máy tính hawojc mạng bị xâm nhập vận hành. Mặc dù hầu hết các cuộc tấn công DoS không dẫn đến mất dữ liệu nhưng chúng khiến tổ chức tốn thời gian, tiền bạc và các tài nguyên khác để khôi phục các hoạt động kinh doanh quan trọng.
- **Lừa đảo – Phishing:** Phishing là hình thức tấn công mạng sử dụng email, SMS, điện thoại, mạng xã hội và những kỹ thuật kỹ nghệ xã hội để dụ dỗ nạn nhân chia sẻ thông tin nhạy cảm của họ ví dụ như mật khẩu hoặc số tài khoản – hoặc là tải phần mềm độc hại, thứ sẽ cài đặt viruses vào thiết bị điện tử của họ.

- **Giả mạo – Spoofing:** Giả mạo là một kỹ thuật mà qua đó tội phạm mạng cải trang thành một nguồn đã biết hoặc đáng tin cậy. Khi làm như vậy, tin tặc có thể tương tác với mục tiêu và truy cập vào hệ thống hoặc thiết bị của họ với mục tiêu cuối cùng là đánh cắp thông tin, tống tiền hoặc cài đặt phần mềm độc hại hoặc phần mềm có hại khác trên thiết bị.
- **Tấn công dựa trên danh tính – Identity-Based Attacks:** Các cuộc tấn công dựa trên danh tính cực kỳ khó phát hiện. Khi thông tin đăng nhập hợp lệ của người dùng bị xâm phạm và kẻ thù đang giả mạo người dùng đó, thường rất khó phân biệt giữa hành vi điển hình của người dùng và hành vi của tin tặc bằng cách sử dụng các công cụ và biện pháp bảo mật truyền thống.
- **Tiêm mã – Code Injection Attacks:** Các cuộc tấn công tiêm mã bao gồm kẻ tấn công tiêm mã độc vào máy tính hoặc mạng dễ bị tấn công để thay đổi hành động của nó.
- **Tấn công chuỗi cung ứng – Supply Chain Attacks:** Tấn công chuỗi cung ứng là một loại tấn công mạng nhắm vào nhà cung cấp bên thứ ba đáng tin cậy cung cấp dịch vụ hoặc phần mềm quan trọng cho chuỗi cung ứng. Các cuộc tấn công chuỗi cung ứng phần mềm tiêm mã độc vào một ứng dụng để lây nhiễm cho tất cả người dùng ứng dụng, trong khi các cuộc tấn công chuỗi cung ứng phần cứng xâm phạm các thành phần vật lý cho cùng mục đích. Chuỗi cung ứng phần mềm đặc biệt dễ bị tổn thương vì phần mềm hiện đại không được viết từ đầu: thay vào đó, nó bao gồm nhiều thành phần sẵn có, chẳng hạn như API của bên thứ ba, mã nguồn mở và mã độc quyền từ các nhà cung cấp phần mềm.
- **Mối đe dọa nội bộ - Insider Threats:** Các tác nhân nội bộ gây ra mối đe dọa cho một tổ chức có xu hướng độc hại về bản chất. Một số động cơ thúc đẩy bao gồm lợi ích tài chính để đổi lấy việc bán thông tin bí mật trên web đen và/hoặc ép buộc tinh thần bằng cách sử dụng các chiến thuật lừa đảo xã hội, chẳng hạn như các cuộc tấn công lấy cớ hoặc xâm phạm email doanh nghiệp (BEC). Mặt khác, một số tác nhân đe dọa nội bộ không có bản chất độc hại mà thay vào đó lại có bản chất cẩu thả, dẫn đến việc thông tin nội bộ bị xâm phạm từ bên ngoài.
- **Đường hầm DNS – DNS Tunneling:** Đường hầm DNS là một loại tấn công mạng tận dụng các truy vấn và phản hồi của hệ thống tên miền (DNS) để vượt qua các biện pháp bảo mật truyền thống cũng như truyền dữ liệu và mã trong

mạng. Sau khi bị nhiễm, hacker có thể tự do tham gia vào các hoạt động ra lệnh và kiểm soát. Đường hầm này cung cấp cho tin tặc một lộ trình để giải phóng phần mềm độc hại và/hoặc trích xuất dữ liệu, IP hoặc thông tin nhạy cảm khác bằng cách mã hóa từng chút một trong một loạt phản hồi DNS.

- **Tấn công dựa trên IoT – IoT-Based Attacks:** Cuộc tấn công IoT là bất kỳ cuộc tấn công mạng nào nhắm vào thiết bị hoặc mạng Internet of Things (IoT). Sau khi bị xâm nhập, tin tặc có thể chiếm quyền kiểm soát thiết bị, đánh cắp dữ liệu hoặc tham gia vào một nhóm thiết bị bị nhiễm để tạo mạng botnet nhằm khởi động các cuộc tấn công DoS hoặc DDoS. [5]

1.5. Các giải pháp phòng chống tấn công mạng

Các giải pháp công nghệ mà có thể giúp bảo vệ doanh nghiệp trước các mối đe dọa an toàn thông tin có thể kể đến là:

- Chống mã độc và thư rác (Antivirus, Antispam)
- Tường lửa (Firewall)
- Phát hiện và ngăn chặn xâm nhập (IDPS)
- Hệ thống bẫy (Honeypot, Honeynet)
- Mạng riêng ảo (VPN)
- Chống rò rỉ dữ liệu (DLP)
- Giám sát an toàn mạng (SIEM)

1.5.1. Chống mã độc và thư rác (Antivirus, Antispam)

Antivirus là một phần mềm được thiết kế để phát hiện, ngăn chặn và loại bỏ các phần mềm độc hại (malware) trong các thiết bị đầu cuối. Đây là thành phần cơ bản trong bảo vệ an toàn các thiết bị trong mạng máy tính.

Chức năng chính của phần mềm antivirus là quét các file, các chương trình và tổng thể hệ thống để tìm ra các mẫu, các đặc trưng của malware. Những đặc trưng này cơ bản là đặc điểm riêng biệt hoặc đoạn code mà có liên quan đến các kiểu malware. Khi phần mềm antivirus phát hiện một tệp tin hoặc chương trình có dấu hiệu phù hợp, phần mềm antivirus sẽ cảnh báo và thực hiện hành động để cách ly, xóa hoặc làm sạch tệp tin bị nhiễm.

Dưới đây là một số tính năng và khả năng chính thường được tìm thấy trong phần mềm antivirus:

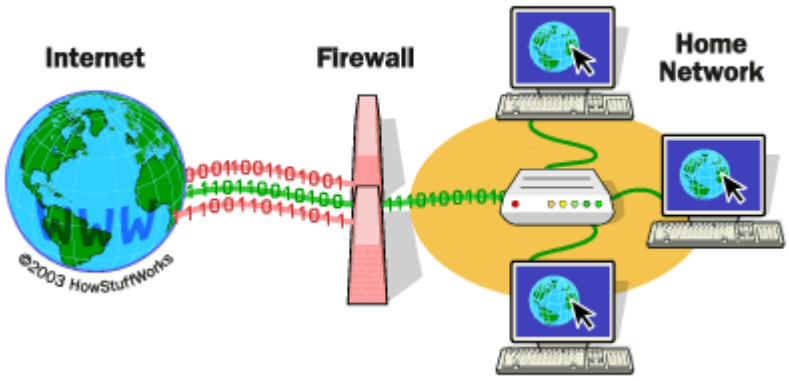
- Quét thời gian thực: Phần mềm antivirus liên tục giám sát các tệp tin và hoạt động trong thời gian thực để phát hiện và chặn malware ngay khi nó xuất hiện. Quét này có thể xảy ra trong quá trình tải xuống tệp tin, cài đặt, tệp đính kèm email và duyệt web.
- Phát hiện dựa trên dấu hiệu: Các chương trình antivirus duy trì một cơ sở dữ liệu chứa các dấu hiệu đã biết về malware. Trong quá trình quét, chúng so sánh các tệp tin và chương trình với cơ sở dữ liệu này để xác định và loại bỏ các mối đe dọa đã biết.
- Phân tích theo cách tiếp cận heuristic: Phần mềm antivirus sử dụng các kỹ thuật heuristic để xác định các hành vi có thể đáng ngờ hoặc độc hại. Nó phân tích mã và hành vi của các tệp tin và chương trình để phát hiện các mối đe dọa chưa biết hoặc zero-day mà không có các dấu hiệu tồn tại.
- Giám sát hành vi: Các chương trình antivirus giám sát hành vi của các tệp tin, quá trình và hoạt động hệ thống để xác định các hành động đáng ngờ có thể chỉ ra sự tồn tại của malware. Điều này bao gồm giám sát các sự thay đổi hệ thống không được ủy quyền, kết nối mạng không mong muốn và các hành vi bất thường khác.
- Cập nhật tự động: Phần mềm antivirus thường cập nhật cơ sở dữ liệu các dấu hiệu malware định kỳ để đối phó với các mối đe dọa mới và đang phát triển. Các cập nhật này cũng bao gồm các bản vá và sửa lỗi để nâng cao hiệu quả của phần mềm và khắc phục các lỗ hổng.
- Cách ly và khắc phục: Khi một tệp tin hoặc chương trình được phát hiện là bị nhiễm, phần mềm antivirus có thể cách ly nó, cô lập khỏi phần còn lại của hệ thống để ngăn chặn thiệt hại tiếp theo. Người dùng sau đó có thể chọn xóa hoặc cố gắng làm sạch tệp tin.
- Quét định kỳ: Phần mềm antivirus cho phép người dùng lên lịch quét hệ thống định kỳ để kiểm tra proactive các mối đe dọa malware. Các quét này có thể được thiết lập để xảy ra vào các thời điểm hoặc khoảng thời gian cụ thể, đảm bảo bảo vệ liên tục.

- Bảo vệ email và web: Nhiều chương trình antivirus bao gồm tính năng quét tệp đính kèm email, tải xuống web và liên kết trang web để phát hiện và chặn malware được lan truyền qua các kênh này.

Trên thị trường hiện đang có đa dạng các nhà cung cấp phần mềm antivirus cả phần mềm miễn phí và phần mềm thương mại. Một số phần mềm antivirus có thể kể đến: Microsoft Defender for Endpoint, ESET PROTECT Advanced, Kaspersky Antivirus, SentinelOne Singularity, Symantec End-user Endpoint Security, McAfee Antivirus.

1.5.2. Tường lửa (Firewall)

Tường lửa (Firewall) là một thiết bị mạng có nhiệm vụ thực thi các chính sách an toàn thông tin dành cho lưu lượng mạng trong hệ thống mạng máy tính. Tường lửa tạo ra một rào chắn giữa các mạng máy tính khác nhau bằng cách đóng vai trò như một trạm kiểm soát mà lưu lượng mạng phải đi qua trước khi lưu lượng mạng đó có thể đi đến mạng máy tính. Tường lửa giúp hạn chế các lưu lượng mạng độc hại tới các thiết bị trong mạng máy tính, ví dụ như một attacker cố gắng thực hiện khai thác các lỗ hổng trong các phần mềm ứng dụng từ Internet.



Hình 1.2. Minh họa việc chặn gói tin tộc hại của tưởng lửa

Tường lửa có thể là thiết bị phần cứng hoặc phần mềm. Tường lửa phần cứng là các thiết bị vật lý thường được cài đặt giữa mạng nội bộ và mạng ngoại vi. Chúng cung cấp một lớp bảo vệ bổ sung bằng cách lọc lưu lượng mạng dựa trên các yếu tố như địa chỉ IP, cổng và giao thức. Trong khi đó, tường lửa phần mềm là các chương trình được cài đặt

trên các thiết bị hoặc server cá nhân. Chúng cung cấp bảo vệ ở mức thiết bị, giám sát và kiểm soát lưu lượng mạng cụ thể cho từng thiết bị.

Tường lửa hoạt động bằng cách kiểm tra các gói dữ liệu di chuyển qua mạng. Chúng so sánh thông tin trong các gói dữ liệu này với một tập hợp quy tắc hoặc chính sách đã được định trước. Nếu gói dữ liệu đáp ứng các tiêu chí xác định, nó được phép đi qua tường lửa và đến điểm đích. Nếu vi phạm bất kỳ quy tắc nào, tường lửa sẽ chặn hoặc loại bỏ gói dữ liệu, ngăn chúng tiếp cận mạng.

Tường lửa cung cấp một số chức năng quan trọng bao gồm:

- Kiểm soát truy cập: Tường lửa có thể hạn chế truy cập vào mạng bằng cách cho phép hoặc chặn địa chỉ IP, cổng hoặc giao thức cụ thể.
- Lọc gói dữ liệu: Chúng kiểm tra từng gói dữ liệu và đưa ra quyết định dựa trên thông tin như địa chỉ IP nguồn và đích, cổng và giao thức.
- Chuyển đổi địa chỉ mạng (NAT): Tường lửa có thể chuyển đổi địa chỉ IP riêng tư thành địa chỉ IP công cộng, cho phép nhiều thiết bị trong mạng riêng tư chia sẻ cùng một địa chỉ IP công cộng.
- Hỗ trợ VPN: Tường lửa thường bao gồm hỗ trợ Mạng Riêng Ảo (VPN), cho phép người dùng từ xa kết nối an toàn với mạng.
- Ngăn chặn xâm nhập: Một số tường lửa tiên tiến có khả năng ngăn chặn xâm nhập, phát hiện và chặn các hoạt động mạng đáng ngờ hoặc độc hại.

Một số tưởng lửa tiêu biểu Cisco ASA, Checkpoint Firewall, Paolo Alto Firewall, Iptable, UFW, Windows Firewall,..

1.5.3. Phát hiện và ngăn chặn xâm nhập (IDPS)

IDPS là viết tắt của Intrusion Detection and Prevention System (tạm dịch: Hệ thống phát hiện và ngăn chặn xâm nhập). Đây là một công nghệ bảo mật mạng được thiết kế để phát hiện và ngăn chặn việc truy cập trái phép, hoạt động độc hại và các mối đe dọa mạng tiềm năng trong một mạng máy tính. IDPS kết hợp cả hệ thống phát hiện xâm nhập (IDS – Intrusion Detection System) và hệ thống ngăn chặn xâm nhập (IPS – Intrusion Prevention System) để cung cấp một cơ chế phòng thủ toàn diện.

Hệ thống Phát hiện xâm nhập (IDS) giám sát lưu lượng mạng, sự kiện hệ thống và hoạt động người 20ung để phát hiện bất kỳ dấu hiệu hành vi độc hại hoặc hoạt động đáng ngờ. Nó phân tích gói tin mạng, nhật ký và các nguồn dữ liệu khác nhau để phát hiện các mẫu hoặc chữ ký liên quan đến các cuộc tấn công đã biết hoặc các sự bất thường có thể chỉ ra một cuộc tấn công đang diễn ra. Khi một hệ thống IDS phát hiện một xâm nhập tiềm ẩn hoặc vi phạm bảo mật, IDS tạo ra cảnh báo hoặc thông báo để thông báo cho quản trị mạng hoặc nhân viên bảo mật.

Trong khi đó, hệ thống Ngăn chặn xâm nhập (IPS) hoạt động một cách chủ động để chặn hoặc ngăn chặn các mối đe dọa được xác định để không gian mạng không bị xâm nhập. Nó vượt xa giai đoạn phát hiện và thực hiện các hành động để ngăn chặn xâm nhập một cách tích cực. IPS có thể tự động loại bỏ, cách ly hoặc chặn lưu lượng mạng hoặc kết nối liên quan đến các mối đe dọa đã biết hoặc các hoạt động độc hại. IPS cũng có thể thực hiện các chính sách bảo mật, phát hiện và ngăn chặn các cuộc tấn công trái phép và bảo vệ khỏi nhiều loại cuộc tấn công, chẳng hạn như tấn công từ chối dịch vụ (DoS), quét cổng hoặc tấn công SQLi,..

Các chức năng chính của một IDPS bao gồm:

- Giám sát và Phân tích: IDPS liên tục giám sát lưu lượng mạng, nhật ký hệ thống và dữ liệu sự kiện để phát hiện các mối đe dọa tiềm ẩn và các sự cố bảo mật.
- Phát hiện Mối đe dọa: Phát hiện và phân tích các mẫu, chữ ký và các sự bất thường trong lưu lượng mạng và hành vi hệ thống để xác định các cuộc tấn công tiềm ẩn hoặc vi phạm bảo mật.
- Cảnh báo và Thông báo: IDPS tạo ra cảnh báo, thông báo hoặc báo động khi phát hiện các hoạt động đáng ngờ, cho phép quản trị mạng thực hiện các biện pháp phù hợp.
- Ngăn chặn và phản ứng: Một IDPS có thể ngăn chặn các mối đe dọa đã biết hoặc lưu lượng mạng đáng ngờ khỏi xâm nhập vào hoặc rời khỏi mạng bằng cách chặn các lưu lượng mạng đó, cung cấp phòng thủ tích cực chống lại các cuộc tấn công.
- Phản ứng sự cố: IDPS đóng vai trò quan trọng trong phản ứng sự cố bằng cách cung cấp thông tin quan trọng về các sự cố bảo mật, giúp đội ngũ bảo mật điều tra và giảm thiểu tác động của một cuộc xâm nhập.

- Phân tích Nhật ký và Báo cáo: IDPS ghi nhật ký và ghi lại các sự kiện bảo mật, cung cấp nguồn thông tin quý giá cho việc phân tích sau sự cố, kiểm tra tuân thủ quy định và mục đích báo cáo.

1.5.4. Hệ thống bẫy (Honeypot)

Honeypot là một kỹ thuật hoặc công cụ bảo mật mạng được thiết kế để thu hút và đánh lừa các kẻ tấn công tiềm năng. Đó là một hệ thống hoặc tài nguyên mạng giả mạo có vẻ như là một mục tiêu hợp lệ, nhưng thực tế, hệ thống honeypot đã được cô lập và theo dõi chuyên sâu bởi các chuyên gia bảo mật. Mục đích của một honeypot là thu thập thông tin về các kẻ tấn công, phương pháp và động cơ của kẻ tấn công, nhằm hiểu rõ hơn và giảm thiểu các mối đe dọa tiềm năng.

Honeypot thường được triển khai như một môi trường được kiểm soát, mô phỏng các hệ thống hoặc dịch vụ thực tế. Hệ thống honeypot được thiết kế có ý đồ để bị tấn công, thu hút kẻ tấn công tương tác với chúng. Honeypot có thể mô phỏng nhiều loại tài nguyên, chẳng hạn như server, ứng dụng hoặc thiết bị mạng, phụ thuộc vào mục tiêu nghiên cứu bảo mật hoặc chiến lược phòng vệ mong muốn.

Khi kẻ tấn công tương tác với một Honeypot, hành vi và hoạt động của họ được ghi lại và phân tích. Điều này bao gồm việc ghi nhận lưu lượng mạng, lưu log lệnh nhập và giám sát bất kỳ cố gắng khai thác lỗ hổng nào. Bằng cách quan sát cẩn thận các hoạt động của kẻ tấn công, các chuyên gia bảo mật có thể thu thập thông tin quý giá về các kỹ thuật, công cụ và ý đồ của họ.

Hệ thống Honeypot có thể phân loại như sau:

- Production Honeypot: Đây là các hệ thống hoặc mạng thực sự được tạo ra mục đích để thu hút kẻ tấn công. Thông thường, chúng được triển khai song song với các hệ thống sản xuất thực tế để phát hiện và đáp ứng các mối đe dọa một cách tích cực.
- Research Honeypot: Các honeypot này được thiết kế cho mục đích nghiên cứu cụ thể. Chúng có thể tập trung vào nghiên cứu các mô hình tấn công cụ thể, phân tích phần mềm độc hại hoặc hiểu rõ ý đồ của kẻ tấn công.
- High-Interaction Honeypot: Những honeypot này cung cấp chức năng và khả năng tương tác rộng, thường mô phỏng các hệ điều hành hoặc dịch vụ đầy đủ. Mục tiêu

của chúng là giữ kẻ tấn công tương tác trong thời gian dài, thu thập thông tin chi tiết hơn về hoạt động của họ.

- Low-Interaction Honeypot: Những honeypot này mô phỏng chức năng và dịch vụ hạn chế, tạo ra bề mặt tấn công nhỏ hơn. Chúng dễ dàng triển khai và bảo trì, nhưng có thể thu hút ít kẻ tấn công kỹ thuật hơn.

Honeypot mang lại nhiều lợi ích trong bảo mật mạng:

- Thông tin về mối đe dọa: Honeypot cung cấp thông tin quý giá về các chiến thuật, kỹ thuật và công cụ mà kẻ tấn công sử dụng. Thông tin này có thể nâng cao thông tin về mối đe dọa và giúp tổ chức cải thiện tổng thể bảo mật.
- Cảnh báo sớm: Honeypot có thể phát hiện và cảnh báo tổ chức về các cuộc tấn công tiềm ẩn ở giai đoạn đầu, cho phép triển khai biện pháp phòng vệ chủ động.
- Đánh lạc hướng: Bằng cách lạc hướng sự chú ý của kẻ tấn công đến honeypot, tổ chức có thể bảo vệ các hệ thống và mạng thực tế khỏi bị tấn công.
- Xác định lỗ hổng: Honeypot có thể giúp xác định các lỗ hổng mới hoặc chưa biết bằng cách thu hút và ghi lại các cuộc tấn công.

1.5.5. Mạng riêng ảo VPN

VPN, viết tắt của Virtual Private Network (tạm dịch: Mạng riêng ảo), là một công nghệ cho phép người 22hem tạo ra một kết nối an toàn và được mã hóa trên mạng công cộng như internet. VPN cho phép người 22hem thiết lập một kết nối mạng riêng từ xa, như khi người 22hem trực tiếp kết nối đến một mạng riêng, ngay cả khi người 22hem truy cập internet thông qua một mạng công cộng hoặc mạng Wi-Fi không an toàn.

Mục đích chính của VPN là nâng cao bảo mật và sự riêng tư bằng cách mã hóa dữ liệu được truyền giữa thiết bị của người 22hem và server hoặc trang web đích. Khi người 22hem kết nối vào VPN, lưu lượng internet của họ được định tuyến qua một đường hầm an toàn, mã hóa dữ liệu và bảo vệ nó khỏi việc truy cập trái phép hoặc chặn bởi các hacker, giám sát chính phủ hoặc các thực thể độc hại khác.

Dưới đây là một số tính năng và lợi ích chính của việc sử dụng VPN:

- Truyền dữ liệu an toàn: VPN mã hóa dữ liệu truyền giữa thiết bị của người 22hem và server VPN, đảm bảo rằng nó không thể dễ dàng bị người khác chặn hoặc truy

cập trái phép. Điều này đặc biệt quan trọng khi sử dụng mạng Wi-Fi công cộng, mà dễ bị nghe trộm và đánh cắp dữ liệu.

- Bảo vệ sự riêng tư: Bằng cách ẩn địa chỉ IP của người 23hem và mã hóa lưu lượng internet của họ, VPN cung cấp một mức độ bảo mật cao hơn. Nó ngăn người cung cấp dịch vụ internet (ISP), các trang web hoặc các thực thể khác từ việc theo dõi và giám sát hoạt động trực tuyến của người 23hem, nâng cao tính nặc danh.
- Vượt qua hạn chế địa lý: VPN cho phép người 23hem vượt qua các hạn chế địa lý được áp đặt bởi các trang web hoặc nền tảng phát trực tuyến. Bằng cách kết nối đến server VPN ở một quốc gia khác, người 23hem có thể truy cập nội dung và dịch vụ mà có thể bị hạn chế hoặc không khả dụng ở vị trí của mình.
- Tăng cường bảo mật cho việc truy cập từ xa: VPN cung cấp một phương pháp an toàn cho người lao động từ xa truy cập vào tài nguyên mạng nội bộ của tổ chức. Bằng cách kết nối vào VPN của tổ chức, nhân viên có thể truyền dữ liệu nhạy cảm và truy cập các nguồn tài nguyên của công ty một cách an toàn, ngay cả khi làm việc từ xa.
- Chia sẻ tập tin P2P: VPN có thể được sử dụng cho việc chia sẻ tập tin peer-to-peer (P2P) an toàn và riêng tư. Bằng cách mã hóa lưu lượng P2P, VPN ngăn ISP hoặc các bên khác theo dõi hoặc giới hạn kết nối dựa trên loại nội dung được chia sẻ.
- Ẩn danh và bảo vệ danh tính: VPN có thể giúp bảo vệ danh tính và hoạt động trực tuyến của người 23hem khỏi việc bị truy tìm lại. Bằng cách ẩn địa chỉ IP và mã hóa dữ liệu của người 23hem, VPN 23hem một lớp bảo vệ và nâng cao mức độ nặc danh.

1.5.6. Chống rò rỉ dữ liệu (DLP)

DLP là viết tắt của Data Loss Prevention (DLP). DLP đề cập đến một tập hợp các công nghệ và phương pháp được thiết kế để ngăn chặn việc tiết lộ hoặc mất mát dữ liệu nhạy cảm trong một tổ chức. Các giải pháp DLP được sử dụng để xác định, giám sát và bảo vệ thông tin nhạy cảm để đảm bảo rằng các thông tin trên không rời khỏi giới hạn của tổ chức.

Mục tiêu chính của DLP là ngăn chặn việc vi phạm dữ liệu, bất kể có ý định hay vô tình, bằng cách giám sát và kiểm soát luồng dữ liệu nhạy cảm qua các kênh và điểm cuối khác nhau. Điều này bao gồm dữ liệu ở trạng thái yên ngủ (dữ liệu đã được lưu trữ), dữ

liệu trong quá trình truyền (dữ liệu đang được truyền) và dữ liệu đang sử dụng (dữ liệu đang được truy cập hoặc xử lý).

Hệ thống DLP thường áp dụng sự kết hợp giữa chính sách, quy tắc và kiểm tra nội dung để xác định và phân loại dữ liệu nhạy cảm. Các hệ thống này có thể phát hiện thông tin nhạy cảm như thông tin cá nhân có thể xác định (PII), dữ liệu tài chính, sở hữu trí tuệ, bí mật thương mại hoặc thông tin bí mật khác dựa trên tiêu chí được xác định trước. Một số phương pháp thông thường được sử dụng trong DLP bao gồm khớp từ khóa, biểu thức chính quy, dấu vân tay dữ liệu và các thuật toán học máy.

Sau khi xác định dữ liệu nhạy cảm, các giải pháp DLP có thể thực hiện các biện pháp bảo mật khác nhau để ngăn chặn mất dữ liệu. Các biện pháp này có thể bao gồm mã hóa, kiểm soát truy cập, chặn hoặc cách ly dữ liệu, thông báo cho quản trị viên hoặc người dùng, hoặc áp dụng kỹ thuật xóa thông tin hoặc che giấu thông tin nhạy cảm.

Các giải pháp DLP có thể được triển khai ở các mức độ khác nhau trong hạ tầng của một tổ chức, bao gồm mức độ mạng, mức độ điểm cuối và mức độ lưu trữ dữ liệu. DLP mức độ mạng giám sát lưu lượng mạng để phát hiện và ngăn chặn rò rỉ dữ liệu qua giao thức mạng, email, ứng dụng web hoặc truyền tệp. DLP mức độ điểm cuối tập trung vào bảo vệ các thiết bị cá nhân như máy tính xách tay, máy tính để bàn hoặc thiết bị di động, đảm bảo rằng dữ liệu được bảo vệ ngay cả khi không kết nối với mạng của tổ chức. DLP mức độ lưu trữ dữ liệu liên quan đến bảo vệ các kho dữ liệu và cơ sở dữ liệu để ngăn chặn việc truy cập trái phép hoặc rò rỉ dữ liệu.

Kết luận chương 1

Chương 1 đã giới thiệu tổng quan về các khái niệm trong an toàn thông tin, thực trạng an toàn thông tin trong năm 2023, và xu hướng an toàn thông tin tại các tổ chức và doang nhiệp tại Việt Nam, cùng với đó cũng đưa a những tính cấp thiết của việc sử dụng các giải pháp giám sát an ninh mạng, xu hướng và những kỹ thuật tấn công phổ biến hiện nay. Từ đó nêu ra ý tưởng của đề tài và đưa ra những giải pháp để giúp bảo đảm an toàn thông tin cho các tổ chức và doanh nghiệp cụ thể có trong chương tiếp theo.

CHƯƠNG 2. CÁC GIẢI PHÁP GIÁM SÁT VÀ ỨNG PHÓ SỰ CỐ TRONG HỆ THỐNG MẠNG MÁY TÍNH

2.1 Các giải pháp giám sát và ứng phó sự cố trong hệ tống mạng máy tính

Hiện nay, các doanh nghiệp thưởng xây dựng những phòng giám sát an ninh mạng (Security Operations Center – SOC), tùy quy mô và điều kiện của tổ chức mà những giải pháp được sử dụng cũng khác nhau, thường sẽ sử dụng những giải pháp như SOAR – Điều phối, tự động hóa và ứng phó bảo mật, SIEM – Giải pháp quản lý và phân tích sự kiện an ninh, EDR – Phát hiện và phản hồi điểm cuối , XDR – Phát hiện và phản hồi mở rộng.

Giải pháp SOAR - điều phối, tự động hóa và phản hồi bảo mật: giúp điều phối thực, hiện và tự động hóa các nhiệm vụ giữa nhiều người và nhiều công cụ trong một nền tảng duy nhất. Điều này cho phép các tổ chức không chỉ phản ứng nhanh chóng với các cuộc tấn công an ninh mạng mà còn quan sát, hiểu và ngăn chặn các sự cố trong tương lai, từ đó cải thiện tình trạng bảo mật tổng thể của họ. Tuy nhiên, việc tích hợp những giải pháp khác vào SOAR khá là phức tạp, them vào đó SOAR không có khả năng đánh giá mức độ trưởng thành của bảo mật ở phạm vi rộng hoặc cạnh tranh hơn cũng như khả năng tích hợp vào chiến lược.

Giải pháp SIEM – Quản lý và phân tích sự kiện an ninh: SIEM cho phép các đơn vị, cơ quan có được cái nhìn toàn cảnh về các sự kiện an ninh. SIEM có thể phân tích một lượng lớn dữ liệu để phát hiện các cuộc tấn công ẩn dấu đằng sau chúng. Tuy nhiện SIEM có những hạn chế về khả năng mở rộng, độ phức tạp, tích hợp dữ liệu, khả năng phát hiện, nhận thức theo ngữ cảnh, phản hồi theo thời gian thực và cách tiếp cận lấy tuân thủ làm trung tâm. Các tổ chức cần nhận thức được những hạn chế này và xem xét các giải pháp bảo mật hiện đại tận dụng các công nghệ tiên tiến, như học máy, AI và tự động hóa, để khắc phục những hạn chế này và nâng cao tình trạng bảo mật của họ.

Giải pháp EDR – Phát hiện và phản hồi điểm cuối: cung cấp cho tổ chức khả năng giám sát các điểm cuối để tìm hành vi đáng ngờ và ghi lại mọi hoạt động và sự kiện đơn lẻ. Sau đó, nó tương quan thông tin để cung cấp ngữ cảnh quan trọng nhằm phát hiện các mối đe dọa nâng cao và cuối cùng chạy hoạt động ứng phó tự động, chẳng hạn như cách ly một điểm cuối bị nhiễm khỏi mạng trong thời gian gần thực tế. các phương thức xếp

hạng cảnh báo và trực quan hóa dữ liệu giúp người quản trị nhanh chóng xác định được mối đe dọa và lên phương án phản ứng.

Giải pháp XDR – Phát hiện và phản hồi mở rộng: là sự phát triển của EDR, Phát hiện điểm cuối và Ứng phó. Trong khi EDR thu thập và tương quan các hoạt động trên nhiều điểm cuối, XDR mở rộng phạm vi phát hiện ngoài các điểm cuối để cung cấp khả năng phát hiện, phân tích và ứng phó trên các điểm cuối, mạng, server, khối lượng công việc đám mây, SIEM, v.v.

2.1.1 Security orchestration, automation and response (SOAR)

Giải pháp điều phối, tự động hóa và phản hồi bảo mật là một giải pháp phần mềm cho phép các nhóm bảo mật tích hợp và điều phối các công cụ riêng biệt vào quy trình ứng phó mối đe dọa được hợp lý hóa.

Bằng cách hợp lý hóa việc phân loại cảnh báo và đảm bảo rằng các công cụ bảo mật khác nhau hoạt động cùng nhau, SOAR giúp SOC giảm thời gian phát hiện trung bình (MTTD) và thời gian phản hồi trung bình (MTTR), cải thiện tình trạng bảo mật tổng thể. Việc phát hiện và ứng phó với các mối đe dọa bảo mật nhanh hơn có thể làm giảm tác động của các cuộc tấn công mạng.

Khả năng điều phối và tự động hóa của SOAR cho phép nó hoạt động như một bảng điều khiển trung tâm để ứng phó với sự cố bảo mật. Các nhà phân tích bảo mật có thể sử dụng SOAR để điều tra và giải quyết sự cố mà không cần di chuyển giữa nhiều công cụ.

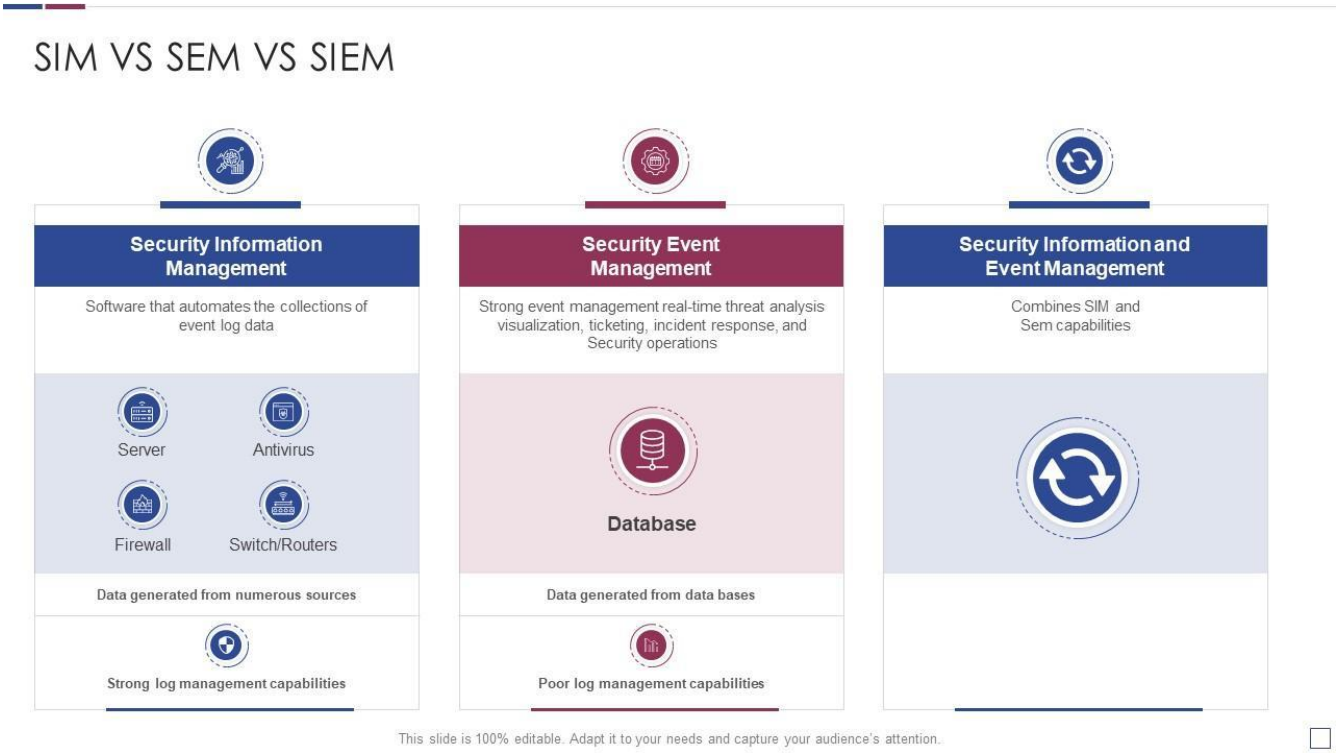
Giống như các nền tảng thông tin về mối đe dọa, SOAR tổng hợp các chỉ số và cảnh báo từ nguồn cấp dữ liệu bên ngoài cũng như các công cụ bảo mật tích hợp trong trang tổng quan trung tâm. Các nhà phân tích có thể đối chiếu dữ liệu từ các nguồn khác nhau, lọc ra các kết quả dương tính giả, ưu tiên cảnh báo và xác định các mối đe dọa cụ thể mà họ đang giải quyết. Sau đó, các nhà phân tích có thể phản hồi bằng cách kích hoạt các kịch bản thích hợp.

2.1.2. Security information and event management (SIEM)

Hệ thống SIEM được viết tắt cụm từ tiếng Anh: Security Information and Event Management. SIEM là hệ thống quản lý nhật ký và sự kiện tập trung, có nhiệm vụ thu thập thông tin nhật ký, sự kiện trong toàn hệ thống doanh nghiệp và tổng hợp tất cả trên 1 giao diện duy nhất.

SIEM là quá trình xác định, giám sát, ghi lại và phân tích các sự kiện hoặc sự cố bảo mật trong môi trường CNTT thời gian thực. Nó cung cấp một cái nhìn toàn diện và tập trung về kịch bản bảo mật của cơ sở hạ tầng CNTT.

SIEM còn được gọi là quản lý sự kiện thông tin bảo mật.



Hình 2.1. SIM, SEM và SIEM

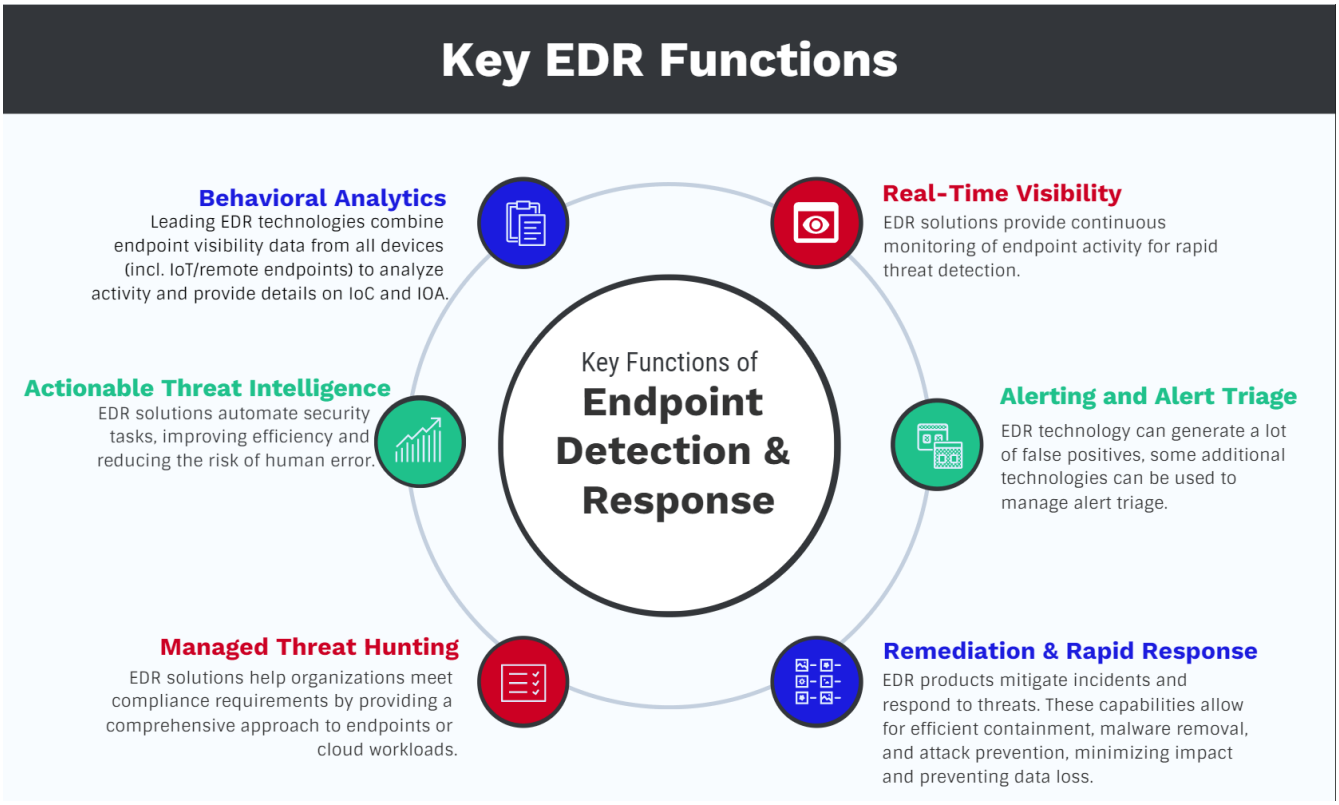
2.1.3. Endpoint Detection and Response (EDR)

Phát hiện và phản hồi điểm cuối (EDR), còn được gọi là Phát hiện và phản hồi mối đe dọa điểm cuối (ETDR), là một giải pháp bảo mật điểm cuối tích hợp kết hợp giám sát và thu thập dữ liệu điểm cuối liên tục theo thời gian thực với khả năng phân tích và phản hồi tự động dựa trên quy tắc. Thuật ngữ này được Anton Chuvakin tại Gartner đề xuất để mô tả các hệ thống bảo mật mới nổi có khả năng phát hiện và điều tra các hoạt động đáng ngờ trên server và thiết bị đầu cuối, sử dụng mức độ tự động hóa cao để cho phép các nhóm bảo mật nhanh chóng xác định và ứng phó với các mối đe dọa.

Các giải pháp bảo mật EDR ghi lại các hoạt động và sự kiện diễn ra trên các điểm cuối và tất cả khối lượng công việc, cung cấp cho các nhóm bảo mật khả năng hiển thị mà họ cần để phát hiện các sự cố mà lẽ ra vẫn không thể phát hiện được. Giải pháp EDR cần

cung cấp khả năng hiển thị liên tục và toàn diện về những gì đang xảy ra trên các điểm cuối trong thời gian thực.

Công cụ EDR phải cung cấp khả năng phát hiện, điều tra và ứng phó mối đe dọa nâng cao - bao gồm phân loại cảnh báo điều tra và tìm kiếm dữ liệu sự cố, xác thực hoạt động đáng ngờ, săn tìm mối đe dọa cũng như phát hiện và ngăn chặn hoạt động độc hại.



Hình 2.2. Các chức năng chính của EDR

2.1.4. Extended Dectecion and Response (XDR)

Bối cảnh kỹ thuật số đã chứng kiến sự gia tăng theo cấp số nhân của các mối đe dọa mạng, khiến các chuyên gia an ninh mạng phải liên tục đổi mới chiến lược phòng thủ của họ. Một trong những đổi mới đáng chú ý nhất xuất hiện trong những năm gần đây là khả năng phát hiện và phản hồi mở rộng (XDR). Phát triển từ phiên bản tiền nhiệm, phát hiện và phản hồi điểm cuối (EDR), XDR thể hiện sự thay đổi mô hình trong an ninh mạng bằng cách cung cấp cách tiếp cận toàn diện và tích hợp để phát hiện, ứng phó và giảm thiểu mối đe dọa.

Kẻ thù đã vượt ra ngoài các cuộc tấn công vectơ đơn lẻ để sắp xếp các chiến dịch phức tạp, đa vectơ nhằm khai thác lỗ hổng trên nhiều điểm truy cập. Các biện pháp bảo

mật truyền thống, thường tập trung vào các lớp phòng thủ biệt lập, không còn có thể theo kịp các cuộc tấn công nâng cao này. XDR thu hẹp những khoảng trống này bằng cách thống nhất dữ liệu bảo mật và cho phép phân tích thời gian thực, phát hiện mối đe dọa và phản hồi nhanh. XDR không chỉ nâng cao khả năng ngăn chặn các mối đe dọa của tổ chức mà còn cung cấp hoạt động bảo mật hợp lý và hiệu quả hơn, giải phóng các tài nguyên có giá trị mà lẽ ra phải chi cho các nhiệm vụ điều tra và ứng phó thủ công.

Tính năng phát hiện và phản hồi mở rộng (XDR) thể hiện sự khác biệt đáng kể so với các giải pháp bảo mật truyền thống, mang lại cách tiếp cận toàn diện và thích ứng hơn cho an ninh mạng. Dưới đây là một số khác biệt chính làm nổi bật những ưu điểm của XDR so với các phương pháp truyền thống:

Tích hợp phạm vi và dữ liệu: XDR tích hợp dữ liệu từ nhiều nguồn, bao gồm điểm cuối, mạng, môi trường đám mây và ứng dụng. Cách tiếp cận toàn diện này cung cấp góc nhìn rộng hơn về các mối đe dọa và cho phép tương quan dữ liệu giữa các vectơ khác nhau, giúp phát hiện các mô hình tấn công phức tạp có thể bị bỏ qua.

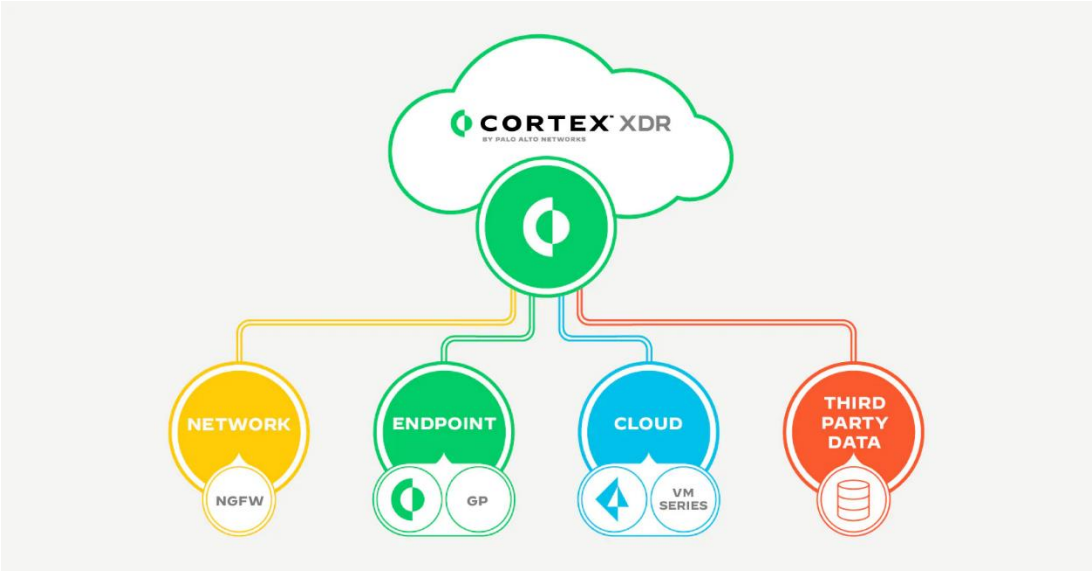
Hiểu biết theo ngữ cảnh: XDR cung cấp thông tin chi tiết theo ngữ cảnh bằng cách phân tích dữ liệu trên các lớp khác nhau của môi trường CNTT. Bối cảnh này giúp các nhóm bảo mật hiểu được chiến thuật, kỹ thuật và quy trình (tactics, techniques, and procedures - TTP) của những kẻ tấn công, cho phép đưa ra phản hồi sáng suốt hơn.

Tự động phát hiện và phản hồi mối đe dọa: XDR sử dụng tự động hóa và học máy để nhanh chóng xác định và ứng phó với các mối đe dọa. Playbook tự động có thể thực hiện các hành động được xác định trước dựa trên mức độ nghiêm trọng của mối đe dọa, giảm thời gian phản hồi và cho phép các nhóm bảo mật tập trung vào các nhiệm vụ chiến lược hơn.

Giám sát theo thời gian thực: XDR cung cấp khả năng giám sát theo thời gian thực và phát hiện mối đe dọa liên tục trên toàn bộ hệ sinh thái CNTT. Cách tiếp cận chủ động này giúp xác định và ngăn chặn các mối đe dọa ở giai đoạn đầu, giảm thiểu thiệt hại tiềm ẩn.

Dịch vụ đám mây và hỗ trợ làm việc từ xa: XDR được xây dựng để xử lý các môi trường đa dạng, bao gồm các hệ thống dựa trên đám mây và thiết bị từ xa. Tính linh hoạt

này cho phép các tổ chức duy trì bảo mật trên các cơ sở hạ tầng phân tán và đang phát triển.



Hình 2.3. Cortex XDR

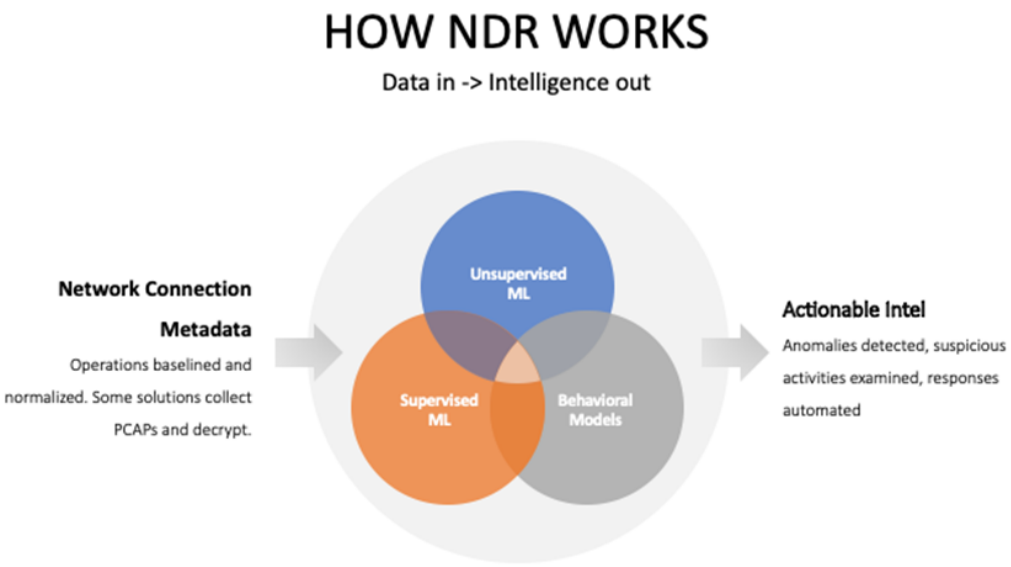
2.2. Các thành phần, kiến trúc và cách thức hoạt động của XDR

2.2.1. Thành phần của XDR

Một trong những điểm mạnh chính của XDR là khả năng cung cấp ngữ cảnh và khả năng hiển thị trên các công cụ bảo mật và nguồn dữ liệu khác nhau. Nó tổng hợp và phân tích dữ liệu từ các giải pháp bảo mật khác nhau, chẳng hạn như phát hiện và phản hồi điểm cuối (EDR), phát hiện và phản hồi mạng (NDR) cũng như các hệ thống quản lý sự kiện và thông tin bảo mật (SIEM).

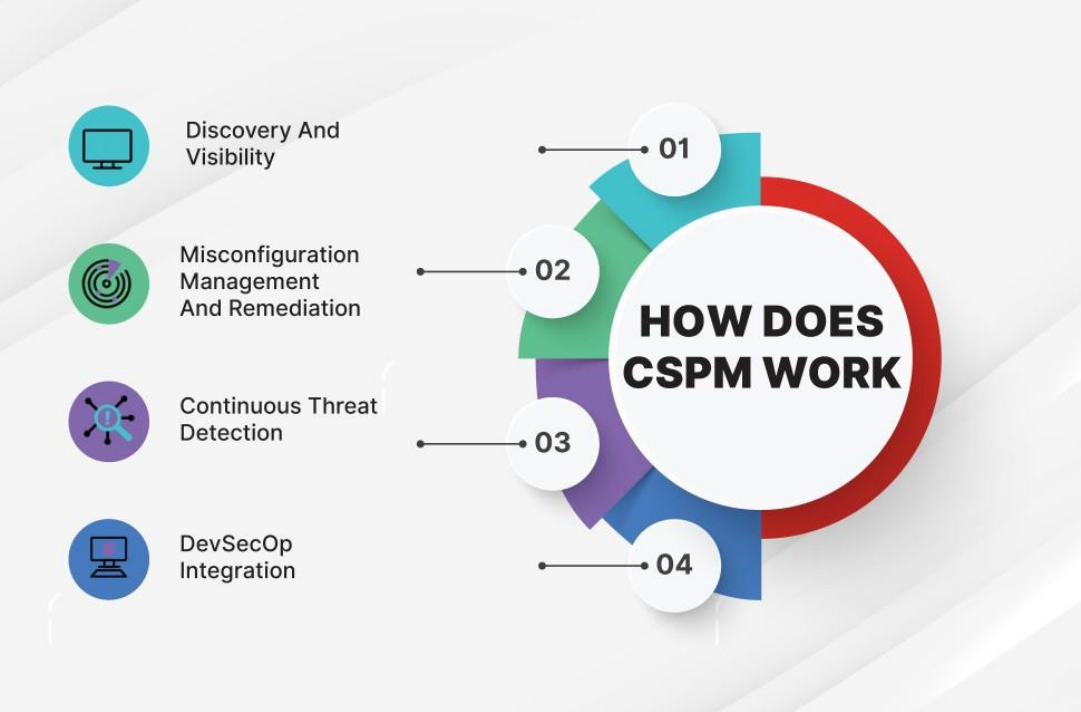
Ngoài 2 giải pháp SIEM và EDR đã được trình bày ở mục 2.1, thành phần của XDR còn bao gồm cả NDR – Giải pháp phát hiện và phản hồi mạng và CSPM – Quản lý tình trạng bảo mật đám mây.

Network Detection and Response – NDR: Phát hiện và phản hồi mạng (NDR) là một phương pháp an ninh mạng tập trung vào việc phát hiện và ứng phó với các mối đe dọa trong mạng máy tính. Nó được thiết kế để cung cấp cho các tổ chức khả năng hiển thị theo thời gian thực về lưu lượng mạng, phát hiện các hoạt động độc hại và cho phép ứng phó sự cố nhanh chóng. Các giải pháp NDR tăng cường khả năng hiển thị mạng, thực hiện phân tích phát hiện mối đe dọa, tận dụng các biện pháp ứng phó sự cố nhanh chóng và sử dụng các kỹ thuật điều tra và pháp lý để bảo mật mạng.



Hình 2.4. Cách thức NDR hoạt động

Cloud Security Posture Management – CSPM: Quản lý tình trạng bảo mật đám mây (CSPM) đề cập đến một bộ công cụ, biện pháp thực hành và quy trình giúp các tổ chức đảm bảo tính bảo mật và tuân thủ của môi trường đám mây của họ. Khi ngày càng nhiều doanh nghiệp áp dụng dịch vụ đám mây cho hoạt động của mình, việc duy trì trạng thái bảo mật mạnh mẽ trở nên quan trọng để bảo vệ dữ liệu nhạy cảm, ngăn chặn cấu hình sai và giảm thiểu rủi ro liên quan đến việc triển khai đám mây. Đây là một thành phần quan trọng của XDR giúp nó khác biệt với các nền tảng bảo mật trước đó.



Hình 2.5. Cách CSPM hoạt động

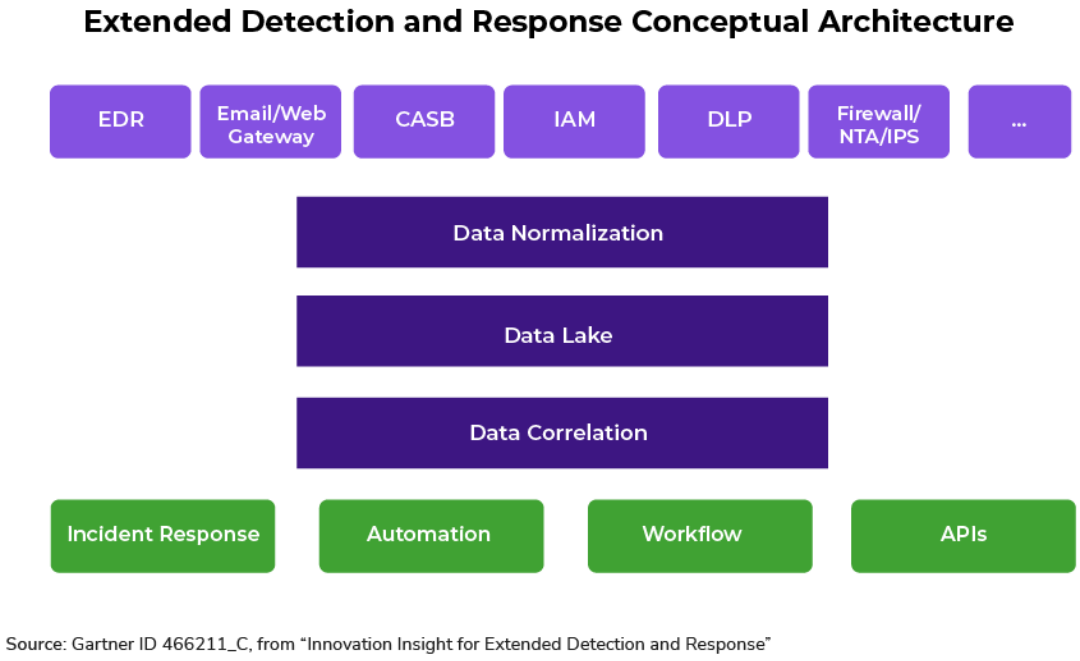
2.2.2. Cách thức hoạt động và kiến trúc của XDR

Quá trình bắt đầu (và tiếp tục) bằng việc thu thập dữ liệu từ các công cụ và hệ thống bảo mật khác nhau trên toàn bộ môi trường điện toán của tổ chức. Các nguồn ví dụ bao gồm các công cụ bảo mật điểm cuối, hệ thống quản lý danh tính và quyền truy cập, thiết bị bảo mật mạng, khối lượng công việc trên đám mây, hệ thống email, ứng dụng, v.v. Tất cả thông tin bảo mật có liên quan phải được nhập vào. Khi xuất hiện, dữ liệu được chuẩn hóa thành định dạng nhất quán và được bổ sung thêm thông tin theo ngữ cảnh.

Bước tiếp theo liên quan đến tương quan và phân tích dữ liệu. Phân tích nâng cao, học máy và trí tuệ nhân tạo được áp dụng để xác định các mối quan hệ và mẫu cũng như phát hiện các điểm bất thường trong dữ liệu. XDR có thể kết hợp nhiều điểm dữ liệu thành một câu chuyện tấn công thống nhất, có thể trải rộng trên nhiều vectơ tấn công. Phân tích này tạo ra những hiểu biết sâu sắc giúp nhóm bảo mật phát hiện cả mối đe dọa đã biết và chưa biết, cũng như các mối đe dọa liên tục nâng cao (APTS), trong thời gian thực.

Sau khi phát hiện được mối đe dọa, giải pháp XDR sẽ cung cấp tất cả thông tin liên quan cho nhóm bảo mật để hỗ trợ quá trình điều tra và ứng phó. Có thể có các hành động phản hồi tự động, chẳng hạn như chặn lưu lượng truy cập độc hại hoặc ngắt kết nối của

một tài khoản đáng ngờ. Các mối đe dọa được ưu tiên theo mức độ nghiêm trọng để giúp nhóm bảo mật quyết định việc cần làm trước tiên.



Hình 2.6. Kiến trúc của XDR

a. Chuẩn hóa dữ liệu – Data Normalization

Trong XDR, dữ liệu được chuyển sang trạng thái phong phú và chuẩn hóa và việc này được thực hiện trước khi dữ liệu được lưu trữ trong hồ dữ liệu. SIEM thường lưu dữ liệu ở dạng ban đầu từ nhật ký gốc hoặc nguồn sự kiện.

Đối với XDR, mối tương quan và phát hiện cảnh báo được AI điều khiển tự động, trong khi SIEM sử dụng các quy tắc tương quan do con người viết ra để tạo ra các sự kiện quan tâm. Điều này thường đòi hỏi kiến thức hoặc kinh nghiệm chuyên môn mà không phải nhân viên nào cũng có được.

Giai đoạn tiền xử lý của XDR giải quyết vấn đề về chất lượng và tiêu chuẩn hóa dữ liệu, cần thiết để xây dựng và duy trì AI có ý nghĩa. Trong bảo mật, điều này chỉ ra rằng dữ liệu phải được tập trung, làm phong phú và chuẩn hóa để giảm thiểu độ phức tạp của dữ liệu. Nếu dữ liệu được mô hình hóa theo nhiều cách khác nhau trong mỗi lần triển khai nền tảng thì sẽ không thể duy trì các mô hình AI.

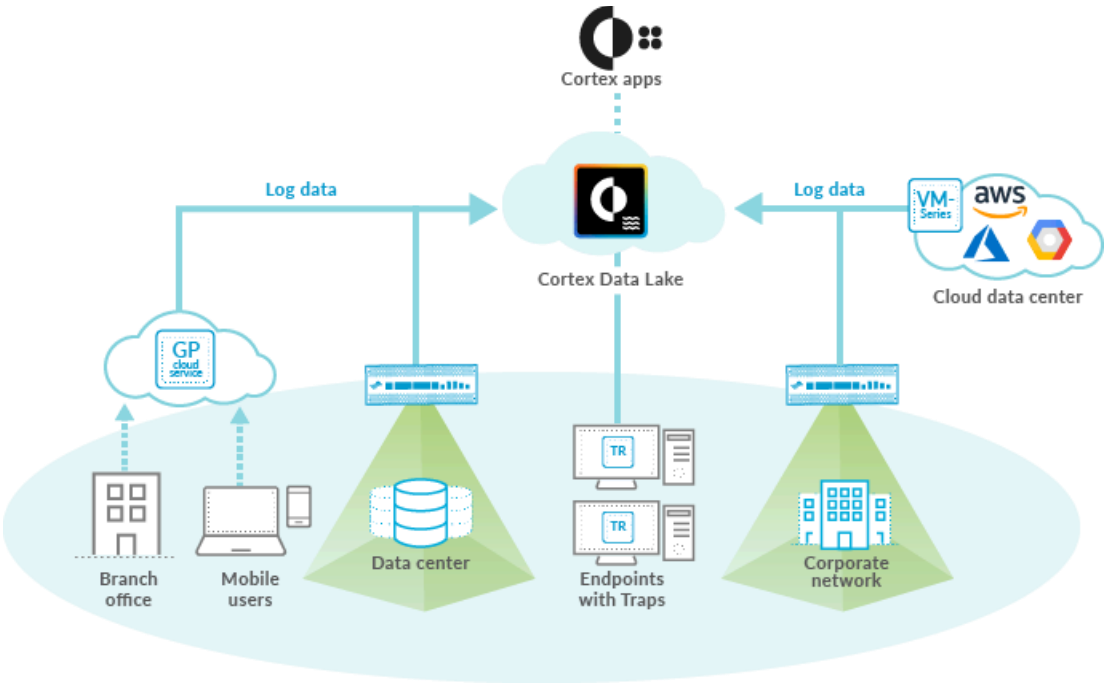
XDR yêu cầu dữ liệu phải được mô hình hóa theo cách giống nhau trong mỗi lần triển khai trước khi đưa vào Hồ dữ liệu; dữ liệu chỉ có sẵn ở trạng thái được làm giàu hoặc chuẩn hóa.

SIEM cung cấp tính năng này dưới dạng chức năng tùy chọn hoặc không cung cấp tính năng này; trong kịch bản tùy chọn, việc làm giàu và chuẩn hóa được tiếp cận như một bước xử lý sau trên dữ liệu thô đã được lưu trữ. Do đó, kiến trúc SIEM gặp khó khăn trong việc tạo ra một công cụ AI có độ trung thực tương đương với XDR. SIEM có thể tận dụng AI, tuy nhiên, việc mở rộng quy mô sẽ khó khăn hơn.

b. Hồ dữ liệu – Data lake

Hồ dữ liệu là vị trí trung tâm chứa một lượng lớn dữ liệu ở định dạng thô, gốc. So với kho dữ liệu phân cấp lưu trữ dữ liệu trong tệp hoặc thư mục, hồ dữ liệu sử dụng kiến trúc phẳng và lưu trữ đối tượng để lưu trữ dữ liệu. Bộ lưu trữ đối tượng lưu trữ dữ liệu bằng thẻ siêu dữ liệu và mã nhận dạng duy nhất, giúp định vị và dễ dàng hơn truy xuất dữ liệu trên các khu vực và cải thiện hiệu suất. Bằng cách tận dụng bộ lưu trữ đối tượng rẻ tiền và các định dạng mở, hồ dữ liệu cho phép nhiều ứng dụng tận dụng dữ liệu.

Hồ dữ liệu được phát triển để đáp ứng những hạn chế của kho dữ liệu. Mặc dù kho dữ liệu cung cấp cho doanh nghiệp các phân tích có hiệu suất cao và có khả năng mở rộng, nhưng chúng đắt tiền và độc quyền, đồng thời không thể xử lý các trường hợp sử dụng hiện đại mà hầu hết các công ty đang tìm cách giải quyết. Hồ dữ liệu thường được sử dụng để hợp nhất tất cả dữ liệu của tổ chức ở một vị trí trung tâm duy nhất, nơi dữ liệu có thể được lưu "nguyên trạng" mà không cần áp đặt lược đồ (tức là cấu trúc chính thức về cách tổ chức dữ liệu) phía trước giống như một kho dữ liệu. Dữ liệu trong tất cả các giai đoạn của quá trình sàng lọc có thể được lưu trữ trong hồ dữ liệu: dữ liệu thô có thể được nhập và lưu trữ ngay bên cạnh các nguồn dữ liệu dạng bảng có cấu trúc của tổ chức (như bảng cơ sở dữ liệu), cũng như các bảng dữ liệu trung gian được tạo trong quá trình tinh chỉnh dữ liệu thô. Không giống như hầu hết các cơ sở dữ liệu và kho dữ liệu, hồ dữ liệu có thể xử lý tất cả các loại dữ liệu — bao gồm cả dữ liệu phi cấu trúc và bán cấu trúc như hình ảnh, video, âm thanh và tài liệu — vốn rất quan trọng đối với các trường hợp sử dụng máy học và phân tích nâng cao ngày nay.



Hình 2.7. Cortex Data Lake

Trong kiến trúc của Extended Detection and Response (XDR), Data Lake đóng vai trò quan trọng như một kho lưu trữ tập trung cho dữ liệu an ninh từ nhiều nguồn khác nhau. Nó hỗ trợ việc tổng hợp, lưu trữ dữ liệu lớn một cách linh hoạt, và cung cấp khả năng mở rộ để xử lý lượng dữ liệu ngày càng tăng. Data Lake giữ nguyên dữ liệu dưới dạng gốc, hỗ trợ lưu trữ dài hạn cho mục đích tuân thủ và phân tích lịch sử. Nó cũng là nền tảng cho việc áp dụng phân tích tiên tiến và machine learning, giúp phát hiện mẫu, bất thường, và mối đe dọa an ninh. Đồng thời, Data Lake tích hợp tốt với hệ thống SIEM, cung cấp cái nhìn toàn diện để hỗ trợ quá trình quản lý và phản ứng đối với các mối đe dọa an ninh.

c. Tương quan dữ liệu – Data correlation

Các giải pháp XDR sử dụng phân tích nâng cao, học máy và trí tuệ nhân tạo để xác định các mẫu và điểm bất thường trong dữ liệu được thu thập. Những hiểu biết sâu sắc này giúp các nhóm bảo mật phát hiện các mối đe dọa trong thời gian thực, cho phép họ phản ứng nhanh hơn và hiệu quả hơn trước các cuộc tấn công tiềm ẩn.

XDR tích hợp thông tin từ nhiều giải pháp an ninh, như phản ứng và phát hiện tại điểm cuối (EDR) và an ninh mạng, và so sánh các logs và sự kiện để xác định các kết nối

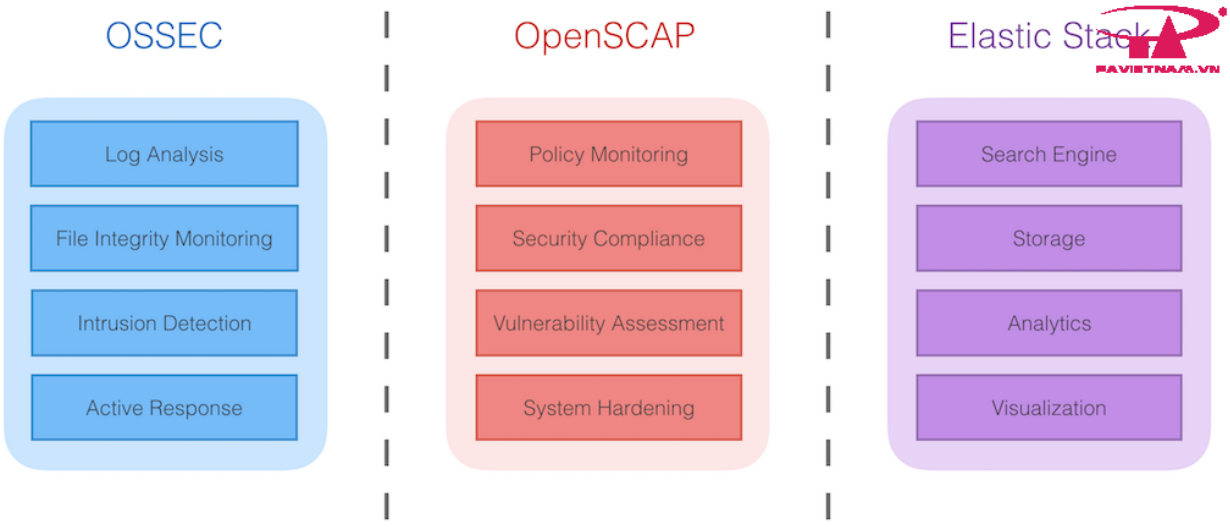
có ý nghĩa. Quá trình này nhằm mục đích phát hiện ra các mô hình và mối quan hệ có thể chỉ ra nguy cơ an ninh.

Ngoài ra, việc liên kết dữ liệu trong XDR đóng vai trò quan trọng trong việc phát hiện bất thường và ưu tiên các sự kiện. Nó giúp phân biệt hành vi bình thường và các hoạt động có thể làm nảy sinh rủi ro an ninh. Bằng cách giảm thiểu các báo động giả mạo và xác nhận thông qua nhiều điểm dữ liệu, XDR nâng cao độ chính xác trong việc nhận diện nguy cơ. Hơn nữa, sự liên kết dữ liệu hỗ trợ phân tích pháp y, cho phép đội ngũ an ninh theo dõi các bước của một cuộc tấn công theo thời gian và đánh giá toàn diện về phạm vi của sự cố an ninh.

2.3. Bộ công cụ Opensource Wazuh, TheHive, Cortex XDR.

2.3.1. Wazuh – Nền tảng bảo mật mã nguồn mở

Wazuh là nền tảng mã nguồn mở hợp nhất của XDR và Security Information and Event Management (SIEM). Được xây dựng từ các thành phần : OSSEC HIDS, OpenSCAP và Elastic Stack.



Hình 2.8. Cách Wazuh được xây dựng

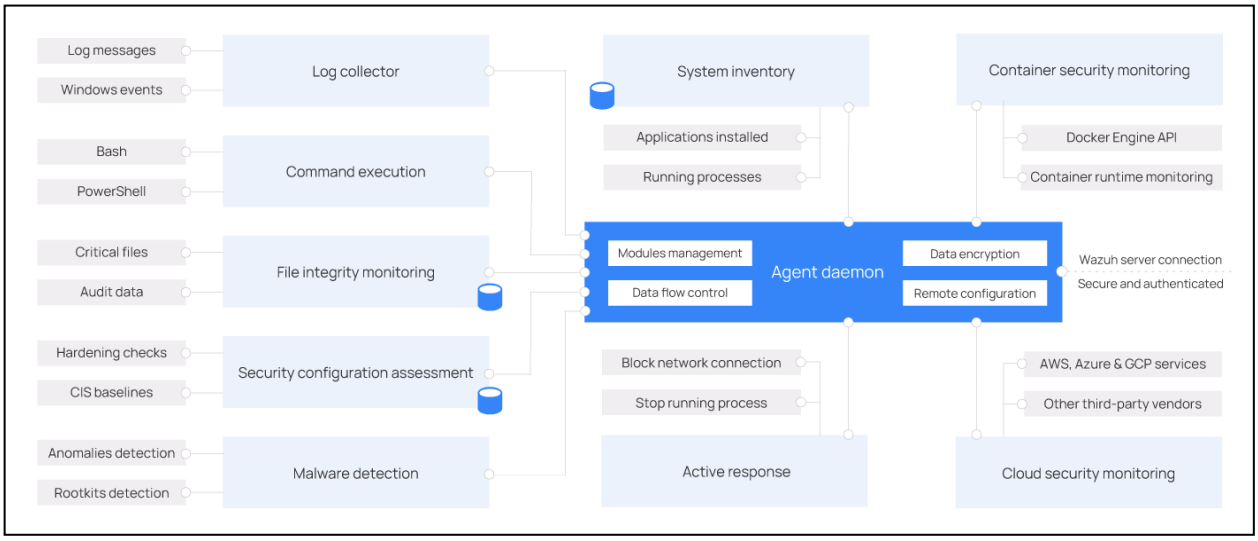
- OSSEC HIDS : host-based Intrusion Detection System (HIDS) được dùng cho việc phát hiện xâm nhập, hiển thị và giám stas. Nó dựa vào 1 multi-platform agent cho việc đẩy dữ liệu hệ thống (log message, file hash và phát hiện bất thường) tới 1 máy quản lý trung tâm, nơi sẽ phân tích và xử lý, dựa trên các cảnh báo an ninh. Các agent truyền event data event data tới máy quản lý trung tâm thông qua kênh được

bảo mật và xác thực. OSSEC HIDS cung cấp syslog server trung tâm và hệ thống giám sát không cần agent, cung cấp việc giám sát tới các event và thay đổi trên các thiết bị không cài được agent như firewall, switch, router, access point, thiết bị mạng….

- OpenSCAP OpenSCAP là 1 OVAL (Open Vulnerability Assesment Language) và XCCDF (Extensible Configuration Checklist Description Format) được dùng để kiểm tra cấu hình hệ thống và phát hiện các ứng dụng dễ bị tấn công. Nó được biết đến như là một công cụ được thiết kế để kiểm tra việc tuân thủ an ninh của hệ thống sử dụng các tiêu chuẩn an ninh dùng cho môi trường doanh nghiệp
- ELK Stack Sử dụng cho việc thu thập, phân tihcs, index, store, search và hiển thị dữ liệu log.

a. Các thành phần của Wazuh

Wazuh agent: Tác nhân Wazuh chạy trên Linux, Windows, macOS, Solaris, AIX và các hệ điều hành khác. Nó có thể được triển khai trên máy tính xách tay, máy tính để bàn, server, phiên bản đám mây, vùng chứa hoặc máy ảo. Tác nhân này giúp bảo vệ hệ thống của bạn bằng cách cung cấp khả năng ngăn chặn, phát hiện và phản hồi mối đe dọa. Nó cũng được sử dụng để thu thập các loại dữ liệu ứng dụng và hệ thống khác nhau mà nó chuyển tiếp đến Wazuh server thông qua kênh được mã hóa và xác thực.



Hình 2.9. Sơ đồ liên kết các module của Wazuh Agent

Các modun của Wazuh agent sẽ có thể cấu hình và thực hiện các tác vụ bảo mật khác nhau. Kiến trúc mô-đun này cho phép bạn bật hoặc tắt từng thành phần theo nhu cầu bảo mật của mình. Dưới đây là các mục đích khác nhau của tất cả các mô-đun agent.

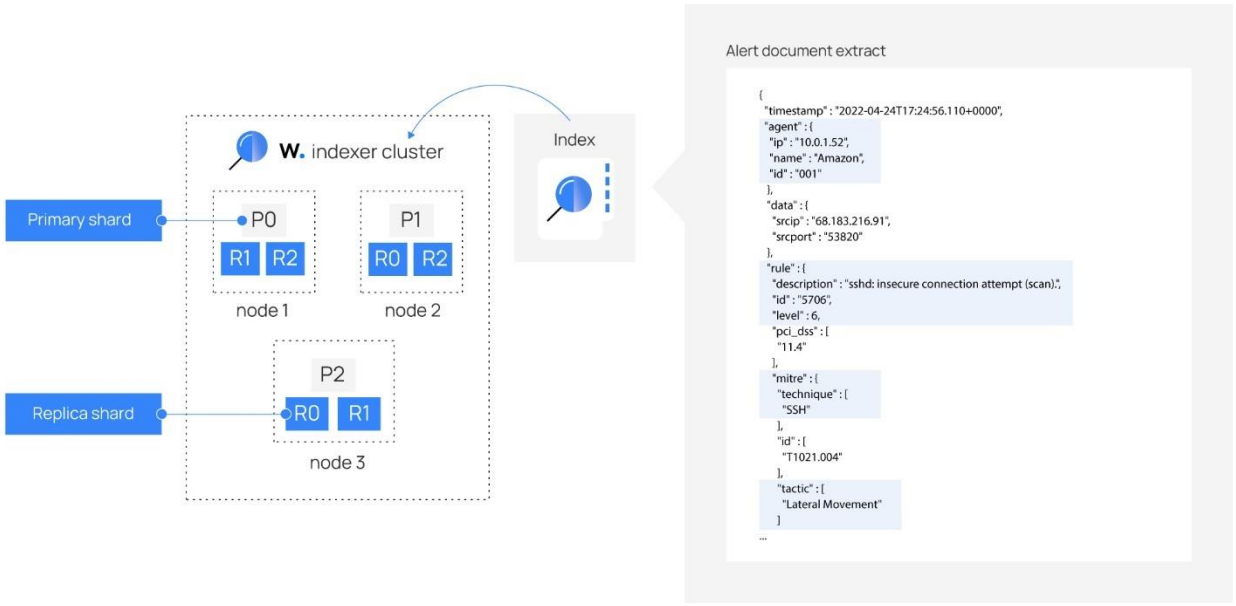
- Trình thu thập nhật ký: Thành phần tác nhân này có thể đọc các tệp nhật ký phẳng và các sự kiện Windows, thu thập thông báo nhật ký ứng dụng và hệ điều hành. Nó hỗ trợ các bộ lọc XPath cho các sự kiện Windows và nhận dạng các định dạng nhiều dòng như nhật ký Kiểm tra Linux. Nó cũng có thể làm phong phú thêm các sự kiện JSON bằng siêu dữ liệu bổ sung.
- Thực thi lệnh: Đại lý chạy các lệnh được ủy quyền định kỳ, thu thập đầu ra của chúng và báo cáo lại cho Wazuh server để phân tích thêm. Bạn có thể sử dụng mô-đun này cho các mục đích khác nhau, chẳng hạn như theo dõi dung lượng ổ cứng còn lại hoặc lấy danh sách những người dùng đăng nhập lần cuối.
- Giám sát tính toàn vẹn của tệp (FIM): Mô-đun này giám sát hệ thống tệp, báo cáo khi tệp được tạo, xóa hoặc sửa đổi. Nó theo dõi những thay đổi về thuộc tính, quyền, quyền sở hữu và nội dung của tệp. Khi một sự kiện xảy ra, nó sẽ ghi lại chi tiết ai, cái gì và khi nào trong thời gian thực. Ngoài ra, mô-đun FIM xây dựng và duy trì cơ sở dữ liệu về trạng thái của các tệp được giám sát, cho phép chạy các truy vấn từ xa.
- Đánh giá cấu hình bảo mật (SCA): Thành phần này cung cấp đánh giá cấu hình liên tục, sử dụng các bước kiểm tra sẵn dùng dựa trên điểm chuẩn của Trung tâm Bảo mật Internet (CIS). Người dùng cũng có thể tạo các bước kiểm tra SCA của riêng mình để giám sát và thực thi các chính sách bảo mật của mình.
- Kiểm kê hệ thống: Mô-đun tác nhân này chạy quét định kỳ, thu thập dữ liệu kiểm kê như phiên bản hệ điều hành, giao diện mạng, quy trình đang chạy, ứng dụng đã cài đặt và danh sách các cổng đang mở. Kết quả quét được lưu trữ trong cơ sở dữ liệu SQLite cục bộ có thể được truy vấn từ xa.
- Phát hiện phần mềm độc hại: Sử dụng phương pháp tiếp cận không dựa trên chữ ký, thành phần này có khả năng phát hiện các điểm bất thường và sự hiện diện có thể có của rootkit. Ngoài ra, nó còn tìm kiếm các tiến trình ẩn, tệp ẩn và cổng ẩn trong khi giám sát các cuộc gọi hệ thống.

- Phản hồi tích cực: Mô-đun này chạy các hành động tự động khi phát hiện mối đe dọa, kích hoạt phản hồi để chặn kết nối mạng, dừng quá trình đang chạy hoặc xóa tệp độc hại. Người dùng cũng có thể tạo phản hồi tùy chỉnh khi cần thiết và tùy chỉnh, chẳng hạn như phản hồi để chạy tệp nhị phân trong hộp cát, nắm bắt lưu lượng truy cập mạng và quét tệp bằng phần mềm chống vi-rút.
- Giám sát bảo mật vùng chứa: Mô-đun tác nhân này được tích hợp với API Docker Engine để giám sát các thay đổi trong môi trường được chứa trong vùng chứa. Ví dụ: nó phát hiện các thay đổi đối với hình ảnh vùng chứa, cấu hình mạng hoặc khối lượng dữ liệu. Ngoài ra, nó còn cảnh báo về các container đang chạy ở chế độ đặc quyền và về việc người dùng thực thi các lệnh trong một container đang chạy.
- Giám sát bảo mật đám mây: Thành phần này giám sát các nhà cung cấp đám mây như Amazon AWS, Microsoft Azure hoặc Google GCP. Nó thực sự giao tiếp với API của họ. Nó có khả năng phát hiện các thay đổi đối với cơ sở hạ tầng đám mây (ví dụ: người dùng mới được tạo, nhóm bảo mật được sửa đổi, phiên bản đám mây bị dừng, v.v.) và thu thập dữ liệu nhật ký dịch vụ đám mây (ví dụ: AWS Cloudtrail, AWS Macie, AWS GuardDuty, Microsoft Entra ID, v.v.).

Wazuh Indexer: Wazuh Indexer là một công cụ phân tích và tìm kiếm toàn văn bản có khả năng mở rộng cao. Thành phần trung tâm Wazuh này lập chỉ mục và lưu trữ các cảnh báo do Wazuh server tạo ra, đồng thời cung cấp khả năng phân tích và tìm kiếm dữ liệu gần như theo thời gian thực. Wazuh Indexer có thể được cấu hình dưới dạng cụm một nút hoặc nhiều nút, cung cấp khả năng mở rộng và tính sẵn sàng cao.

Wazuh Indexer lưu trữ dữ liệu dưới dạng tài liệu JSON. Mỗi tài liệu tương quan với một tập hợp khóa, tên trường hoặc thuộc tính với các giá trị tương ứng của chúng có thể là chuỗi, số, boolean, ngày tháng, mảng giá trị, vị trí địa lý hoặc các loại dữ liệu khác.

Indexer là tập hợp các tài liệu có liên quan với nhau. Các tài liệu được lưu trữ trong Wazuh Indexer được phân phối trên các vùng chứa khác nhau được gọi là phân đoạn. Bằng cách phân phối tài liệu trên nhiều phân đoạn và phân phối các phân đoạn đó trên nhiều nút, Wazuh Indexer có thể đảm bảo tính dư thừa. Điều này bảo vệ hệ thống của bạn khỏi các lỗi phần cứng và tăng khả năng truy vấn khi các nút được thêm vào một cụm.



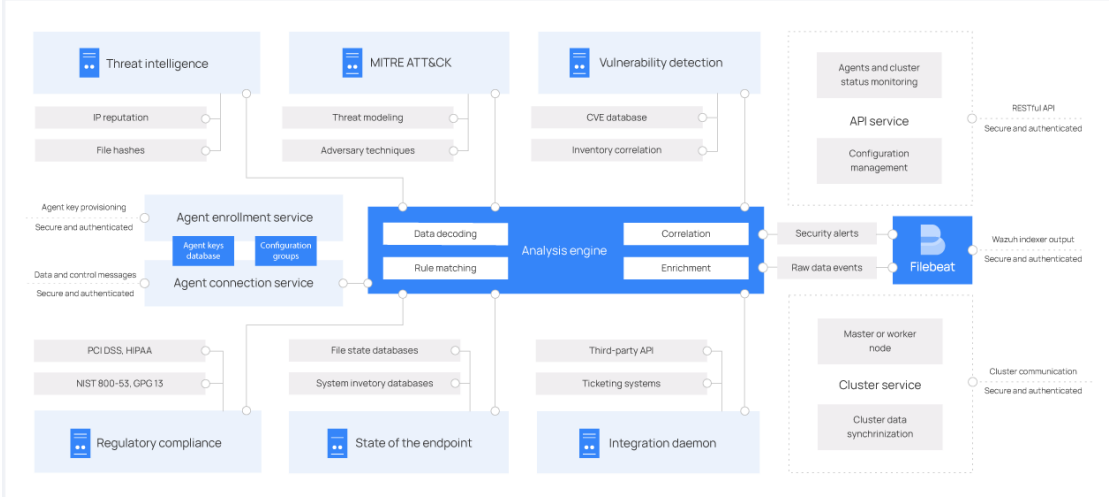
Hình 2.10. Sơ đồ cấu hình các cluster trong Wazuh Indexer

Wazuh server: Thành phần Wazuh server phân tích dữ liệu nhận được từ các tác nhân, kích hoạt cảnh báo khi phát hiện các mối đe dọa hoặc sự bất thường. Nó cũng được sử dụng để quản lý cấu hình tác nhân từ xa và theo dõi trạng thái của chúng.

Wazuh server sử dụng các nguồn thông tin về mối đe dọa để cải thiện khả năng phát hiện. Nó cũng làm phong phú thêm dữ liệu cảnh báo bằng cách sử dụng khung MITER ATT&CK và các yêu cầu tuân thủ quy định như PCI DSS, GDPR, HIPAA, CIS và NIST 800-53, cung cấp bối cảnh hữu ích cho phân tích bảo mật.

Ngoài ra, Wazuh server có thể được tích hợp với phần mềm bên ngoài, bao gồm các hệ thống bán vé như ServiceNow, Jira và PagerDuty, cũng như các nền tảng nhắn tin tức thời như Slack. Những tích hợp này thuận tiện cho việc hợp lý hóa các hoạt động bảo mật.

Wazuh server chạy công cụ phân tích, API Wazuh RESTful, dịch vụ đăng ký tác nhân, dịch vụ kết nối tác nhân, daemon cụm Wazuh và Filebeat. Máy chủ được cài đặt trên hệ điều hành Linux và thường chạy trên máy vật lý độc lập, máy ảo, bộ chứa docker hoặc phiên bản đám mây.



Hình 2.11. Wazuh Architecture

Wazuh server bao gồm một số thành phần được liệt kê bên dưới có các chức năng khác nhau, chẳng hạn như đăng ký tác nhân mới, xác thực danh tính từng tác nhân và mã hóa thông tin liên lạc giữa tác nhân Wazuh và Wazuh server.

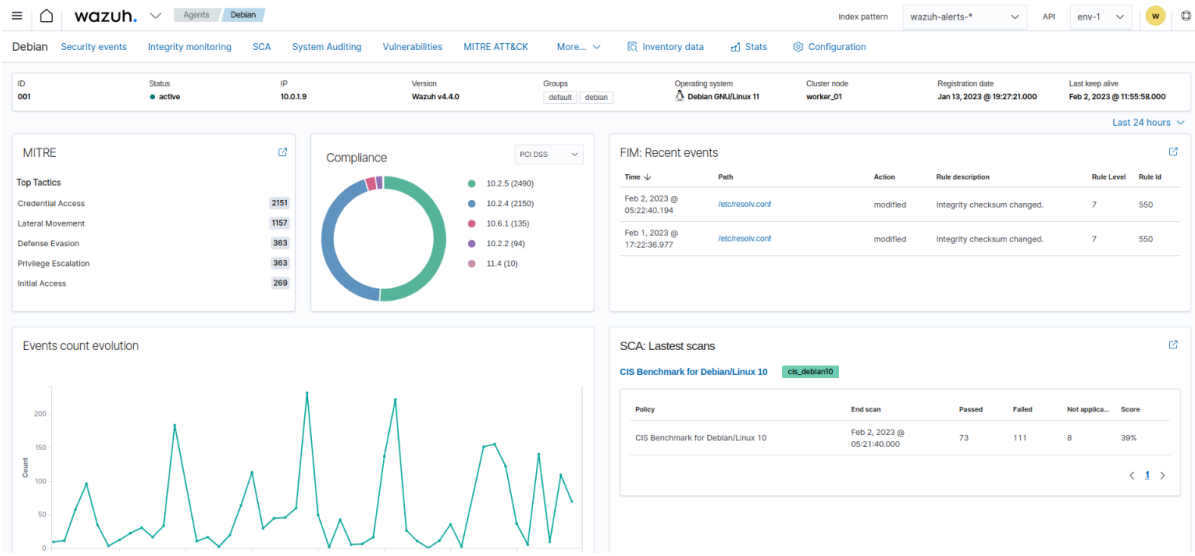
- Dịch vụ tuyển đại lý: Dùng để tuyển đại lý mới. Dịch vụ này cung cấp và phân phối các khóa xác thực duy nhất cho mỗi tác nhân. Quá trình này chạy như một dịch vụ mạng và hỗ trợ xác thực thông qua chứng chỉ TLS/SSL hoặc bằng cách cung cấp mật khẩu cố định.
- Dịch vụ kết nối đại lý: Dịch vụ này nhận dữ liệu từ các đại lý. Nó sử dụng các khóa được chia sẻ bởi dịch vụ đăng ký để xác thực từng danh tính tác nhân và mã hóa thông tin liên lạc giữa tác nhân Wazuh và Wazuh server. Ngoài ra, dịch vụ này còn cung cấp khả năng quản lý cấu hình tập trung, cho phép bạn đẩy cài đặt tác nhân mới từ xa.
- Công cụ phân tích: Đây là thành phần server thực hiện phân tích dữ liệu. Nó sử dụng bộ giải mã để xác định loại thông tin đang được xử lý (sự kiện Windows, nhật ký SSH, nhật ký server web và các thông tin khác). Các bộ giải mã này cũng trích xuất các phần tử dữ liệu có liên quan từ thông điệp tường trình, chẳng hạn như địa chỉ IP nguồn, ID sự kiện hoặc tên người dùng. Sau đó, bằng cách sử dụng các quy tắc, công cụ này xác định các mẫu cụ thể trong các sự kiện được giải mã có thể kích hoạt cảnh báo và thậm chí có thể yêu cầu các biện pháp đối phó tự động (ví dụ: cấm địa chỉ IP, dừng quy trình đang chạy hoặc xóa phần mềm độc hại).

- API Wazuh RESTful: Dịch vụ này cung cấp giao diện để tương tác với cơ sở hạ tầng Wazuh. Nó được sử dụng để quản lý cài đặt cấu hình của các tác nhân và server, theo dõi trạng thái cơ sở hạ tầng và tình trạng tổng thể, quản lý và chỉnh sửa bộ giải mã và quy tắc Wazuh cũng như truy vấn về trạng thái của các điểm cuối được giám sát. Bảng điều khiển Wazuh cũng sử dụng nó.
- Daemon cụm Wazuh: Dịch vụ này được sử dụng để mở rộng quy mô Wazuh server theo chiều ngang, triển khai chúng dưới dạng cụm. Loại cấu hình này, kết hợp với bộ cân bằng tải mạng, mang lại tính khả dụng và cân bằng tải cao. Daemon cụm Wazuh là thứ mà các Wazuh server sử dụng để liên lạc với nhau và duy trì đồng bộ hóa.
- Filebeat: Nó được sử dụng để gửi các sự kiện và cảnh báo đến Wazuh Indexer. Nó đọc đầu ra của công cụ phân tích Wazuh và gửi các sự kiện theo thời gian thực. Nó cũng cung cấp khả năng cân bằng tải khi được kết nối với cụm Wazuh Indexer nhiều nút.
- Wazuh dashboard: Bảng điều khiển Wazuh là giao diện người dùng web linh hoạt và trực quan để khai thác, phân tích và trực quan hóa dữ liệu cảnh báo và sự kiện bảo mật. Nó cũng được sử dụng để quản lý và giám sát nền tảng Wazuh. Ngoài ra, nó còn cung cấp các tính năng kiểm soát truy cập dựa trên vai trò (RBAC) và đăng nhập một lần (SSO).

Giao diện web giúp người dùng điều hướng qua các loại dữ liệu khác nhau được thu thập bởi tác nhân Wazuh, cũng như các cảnh báo bảo mật do máy chủ Wazuh tạo ra. Người dùng cũng có thể tạo báo cáo và tạo trực quan hóa và bảng chỉ số tùy chỉnh.

Ví dụ: Wazuh cung cấp bảng thông tin sẵn dùng để tuân thủ quy định như PCI DSS, GDPR, HIPAA và NIST 800-53. Nó cũng cung cấp một giao diện để điều hướng qua khung MITER ATT&CK và các cảnh báo liên quan.

Bảng điều khiển Wazuh cho phép người dùng quản lý cấu hình đại lý và theo dõi trạng thái của họ. Ví dụ: đối với mỗi điểm cuối được giám sát, người dùng có thể xác định mô-đun tác nhân nào sẽ được bật, tệp nhật ký nào sẽ được đọc, tệp nào sẽ được giám sát để phát hiện các thay đổi về tính toàn vẹn hoặc kiểm tra cấu hình nào sẽ được thực hiện.

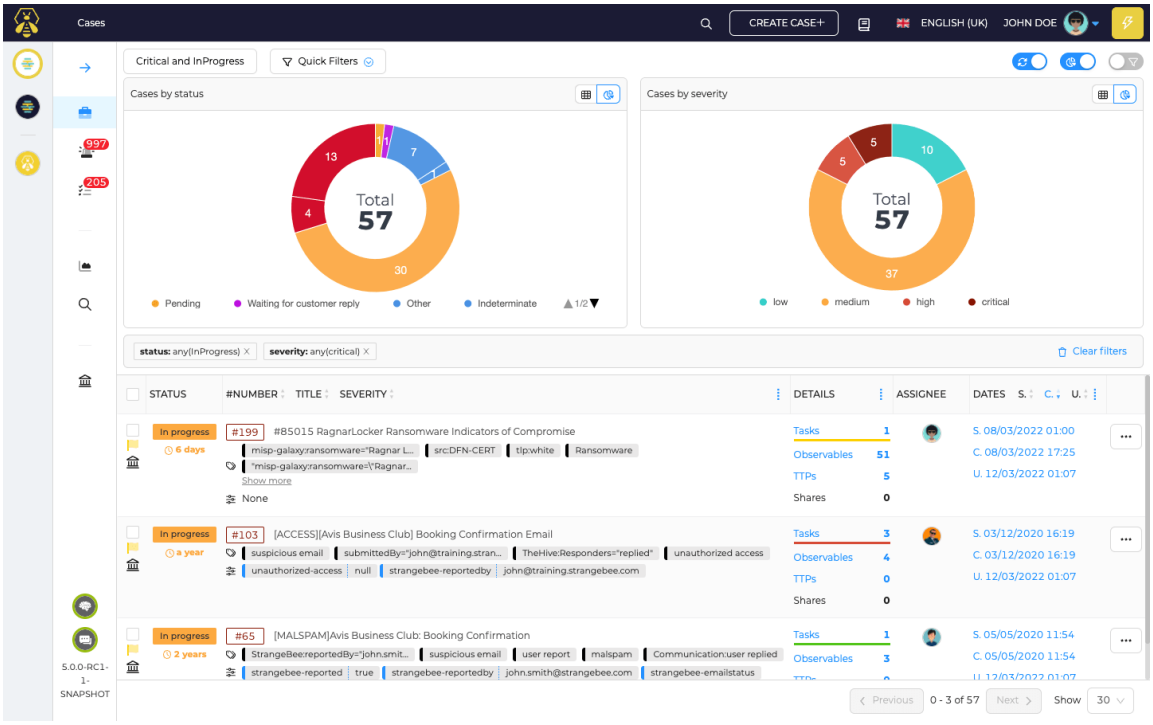


Hình 2.12. Giao diện bảng thống kê của Wazuh dashboard

2.3.2. TheHive – Nền tảng ứng phó sự cố bảo mật

TheHive là một Nền tảng ứng phó sự cố bảo mật (SIRP) nguồn mở, có thể mở rộng và hợp tác được thiết kế để hỗ trợ quản lý và phân tích các sự cố bảo mật. Được phát triển chủ yếu cho Nhóm ứng phó sự cố bảo mật máy tính (CSIRT) và Trung tâm điều hành bảo mật (SOC), TheHive hợp lý hóa và nâng cao quy trình xử lý sự cố bằng cách cung cấp một nền tảng tập trung để quản lý trường hợp, phân công nhiệm vụ và cộng tác theo thời gian thực.

Với bộ tính năng phong phú, bao gồm bảng điều khiển có thể tùy chỉnh, thiết bị quan sát tích hợp và tích hợp với các công cụ tình báo về mối đe dọa phổ biến như MISP và Cortex, TheHive cho phép các chuyên gia bảo mật phân loại, phân tích và ứng phó hiệu quả với các sự kiện bảo mật. Ví dụ: nhà phân tích SOC xử lý chiến dịch lừa đảo có thể sử dụng TheHive để tạo trường hợp, phân công nhiệm vụ cho các thành viên trong nhóm và tận dụng dữ liệu tình báo về mối đe dọa tích hợp tốt hơn để hiểu rõ hơn về bản chất và phạm vi của cuộc tấn công. Kiến trúc mô-đun của TheHive và sự hỗ trợ cộng đồng tích cực khiến nó trở thành một công cụ linh hoạt và có giá trị trong bối cảnh an ninh mạng không ngừng phát triển.



Hình 2.13. Giao diện trang thống kê của TheHive

TheHive hoạt động như một nền tảng dựa trên web tập trung quản lý sự cố và cộng tác cho các nhóm bảo mật. Các thành phần chính của nó là các trường hợp, tác vụ, quan sát và phân tích, được sử dụng để quản lý và phân tích các sự cố bảo mật một cách hiệu quả. Dưới đây là bảng phân tích về cách thức hoạt động TheHive:

- Các trường hợp : Các nhà phân tích bảo mật tạo các trường hợp để đại diện cho các sự cố riêng lẻ. Mỗi trường hợp chứa một bản tóm tắt, mức độ nghiêm trọng, thẻ và siêu dữ liệu có liên quan khác. Các trường hợp cho phép các nhà phân tích duy trì cách tiếp cận có cấu trúc trong khi xử lý đồng thời nhiều sự cố.
- Nhiệm vụ : Nhà phân tích có thể tạo và giao nhiệm vụ cho các thành viên trong nhóm trong từng trường hợp. Nhiệm vụ có thể được chỉ định mức độ ưu tiên, ngày đến hạn và mô tả, giúp theo dõi tiến độ và đảm bảo trách nhiệm rõ ràng.
- Có thể quan sát : Có thể quan sát là các điểm dữ liệu hoặc chỉ báo liên quan đến một sự cố, chẳng hạn như địa chỉ IP, tên miền, địa chỉ email hoặc mã băm tệp. Các nhà phân tích có thể thêm các dữ liệu có thể quan sát được vào một trường hợp, làm phong phú chúng bằng các công cụ tình báo về mối đe dọa tích hợp (như MISP và Cortex) và nhanh chóng xác định các mối đe dọa tiềm ẩn hoặc các hoạt động độc hại.

- Phân tích : TheHive cung cấp khả năng trực quan hóa và báo cáo giúp các nhóm bảo mật phân tích dữ liệu sự cố và xác định các mẫu, xu hướng hoặc mối tương quan. Bảng điều khiển có thể tùy chỉnh cung cấp một cái nhìn toàn diện về các sự cố đang diễn ra và tạo điều kiện cho việc ra quyết định hiệu quả.
- Tích hợp : TheHive hỗ trợ tích hợp với nhiều công cụ và dịch vụ của bên thứ ba, cho phép các nhóm tận dụng cơ sở hạ tầng an ninh mạng hiện có của họ. Các tích hợp phổ biến bao gồm nhập cảnh báo từ các hệ thống SIEM, hệ thống bán vé để báo cáo trường hợp và các công cụ phản hồi tự động để khắc phục sự cố.
- Cộng tác : Cộng tác trong thời gian thực là nền tảng của TheHive. Các thành viên trong nhóm có thể giao tiếp, chia sẻ kết quả và cập nhật chi tiết trường hợp trên nền tảng, hợp lý hóa việc liên lạc và đảm bảo rằng mọi người đều được thông báo.

TheHive cung cấp một bộ tính năng hỗ trợ cộng tác và quản lý sự cố hiệu quả cho các nhóm bảo mật. Trọng tâm của chức năng này là tạo và sắp xếp các trường hợp, lưu trữ thông tin quan trọng về các sự cố bảo mật, bao gồm mức độ nghiêm trọng và siêu dữ liệu liên quan khác. Khi các trường hợp phát triển, các thành viên trong nhóm có thể phân công và theo dõi các nhiệm vụ, đảm bảo phân công trách nhiệm rõ ràng và phản hồi kỹ lưỡng cho từng sự cố.

Một trong những khía cạnh có giá trị nhất của TheHive là khả năng xử lý các vật thể quan sát được hoặc các chỉ số thỏa hiệp. Các điểm dữ liệu này có thể được bổ sung thông qua tích hợp với các công cụ của bên thứ ba như MISP và Cortex, cung cấp cho các nhà phân tích bối cảnh và hiểu biết sâu sắc hơn. Bảng điều khiển có thể tùy chỉnh cho phép giám sát và trực quan hóa dữ liệu trường hợp theo thời gian thực, cho phép các nhóm xác định xu hướng và điểm bất thường một cách nhanh chóng.

Sự nhấn mạnh của TheHive vào cộng tác trong thời gian thực là một lợi thế quan trọng, thúc đẩy giao tiếp liền mạch giữa các thành viên trong nhóm. Khả năng tích hợp với các công cụ và dịch vụ bảo mật khác, chẳng hạn như hệ thống SIEM hoặc nền tảng bán vé, thông qua API RESTful của TheHive, tiếp tục mở rộng khả năng của nó.

Hơn nữa, TheHive cung cấp một hệ thống kiểm soát truy cập dựa trên vai trò để đảm bảo dữ liệu nhạy cảm được bảo mật trong khi duy trì quá trình kiểm tra toàn diện các hoạt động cho mục đích phân tích tuân thủ và sau sự cố. Nói chung, các tính năng này làm cho

TheHive trở thành một công cụ không thể thiếu cho các nhóm bảo mật điều hướng thế giới an ninh mạng phức tạp và có nhịp độ nhanh.

2.3.3. Cortex XDR – Giải pháp phát hiện và phản hồi mở rộng

Cortex XDR sử dụng dữ liệu từ Lớp dữ liệu để cung cấp khả năng lưu trữ dựa trên đám mây trong đối tượng thuê Cortex XDR, bao gồm tất cả các nguồn được truyền vào Cortex XDR — điểm cuối, tường lửa, nguồn đám mây và dữ liệu của bên thứ ba. Cortex XDR có thể tương quan và ghép dữ liệu này lại với nhau từ nhật ký trên các cảm biến nhật ký khác nhau của bạn để rút ra quan hệ nhân quả và dòng thời gian của sự kiện.

Cortex XDR—Ứng dụng Cortex XDR cung cấp khả năng hiển thị đầy đủ về tất cả dữ liệu của bạn trong Lớp dữ liệu. Ứng dụng này cung cấp một giao diện duy nhất mà từ đó bạn có thể điều tra và phân loại cảnh báo, thực hiện hành động khắc phục và xác định chính sách để phát hiện hoạt động độc hại trong tương lai.

Với Phát hiện và phản hồi điểm cuối (EDR), doanh nghiệp dựa vào dữ liệu điểm cuối như một phương tiện để kích hoạt các sự cố an ninh mạng. Khi tội phạm mạng và chiến thuật của chúng ngày càng tinh vi hơn, thời gian để xác định và ngăn chặn các vi phạm cũng tăng lên. Phát hiện và phản hồi mở rộng Cortex (XDR) vượt xa phương pháp EDR truyền thống là chỉ sử dụng dữ liệu điểm cuối để xác định và ứng phó với các mối đe dọa bằng cách áp dụng học máy trên tất cả dữ liệu doanh nghiệp, mạng, đám mây và điểm cuối của bạn. Cách tiếp cận này cho phép bạn nhanh chóng tìm và ngăn chặn các cuộc tấn công có mục tiêu cũng như hành vi lạm dụng nội bộ, đồng thời khắc phục các điểm cuối bị xâm phạm.

Để cung cấp bức tranh đầy đủ và toàn diện về các sự kiện và hoạt động xung quanh một sự kiện, Cortex XDR tương quan với nhật ký mạng tường lửa, dữ liệu thô điểm cuối và dữ liệu đám mây trên các cảm biến phát hiện của bạn. Hành động tương quan các nhật ký từ các nguồn khác nhau được gọi là Log Stitching và giúp xác định nguồn và đích của các quy trình và kết nối bảo mật được thực hiện qua mạng.

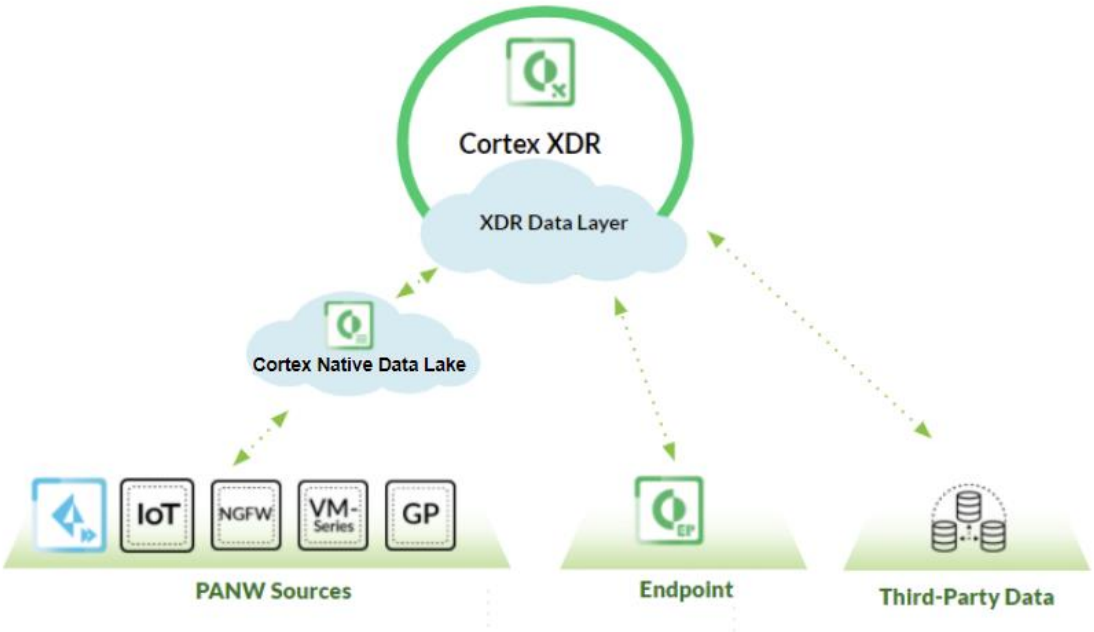
Log Stiting cho phép:

- Chạy các truy vấn điều tra dựa trên nhật ký điểm cuối và mạng được ghép.
- Tạo BIOC chi tiết và Quy tắc tương quan trên nhật ký từ Tường lửa thế hệ tiếp theo của Palo Alto Networks và dữ liệu điểm cuối thô.

- Điều tra các sự kiện mạng và điểm cuối tương quan trong Chế độ xem nguyên nhân mạng.
- Điều tra các cảnh báo Cortex XDR trên đám mây và Nhật ký kiểm tra đám mây trong Chế độ xem nguyên nhân đám mây.
- Điều tra các cảnh báo liên quan đến phần mềm dưới dạng dịch vụ (SaaS) cho các câu chuyện kiểm tra, chẳng hạn như nhật ký kiểm tra Office 365 và nhật ký chuẩn hóa, trong Chế độ xem quan hệ nhân quả của SaaS.

Việc ghép nhật ký giúp hợp lý hóa việc phát hiện và giảm thời gian phản hồi bằng cách loại bỏ nhu cầu phân tích thủ công trên các cảm biến dữ liệu khác nhau. Việc kết hợp dữ liệu qua tường lửa và điểm cuối cho phép bạn lấy dữ liệu từ các cảm biến khác nhau trong một chế độ xem thống nhất, mỗi cảm biến sẽ bổ sung thêm một lớp khả năng hiển thị khác. Ví dụ: khi một kết nối được nhìn thấy qua tường lửa và điểm cuối, điểm cuối có thể cung cấp thông tin về các quy trình liên quan và chuỗi thực thi trong khi tường lửa có thể cung cấp thông tin về lượng dữ liệu được truyền qua kết nối và các ID ứng dụng khác nhau có liên quan.

Khi phát hiện một tệp, hành vi hoặc kỹ thuật độc hại, Cortex XDR sẽ liên kết dữ liệu có sẵn trên các cảm biến phát hiện của bạn để hiển thị chuỗi hoạt động dẫn đến cảnh báo. Chuỗi sự kiện này được gọi là chuỗi nhân quả - causality chain. Chuỗi quan hệ nhân quả được xây dựng từ các quy trình, sự kiện, hiểu biết sâu sắc và cảnh báo liên quan đến hoạt động. Trong quá trình điều tra cảnh báo, bạn nên xem lại toàn bộ chuỗi quan hệ nhân quả để hiểu đầy đủ lý do xảy ra cảnh báo.



Hình 2.14. Cortex XDR architecture

Kiến trúc Cortex XDR hơi khác nhau giữa các phiên bản sản phẩm nhưng bao gồm một số thành phần tiêu chuẩn. Cả hai phiên bản đều dựa trên Cortex Data Lake và được thiết kế để tương quan với dữ liệu nhật ký trên các thiết bị của bạn.

Các thành phần nền tảng cơ bản bao gồm:

Ứng dụng Cortex XDR—giao diện người dùng (UI) cung cấp khả năng hiển thị vào Data Lake của bạn. Từ giao diện người dùng này, bạn có thể phân loại và điều tra các cảnh báo, thực hiện hành động khắc phục cũng như xác định chính sách phát hiện và phản hồi của mình.

Cortex Data Lake—tài nguyên lưu trữ để ghi nhật ký dựa trên đám mây được thiết kế để lưu giữ dữ liệu nhật ký của bạn từ tất cả các nguồn. Hồ dữ liệu tập trung dữ liệu của bạn, cho phép công cụ XDR tương quan với các sự kiện và tạo cảnh báo.

Các thành phần nền tảng nâng cao bao gồm:

Công cụ phân tích - một dịch vụ bảo mật sử dụng dữ liệu mạng và điểm cuối để phát hiện và ứng phó với các mối đe dọa. Nó áp dụng phân tích hành vi để xác định cả mối đe

dọa đã biết và chưa biết thông qua so sánh với hành vi của người dùng hoặc thiết bị đã biết và được chấp nhận.

Tường lửa thế hệ tiếp theo - tường lửa ảo hoặc tại chỗ mà bạn có thể sử dụng để thực thi các chính sách lưu lượng truy cập an toàn trong mạng của mình. Các tường lửa này bao gồm các công nghệ máy học để giúp phát hiện các mối đe dọa đã biết và chưa biết.

Prisma Access và GlobalProtect - các dịch vụ bạn có thể sử dụng để mở rộng khả năng bảo vệ tường lửa của mình cho người dùng di động và từ xa. Các dịch vụ này cho phép bạn chuyển tiếp nhật ký lưu lượng truy cập từ xa tới Hồ dữ liệu của mình để cho phép tương quan chung với nhật ký cục bộ.

Tường lửa và cảnh báo bên ngoài- thông qua tích hợp, bạn có thể nhập nhật ký và cảnh báo tường lửa bên ngoài vào hệ thống Cortex XDR của mình. Điều này có thể thực hiện được thông qua API Cortex XDR. Sau đó, những điểm dữ liệu này có thể được kết hợp với dữ liệu Cortex của bạn để cung cấp thêm ngữ cảnh cho các sự kiện và cho phép phản hồi kỹ lưỡng hơn.

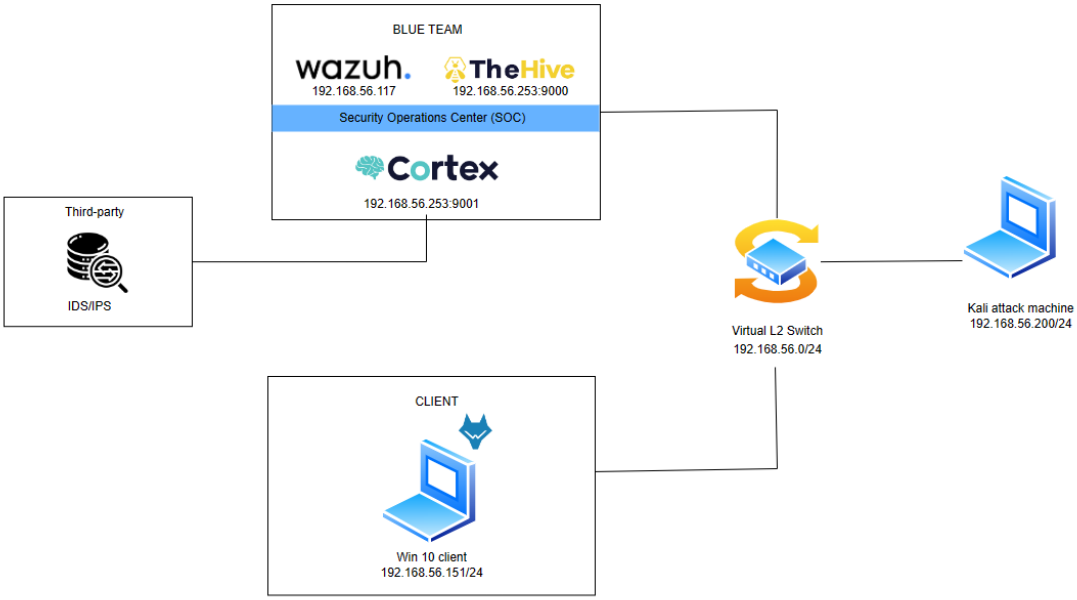
Cortex XDR agents- phần mềm được cài đặt trên các điểm cuối dùng để thu thập và chuyển tiếp dữ liệu. Các tác nhân này cũng có thể thực hiện phân tích cục bộ và có thể sử dụng thông tin về mối đe dọa của WildFire để cải thiện khả năng phát hiện các mối đe dọa. Tất cả dữ liệu được thu thập cũng được gửi đến Hồ dữ liệu để cùng phân tích.

Kết luận chương 2:

Chương 2 đã giới thiệu tổng quan về các giải pháp giám sát và ứng phó sự cố trong hệ thống mạng máy tính, đồng thời cũng đưa ra những giải pháp phổ biến như EDR, XDR, SIEM, SOAR. Ngoài ra cũng đã đề cập chi tiết đến những bô cộng cụ tương ứng với cấu trúc đề ra. Chương tiếp theo sẽ trình bày chi tiết về xây dựng và thiết kế một hệ thống giám sát tích hợp cùng với XDR cho một có quan, tổ chức hay doanh nghiệp dựa vào các thành phần đã đưa ra ở chương 2.

CHƯƠNG 3. TRIỂN KHAI VÀ THỰC NGHIỆM HỆ THỐNG

3.1. Mô hình hệ thống giám sát



Hình 3.1. Sơ đồ tổng quát hệ thống

Hệ thống giám sát được chia làm 4 thành phần chính:

- Blue team: bao gồm các giải pháp Wazuh, TheHive, và Cortex XDR.
- Kali attacker: Cài Sliver C2
- Client Win 10.

Bảng 3.1: Yêu cầu cầu hình của hệ thống giám sát

Máy chủ	CPU	RAM	Hệ điều hành
TheHive Cortex XDR	2	6 GB	Ubuntu 22.04
Kali attacker	4	2 GB	Kali linux
Client	2	2 GB	Win 10
Wazuh	4	6 GB	Ubuntu 22.04

Quá trình cài đặt nằm trong phần phụ lục

3.2. Kịch bản thử nghiệm

Phát hiện Framework Sliver C2

Sliver C2 là một Framework Command and Control (C2) được sử dụng để điều khiển từ xa các thiết bị endpoint bị xâm phạm. Sliver C2 là một giải pháp thay thế mã nguồn mở cho các khung C2 khác như Cobalt Strike và Metasploit. Sliver hỗ trợ các hệ điều hành Windows, macOS và Linux.

Sliver C2 Framework đã được liên kết đến nhiều mối đe dọa và malware theo các báo cáo ATTT của các tổ chức trên thế giới. Sliver C2 cho phép Attacker điều khiển và kết nối với các thiết bị đầu cuối bị xâm phạm, thực thi các câu lệnh từ xa trên các endpoint của nạn nhân, và trích xuất các dữ liệu nhạy cảm.

Trong quá khứ, Russian SVR (Cơ quan tình báo nước ngoài Nga) đã được báo cáo là đã sử dụng Sliver C2 để đảm bảo, duy trì kết nối với các mạng máy tính bị xâm phạm, trong khi BumbleBee loader đã được quan sát hành vi thả Sliver C2 implants sau khi hoàn thành bước xâm nhập ban đầu. Các mối đe dọa như APT29 và TA551 cũng được báo cáo là có liên quan đến việc sử dụng Framework.

Kịch bản sau đây sẽ minh họa cơ bản cách các mối đe dọa sử dụng Sliver C2 framework để thực thi các cuộc tấn công mạng và cách hệ thống phòng thủ phát hiện ra các hoạt động của Sliver trong hệ thống mạng. Trong kịch bản tấn công này sẽ không đề cập đến bước xâm nhập ban đầu (intial access) tới môi trường của nạn nhân vì Sliver C2 được thiết kế và sử dụng ở các bước (Command & Control) trong mô hình Cyber Kill Chain.

Khả năng của Sliver có trong danh sách sau đây:

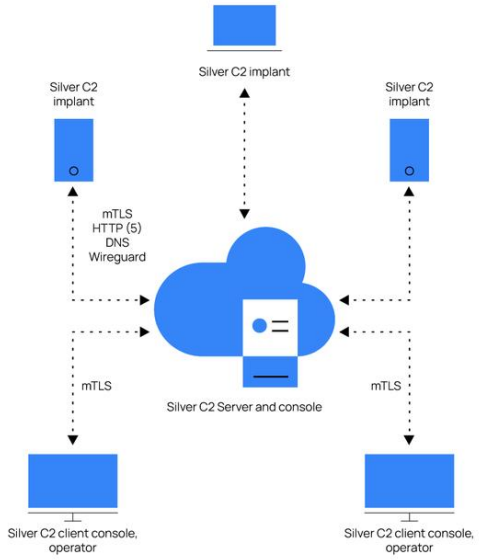
- **Shell**: Tính năng này cung cấp một reverse shell để giúp attacker tương tác với hệ thống nạn nhân và tương ứng với tactic execution trong MITRE ATT&CK framework.
- **UAC bypass**: Tính năng này được sử dụng để leo thang đặc quyền tương ứng với kĩ thuật bypass user account control (T1548).
- **Getsystem**: Tính năng này cũng được sử dụng để leo thang đặc quyền và tương ứng với kĩ thuật access token manipulation (T1134).

- **Migrate**: Tính năng này được sử dụng cho việc tránh bị phát hiện bởi hệ thống và dựa vào kĩ thuật process injection (T1055).
- **PsExec**: Tính năng này được sử dụng cho việc mở rộng phạm vi khai thác (lateral movement) tương ứng với kĩ thuật service execution (T1569).
- **Specific network port and use of SOCKS:** Tính năng này được sử dụng cho việc điều khiển và thực thi lệnh từ xa lần lượt tương ứng với kĩ thuật non-standard port and proxy.

Trong kịch bản này sẽ minh họa việc phát hiện các tính năng **shell, migration, và specific Network Port** của Sliver.

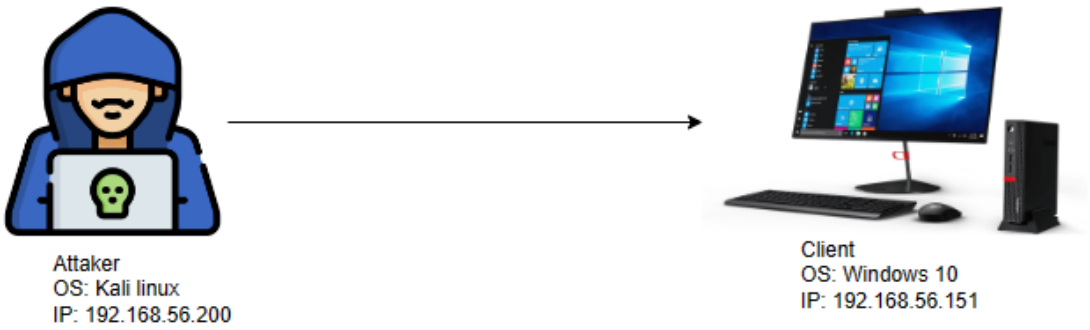
Thành phần của Sliver C2 Framework

- **Server console**: Giao diện chính khi file thực thi Sliver server được chạy
- **Sliver C2 server**: Sliver C2 server là một phần của file thực thi Sliver server có nhiệm vụ quản lý database nội bộ. Sliver C2 server khởi động và dừng các network listeners.
- **Client console**: Giao diện cho phép người dùng tương tác với C2 server.
- **Implant**: Đoạn mã độc được chạy trên endpoint của mục tiêu để cung cấp kết nối từ xa tới attacker



Hình 3.2. Các thành phần của Silver C2 Framework

Mô hình thử nghiệm



Hình 3.3. Mô hình thử kiệm Sliver C2

*Lưu ý: Để đảm bảo tấn công thành công, Windows Defender trên máy Windows 10 sẽ được tắt. Đồng thời để phát hiện các hành vi của Sliver, Windows 11 sẽ cài đặt thêm YARA và Sysmon kết hợp với Wazuh Agent.

Quá trình cài đặt YARA, Sysmon kết hợp với Wazuh Agent sẽ được đề cập ở phần phụ lục.

Wazuh rules phát hiện hành vi **Shell**:

Rule ID: 107000 kích hoạt khi tiến trình Sliver tạo ra reverse shell.

```
<group name="sliver,">
  <!-- Rule for detecting potential sliver shell execution -->
  <rule id="107000" level="12">
    <if_sid>61603</if_sid>
    <field name="win.eventdata.parentImage" type="pcre2">.exe</field>
    <field name="win.eventdata.image" type="pcre2">powershell.exe</field>
    <field name="win.eventdata.commandLine" type="pcre2"> -NoExit -Command \[Console\]::OutputEncoding=\[Text.UTF8Encoding]::UTF8</field>
    <description>Possible Sliver C2 activity: shell executed: $(win.eventdata.commandLine).</description>
    <mitre>
      <id>T1086</id>
    </mitre>
  </rule>
```

Giải thích rule:

```
<rule id="107000" level="12">
```

Dòng này định nghĩa rule có ID là “107000” trong hệ thống SIEM và có mức độ nghiêm trọng là “12”

```
<if_sid>61603</if_sid>
```

Dòng này chỉ định điều kiện của rule: Rule chỉ được kích hoạt nếu sự kiện liên quan thỏa mãn rule cha có ID là “61603” - Sự kiện là sự kiện của Sysmon có Event ID là 1 – Process creation

```
<rule id="61603" level="0">
  <if_sid>61600</if_sid>
  <field name="win.system.eventID">^1$</field>
  <description>Sysmon - Event 1: Process creation $(win.eventdata.description)</description>
  <options>no_full_log</options>
  <group>sysmon_event1,</group>
</rule>
```

```
<field name="win.eventdata.parentImage" type="pcre2">.exe</field>
```

Dòng này định nghĩa trường sẽ được rule kiểm tra chỉ định sẵn là trường "parentImage" – Tên image của tiến trình cha có kiểu mẫu chuỗi chứa .exe sử dụng regular expression (biểu thức chính quy) loại pcre2. Kiểu chuỗi mang ý nghĩa nếu nếu tiến trình cha là bất kì file thực thi nào có kiểu file là exe.

```
<field name="win.eventdata.image" type="pcre2">powershell.exe</field>
```

Dòng này định nghĩa rule kiểm tra thêm một trường nữa đó là trường "image", đại diện cho tên của tiến trình đã được thực thi có kiểu mẫu chuỗi là "powershell.exe". Kiểu chuỗi này mang ý nghĩa nếu tiến trình được thực thi là Powershell.

```
<field name="win.eventdata.commandLine" type="pcre2"> -NoExit -Command
¥[Console¥]::OutputEncoding=¥[Text.UTF8Encoding]::UTF8</field>
```

Dòng này định nghĩa trường thứ ba mà rule sẽ kiểm tra là trường "commandLine". Trường này chứa câu lệnh mà được thực thi bởi tiến trình nếu có khớp với biểu thức chính quy "-NoExit -Command ¥[Console¥]::OutputEncoding=¥[Text.UTF8Encoding]::UTF8". Biểu thức chính quy này mang ý nghĩa nếu tham số câu lệnh được thực thi liên quan đến việc mã hóa UTF-8.

```
<description>Possible Sliver C2 activity: shell executed:
$(win.eventdata.commandLine).</description>
```

Dòng này cung cấp mô tả cho rule: Phát hiện hành vi liên quan đến Sliver C2 có thể là thực thi shell. Phần mô tả có chứa một biến $(win.eventdata.commandLine) mà sẽ được thay thế khi rule được kích hoạt

```
<mitre>
    <id>T1086</id>
</mitre>
```

Phần này cung cấp thêm thông tin về kĩ thuật phát hiện trong rule có ID (T1086 - Command and Scripting Interpreter: PowerShell) của kĩ thuật được tham chiếu tới khung MITRE ATT&CK.

Rule ID: 107001 kích hoạt khi một tiến trình tạo ra một remote thread.

```
<rule id="107001" level="9">
  <if_sid>61610</if_sid>
  <field name="win.eventdata.sourceImage" type="pcre2">.exe</field>
  <field name="win.eventdata.targetImage" type="pcre2">C:\\\\Program\ Files\\\\D*[A-Za-z0-9_.]*\\\\[A-Za-z0-9_.]*\\\\[A
-Za-z0-9_.]*\\\\[A-Za-z0-9_.]*.exe$</field>
  <description>Suspicious process injection activity detected from $(win.eventdata.sourceImage) on $(win.eventdata
.targetImage).</description>
  <mitre>
    <id>T1055</id>
  </mitre>
</rule>
```

Giải thích rule:

```
<rule id="107001" level="9">
```

Dòng này định nghĩa rule có ID là "107001" trong hệ thống SIEM và có mức độ nghiêm trọng là "9"

```
<if_sid>61610</if_sid>
```

Dòng này chỉ định điều kiện của rule: Rule chỉ được kích hoạt nếu sự kiện liên quan thỏa mãn rule cha có ID là "61610" - Sự kiện là sự kiện của Sysmon có Event ID là 8 – CreateRemoteThread.

```
<rule id="61610" level="0">
  <if_sid>61600</if_sid>
  <field name="win.system.eventID">^8$</field>
  <description>Sysmon - Event 8: CreateRemoteThread by $(win.eventdata.sourceImage) on $(win.eventdata.targetImage), possible process injection</description>
  <options>no_full_log</options>
  <group>sysmon_event8,</group>
</rule>
```

```
<field name="win.eventdata.sourceImage" type="pcre2">.exe</field>
```

Dòng này định nghĩa trường sẽ được rule kiểm tra chỉ định sẵn là trường "parentImage" – Tên image của tiến trình cha có kiểu mẫu chuỗi chứa .exe sử dụng regular expression (biểu thức chính quy) loại pcre2. Kiểu chuỗi mang ý nghĩa nếu nếu tiến trình cha là bất kì file thực thi nào có kiểu file là exe.

```
<field name="win.eventdata.targetImage" type="pcre2">C:¥¥¥¥Program¥ Files¥¥¥¥D*[A-Za-z0-
9_.]*¥¥¥¥[A-Za-z0-9_.]*¥¥¥¥[A-Za-z0-9_.]*¥¥¥¥[A-Za-z0-9_.]*.exe$</field>
```

Dòng này định nghĩa rule kiểm tra thêm trường "targetImage", đây là đường dẫn của tiến trình được chọn, nếu đường dẫn đó thỏa mãn biểu thức chính quy trên.

Biểu thức chính quy này tìm kiếm một đường dẫn của một file thực thi .exe nằm trong thư mục "Program Files"

```
<description>Suspicious process injection activity detected from
$(win.eventdata.sourceImage) on $(win.eventdata.targetImage).</description>
```

Dòng này cung cấp mô tả cho rule: Phát hiện hành vi nghi ngờ process injection. Phần mô tả có chứa hai biến `$(win.eventdata.sourceImage)` và `$(win.eventdata.targetImage)` mà sẽ được thay thế khi rule được kích hoạt.

```
<mitre>
    <id>T1055</id>
</mitre>
```

Phần này cung cấp thêm thông tin về kĩ thuật phát hiện trong rule có ID (T1055 - Process Injection) của kĩ thuật được tham chiếu tới khung MITRE ATT&CK.

YARA Rule để phát hiện Sliver C2 Implant:

```
rule sliver_client : c2 implant {
    meta:
        description = "Sliver C2 Implant"
        url = "https://github.com/BishopFox/sliver"

    strings:
        $s1 = "sliverpb"
        $s2 = "/sliver/"
        $s3 = "github.com/bishopfox/sliver/"
        $p1 = {66 81 ?? 77 67}
        $p2 = { 81 ?? 68 74 74 70 [2-32] 80 ?? 04 73 }
        $p3 = { 66 81 ?? 64 6E [2-20] 80 ?? 02 73 }
        $p4 = {  81 ?? 6D 74 6C 73  }

    condition:
        2 of ($p*) or any of ($s1,$s2,$s3) and filesize < 50MB
}
```

Giải thích về YARA rule trên:

```
rule sliver_client : c2 implant
```

Dòng đầu tiên định danh tên của rule có tên là "sliver_client". Rule bao gồm cả comment "c2 implant" để cung cấp thêm thông tin về mục đích của rule

```
    meta:
        description = "Sliver C2 Implant"
        url = "https://github.com/BishopFox/sliver"
```

Các dòng này định nghĩa metadata (tạm dịch: siêu dữ liệu) bao gồm một description (mô tả) về rule và một URL trỏ đến trang Github nơi mà có thể cung cấp chi tiết thông tin về Sliver C2.

```
    strings:
        $s1 = "sliverpb"
        $s2 = "/sliver/"
        $s3 = "github.com/bishopfox/sliver/"
```

```
        $p1 = {66 81 ?? 77 67}
        $p2 = { 81 ?? 68 74 74 70 [2-32] 80 ?? 04 73 }
        $p3 = { 66 81 ?? 64 6E [2-20] 80 ?? 02 73 }
        $p4 = {  81 ?? 6D 74 6C 73  }
```

Trong phần này, một số chuỗi được định nghĩa dưới dạng các biến bằng cách sử dụng kí tự dollar ($). Các biến từ $s1 đến $s3 đại diễn cho mội chuỗi văn bản bản rõ và các biến từ $p1 đến $p4 đại diện cho kiểu chuỗi dưới dạng hệ thập lục phân.

```
condition:
        2 of ($p*) or any of ($s1,$s2,$s3) and filesize < 50MB
```

Dòng này định nghĩa điều kiện của rule. Điều kiện được định nghĩa ở đây là:

Nếu hai trong nhóm biến $p được tìm thấy trong file (‘**2 of ($p*)**’) hoặc bất kì chuỗi nào trong nhóm biến $s1, $s2, $s3 (‘**any of ($s1, $s2, $s3)**’) được tìm thấy. Điều kiện thứ hai là kiểm tra xem dung lượng của file có nhỏ hơn 50mb (‘**filesize < 50MB**’)

Wazuh rule phát hiện ra **Sliver C2 Implants**:

Rule ID: 100770 kích hoạt khi một file được thay đổi trong đường dẫn C:\Users\vboxuser\Downloads

```
<group name="syscheck,">
  <rule id="100770" level="7">
    <if_sid>550</if_sid>
    <field name="file">C:\\Users\\vboxuser\\Downloads</field>
    <description>File modified in C:\Users\vboxuser\Downloads directory.</description>
  </rule>
  <rule id="100771" level="7">
    <if_sid>554</if_sid>
    <field name="file">C:\\Users\\vboxuser\\Downloads</field>
    <description>File added to C:\Users\vboxuser\Downloads  directory.</description>
  </rule>
</group>
```

```
<if_sid>550</if_sid>
```

Dòng này chỉ định điều kiện của rule: Rule chỉ được kích hoạt nếu sự kiện liên quan thỏa mãn rule cha có ID là “550” – Integrity checksum changed (Giám sát sự thay đổi file)

```
<rule id="550" level="7">
  <category>ossec</category>
  <decoded_as>syscheck_integrity_changed</decoded_as>
  <description>Integrity checksum changed.</description>
  <mitre>
    <id>T1565.001</id>
  </mitre>
  <group>syscheck,syscheck_entry_modified,syscheck_file,pci_
      ,</group>
</rule>
```

```
<field name="file">C:¥¥Users¥¥vboxuser¥¥Downloads</field>
```

Dòng này chỉ ra đường dẫn thư mục cần được giám sát, trong trường hợp này là giám sát tất cả các file có trong thư mục C:\Users\vboxuser\Downloads

```
<description>File modified in C:¥Users¥Bob¥Downloads directory.</description>
```

Dòng này đưa ra mô tả của rule: Phát hiện có sự thay đổi file trong đường dẫn C:\Users\Bob\Downloads

Rule ID: 100771 kích hoạt khi một file được thêm vào đường dẫn C:\Users\Bob\Downloads

```
<rule id="100771" level="7">
    <if_sid>554</if_sid>
    <field name="file">C:¥¥Users¥¥vboxuser¥¥Downloads</field>
    <description>File added to C:¥Users¥vboxuser¥Downloads  directory.</description>
</rule>
```

```
<if_sid>554</if_sid>
```

Dòng này chỉ định điều kiện của rule: Rule chỉ được kích hoạt nếu sự kiện liên quan thỏa mãn rule cha có ID là “554” – Filed added to the system (File được add vào hệ thống)

```
<group name="syscheck,">

  <rule id="554" level="5">
    <category>ossec</category>
    <decoded_as>syscheck_integrity_changed</decoded_as>
    <description>File added to system </description>
    <group>syscheck,syscheck_entry_modified,syscheck_file, pci_dss,</group>
  </rule>
```

```
<field name="file">C:¥¥Users¥¥vboxuser¥¥Downloads</field>
```

Dòng này chỉ ra đường dẫn thư mục cần được giám sát, trong trường hợp này là giám sát tất cả các file có trong thư mục C:\Users\vboxuser\Downloads

```
<description>File added to C:¥Users¥vboxuser¥Downloads directory.</description>
```

Dòng này đưa ra mô tả của rule: Phát hiện có file được thêm vào trong đường dẫn C:\Users\vboxuser\Downloads

Rule ID: 100881 kích hoạt khi một file được xác định là malware bởi YARA

```
<group name="yara,">
  <rule id="100880" level="0">
    <decoded_as>yara_decoder</decoded_as>
    <description>Yara grouping rule</description>
  </rule>

  <rule id="100881" level="12">
    <if_sid>100880</if_sid>
    <match>wazuh-yara: INFO - Scan result: </match>
    <description>File "$(yara_scanned_file)" is a positive match. Yara rule: $(yara_rule)</description>
  </rule>
</group>
```

```
<if_sid>100880</if_sid>
```

Dòng này chỉ định điều kiện của rule: Rule chỉ được kích hoạt nếu sự kiện liên quan thỏa mãn rule cha có ID là "100880" – Yara grouping rule

Sử dụng câu lệnh giám sát để phát hiện kết nối mạng:

Sliver C2 server sẽ lắng nghe trên các cổng mặc định nếu mà không được cấu hình. Các Implants được tạo ra sử dụng HTTPS, mTLS và Wireguard để giao tiếp tương ứng thông qua cổng TCP/443, TCP/8888 và UDP/51820.

Sử dụng câu lệnh giám sát thông qua tính năng Active Response của Wazuh để phát hiện cổng mà Sliver C2 đang lắng nghe

Cấu hình file agent trên Windows 10:

```
<!-- Sliver mtls listening on port 8888 -->
  <localfile>
    <log_format>full_command</log_format>
    <command>netstat -ano | findstr :8888 </command>
    <alias>Detecting possible Sliver communication</alias>
    <frequency>300</frequency>
  </localfile>
```

Giải thích cấu hình trên:

```
<localfile>
..
</localfile>
```

Tag <localfile> và kết thúc </localfile> khai báo khối cấu hình với ý nghĩa là cấu hình này cho localfile. Một khối cấu hình localfile cho phép Wazuh agent có thể giám sát các file log hoặc thực thi câu lệnh trên hệ thống mà agent được cài đặt

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

Dòng này định nghĩa kiểu định dạng của dữ liệu log sẽ được thu thập. Trong trường hợp này "full_command" được sử dụng, có nghĩa là tất cả command được thực thi bởi agent sẽ được bao gồm vào trong dữ liệu log

```
<command>netstat -ano | findstr :8888 </command>
```

Dòng này định nghĩa câu lệnh mà Wazuh Agent sẽ thực thi để thu thập log. Trong trường hợp này câu lệnh là "netstat -ano | findstr :8888 " đây là câu lệnh Windows dùng để lấy thống kê về kết nối network và lọc ra các kết quả để tìm các tiến trình đang lắng nghe trên cổng 8888

```
<alias>Detecting possible Sliver communication</alias>
```

Dòng này cung cấp tên và mô tả mục đích của khối cấu hình này: Phát hiện kết nối mang signature của Sliver.

```
<frequency>300</frequency>
```

Dòng này định nghĩa tần xuất mà Wazuh Agent sẽ thực thi command để thu thập dữ liệu log. Trong trường hợp này, tần xuất được đặt là 300 giây, agent sẽ thực thi command được định nghĩa ở trên mỗi 300 giây để kiểm tra xem có tiến trình nào đang lắng nghe trên cổng 8888.

Rule ID: 107002 kích hoạt khi port 8888 được sử dụng để lắng nghe kết nối

```
<group name="mtls-port,">
<!-- Command monitoring rule for specific Sliver C2 port communication -->
  <rule id="107002" level="10">
    <if_sid>530</if_sid>
    <match>ossec: output: 'Detecting possible Sliver communication'</match>
    <description>Possible Sliver C2 activity: Detected port 8888
listening.</description>
  </rule>
```

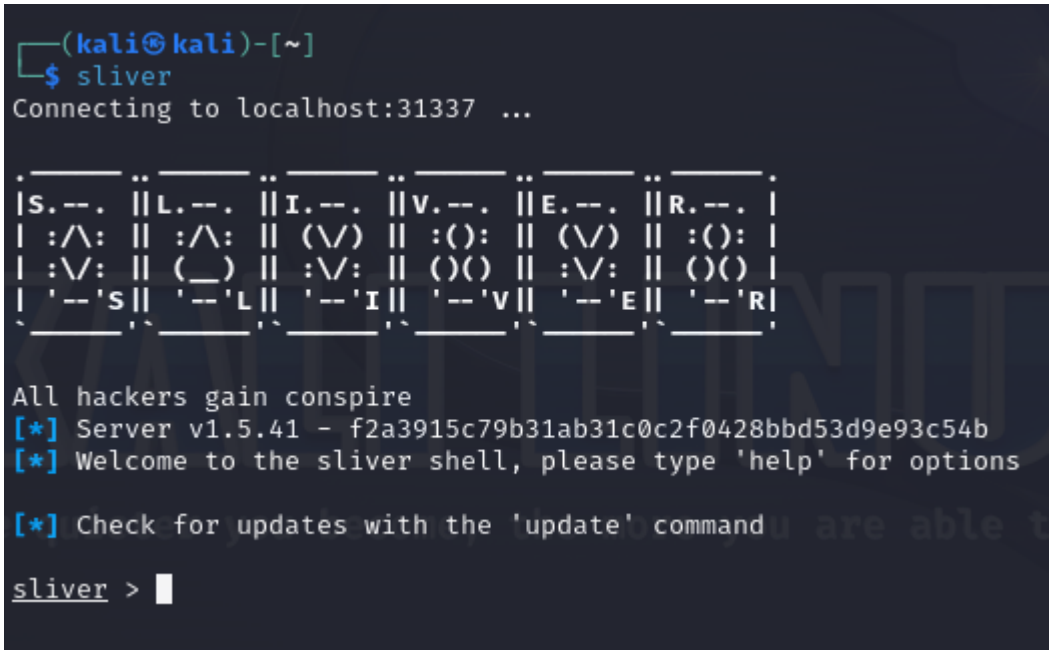
```
</group>
```

Rule này có ý nghĩa tạo ra cảnh báo có tiêu đề "Possible Sliver C2 activity: Detected port 8888 listening" khi phát hiện có chuỗi "Detecting possible Sliver communication" khi log của OSSEC được đẩy về Wazuh

Giả lập tấn công:

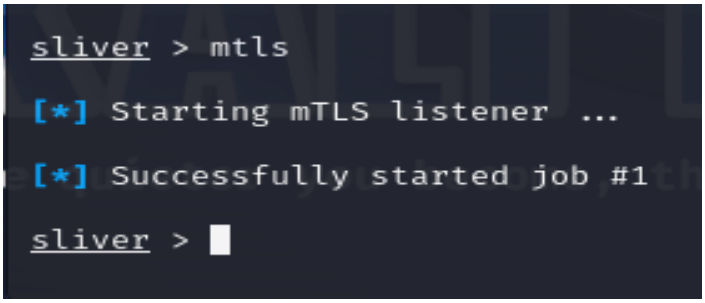
Trên máy Kali Linux:

Tiến hành tải và khởi chạy Sliver C2 Framework:



Hình 3.4. Khởi chạy Sliver C2 Framework

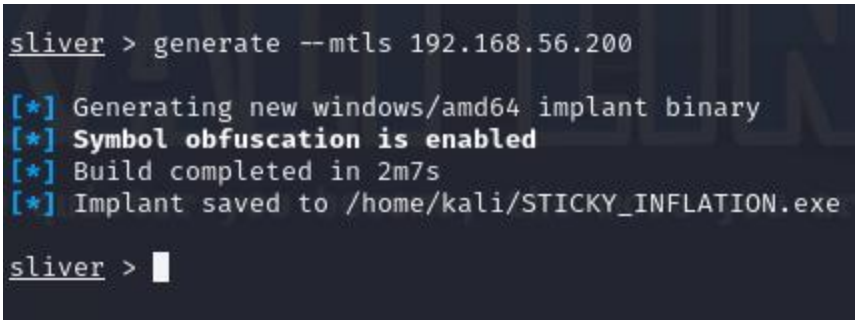
Tạo listener



Hình 3.5. Tạo listener trong Sliver

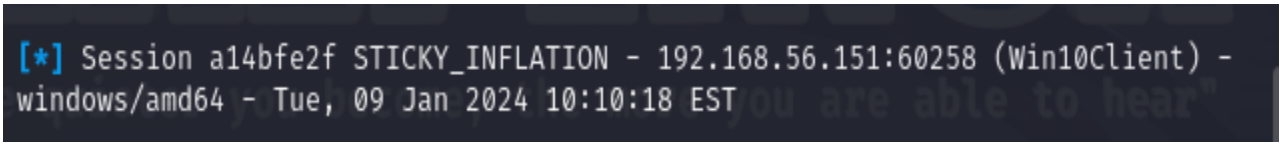
Tạo một mTLS implant để được thực thi trên endpoint của nạn nhân:

Cú pháp: generate –mtls <ip_attacker>



Hình 3.6. Tạo payload để chạy trên máy nạn nhân

Thực hiện tải file vừa được tạo ra bởi Sliver sang máy nạn nhân - Windows 11 và thực thi file đó. Khi mà việc thực thi file implant được hoàn thành, Attacker sẽ có được một session trên terminal của Kali Linux

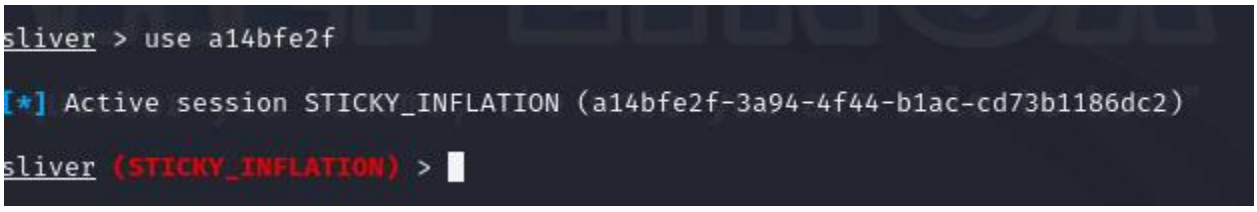


Hình 3.7. Kết quả sau khi thực thi file trên máy user

Liệt kê các sessions hiện có và chọn session để sử dụng:

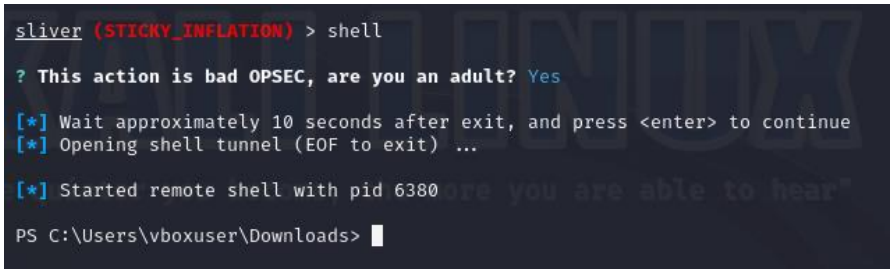
Hình 3.8. Liệt kê các sessions hiện có

Kích hoạt sử dụng session được tạo ra bởi Windows 10 endpoint:



Hình 3.9. Sử dụng phiên kết nối đến windows 10

Tạo reverse shell



Hình 3.10. Tạo reverse shell

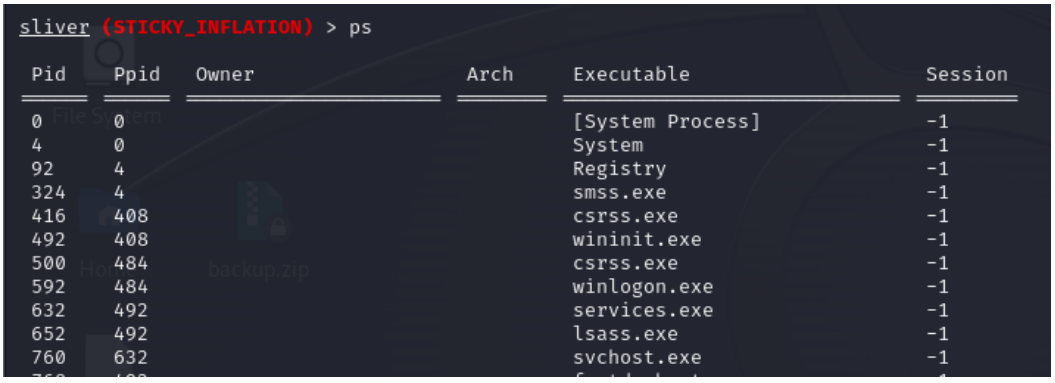
Process Injection:

Sliver C2 có khả năng inject file thực thi implant vào một tiến trình khác bằng cách tạo ra một remote threat trong process hợp lệ.

Thực hiện list ra danh sách các tiến trình đang chạy trên máy Windows.

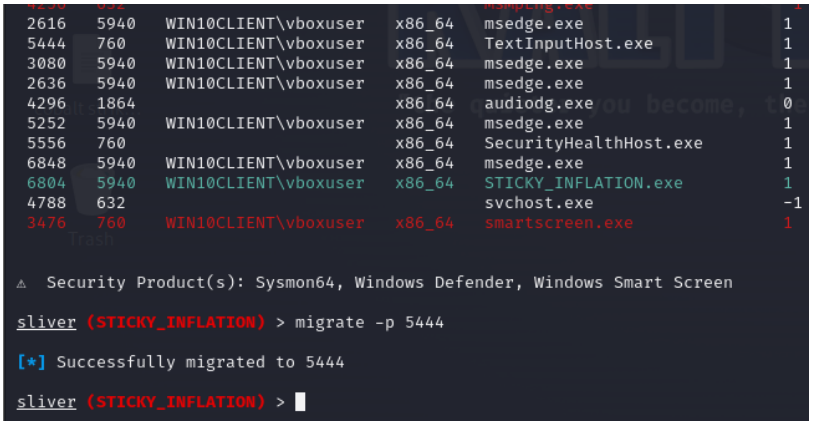
Tìm ra PID của một tiến trình mà Attacker muốn inject vào, trong kịch bản này sẽ là tiến trình TextInputHost.exe

Kiểm tra PID trên máy nạn nhân:



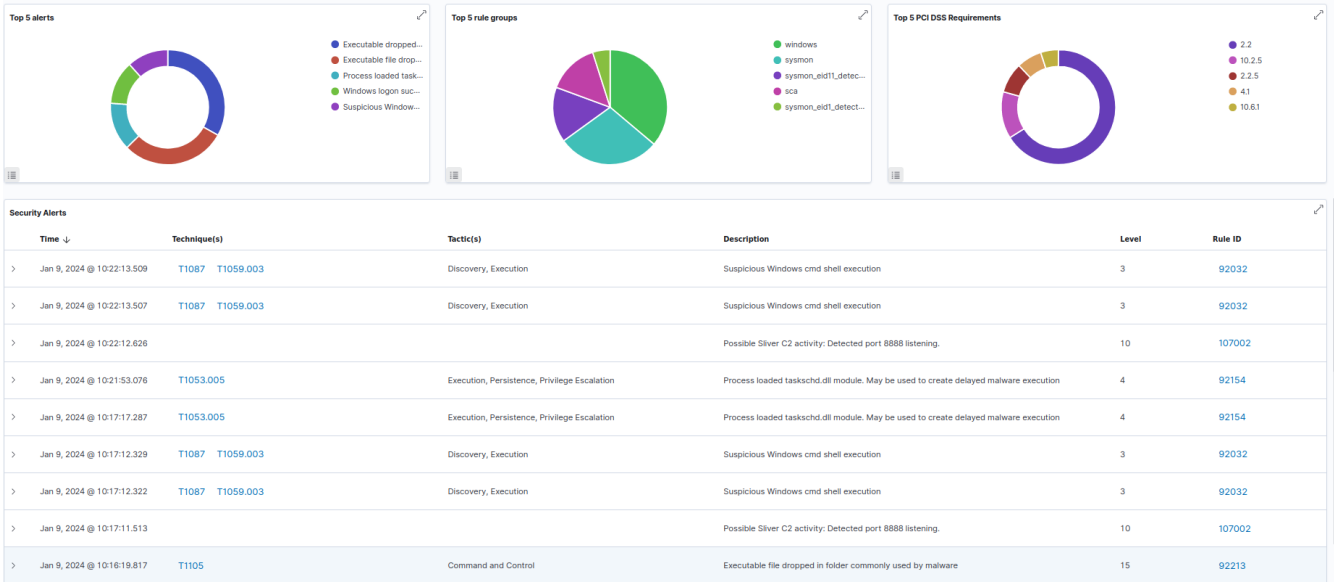
Hình 3.11. PID trên Windows 10

Sau khi kiểm tra PID trên máy nạn nhân, ta thực hiện inject vào process Text InputHost.exe.



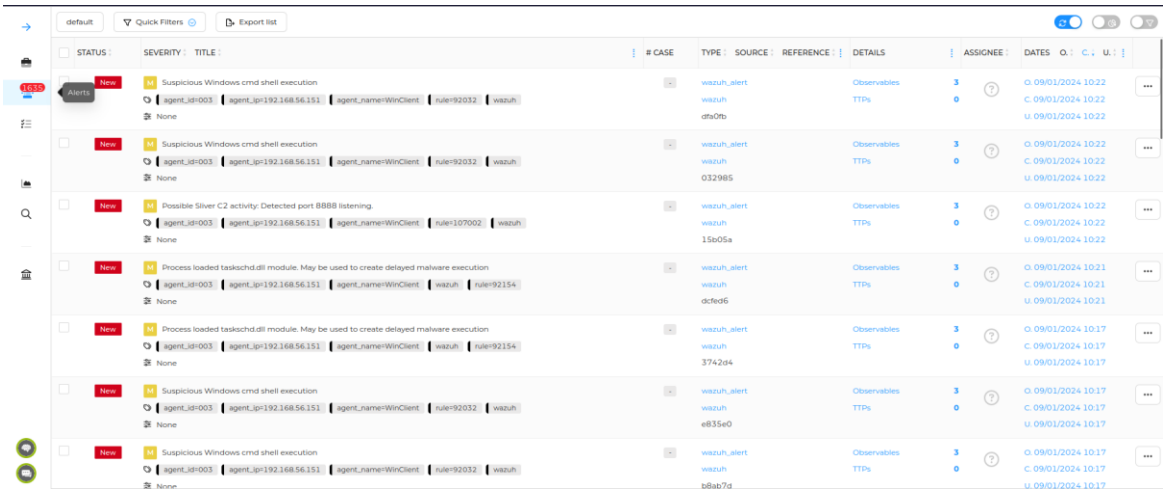
Hình 3.12. Inject vào tiến trình TextInputHost.exe

Kết quả trên hệ thống giám sát:



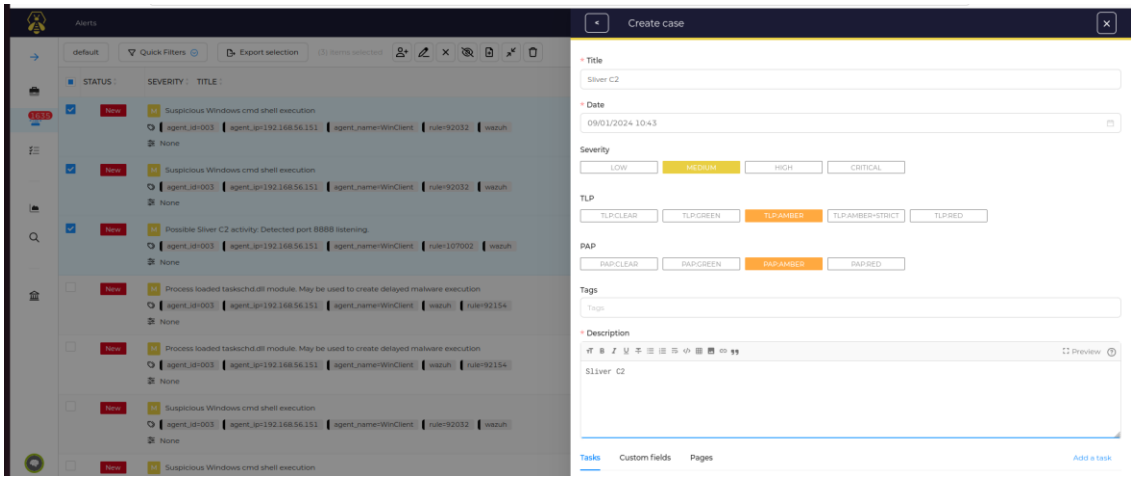
Hình 3.13. Kết quả trên Wazuh siem

Các alert trên hệ thống giám sát được đẩy về trình quản lý TheHive



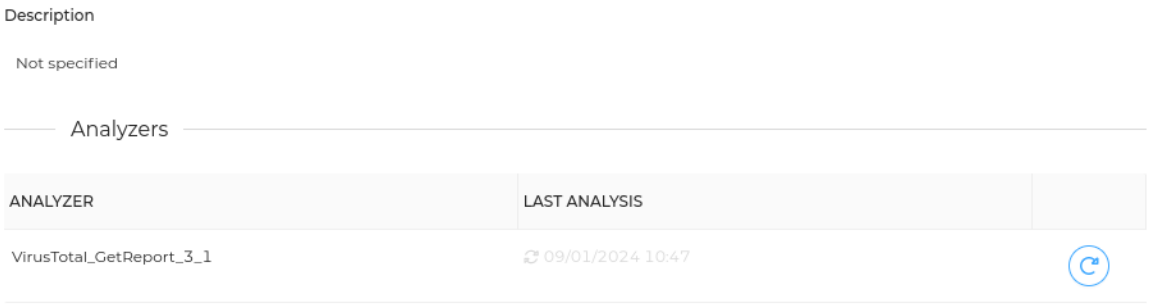
Hình 3.14. Alerts được đẩy về TheHive

Sau khi các alerts từ phía Wazuh siem được đẩy về TheHive, ta tiến hành tạo những incidents để thực hiện quá trình phân tích và xử lý sự cố.

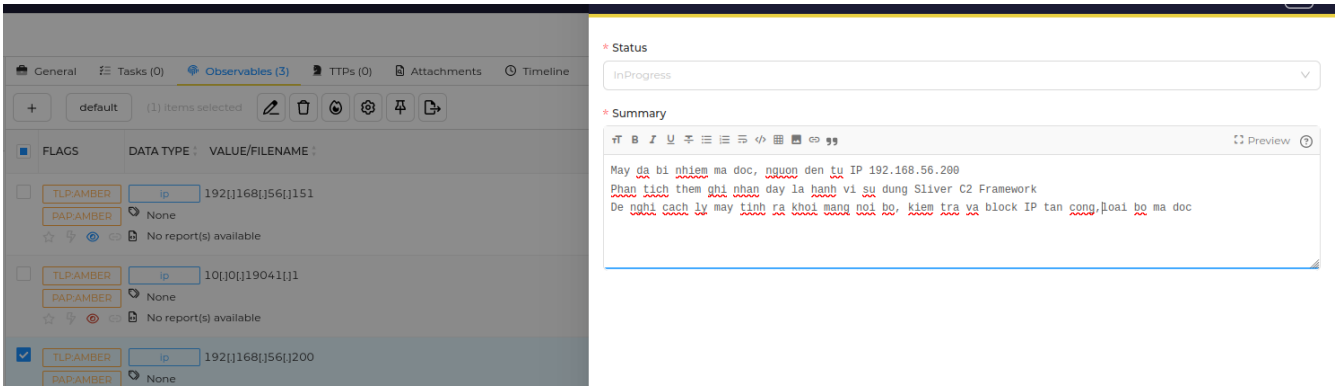


Hình 3.15. Tạo incident từ những alert được đẩy về

Sau khi incident được tạo, phần Observable sẽ có những IOC liên quan đến được them vào, lúc này là lúc Cortex XDR thực hiện chạy các Analyzers để kiểm tra các IOC xem có dấu hiệu độc hại hay không

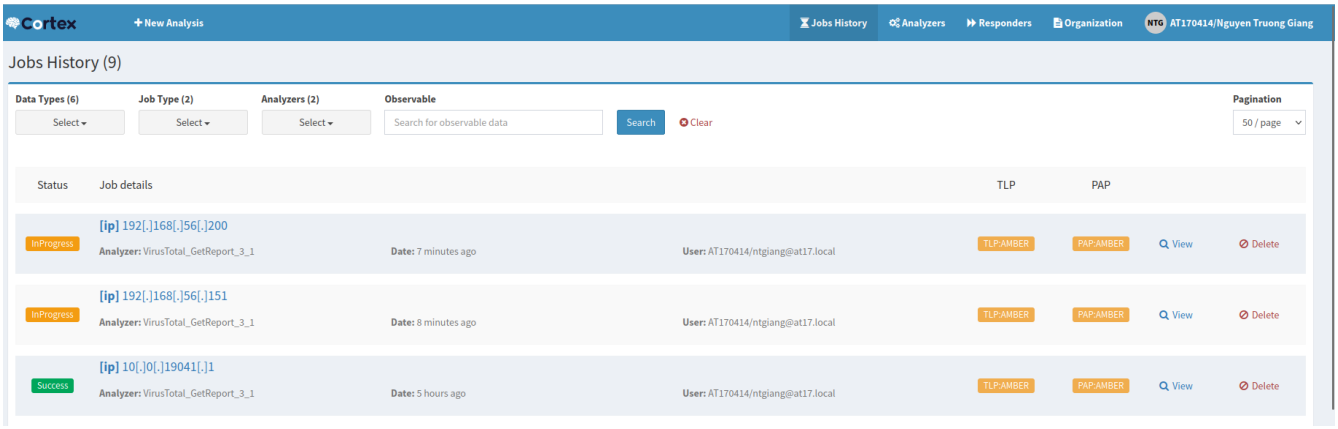


Hình 3.16. Analyzer VirusTotal_GetReport_3_1



Hình 3.17. Giao diện note incident trên TheHive

Phía bên Cortex, ghi nhận đã có những Jobs được tạo ra từ yêu cầu phân tích của TheHive.



Hình 3.18. Jobs history của Cortex XDR

3.3. Đánh giá kết quả

Từ kịch bản tấn công trên, báo cáo chuyên đề kỹ nghệ đã trình bày khả năng thu thập, giám sát và phân tích của hệ thống SOC (Security Operation Center) dựa trên các công cụ mã nguồn mở trong mạng doanh nghiệp. Hệ thống giám sát an toàn đề xuất hoàn toàn đảm nhiệm tốt vai trò giám sát của mình trong hệ thống mạng và đảm bảo mọi hoạt động trong hệ thống được diễn ra trơn tru và an toàn nhất. Điều này có thể được cải thiện và đánh giá tốt hơn phụ thuộc vào trình độ của giám sát viên và quản trị viên của hệ thống giám sát.

Kết luận chương 3

Chương 3 đã đưa mô hình triển khai thực tế của hệ thống, đưa ra kịch bản thử nghiệm và diễn giải chi tiết nội dung của các bộ luật dùng để phát hiện qua đó thể hiện được khả năng giám sát và phát hiện của hệ thống đề xuất đồng thời cũng đã kiểm tra được khả năng phân tích tự động mạnh mẽ của Cortex XDR.

KẾT LUẬN CHUYÊN ĐỀ

Kết quả đạt được chuyên đề

Sau quá trình nghiên cứu và xây dựng giải pháp "Nghiên cứu giải pháp phát hiện vả phản hồi mở rộng Extended Detection and Response (XDR)", chuyên đề đã thực hiện cấu hình hệ thống bởi các công cụ giám sát và xử lý sự cố sử dụng công cụ mã nguồn mở.

Hệ thống đã hoàn thành mục tiêu đã đề ra là xây dựng được hệ thống giám sát sử dụng các giải pháp như Wazuh, TheHive và tích hợp Cortex XDR, đem lại khả năng tương thích và khả năng mở rộng cao mà không phụ thuộc vào các hệ sinh thái có sẵn của các giải pháp thương mại như Qradar, Arbor, Splunk,…

Các hạn chế của chuyên đề

Trong quá trình thực hiện, nhóm em nhận thấy chuyên đề còn tồn tại một số hạn chế như sau:

- Chưa phát huy tối đa khả năng mạnh mẽ của XDR, thực nghiệm chưa đào sâu vào những chức năng của XDR
- Chưa khai thác được hết các chức năng mà các giải pháp đem lại
- Cần ghi nhận thông số vận hành thực tế để có thể lựa chọn cấu hình cho hệ thống giám sát tránh tình trạng cao tải của hệ thống
- Bộ luật cần được xây dựng để có thể phủ được đa phần các mối đe dọa dựa trên mô hình ATTCK

Hướng phát triển tương lai

Dựa trên các hạn chế được ra ở phía trên, nhóm em có một số hướng phát triển để có thể khiến giải pháp đề xuất toàn diện hơn và đáp ứng được nhu cầu của các tổ chức:

- Tích hợp phần mềm mã nguồn mở Velociraptor, Sublime Security để mở rộng phạm vi giám sát hệ thống, thực hiện điều tra chuyên sâu các thiết bị đồng thời thực hiện ứng phó nhanh khi có sự cố diễn ra
- Tích hợp thêm các nguồn Threat Inteligence như VirusTotal, AbuseIPDB,.. để hỗ trợ trong việc phân tích và xử lí các sự cố
- Tối ưu các bộ luật trên SIEM để giảm thiểu tỉ lệ False Postive.

TÀI LIỆU THAM KHẢO

[1] An toàn thông tin là gì, *Tek4.vn*, https://tek4.vn/an-toan-thong-tin-la-gi

[2] An ninh mạng là gì, giải thích về an ninh mạng, https://aws.amazon.com/vi/what-is/cybersecurity/

[3] Bảo mật an ninh mạng khi chuyển đổi số, *Báo Nhân dân, Anh Tuấn,* https://nhandan.vn/bao-mat-an-ninh-mang-khi-chuyen-doi-so-post769699.html

[4] Chỉ 17% doanh nghiệp tại Việt Nam sẵn sàng trước các mối đe dọa an ninh mạng, *Ban cơ yếu chính phủ - An toàn thông tin, theo VTV,* https://antoanthongtin.vn/an-toan-thong-tin/chi-17-doanh-nghiep-tai-viet-nam-san-sang-truoc-cac-moi-de-doa-an-ninh-mang-108822

[5] 10 loại hình tấn công mạng phổ biến, *Kurt Baker,* https://www.crowdstrike.com/cybersecurity-101/cyberattacks/most-common-types-of-cyberattacks/

[6] What is SOAR, security orchestration automation response, https://www.ibm.com/topics/security-orchestration-automation-response

[7] What is SIEM, https://www.ibm.com/topics/siem

[8] What is Enpoint Detection and Response (EDR), https://www.crowdstrike.com/cybersecurity-101/endpoint-security/endpoint-detection-and-response-edr/

[9] What is Extended Dectection and Response (XDR), *Paloalto Networks* https://www.paloaltonetworks.com/cyberpedia/what-is-extended-detection-response-XDR

[10] What is XDR concepts and benefits, https://securityboulevard.com/2023/06/what-is-xdr-concepts-and-benefits/

[11] What is XDR, https://www.ontinue.com/what-is-xdr/

[12] XDR, understanding open xdr, https://www.exabeam.com/explainers/xdr/understanding-open-xdr/

[13] Components of Wazuh. https://documentation.wazuh.com/current/getting-started/components/index.html

[14] Tổng quan về TheHive, https://appmaster.io/vi/blog/tong-quan-ve-to-ong

[15] Cortex XDR overview, https://docs-cortex.paloaltonetworks.com/r/Cortex-XDR/Cortex-XDR-Pro-Administrator-Guide/Overview

[16] Cortex XDR paloalto architecture capabilities overview, *Shani Verma,* https://www.linkedin.com/pulse/cortex-xdr-palo-alto-architecture-capabilities-overview-shani-verma/

[17] Detecting Sliver C2 framework with Wazuh, https://wazuh.com/blog/detecting-sliver-c2-framework-with-wazuh/

PHỤ LỤC

Phụ lục 1. Các file được sử dụng trong quá trình cài đặt

1. Mã nguồn file cài đặt TheHive,Cortex và MISP

```
version: "3.7"
services:
  thehive:
    image: strangebee/thehive:5.2
    restart: unless-stopped
    depends_on:
      - cassandra
      - elasticsearch
      - minio
      - cortex.local
    mem_limit: 1500m
    ports:
      - "0.0.0.0:9000:9000"
    environment:
      - JVM_OPTS="-Xms1024M -Xmx1024M"
    command:
      - --secret
      - "lab123456789"
      - "--cql-hostnames"
      - "cassandra"
      - "--index-backend"
      - "elasticsearch"
      - "--es-hostnames"
      - "elasticsearch"
      - "--s3-endpoint"
      - "http://minio:9002"
      - "--s3-access-key"
      - "minioadmin"
      - "--s3-secret-key"
      - "minioadmin"
      - "--s3-use-path-access-style"
    #If you are familiar with the previous docker compose file you will note that the Cortex
ports and keys have been omitted this is because we can now
    #complete the integration from TheHive GUI directly.
    volumes:
      - thehivedata:/etc/thehive/application.conf
    networks:
      - SOC_NET
```

```
cassandra:
  image: 'cassandra:4'
  restart: unless-stopped
  ports:
    - "0.0.0.0:9042:9042"
  environment:
    - CASSANDRA_CLUSTER_NAME=TheHive
  volumes:
    - cassandradata:/var/lib/cassandra
  networks:
    - SOC_NET

elasticsearch:
  image: docker.elastic.co/elasticsearch/elasticsearch:7.17.9
  restart: unless-stopped
  mem_limit: 512m
  ports:
    - "0.0.0.0:9200:9200"
  environment:
    - discovery.type=single-node
    - xpack.security.enabled=false
    - cluster.name=hive
    - http.host=0.0.0.0
    - "ES_JAVA_OPTS=-Xms256m -Xmx256m"
  volumes:
    - elasticsearchdata:/usr/share/elasticsearch/data
  networks:
    - SOC_NET

minio:
  image: quay.io/minio/minio
  restart: unless-stopped
  command: ["minio", "server", "/data", "--console-address", ":9002"]
  environment:
    - MINIO_ROOT_USER=minioadmin
    - MINIO_ROOT_PASSWORD=minioadmin
  ports:
    - "0.0.0.0:9002:9002"
  volumes:
    - "miniodata:/data"
  networks:
    - SOC_NET
```

```
#appended .local onto the container name because when we integrate cortex with TheHive using the new GUI menu it only accept a FQDN.
  cortex.local:
    image: thehiveproject/cortex:latest
    restart: unless-stopped
    environment:
      - job_directory=/tmp/cortex-jobs
      - docker_job_directory=/tmp/cortex-jobs
    volumes:
      #For analyzers and responders (called neurons, also based on docker containers) to work, we need to bind the hosts docker socket into the cortex container
      #so it can use the docker service of the host, and share the job directory between the container and the host.
      #An alternative way of doing this would be to run docker (neurons) within the cortex docker container (docker-ception), the container will need to be run in
      #privileged mode and you will need the --start-docker parameter for this work. It is however not advised to run docker containers in priviliged mode because it
      #grants the docker container root capabilities over the host system which is a security risk.
      - /var/run/docker.sock:/var/run/docker.sock
      - /tmp/cortex-jobs:/tmp/cortex-jobs
      - ./cortex/logs:/var/log/cortex
      - ./cortex/application.conf:/cortex/application.conf
    depends_on:
      - elasticsearch
    ports:
      - "0.0.0.0:9001:9001"
    networks:
      - SOC_NET
  #appended .local onto the container name because when we integrate MISP with TheHive using the new GUI menu it only accepts a FQDN.
  misp.local:
    image: coolacid/misp-docker:core-latest
    restart: unless-stopped
    depends_on:
      - misp_mysql
    ports:
      - "0.0.0.0:80:80"
      - "0.0.0.0:443:443"
    volumes:
      - "./server-configs/:/var/www/MISP/app/Config/"
      - "./logs/:/var/www/MISP/app/tmp/logs/"
      - "./files/:/var/www/MISP/app/files"
      - "./ssl/:/etc/nginx/certs"
```

```
    environment:
      - MYSQL_HOST=misp_mysql
      - MYSQL_DATABASE=mispdb
      - MYSQL_USER=mispuser
      - MYSQL_PASSWORD=misppass
      - MISP_ADMIN_EMAIL=mispadmin@lab.local
      - MISP_ADMIN_PASSPHRASE=mispadminpass
      - MISP_BASEURL=localhost
      - TIMEZONE=Europe/London
      - "INIT=true"
      - "CRON_USER_ID=1"
      - "REDIS_FQDN=redis"
      - "HOSTNAME=https://192.168.56.253"
    networks:
      - SOC_NET

  misp_mysql:
    image: mysql/mysql-server:5.7
    restart: unless-stopped
    volumes:
      - mispsqldata:/var/lib/mysql
    environment:
      - MYSQL_DATABASE=mispdb
      - MYSQL_USER=mispuser
      - MYSQL_PASSWORD=misppass
      - MYSQL_ROOT_PASSWORD=mispass
    networks:
      - SOC_NET
  redis:
    image: redis:latest
    networks:
      - SOC_NET
  misp-modules:
    image: coolacid/misp-docker:modules-latest
    environment:
      - "REDIS_BACKEND=redis"
    depends_on:
      - redis
      - misp_mysql
    networks:
      - SOC_NET
  #removed the cortex volumes as we no longer require it, cortex will share the /tmp
directory for jobs, the logs and application files will be stored in the cortex folder
```

```
  #in the same directory on the host where the docker-compose.yml resides for ease of
access.
volumes:
  miniodata:
  cassandradata:
  elasticsearchdata:
  thehivedata:
  mispsqldata:

networks:
    SOC_NET:
          driver: bridge
```

2. Mã nguồn file custom-w2thive.py

```
#!/var/ossec/framework/python/bin/python3
import json
import sys
import os
import re
import logging
import uuid
from thehive4py.api import TheHiveApi
from thehive4py.models import Alert, AlertArtifact
#start user config
# Global vars
#threshold for wazuh rules level
lvl_threshold=0
#threshold for suricata rules level
suricata_lvl_threshold=3
debug_enabled = False
#info about created alert
info_enabled = True
#end user config
# Set paths
pwd = os.path.dirname(os.path.dirname(os.path.realpath(__file__)))
log_file = '{0}/logs/integrations.log'.format(pwd)
logger = logging.getLogger(__name__)
#set logging level
logger.setLevel(logging.WARNING)
if info_enabled:
    logger.setLevel(logging.INFO)
if debug_enabled:
    logger.setLevel(logging.DEBUG)
# create the logging file handler
fh = logging.FileHandler(log_file)
formatter = logging.Formatter('%(asctime)s - %(name)s - %(levelname)s - %(message)s')
fh.setFormatter(formatter)
logger.addHandler(fh)

def main(args):
    logger.debug('#start main')
    logger.debug('#get alert file location')
```

```
    alert_file_location = args[1]
    logger.debug('#get TheHive url')
    thive = args[3]
    logger.debug('#get TheHive api key')
    thive_api_key = args[2]
    thive_api = TheHiveApi(thive, thive_api_key )
    logger.debug('#open alert file')
    w_alert = json.load(open(alert_file_location))
    logger.debug('#alert data')
    logger.debug(str(w_alert))
    logger.debug('#gen json to dot-key-text')
    alt = pr(w_alert,'',[])
    logger.debug('#formatting description')
    format_alt = md_format(alt)
    logger.debug('#search artifacts')
    artifacts_dict = artifact_detect(format_alt)
    alert = generate_alert(format_alt, artifacts_dict, w_alert)
    logger.debug('#threshold filtering')
    if w_alert['rule']['groups']==['ids','suricata']:
        #checking the existence of the data.alert.severity field
        if 'data' in w_alert.keys():
            if 'alert' in w_alert['data']:
                #checking the level of the source event
                if int(w_alert['data']['alert']['severity'])<=suricata_lvl_threshold:
                    send_alert(alert, thive_api)
    elif int(w_alert['rule']['level'])>=lvl_threshold:
        #if the event is different from suricata AND suricata-event-type: alert check
lvl_threshold
        send_alert(alert, thive_api)

def pr(data,prefix, alt):
    for key,value in data.items():
        if hasattr(value,'keys'):
            pr(value,prefix+'.'+str(key),alt=alt)
        else:
            alt.append((prefix+'.'+str(key)+'|||'+str(value)))
    return alt

def md_format(alt,format_alt=''):
    md_title_dict = {}
    #sorted with first key
    for now in alt:
        now = now[1:]
        #fix first key last symbol
        dot = now.split('|||')[0].find('.')
        if dot==-1:
            md_title_dict[now.split('|||')[0]] =[now]
        else:
            if now[0:dot] in md_title_dict.keys():
                (md_title_dict[now[0:dot]]).append(now)
            else:
                md_title_dict[now[0:dot]]=[now]
    for now in md_title_dict.keys():
        format_alt+='### '+now.capitalize()+'\n'+'| key | val |\n| ------ | ------ |\n'
        for let in md_title_dict[now]:
            key,val = let.split('|||')[0],let.split('|||')[1]
            format_alt+='| **' + key + '** | ' + val + ' |\n'
```

```
    return format_alt

def artifact_detect(format_alt):
    artifacts_dict = {}
    artifacts_dict['ip'] = re.findall(r'\d+\.\d+\.\d+\.\d+',format_alt)
    artifacts_dict['url'] =  re.findall(r'http[s]?://(?:[a-zA-Z]|[0-9]|[$-
_@.&+]|[!*\(\),]|(?:%[0-9a-fA-F][0-9a-fA-F]))+',format_alt)
    artifacts_dict['domain'] = []
    for now in artifacts_dict['url']:
artifacts_dict['domain'].append(now.split('//')[1].split('/')[0])
    return artifacts_dict

def generate_alert(format_alt, artifacts_dict,w_alert):
    #generate alert sourceRef
    sourceRef = str(uuid.uuid4())[0:6]
    artifacts = []
    if 'agent' in w_alert.keys():
        if 'ip' not in w_alert['agent'].keys():
            w_alert['agent']['ip']='no agent ip'
    else:
        w_alert['agent'] = {'id':'no agent id', 'name':'no agent name'}
    for key,value in artifacts_dict.items():
        for val in value:
            artifacts.append(AlertArtifact(dataType=key, data=val))
    alert = Alert(title=w_alert['rule']['description'],
              tlp=2,
              tags=['wazuh',
              'rule='+w_alert['rule']['id'],
              'agent_name='+w_alert['agent']['name'],
              'agent_id='+w_alert['agent']['id'],
              'agent_ip='+w_alert['agent']['ip'],],
              description=format_alt ,
              type='wazuh_alert',
              source='wazuh',
              sourceRef=sourceRef,
              artifacts=artifacts,)
    return alert

def send_alert(alert, thive_api):
    response = thive_api.create_alert(alert)
    if response.status_code == 201:
        logger.info('Create TheHive alert: '+ str(response.json()['id']))
    else:
        logger.error('Error create TheHive alert: {}/{}'.format(response.status_code,
response.text))

if __name__ == "__main__":
    try:
       logger.debug('debug mode') # if debug enabled
       # Main function
       main(sys.argv)
    except Exception:
       logger.exception('EGOR')
```

3. Mã nguồn file custom-w2thive

```
#!/bin/sh
# Copyright (C) 2015-2020, Wazuh Inc.
# Created by Wazuh, Inc. <info@wazuh.com>.
# This program is free software; you can redistribute it and/or modify it under the
terms of GP>
WPYTHON_BIN="framework/python/bin/python3"
SCRIPT_PATH_NAME="$0"
DIR_NAME="$(cd $(dirname ${SCRIPT_PATH_NAME}); pwd -P)"
SCRIPT_NAME="$(basename ${SCRIPT_PATH_NAME})"
case ${DIR_NAME} in
    */active-response/bin | */wodles*)
        if [ -z "${WAZUH_PATH}" ]; then
            WAZUH_PATH="$(cd ${DIR_NAME}/../..; pwd)"
        fi
    PYTHON_SCRIPT="${DIR_NAME}/${SCRIPT_NAME}.py"
    ;;
    */bin)
    if [ -z "${WAZUH_PATH}" ]; then
        WAZUH_PATH="$(cd ${DIR_NAME}/..; pwd)"
    fi
    PYTHON_SCRIPT="${WAZUH_PATH}/framework/scripts/${SCRIPT_NAME}.py"
    ;;
     */integrations)
        if [ -z "${WAZUH_PATH}" ]; then
            WAZUH_PATH="$(cd ${DIR_NAME}/..; pwd)"
        fi
    PYTHON_SCRIPT="${DIR_NAME}/${SCRIPT_NAME}.py"
    ;;
esac
${WAZUH_PATH}/${WPYTHON_BIN} ${PYTHON_SCRIPT} $@
```

Phụ lục 2: Quá trình cài đặt các giải pháp

1. Cài đặt Yara rule và Wazuh agent trên Windows 10

Cài đặt Wazuh Agent lên windows 11

```
Invoke-WebRequest -Uri https://packages.wazuh.com/4.x/windows/wazuh-agent-4.7.1-1.msi -OutFile
${env.tmp}\wazuh-agent; msiexec.exe /i ${env.tmp}\wazuh-agent /q WAZUH_MANAGER='192.168.56.117'
WAZUH_AGENT_NAME='WebServer' WAZUH_REGISTRATION_SERVER='192.168.56.117'
```

```
NET START WazuhSvc
```

Tải và cài đặt yara rules

```
Invoke-WebRequest -Uri
https://github.com/VirusTotal/yara/releases/download/v4.2.3/yara-4.2.3-2029-win64.zip
-OutFile v4.2.3-2029-win64.zip
Expand-Archive v4.2.3-2029-win64.zip; Remove-Item v4.2.3-2029-win64.zip
```

Tạo file yara.bat trong thư mục C:\Program Files (x86)\ossec-agent\active-response\bin\ với nội dung:

```
@echo off

setlocal enableDelayedExpansion

reg Query "HKLM\Hardware\Description\System\CentralProcessor\0" | find /i "x86" > NUL
&& SET OS=32BIT || SET OS=64BIT


if %OS%==32BIT (
    SET log_file_path="%programfiles%\ossec-agent\active-response\active-
responses.log"
)

if %OS%==64BIT (
    SET log_file_path="%programfiles(x86)%\ossec-agent\active-response\active-
responses.log"
)

set input=
for /f "delims=" %%a in ('PowerShell -command "$logInput = Read-Host; Write-Output
$logInput"') do (
    set input=%%a
```

```
)


set json_file_path="C:\Program Files (x86)\ossec-agent\active-response\stdin.txt"
set syscheck_file_path=
echo %input% > %json_file_path%

for /F "tokens=* USEBACKQ" %%F in (`Powershell -Nop -C "(Get-Content 'C:\Program Files
(x86)\ossec-agent\active-response\stdin.txt'|ConvertFrom-
Json).parameters.alert.syscheck.path"`) do (
set syscheck_file_path=%%F
)

del /f %json_file_path%
set yara_exe_path="C:\Program Files (x86)\ossec-agent\active-
response\bin\yara\yara64.exe"
set yara_rules_path="C:\Program Files (x86)\ossec-agent\active-
response\bin\yara\rules\yara_rules.yar"
echo %syscheck_file_path% >> %log_file_path%
for /f "delims=" %%a in ('powershell -command "& \"%yara_exe_path%\"
\"%yara_rules_path%\" \"%syscheck_file_path%\""') do (
    echo wazuh-yara: INFO - Scan result: %%a >> %log_file_path%
)

exit /b
```

Tạo file yara trong thư mục C:\Program Files (x86)\ossec-agent\active-response\bin\ và copy file yara64.exe vào trong:

```
mkdir 'C:¥Program Files (x86)¥ossec-agent¥active-response¥bin¥yara¥'
> cp .¥v4.2.3-2029-win64¥yara64.exe 'C:¥Program Files (x86)¥ossec-agent¥active-
response¥bin¥yara¥'
```

Tạo file download_yara_rules.py có nội dung:

```
from valhallaAPI.valhalla import ValhallaAPI
v = ValhallaAPI(api_key="1111111111111111111111111111111111111111111111111111111111111111")
response = v.get_rules_text()
with open('yara_rules.yar', 'w') as fh:
    fh.write(response)
```

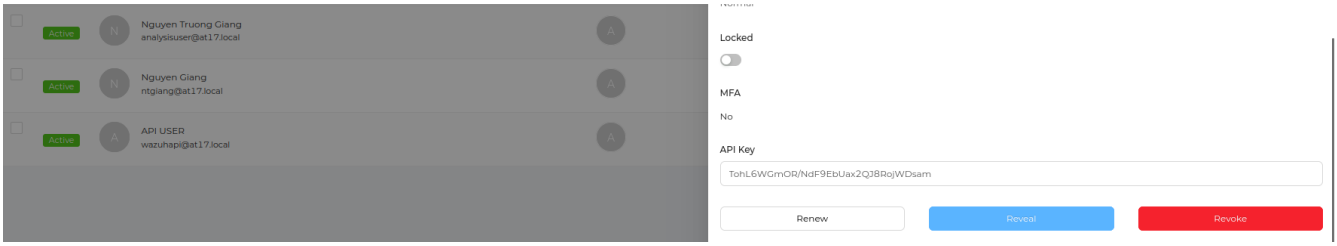
Thực thi file download_yara_rules.py

Tạo folder rules trong thư mục C:\Program Files (x86)\ossec-agent\active-response\bin\yara\ copy file yara_rules.yar vào trong thư mục này

```
mkdir 'C:¥Program Files (x86)¥ossec-agent¥active-response¥bin¥yara¥rules'
> mv .¥yara_rules.yar 'C:¥Program Files (x86)¥ossec-agent¥active-response¥bin¥yara¥rules'
```

2. Kết nối Wazuh với TheHive

Tạo tài khoản với quyền analysis trên TheHive, thực hiện tạo API key. Ghi nhớ API key này.



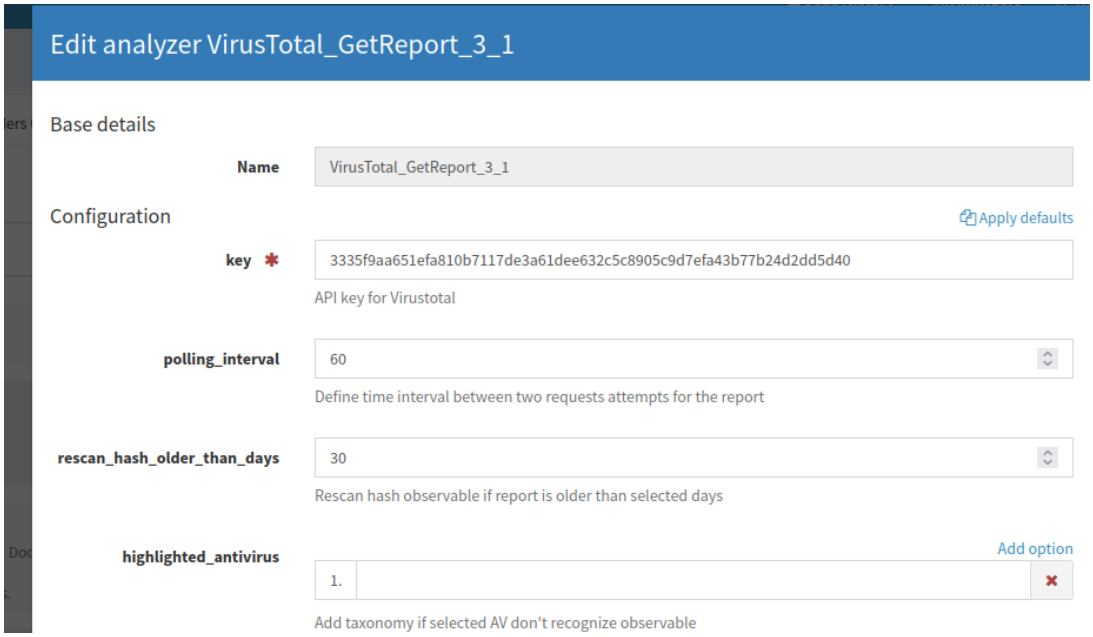
Trong thư mục /var/ossec/etc/ossec.conf, ta thêm đoạn mã này vào, với API key vừa được tạo ở trên.

```
<ossec_config>
…
  <integration>
    <name>custom-w2thive</name>
    <hook_url>http://TheHive_Server_IP:9000</hook_url>
    <api_key>RWw/Ii0yE6l+Nnd3nv3o3Uz+5UuHQYTM</api_key>
    <alert_format>json</alert_format>
  </integration>
…
</ossec_config>
```

3. Kết nối Cortex đến các Analyzers

Thực hiện tạo tài khoản trên các nền tảng Virustotal và MalwareBazza do ở đây chúng ta sử dụng 2 Analyzers là VirusTotal_GetReport_3_1 và MalwareBaaza_1_0, và lấy API key của những tài khoản vừa tạo.

Vào phần Organization -> Analyzers và tìm kiếm Analyzer mình muốn thêm, sau đó chọn enable rồi paste APT key vừa lấy được vào.



Tùy chỉnh các option phù hợp rồi chọn save.