HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

HOÀNG VĂN TÙNG

NGHIÊN CỨU FILE LOG VÀ ỨNG DỤNG TRONG BẢO MẬT SERVER

LUẬN VĂN THẠC SĨ KỸ THUẬT

(*Theo định hướng ứng dụng*)

HÀ NỘI – 2020

HỌC VIỆN CÔNG NGHỆ BƯU CHÍNH VIỄN THÔNG

HOÀNG VĂN TÙNG

NGHIÊN CỨU FILE LOG VÀ ỨNG DỤNG TRONG BẢO MẬT SERVER

Chuyên ngành : Hệ thống thông tin

MÃ SỐ: 8.48.01.04

LUẬN VĂN THẠC SĨ KỸ THUẬT

(*Theo định hướng ứng dụng*)

NGƯỜI HƯỚNG DẪN KHOA HỌC: TS. PHAN THỊ HÀ

HÀ NỘI – 2020

LỜI CAM ĐOAN

Tôi cam đoan đây là công trình nghiên cứu của riêng tôi.

Các số liệu, kết quả nêu trong luận văn là trung thực và chưa từng được ai công bố trong bất kỳ công trình nào khác.

Học viên luận văn ký và ghi rõ họ tên

Hoàng Văn Tùng

LỜI CẢM ƠN

Để hoàn thành được luận văn, ngoài sự nghiên cứu và cố gắng của bản thân, tôi xin cảm ơn cô giáo TS Phan Thị Hà - người cô trực tiếp hướng dẫn, tận tình chỉ bảo và định hướng cho tôi trong suốt quá trình thực hiện luận văn. Một lời cảm ơn chắc chắn không thể diễn tả hết lòng biết ơn sâu sắc của tôi tới cô – một người cô của tôi trên mọi phương diện!

Tôi xin gửi lời cảm ơn chân thành cảm ơn tất cả các thầy cô giáo của Học viện Công nghệ Bưu chính Viễn thông đã giảng dạy, quan tâm nhiệt tình và dìu dắt tôi trong trong suốt quá trình học tập tại trường.

Cuối cùng, tôi xin gửi lời cảm ơn tới gia đình, bạn bè và những người đã luôn ở bên tôi cổ vũ, động viên, tạo điều kiện thuận lợi cho tôi học tập, tạo động lực tinh thần vô giá để tôi hoàn thiện luận văn này và ngày một hoàn thiện chính bản thân mình.

Trong quá trình nghiên cứu và thực hiện luận văn, mặc dù được sự hướng dẫn nhiệt tình của cô giáo TS Phan Thị Hà và những nỗ lực của bản thân nhưng cũng không thể tránh khỏi những thiếu sót hạn chế. Tôi rất mong nhận được ý kiến đóng góp, sửa chữa từ quý Thầy, Cô và các bạn bè đồng nghiệp để luận văn được hoàn thiện hơn.

Trân trọng cảm ơn!

Học viên

Hoàng Văn Tùng

MỤC LỤC

DANH MỤC BẢNG BIỂU

DANH MỤC HÌNH VẼ

MỞ ĐẦU

Hiện nay, Internet phát triển một cách mạnh mẽ và có nhiều bước chuyển mình vượt bậc đã đóng góp tích cực trong việc phát triển kinh tế, xã hội và đặc biệt là con người. Sự phát triển của internet kéo theo các website trực tuyến được tạo ra ngày càng một nhiều hơn. Và tất nhiên các server lưu trữ web (như hosting, vps) cũng tăng nhanh chóng nhằm đáp ứng nhu cầu tạo website, nhưng cũng đồng thời tạo ra thêm môi trường thu lợi đối với các tin tặc. Với nhiều thủ đoạn lẫn cách thức tinh vi, việc đảm bảo an toàn cho các server lưu trữ web luôn là một bài toán vô cùng nan giải.

Tháng 2/2019, Tập đoàn Bkav đã phát đi cảnh báo về việc đang có một chiến dịch tấn công có chủ đích của các hacker nước ngoài nhằm vào các server public của Việt Nam. Hàng trăm cơ quan, tổ chức tại Việt Nam đã bị hacker tấn công, xâm nhập máy chủ, sau đó thực hiện mã hóa toàn bộ dữ liệu trên server [1]. Hay các cuộc tấn công DDoS vào Wikipedia – website bách khoa hàng đầu thế giới – khiến cho website này bị ngừng hoạt động gần một ngày [2]. Theo Báo cáo an ninh mạng hàng năm của năm 2019 từ Bulletproof, một cuộc tấn công DoS hoặc DDoS có thể gây thiệt hại cho một công ty doanh nghiệp hơn 2 triệu đô la hoặc lên tới 120.000 đô la cho một công ty nhỏ [4].

Khi bị tin tặc tấn công, các server sẽ bị ảnh hưởng rất nhiều có thể dẫn đến việc hoạt động của hệ thống bị ngưng trệ hay mất mát dữ liệu. Điều này không chỉ làm tốn thời gian khắc phục cho người quản trị hệ thống mà kéo theo các hệ lụy về kinh tế hoặc là cả an ninh. Do vậy mà cần sự phát hiện sớm những cuộc tấn công vào server.

Để ngăn chặn nhanh nhất những cuộc tấn công vào server từ tin tặc, người quản trị cần sớm biết những mối nguy hại tiềm ẩn có thể tác động đến hệ thống của mình. Phân tích log của server sẽ giúp người quản trị biết được có những gì đang tác động vào server và đưa ra cách xử lý nhanh chóng [6].

Đề tài này được xây dựng nhằm mục đích nghiên cứu cũng như xây dựng một hệ thống có thể phân tích được các dấu hiệu bất thường trên sever thông qua file log để có thể có những phương án phòng bị cũng như ngăn chặn việc tấn công server.

Việc xây dựng được hệ thống phân tích log cũng giúp cho các quản trị viên có thể nắm bắt được tình hình của server nhanh chóng và chính xác nhất. Điều này không chỉ giúp ngăn chặn hay hiểu rõ cách thức tấn công nhanh hơn bình thường, mà còn giúp giảm chi phí nhân lực để quản trị server.

Xuất phát từ yêu cầu thực tế, học viên đã nghiên cứu áp dụng hệ thống phân tích log vào việc tăng cường bảo mật cho server, luận văn có tựa đề: "Nghiên cứu file log server và ứng dụng trong bảo mật server". Luận văn tập trung vào những vấn đề liên quan đến file log trong server và cách mà file log server có thể giúp người quản trị trong việc bảo mật.

Đề tài nghiên cứu về dữ liệu trong file log của một web server từ đó tìm ra những vấn đề có thể gây ảnh hưởng đến server này. Cụ thể hơn là dựa trên những dữ liệu thu được từ những lưu lượng truy cập, những thay đổi được tạo ra trong server,… để phát hiện ra những bất thường có thể gây tổn hại đến server.

Log là một file ghi lại liên tục các thông báo về hoạt động của hệ thống server hoặc của các dịch vụ được triển khai trên hệ thống hay những file tương ứng. Log file thường là các file văn bản thông thường biểu diễn dưới dạng "clear text" tức là người quản trị có thể dễ dàng đọc được nó, vì vậy mà có thể sử dụng các trình soạn thảo văn bản (vi, vim, nano...) hoặc các trình xem văn bản thông thường (cat, tailf, head...) để kiểm tra những file log server này.

File log server cung cấp cho quản trị viên toàn bộ các thông tin hoạt động của server, hỗ trợ giải quyết các rắc rối mà server gặp phải miễn là họ biết phân tích và ứng dụng các thông tin nhận được vào khắc phục.

Tác dụng của log là vô cùng to lớn, nó có thể giúp quản trị viên theo dõi hệ thống của mình tôt hơn, hoặc giải quyết các vấn đề gặp phải với hệ thống hoặc

service. Điều này đặc biệt quan trọng với các hệ thống cần phải online 24/24 để phục vụ nhu cầu của mọi người dùng.

Việc áp dụng phân tích dữ liệu log vào bảo mật của server là một lĩnh vực rất rộng lớn khó để có thể làm chi tiết do vậy trong khuôn khổ của luận đề tài này, học viên sẽ tập trung vào tìm hiểu file log server thông qua đó xây dựng hệ thống phát hiện bất thường và cảnh báo tới người quản trị.

Đề tài sẽ sử dụng ELK Stack làm hệ thống quản lý log bởi rất nhiều điểm mạnh như mã nguồn mở, có thể thu thập được log từ nhiều nguồn khác, có một nền tảng tìm kiếm mạnh mẽ (ElasticSearch), hỗ trợ phân tích số liệu thu thập được (Analytic), quản lý logs tập trung, tìm kiếm và thông báo lỗi một cách tự động.

Mục đích, đối tượng, phạm vi và phương pháp nghiên cứu

Mục đích nghiên cứu

Mục đích của luận văn đó là tìm hiểu về file log trong server và ứng dụng vào trong việc bảo mật của server.

Đối tượng nghiên cứu

Học viên xác định đối tượng nghiên cứu trong đề tài này là các lý thuyết về file log server cũng như ứng dụng của nó. Bên cạnh đó đề tài còn tập trung vào nghiên cứu **công nghệ mã nguồn mở ELK** (ElasticSearch, LogStash và Kibana) và đưa vào áp dụng trên server dịch vụ lưu trữ công ty iNET.

Phạm vi và phương pháp nghiên cứu

- Phạm vi nghiên cứu
    - Nghiên cứu về file log trong server.
    - Phân tích file log server trên server lưu trữ web của công ty iNET.
- Phương pháp nghiên cứu

Tìm hiểu về file log server, các kỹ thuật phân tích file log. Sau đó nghiên cứu công nghệ mã nguồn mở ELK (ElasticSearch, LogStash và Kibana) để đưa vào áp dụng server lưu trữ nhằm cảnh báo sớm tới quản trị viên khi có bất thường xảy ra.

Cấu trúc luận văn

Với mục tiêu đặt ra như vậy, nội dung và kết quả của luận văn sẽ được chia thành 3 chương như sau:

Chương I: Tổng quan về file log server

Trong chương này, luận văn sẽ đi vào tìm hiểu về file log trong server cũng như bài toán ứng dụng file log này vào việc bảo mật server.

Chương II: Nghiên cứu và thiết kế hệ thống phân tích log server

Những vấn đề trong việc xử lý file log server thông qua các công cụ và nền tảng sẽ được trình bày trong chương 2 này. Trong chương này, học viên cũng sẽ đề cập đến mô hình cũng như các kỹ thuật phân tích file log server.

Chương III: Áp dụng thử nghiệm hệ thống phân file log server thực tiễn

Tại chương này học viên sẽ áp dụng triển khai thử nghiệm hệ thống phân tích log trên một server cung cấp dịch vụ lưu trữ nội dung cho website. Từ việc áp dụng triển khai sẽ thu thập các thông tin từ file log server nhận được và đề xuất phương pháp bảo mật cho server (nếu có).

Chương 1.TỔNG QUAN VỀ FILE LOG SERVER

Chương này trình bày về tổng quan và ý nghĩa của bài toán ứng dụng phân tích file log server trong việc bảo mật server. Đồng thời chương 1 giới thiệu chi tiết về file log trong server, những ứng dụng của file log server đối với quản trị viên trong quá trình hoạt động lẫn bảo mật hệ thống. Những hiểu biết về hệ thống phân tích file log server cũng sẽ được trình bày trong chương này.

1.1 Giới thiệu bài toán

1.1.1 Tổng quan về bảo mật server

Bảo mật server là việc bảo vệ dữ liệu và tài nguyên được lưu trên các máy chủ khỏi việc bị truy cập trái phép, chỉnh sửa hoặc đánh cắp,…. Trong một thế giới lý tưởng, dữ liệu và tài nguyên trên server phải luôn được giữ bí mật, ở trạng thái chính xác và sẵn sàng để sử dụng. Tuy nhiên bài toán bảo mật server lưu trữ dữ liệu hiện đang là vấn đề gây nhức nhối hiện nay của các quản trị viên.

Với việc mạng Internet đang được phổ cập mạnh mẽ như hiện nay việc bảo mật server sẽ càng khó khăn hơn với việc thông tin hướng dẫn tấn công server được cung cấp vô số trên mạng. Những lỗ hổng hàng loạt, lỗ hổng riêng biệt, lỗ hổng từ phần mềm, phần cứng, vô số những tác nhân khiến cho server thành miếng bánh ngon. Tuy nhiên đây cũng là điều không thể tránh khỏi khi server được đưa lên mạng Internet.

Khi server sẽ bị tấn công có thể dẫn đến nhiều hậu quả như ngừng hoạt động, tài nguyên bị lạm dụng,… gây ảnh hưởng đến việc duy trì dịch vụ và hệ thống. Do vậy mà việc có những thông báo sớm đến quản trị viên khi hành động tấn công vừa mới chớm nở để có những phương án phòng thủ thích hợp là lựa chọn tối ưu nhất cho doanh nghiệp. Tuy nhiên câu hỏi đặt ra là làm thế nào để có thể phát hiện sớm được những tác động xấu gây ảnh hưởng đến server? Câu trả lời là thông qua việc phân tích file log.

1.1.2 Bài toán ứng dụng phân tích log server vào bảo mật

Bản thân file log server không có khả năng ngăn chặn tấn công hoặc bảo mật server nhưng những thông tin từ các log file là rất đa dạng và phong phú, khi xem xét file log server quản trị viên sẽ có một cái nhìn toàn cảnh về những thay đổi trên hệ thống. Xem xét log server thường xuyên có thể giúp quản trị viên xác định được các cuộc tấn công độc hại trên hệ thống. Do vậy bài toán đặt ra là làm sao có thể dựa vào việc phân tích log server để có thể phát hiện sớm những cuộc tấn công nhằm vào hệ thống giúp việc bảo mật server trở nên chủ động hơn.

Việc xây dựng một hệ thống có thể hỗ trợ người quản trị trong việc quản lý log lẫn thông báo khi có một sự cố gì phát sinh cũng sẽ là một mục tiêu cần hướng đến trong luận văn nhằm tối ưu thời gian hơn thay vì phân tích thủ công. Những yếu tố cần thiết trong hệ thống phản tích log bao gồm:

- Xây dựng một hệ thống phục vụ việc phân tích dữ liệu log server trực quan, theo thời gian thực.
- Tự động thông báo cho quản trị viên khi có các sự cố xảy ra với server nhằm có những cách xử lý kịp thời và nhanh chóng hạn chế rủi ro về bảo mật.

Ý nghĩa bài toán

Bài toán ứng dụng phân tích log server vào bảo mật server đều mang tính ứng dụng rất cao trong thời đại công nghệ số hiện nay.

Bài toán có ý nghĩa trong nghiên cứu bảo mật hệ thống server giúp cho việc vận hành hệ thống trở nên được thông suốt nhất, giúp cho các doanh nghiệp có thể giảm thiểu được những chi phí không đáng có khi khắc phục các vấn đề do bị tấn công. Không chỉ vậy, nó còn giúp cho quản trị viên có thể nắm rõ hơn được về hệ thống thay vì phân tích thủ công thì có thể dựa vào các số liệu phân tích sẵn để có những thay đổi kịp thời cho server.

1.2 Giới thiệu về file log server

1.2.1 File log server

Log server là một tài liệu văn bản đơn giản chứa tất cả các hoạt động của một máy chủ cụ thể trong một khoảng thời gian nhất định (ví dụ: một ngày). Log server được máy chủ tự động tạo, duy trì và có thể cung cấp cho người quản trị cái nhìn chi tiết về cách thức, thời gian trang web hoặc ứng dụng được cài đặt trên server đó hoạt động như thế nào.

Tuy nhiên không phải file log nào cũng có cấu trúc giống nhau, với mỗi hệ điều hành và ứng dụng khác nhau lại có những định dạng log riêng. Sau đây là một số định dạng file log server phổ biến hiện nay:

- **Common Log Format**

Common Log Format (CLF) hay còn gọi là NCSA Common Log Format là là một định dạng tập tin văn bản tiêu chuẩn được sử dụng bởi các máy chủ web khi tạo file log server. Định dạng log CLF được đặt tên theo NCSA_HTTPd, một phần mềm web server đã ngừng hoạt động.

Common Log Format được viết ở định dạng ASCII cố định (không thể tùy chỉnh) và ghi lại thông tin cơ bản về các gói http request có cấu trúc như:

Host Ident Authuser Date Request Status Bytes

Trong đó:

- Host: cung cấp địa chỉ IP address đang gửi request lên server
- Ident: danh tính của client
- Authuser: user id của client đang gửi request
- Date: Ngày và giờ gửi yêu cầu
- Request: Dòng yêu cầu từ client, được đặt trong dấu ngoặc kép (“”)
- Status: Mã trạng thái HTTP được trả về client(gồm ba chữ số)
- Bytes: kích thước của dữ liệu được trả về client được đo bằng bytes.

Trong định dạng này nếu bất kỳ trường dữ liệu nào thiếu sẽ biểu thị bằng một dấu gạch ngang (-).

Ví dụ một request được ghi lại ở định dạng CLF:

110.53.221.34 - - [21/Mar/2020:02:06:55 +0700] "GET / HTTP/1.1" 200 163 "http://210.211.111.88:80" "Mozilla/4.0 (compatible; MSIE 9.0; Windows NT 6.1)"

- **Common Event Format**

Common Event Format (CEF) là định dạng log có thể mở rộng được HP Arcsight giới thiệu và đề xuất giúp đơn giản hóa việc quản lý event log. CEF đã được tạo ra với mục đích xây dựng một tiêu chuẩn event log chung cho các thông tin bảo mật của các thiết bị mạng, ứng dụng và công cụ từ các nhà cung cấp khác nhau. Cấu trúc của CEF cho các event log bao gồm một tiêu đề tiêu chuẩn và một biến văn bản.

Tiêu đề tiêu chuẩn của CEF thường có dạng ngày và tên máy chủ :

Feb 23 12:54:06 host message

Biến văn bản sẽ được định dạng như bên dưới bao gồm các tham số được phân tách nhau bằng dấu |.

CEF:Version|Device Vendor|Device Product|Device Version|Signature ID|Name |Severity|Extension

- Version: một số nguyên xác định phiên bản của định dạng CEF
- Device Vendor, Device Product and Device Version : trường xác định loại thiết bị. Loại thiết bị phải đáp ứng được yêu cầu không tồn tại hai sản phẩm có thể sử dụng cùng một cặp device-vendor lẫn device-product.
- Signature ID là Mã định danh duy nhất cho mỗi loại sự kiện có thể là một chuỗi hoặc một số nguyên.
- Name là một trường đại diện giúp quản trị viên có thể đọc và hiểu rõ hơn về sự kiện. Tên sự kiện không được chứa thông tin được đề cập cụ thể trong các lĩnh vực khác.
- Severity là một số nguyên và phản ánh mức độ nghiêm trọng của sự kiện. Mức độ nghiêm trọng nằm trong khoảng từ 0 đến 10, trong đó số 10 phản ánh mức độ nghiêm trọng cao nhất.

- Extension là tập hợp các cặp khóa-giá trị được mô tả trong trường mở rộng CEF.

Ví dụ : Sep 19 08:26:10 host CEF:0|security|threatmanager|1.0|100|worm successfully stopped|10|src=10.0.0.1 dst=2.1.2.2 spt=1232

- **JSON Log Format**

JSON Log Format là định dạng log được viết theo dạng chuỗi JSON. JSON Log Format giúp quản trị viên có thể phân tích nhật ký theo dạng big data. Định dạng này không chỉ giúp nội dung có thể dễ dàng đọc được mà còn hỗ trợ việc tạo một cơ sở dữ liệu có thể dễ dàng truy vấn, điều này phép tóm tắt và phân tích diễn ra nhanh hơn giúp quản trị viên giám sát ứng dụng của mình và khắc phục sự cố nhanh hơn.

Ví dụ:

```
{ "timestamp": "2018-05-24 23:15:07",
  "id": 0,
  "class": "connection",
  "event": "connect",
  "connection_id": 12,
  "account": { "user": "user",
               "host": "localhost" },
  "login": { "user": "user",
             "os": "",
             "ip": "::1",
             "proxy": "" },
  "connection_data": { "connection_type": "tcp/ip",
                       "status": 0,
                       "db": "bank_db" } }
```

Ý nghĩa của các trường được định nghĩa như sau:

- Timestamp - Giá trị ngày và thời gian.
- Id - Bộ đếm sự kiện nhận dạng liệt kê các sự kiện có cùng giá trị dấu thời gian.
- Class - Tên lớp của sự kiện.
- Event - Tên sự kiện của lớp được chỉ định.
- Connection_id - Mã định danh của kết nối (phiên) đã kích hoạt sự kiện.
- Account - Tài khoản phiên hiện tại đang sử dụng. Điều này tương ứng với người dùng và máy chủ lưu trữ trong mysql.userbảng.
- Login - Chi tiết đăng nhập được sử dụng để kết nối máy khách.

Các trường còn lại của sự kiện phụ thuộc vào loại lớp sự kiện nằm trong file log để có thêm vào.

- **W3C Extended Log Format**

W3C Extended Log Format là định dạng có thể tùy chỉnh được sử dụng bởi Microsoft Internet Information Server (IIS) phiên bản 4.0 và 5.0. Với tính năng tùy chỉnh, người dùng có thể thêm hoặc bỏ qua các trường khác nhau theo tùy theo nhu cầu công việc và phân tích. Việc tùy chỉnh cũng giúp người quản trị có thể tăng hoặc giảm kích thước của tệp để phù hợp hơn với hệ thống.

Ví dụ: #Software: Microsoft Internet Information Server 6.0

#Version: 1.0

#Date: 1998-11-19 22:48:39

#Fields: date time c-ip cs-username s-ip cs-method cs-uri-stem cs-uri-query sc-status sc-bytes cs-bytes time-taken cs-version cs(User-Agent) cs(Cookie) cs(Referrer)

1998-11-19 22:48:39 206.175.82.5 - 208.201.133.173 GET /global/images/navlineboards.gif - 200 540 324 157 HTTP/1.0 Mozilla/4.0+(compatible;+MSIE+4.01;+Windows+95) USERID=CustomerA;+IMPID=01234 http://www.exampledomain.com

Các dòng bắt đầu bằng ký tự # chứa các chỉ thị. Các chỉ thị được định nghĩa như sau:

- Version: <*integer*>.<*integer*>
- Phiên bản của định dạng tệp log mở rộng được sử dụng.
- Fields: [<*specifier*>...]
- Chỉ định các trường được ghi trong nhật ký.
- Software: *string*
- Xác định phần mềm tạo nhật ký
- Start-Date: <*date*> <*time*>
- Ngày và thời gian mà log được bắt đầu.
- End-Date:<*date*> <*time*>
- Ngày và thời gian mà log kết thúc.
- Date:<*date*> <*time*>
- Ngày và thời gian mà mục nhập được thêm vào.

Định nghĩa trường mở rộng W3C cho file log

s-	Hành động của server
c-	Hành động của client
cs-	Hành động của client tới server
sc-	Hành động của server tới client

Bảng 1.1 Định nghĩa các tiền tố mở rộng trong W3C

Date	Date	Ngày hoạt động xảy ra.
Time	Time	Giờ hoạt động xảy ra..
Client IP Address	c-ip	Địa chỉ IP của client truy cập máy chủ
User Name	cs-username	Tên của người dùng được xác thực đã truy cập máy chủ. Nếu không tìm thấy sẽ thay thế bằng dấu (-)

Service Name	s-sitename	Dịch vụ Internet và số hiệu được khách hàng truy cập.
Server Name	s-computername	Tên của máy chủ
Server IP Address	s-ip	Địa chỉ IP của máy chủ nơi log được tạo.
Server Port	s-port	Cổng mà client truy cập
Method	cs-method	Hành động client gửi đến (ví dụ phương thức GET).
URI Stem	cs-uri-stem	Tài nguyên được truy cập
URI Query	cs-uri-query	URL truy vấn đến
Protocol Status	sc-status	Trạng thái của hành động
Win32® Status	sc-win32-status	Trạng thái của hành động được sử dụng bởi Windows
Bytes Sent	sc-bytes	Số lượng byte được gửi đến máy chủ
Bytes Received	cs-bytes	Số lượng byte nhận được bởi máy chủ.
Time Taken	time-taken	Khoảng thời gian hoạt động, tính bằng mili giây
Protocol Version	cs-version	Phiên bản giao thức (HTTP, FTP) được sử dụng bởi máy khách. Đối với HTTP, đây sẽ là HTTP 1.0 hoặc HTTP 1.1.
Host	cs-host	Hiển thị nội dung của host header.
User Agent	cs(User-Agent)	Trình duyệt được sử dụng trên máy khách.
Cookie	cs(Cookie)	Nội dung của cookie được gửi hoặc nhận, nếu có.

Referrer	cs(Referer)	Trang web giới thiệu client đến

Bảng 1.2 Định nghĩa các trường mở rộng trong W3C

1.2.2 Ứng dụng của File log server

Từ việc xử lý các vấn đề bảo mật đến xử lý sự cố bất thường về hiệu suất của ứng dụng hoặc tuân thủ các quy định, file log cung cấp cho người quản trị rất nhiều thông tin.

- **Xử lý hệ thống tốt hơn**

Một trong những trường hợp sử dụng rõ ràng nhất để phân tích log có lẽ là trong việc khắc phục sự cố máy chủ, mạng hoặc hệ thống, từ sự cố ứng dụng đến sự cố cấu hình và lỗi phần cứng.

Khắc phục sự cố với phân tích log server thường được sử dụng trong việc quản trị vì nó cho phép DevOps phát hiện và giải quyết các lỗi hệ thống quan trọng nhanh hơn, cải thiện hiệu quả hoạt động.

- **Ứng phó tốt hơn với các vi phạm dữ liệu và các sự cố bảo mật khác**

Khi nói đến an ninh mạng, log server cung cấp một kho thông tin về những kẻ tấn công của bạn, chẳng hạn như địa chỉ IP, các request giữa client /server, mã trạng thái HTTP, v.v. Tuy nhiên, nó vẫn bị đánh giá thấp. Rất nhiều công ty không hiểu giá trị của việc phân tích log mà vẫn chỉ dựa vào tường lửa cơ bản hoặc phần mềm bảo mật khác để bảo vệ dữ liệu của họ trước các cuộc tấn công DNS. Tuy nhiên, không có file log server, bạn không thể hiểu rủi ro bảo mật và phản hồi tương ứng.

Log server là một kim chỉ nam cho những ai muốn hiểu về server, với việc phân tích log thì quản trị viên có thể theo dõi các hoạt động đáng ngờ và thiết lập các ngưỡng, quy tắc và tham số để bảo vệ hệ thống khỏi các mối đe dọa tương tự trong tương lai. Với phân tích log, quản trị viên thậm chí có thể hỗ trợ chặn kẻ tấn công bằng địa chỉ IP của họ. Trong trường hợp cần điều tra các lỗ hổng bảo mật, file log có thể cung cấp thời gian và địa điểm của mọi sự kiện xảy ra trong mạng hoặc hệ thống của bạn.

- **Phát hiện hành vi người dùng trực tuyến**

Phân tích Log server là một trong những cách tốt nhất để hiểu hành vi của khách truy cập ứng dụng cài đặt trên Server [3]. Nó không chỉ cho biết bạn có bao nhiêu khách truy cập mà còn cho phép bạn theo dõi lại hành trình chính xác của họ và hiểu về những trang họ đã dành nhiều thời gian nhất, họ đã làm gì trên trang web của bạn, tại sao có sự thay đổi về số lượng khách truy cập, v.v. .

1.3 Ứng dụng của file log trong bảo mật server

1.3.1 Tổng quan về phân tích log

Phân tích log server là quá trình tìm hiểu ý nghĩa của thông điệp tường trình do server tạo ra, còn được gọi là log events, audit trail records hoặc đơn giản là logs. Phân tích log cung cấp các số liệu hữu ích vẽ nên một bức tranh rõ ràng về những gì đã xảy ra trên cơ sở hạ tầng. Quản trị viên hoàn toàn có thể sử dụng dữ liệu này để cải thiện hoặc giải quyết các vấn đề về hiệu suất trong một ứng dụng hoặc trên tổng thể server.

Phân tích log server là một công việc bắt buộc đối với người quản trị server nào. Họ phải thường xuyên theo dõi và phân tích Log server hệ thống để tìm kiếm lỗi, sự bất thường hoặc hoạt động đáng ngờ hoặc trái phép đi lệch khỏi định mức. Điều này cho phép họ tạo lại chuỗi sự kiện dẫn đến sự cố và khắc phục sự cố một cách hiệu quả.

Trong trường hợp xảy ra sự cố bảo mật, chẳng hạn như khi cơ sở dữ liệu bị chỉnh sửa, các trang web bị xóa hoặc xóa các tệp, log server có thể là bằng chứng duy nhất về những gì đã xảy ra.

1.3.2 Một số cuộc tấn công có thể nhận biết được qua file log

- **Cross Site Scripting (XSS)**

Các cuộc tấn công Cross Site Scripting hoạt động bằng cách nhúng các thẻ script chứa mã độc vào URL / HTTP request và lôi kéo người dùng nhấp vào để đảm bảo rằng các Javascript độc hại hoạt động trên máy của nạn nhân [10]. Các cuộc tấn công Cross Site Scripting hoạt động nhờ vào sự tin tưởng giữa người dùng và máy chủ trong khi thực tế không hề có sự kiểm tra đầu vào / đầu ra trên máy chủ

để từ chối các lệnh Javascript độc hại. Các cuộc tấn công đơn giản thường chứa các thẻ HTML như <h1> hoặc <script>. Một ví dụ thường được sử dụng là <script> alert ('XSS') </ script>.

Dưới đây là danh sách một số thẻ có thể sử dụng trong tấn công XSS:

- *javascript, vbscript, expression, applet, meta, xml, blink, link, style, script, embed, object, iframe, frame, frameset, ilayer, layer, bgsound, title, base*

  Hoặc xử lý sự kiện Javascript như:

- *onabort, onactivate, onafterprint, onafterupdate, onsubmit, onunload, ...*

Khi phân tích log server quản trị viên có thể tìm kiếm một số mẫu như thẻ HTML, tham số 'src' của thẻ 'img' và một số trình xử lý sự kiện Javascript như trên nhằm kiểm tra xem có vấn đề bất thường nào xảy ra hay không [8]. Nhưng việc tìm kiếm tất cả các biểu thức trên không đảm bảo tìm thấy tất cả các XSS injection, phân tích log chỉ có thể nhận ra được một số loại tấn công đơn giản. XSS được tạo ra theo bối cảnh. Do vậy mà XSS injection thường rất khó phát hiện.

Ví dụ đơn giản về tấn công XSS:

```
217.160.165.173 - - [12/Mar/2004:22:31:12 -0500]
"GET /foo.jsp?<SCRIPT>foo</SCRIPT>.jsp HTTP/1.1" 200 578 "-" "Mozilla/4.75
[en] (X11, U; Nessus)"
217.160.165.173 - - [12/Mar/2004:22:37:17 -0500]
"GET /cgibin/cvslog.cgi?file=<SCRIPT>window.alert</SCRIPT> HTTP/1.1" 403
302 "-" "Mozilla/4.75 [en] (X11, U; Nessus)"
```

Đây là hai yêu cầu đến từ Nessus khi cố gắng tìm các tập lệnh có thể là tấn công XSS. Theo mã trạng thái HTTP, trong yêu cầu đầu tiên, web sever đã phản hồi với 200, có nghĩa là có một script foo.jsp và hoạt động trên trang, và có thể cần kiểm tra thủ công lại để chắc chắn. Yêu cầu thứ hai (cvslog.cgi) không thành công, máy chủ đã phản hồi 403, có nghĩa là máy chủ web đã từ chối truy cập nên quản trị viên không cần phải kiểm tra lại.

- **Injection Flaws**

Code injection có thể là bất kỳ loại mã nào như SQL, LDAP, XPath, XSLT, HTML, XML hay OS command injection. Cross Site Scripting trên thực tế là một tập hợp con của HTML injection. Tại đây thì học viên sẽ ví dụ về injection phổ biến nhất, SQL injection. Đối với SQL injection kẻ tấn công phải ngắt được câu lệnh SQL gốc. Điều này thường được thực hiện bằng singlequote (') hoặc doublebledash (--). Singlequote hoạt động như một dấu phân cách cho truy vấn SQL; các doubledash là ký tự nhận xét trong Oracle và MS SQL. Ngoài ra cũng có thể có một số từ khóa có thể sử dụng khi tìm kiếm:

- @@version
- varchar
- char
- exec()
- execute
- declare
- cast

Khi phân tích log, quản trị viên có thể tạo ra các bộ lọc dựa theo từ khóa trên để kiểm tra xem có hành vi tác động đến lên dữ liệu hay không, từ đó có những hướng giải quyết tiếp theo.

Ví dụ:

```
ex121209.log 414 2012-12-09 13:17:34 W3SVC100000 WEB151 216.177.71.6
GET /search.aspx home=177&id=1%27%20or%201=@@version-- 80 - 8.8.8.8
HTTP/1.0 Mozilla/4.0+(compatible;+Synapse) - -
www.yourhosteddomainname.com 500 0 0 7639 354 531
```

- **Information Leakage và Improper Error Handling**

Information leakage and improper error handling thường xảy ra khi các ứng dụng web không giới hạn lượng thông tin họ trả về cho người dùng của họ. Các thông tin về lỗi được ứng dụng trên server tạo và hiển thị chúng cho người dùng,

những thông báo lỗi này khá hữu ích cho những kẻ tấn công, vì chúng tiết lộ chi tiết triển khai hoặc thông tin hữu ích trong việc khai thác lỗ hổng.

Để giảm thiểu Information leakage and improper error handling quản trị viên cần phân tích các mã trạng thái HTTP (các mã trạng thái HTTP được sử dụng chủ yếu cho giám sát các máy chủ và mục đích gỡ lỗi) nhằm phát hiện ra các đường dẫn nào đang cung cấp quá nhiều thông tin về lỗi hay phản hồi mã lỗi sai. Dựa vào phản hồi này quản trị viên có thể phân tích chúng để phát hiện các cuộc tấn công.
Dưới đây là tổng quan về mã trạng thái HTTP:

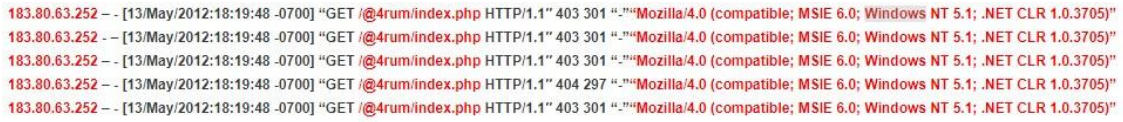
Status	Meaning
1xx	Thông tin, request đã được server tiếp nhận và quá trình đang được diễn ra
2xx	Success, request đã được server nhận và xử lý thành công
3xx	Redirection, request cần có thêm hành động từ client để hoàn thành
4xx	Client Error, request chứa cú pháp sai hoặc không thể thực hiện
5xx	Server Error, phía server không thể thực hiện được request

Bảng 1.3 Mã trạng thái HTTP

- **DDoS**

Tấn công DDoS là tấn công gây ngập tràn tài nguyên của server bằng cách phối hợp hàng ngàn máy cùng một thời điểm. Không khó để xâm nhập tài nguyên như sử dụng brute-force attacks hoặc SQL injection, một cuộc tấn công DDoS thường diễn ra nhanh chóng và không có những dấu hiệu rõ ràng nhưng vẫn mang lại những hậu quả nghiêm trọng như khiến server bị treo.

Khi xác định có một cuộc tấn công DDoS, quản trị viên cần nhanh chóng xem xét file log server. Server bị DDoS thường sẽ có những lưu lượng truy cập tăng cao bất thường khác với những thời điểm khác, do vậy cần nắm được rõ những địa chỉ và request nào đang tăng cao đột biến hoặc bị phản hồi các mã lỗi 5xx hay 4xx hơn thông thường. Sau khi xác định được quản trị viên sẽ có cách thức để xử lý dễ dàng hơn.



```
183.80.63.252 – - [13/May/2012:18:19:48 -0700] “GET /@4rum/index.php HTTP/1.1” 403 301 “-”“Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; .NET CLR 1.0.3705)”
183.80.63.252 - – [13/May/2012:18:19:48 -0700] “GET /@4rum/index.php HTTP/1.1” 403 301 “-”“Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; .NET CLR 1.0.3705)”
183.80.63.252 – - [13/May/2012:18:19:48 -0700] “GET /@4rum/index.php HTTP/1.1” 403 301 “-”“Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; .NET CLR 1.0.3705)”
183.80.63.252 – - [13/May/2012:18:19:48 -0700] “GET /@4rum/index.php HTTP/1.1” 404 297 “-”“Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; .NET CLR 1.0.3705)”
183.80.63.252 – - [13/May/2012:18:19:48 -0700] “GET /@4rum/index.php HTTP/1.1” 403 301 “-”“Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.1; .NET CLR 1.0.3705)”
```

Hình 1.1 Dữ liệu log khi Server bị DDos

Qua ảnh trên ta có thể nhận thấy rằng IP 183.80.63.252 chỉ trong vòng 1s đã gửi nhiều request đến server. Do vậy cần có một hệ thống phân tích log server và gửi cảnh báo sớm đến các quản trị viên, thông qua việc phát hiện kịp thời những hành động này, người quản trị hoàn toàn có thể chặn đứng các IP này hoặc có những những phương án xử lý khác nhanh chóng lại nhằm tránh việc quá nhiều request dẫn đến server bị down.

- **Brute Force Attacks**

Một cuộc tấn công brute force là một trong những phương pháp tấn công đơn giản và ít phức tạp nhất. Lý thuyết đằng sau Brute Force Attack là một cuộc tấn công hoạt động bằng việc tiến hành đăng nhập nhiều lần để đoán mật khẩu, cuối cùng bạn chắc chắn sẽ đúng. Kẻ tấn công nhằm mục đích mạnh mẽ giành quyền truy cập vào tài khoản người dùng bằng cách cố gắng đoán tên người dùng / email và mật khẩu. Thông thường, động cơ đằng sau nó là sử dụng tài khoản bị vi phạm để thực hiện một cuộc tấn công quy mô lớn, đánh cắp dữ liệu nhạy cảm, tắt hệ thống hoặc kết hợp cả ba.

Các cuộc tấn công Brute force attack không yêu cầu nhiều về khả năng liên tưởng hoặc kiến thức nên được rất nhiều tin tặc sử dụng làm lớp tấn công server đầu tiên trước khi thử nghiệm sang các hình thức tấn công khác. Hiện nay có những công cụ tự động có sẵn có thể tiến hành gửi hàng nghìn lần thử mật khẩu mỗi giây giúp cho việc tấn công trở nên dễ dàng hơn bao giờ hết.

Một cuộc tấn công brute force không khó để xác định và điều tra. Quả trị viên hoàn toàn có thể phát hiện ra chúng bằng cách xem xét file log server . Cuộc tấn công sẽ để lại một loạt các nỗ lực đăng nhập không thành công, như được thấy dưới đây:

Sep 21 20:10:10 host proftpd[25197]: yourserver (usersip[usersip]) - USER theusername (Login failed): Incorrect password.

Ngoài những phương thức tấn công kể trên server còn chịu rất nhiều tác động có thể gây hại. Do đó cần có một hệ thống phân tích log server có thể báo cáo sớm các cuộc tấn công tới quản trị viên.

1.3.3 Hệ thống phân tích log server

Thông thường, quản trị viên sẽ tiến hành phân tích log theo định kỳ hoặc khi hệ thống gặp sự cố. Cùng với việc phân tích thủ công như thường lệ, số lượng dữ liệu log có thể phân tích được là vô cùng nhỏ so với khối lượng log được tạo ra hàng ngày. Điều này tạo ra nhiều vấn đề tồn động như dữ liệu log phân tích có phản ánh hết toàn bộ vấn đề của hệ thống hay không, thời gian để có thể xử lý toàn bộ các dữ liệu log, dữ liệu log phân tích quá nhiều chi tiết dư thừa gây tốn thời gian để phân tích,…

Hệ thống phân tích file log được phát triển nhằm bù đắp các vấn đề tồn đọng trên bằng cách sử dụng các quy tắc để thu thập, xem xét các dữ liệu log thu được và chỉ ra các sự kiện có thể đại diện cho các vấn đề hoặc mối đe dọa [9]. Ngoài ra một số tính năng mà hệ thống phân tích file log mang đến cho quản trị viên có thể kể đến như:

- **Lưu trữ log tập trung**

Một trong những lợi thế chính của hệ thống phân tích log server nằm ở việc lưu trữ log tập trung, đây là một yếu tố vô cùng quan trọng trong quản lý log. Việc lưu trữ toàn bộ log ở cùng một nơi giúp quản trị viên dễ dàng phân tích dù ở bất kỳ thư mục nào, nhưng ưu điểm thực sự của việc lưu trữ log tập rung là tăng cường bảo mật hệ thống của bạn. Trong quản trị hệ thống, thời gian từ khi sự cố bắt đầu xảy ra đến lúc khắc phục tạo ra những khác biệt đáng kể, việc lưu trữ log tập trung sẽ góp phần đẩy thời gian phát hiện lên nhanh nhất có thể. Lưu trữ log tập trung cũng đồng nghĩa với việc các dữ liệu log đã được chuẩn hóa về cùng một định

dạng, điều này cũng giúp tiết kiệm thời gian khi tìm kiếm thông tin trên file log từ các nguồn khác nhau.

- **Giám sát và hệ thống cảnh báo**

Hệ thống phân tích log cung cấp các cảnh báo theo thời gian thực khi hệ thống xảy ra những bất thường, điều này rất cần thiết trong phát hiện xâm nhập và tấn công nâng cao khả năng phòng bị cho quản trị viên.

Quản trị viên cũng có thể cài đặt giám sát để theo dõi lựa chọn các sự kiện tùy chỉnh, rất hữu ích cho bảo mật lẫn xử lý sự cố hay khi có sự thay đổi. Ngoài ra hệ thống còn cho phép được tạo các thông báo sự cố đến nhiều nguồn để đảm bảo không bỏ lỡ bất kỳ sự kiện quan trọng nào.

- **Nhóm dữ liệu**

Hệ thống phân tích log cho phép phân chia dữ liệu log thành các phần thông tin nhỏ hơn để lưu trữ và thao tác dễ dàng hơn với chức năng nhận ra các file log có cấu trúc tương tự nhau rồi nhóm lại theo các cấu trúc đó. Chẳng hạn, xác định tất cả các dấu thời gian trong file log và thu thập các log từ một khung thời gian nhất định hoặc chỉ theo dõi hoạt động của một IP nào đó.

- **Khả năng hiển thị dữ liệu**

Các dữ liệu log server vốn được coi là khô khan khi chỉ là những dòng thông báo khô khan. Hệ thống phân tích log cung cấp các giao diện đồ họa trực quan với các biểu đồ, đồ thị cho phép quản trị viên dễ dàng nhận ra được những sự tăng đột biến bất thường khi so sánh với các dữ liệu cũ.

- **Tăng cường khả năng phân tích dữ liệu**

Không phải toàn bộ các dữ liệu được tạo ra trên log đều có ích khi phân tích dữ liệu log server. Rất nhiều thông tin được gói trong các file log gây mất thời gian khi phải xem xét toàn bộ các thông tin đó, do vậy hệ thống phân tích log cho phép quản trị viên cài đặt việc chỉ lấy những dữ liệu quan trọng để phân tích.

Với rất nhiều tính năng hỗ trợ cho quản trị dữ liệu log, hệ thống phân tích log đang là lựa chọn của hầu hết các công ty lớn hiện nay và là xu hướng của các công ty nhỏ. Có thể kể đến một số ví dụ như:

- Splunk: giải pháp hệ thống phân tích log thương mại hàng đầu thế giới hiện nay [16]. Một số ví dụ đang sử dụng Splunk có thể kể đến như Telenor (nhà cung cấp dịch vụ viễn thông lớn nhất của Na Uy), Bộ Giao thông Vận tải Nevada,…
- Graylog: giải pháp xây dựng phân tích log mã nguồn mở miễn phí cho phép quản lý nhật ký dễ dàng [17]. Một số ví dụ đang sử dụng Graylog có thể kể đến như T-System (một công ty tư vấn và dịch vụ công nghệ toàn cầu của Đức), Petronas (một công ty dầu khí lớn của Malaysia),…
- ELK stack: một giải pháp phân tích log mã nguồn mở phổ biến hiện nay [18]. Một số ví dụ sử dụng ELK để xây dựng hệ thống phân tích log: Airbus, Docker, Github,…

1.4 Kết luận

Trong chương này, học viên đã tóm lược ngắn gọn về file log trong server cũng như những ứng dụng của nó. Đồng thời qua đó giới thiệu về bài toán ứng dụng phân tích file log server trong việc bảo mật server. Tiếp theo trong chương 2 học viên sẽ nghiên cứu về công nghệ, kỹ thuật dùng cho việc thiết kế hệ thống phân tích log server.

Chương 2. NGHIÊN CỨU VÀ THIẾT KẾ HỆ THỐNG PHÂN TÍCH LOG SERVER

Chương 2 sẽ giới thiệu về công nghệ được sử dụng để xây dựng hệ thống phân tích log server, các bước và các kỹ thuật phân tích log server đang được áp dụng hiện nay. Cuối chương học viên sẽ phác họa tổng quát một mô hình hệ thống phân tích log server để áp dụng vào thực tiễn.

2.1 Giới thiệu nền tảng và các công cụ phân tích log

Để giải quyết bài toán quản lý dữ liệu log server và tự động cảnh báo khi có bất thường xảy ra, quản trị viên có thể lựa chọn rất nhiều giải pháp công nghệ. Một số giải pháp bao gồm trả phí lẫn mã nguồn mở có thể kể đến như Splunk, Graylog hay ELK stack. Các tính năng nổi trội của các giải pháp giám sát và phân tích dữ liệu log hệ thống ngày nay phải kể đến như khả năng tìm kiếm mạnh mẽ, xây dựng được màn hình giám sát thời gian thực, báo cáo, cảnh bảo ngưỡng, phân tích dữ liệu lịch sử, truy tìm vết, …

Tuy nhiên sau khi cân nhất học viên quyết định lựa chọn nền tảng công nghệ ELK làm giải pháp công nghệ cho bài toán quản lý và phân tích dữ liệu log được nêu ở phần 1. ELK giúp quản trị viên dễ dàng thu thập thông tin thông qua file log từ đó có cơ sở để phát hiện những bất thường được nhanh hơn giảm thiểu việc bị ảnh hưởng do bị tấn công bởi tin tặc, một số ưu điểm của ELK stack có thể kể đến như:

- Hỗ trợ thu thập log từ nhiều nguồn, nhiều server khác nhau giúp cho việc lấy dữ liệu log từ nhiều cụm máy chủ trở nên nhanh chóng hơn mà không bị thất thoát dữ liệu
- ElasticSearch cung cấp khả năng lập chỉ mục dữ liệu mạnh mẽ cho phép quản trị viên hoàn toàn có thể lọc, tìm kiếm nội dung log hay những thay đổi trên hệ thống nhanh hơn so với truyền thống

- Thu thập log tự động đảm bảo tính sẵn sàng của dữ liệu, quản trị viên hoàn toàn có đọc được log ngay khi hệ thống sinh ra trên Kibana
- Thông báo cho quản trị viên những bất thường thay đổi trên server thông qua nhiều nền tảng như email, slack,... giúp việc ngăn ngừa lỗi trên server được chủ động hơn
- Quản trị viên hoàn toàn có thể thay kết nối cổng trên ELK stack đảm bảo tính bí mật của dữ liệu log thu thập được
- Dữ liệu được hiển thị trực quan dễ dàng phân tích hơn so với những dữ liệu thô trên server

Một số công ty lớn sử dụng ELK stack:

- **NetFlix:** Netflix phụ thuộc rất nhiều vào ELK stack. Công ty này sử dụng ELK để theo dõi và phân tích Log server bảo mật của hoạt động dịch vụ khách hàng. ELK stack cho phép NetFlix lập chỉ mục, lưu trữ và tìm kiếm tài liệu từ hơn mười lăm cụm bao gồm gần 800 nút.

- **LinkedIn:** Trang web tiếp thị truyền thông xã hội nổi tiếng LinkedIn sử dụng ELK stack để theo dõi hiệu suất và bảo mật. Nhóm CNTT của LinkedIn đã tích hợp ELK với Kafka để hỗ trợ việc lấy dữ liệu log server của họ trong thời gian thực. Hoạt động ELK của họ bao gồm hơn 100 cụm trên sáu trung tâm dữ liệu khác nhau.

- **Tripwire:** Tripwire là một hệ thống quản lý sự kiện thông tin bảo mật trên toàn thế giới. Công ty sử dụng ELK để hỗ trợ phân tích gói thông tin Log server.

- **Medium:** Medium là một nền tảng xuất bản blog nổi tiếng. Họ sử dụng ELK stack để gỡ lỗi các vấn đề trong việc hoạt động của họ. Công ty cũng sử dụng ELK để phát hiện các điểm nóng của DynamoDB. Hơn nữa, với việc sử dụng ELK stack, công ty có thể hỗ trợ 25 triệu độc giả cũng như hàng ngàn bài đăng được xuất bản mỗi tuần hoạt động trơn tru.

Hệ thống phân tích log ELK stack được xây dựng gồm 3 phần:

- ElasticSearch: Elasticsearch là một công cụ tìm kiếm dựa trên phần mềm Lucene.

- Logstash: Logstash là một công cụ nguồn mở được thiết lập để thu thập nhật ký, phân tích cú pháp và đưa chúng ra các hệ thống khác.
- Kibana: Kibana là một công cụ giao diện trực quan cho phép bạn khám phá, trực quan hóa và xây dựng một bảng điều khiển trên dữ liệu nhật ký được tập trung trong Elaticsearch.

2.1.1. Giới thiệu ElasticSearch

Elaticsearch là một máy chủ cơ sở dữ liệu độc lập, mã nguồn mở được phát triển bằng Java. Về cơ bản, nó là một công cụ để tìm kiếm và phân tích văn bản. ElasticSearch lấy dữ liệu phi cấu trúc từ nhiều nguồn khác nhau và lưu trữ nó ở định dạng được tối ưu hóa cao cho các tìm kiếm dựa trên ngôn ngữ. Elaticsearch hoạt động dựa trên thư viện Apache Lucene nổi tiếng, biểu thị dữ liệu dưới dạng các tài liệu có cấu trúc JSON. Người dùng có thể sử dụng ElasticSearch thông qua API RESTful của nhà phát triển và hay các ngôn ngữ PHP, Python và Ruby để lưu trữ, tìm kiếm và phân tích khối lượng dữ liệu lớn một cách nhanh chóng và hiệu quả. Nó đặc biệt hữu ích trong việc xử lý dữ liệu là ngôn ngữ tự nhiên [12].

Một số tổ chức lớn sử dụng ElasticSearch:

- Trang Wikipedia sử dụng ElasticSearch để cung cấp máy tìm kiếm toàn văn (full-text search) với kết quả tìm kiếm được tô sáng.
- Trang The Guardian sử dụng ElasticSearch để kết hợp dữ liệu của người đọc với dữ liệu mạng xã hội để cung cấp các hồi đáp thời gian thực giúp tăng trải nghiệm người dùng.
- Trang cộng đồng cho các nhà phát triển phần mềm nổi tiếng Stack Overflow sử dụng ElasticSearch làm máy tìm kiếm toàn văn kết hợp với vị trí địa lý của người dùng để đưa ra các kết quả tìm kiếm chính xác nhất cho câu truy vấn của người dùng.
- Trang quản lý mã nguồn mở nổi tiếng GitHub sử dụng ElasticSearch để quản lý hơn 130 triệu dòng codes.

2.1.2 Giới thiệu LogStash

LogStash là phần mềm thu thập dữ liệu mã nguồn mở được viết trên nền tảng Java với khả năng thu thập dữ liệu thời gian thực (realtime). LogStash có chức năng thu thập dữ liệu log từ nhiều nguồn khác nhau sau đó định hình lại rồi gửi đến một cơ sở dữ liệu khác. Ngoài ra LogStash còn được sử dụng để lọc dữ liệu log phục vụ cho các bài toán phân tích và trực quan hóa dữ liệu [13].

Logstash trước đây được sử dụng để xử lý log từ các ứng dụng và gửi chúng đến Elaticsearch. Nhưng hiện nay Logstash đã phát triển thành một công cụ có mục đích hơn là một đường ống xử lý dữ liệu. Dữ liệu mà Logstash nhận được sẽ được xử lý dưới dạng sự kiện, có thể là bất cứ điều gì bạn chọn như các mục file log, đơn đặt hàng thương mại điện tử, khách hàng, tin nhắn trò chuyện, v.v.. và gửi đến cơ sở dữ liệu để chứa các dữ liệu trên.

Logstash bao gồm ba thành phần:

- Input : chuyển các dữ liệu từ các nguồn vào Logstash
- Bộ lọc : Tập hợp các cách Logstash sẽ xử lý các sự kiện nhận được từ các plugin giai đoạn đầu vào
- Output : Nơi xử lý dữ liệu sự kiện hoặc Log server.

Các tính năng của Logstash:

- Các sự kiện được truyền qua từng giai đoạn bằng cách sử dụng hàng đợi nội bộ
- Cho phép thu thập nhiều định dạng Log server của bạn
- Lọc / phân tích cú pháp cho Log server của bạn
- Cung cấp hỗ trợ tập trung xử lý dữ liệu
- Nó phân tích một lượng lớn dữ liệu và sự kiện có cấu trúc / không cấu trúc
- Cung cấp các plugin để kết nối với nhiều loại nguồn và nền tảng đầu vào khác nhau

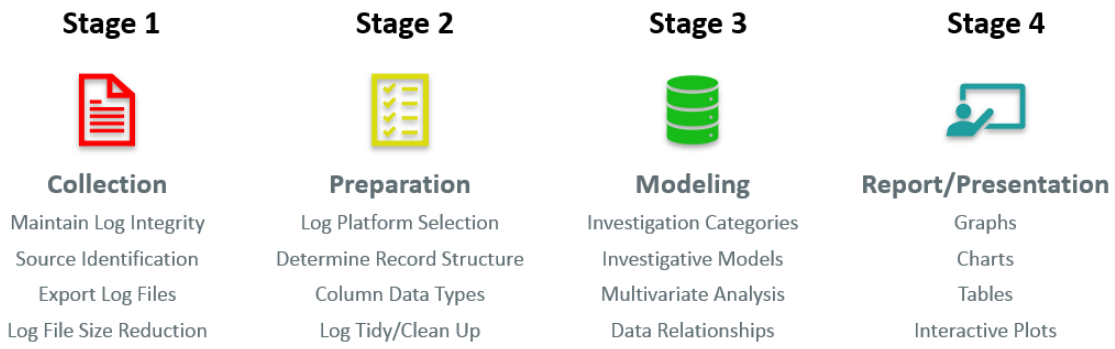
2.1.3 Giới thiệu Kibana

Kibana là một phần mềm mã nguồn mở được sử dụng để trừu tượng hóa dữ liệu, xây dựng các biểu đồ, báo cáo, màn hình giám sát và phân tích dữ liệu thời gian thực từ nguồn dữ liệu trong ElasticSearch. Kibana cung cấp giao diện cho phép người dùng có thể thực hiện truy vấn toàn văn trên hệ truy hồi thông tin ElasticSearch, xây dựng biểu đồ, các màn hình điều khiển từ dữ liệu chỉ mục trên ElasticSearch mộtcách nhanh chóng [14].

- Bảng điều khiển giao diện mạnh mẽ có khả năng hiển thị thông tin được lập chỉ mục từ elastic.
- Cho phép tìm kiếm thông tin theo chỉ mục thời gian thực
- Bạn có thể tìm kiếm, xem và tương tác với dữ liệu được lưu trữ trong Elaticsearch
- Thực hiện truy vấn trên dữ liệu và trực quan hóa kết quả trong biểu đồ, bảng và bản đồ
- Bảng điều khiển có thể định cấu hình để cắt và ghi Log server logstash trong elaticsearch
- Có khả năng cung cấp dữ liệu dưới dạng graphs, charts, v.v.
- Bảng điều khiển thời gian thực có thể dễ dàng cấu hình

2.2 Mô hình xử lý file log server

Quá trình phân tích file log server đòi hỏi quản trị viên phải tuân theo một mô hình phân tích Log server. Hình 2.1 trình bày một mô hình phân tích Log server thường thấy để tuân theo khi tiến hành phân tích file log server.



Hình 2.1. Bốn giai đoạn phân tích log (nguồn: hello.global.ntt)

- **Log Collection**

Giai đoạn này liên quan đến việc xác định các thiết bị hoặc các nguồn (ví dụ: các ứng dụng, hệ thống, thiết bị mạng, an ninh) và tách file log đó ra để phân tích. Chi tiết của các tệp log server được quyết định trong giai đoạn yêu cầu và thiết kế của các hệ thống được triển khai.

Phân tích log ban đầu có thể được thực hiện trên chính các server chứa nó với điều kiện có sẵn các công cụ thích hợp, nếu không các Log server sẽ phải được xuất để phân tích trên một nền tảng riêng. Trong hầu hết các trường hợp, quản trị viên cần xuất file log từ server, tại thời điểm đó phải xác định định dạng Log server xuất (ví dụ: csv, json) và các nguồn thứ cấp liên quan được hỗ trợ.

Trong giai đoạn Log Collection này, người quản trị cũng cần có các kỹ thuật log reduction (lọc) để đảm bảo tệp Log server được trích xuất chứa thông tin liên quan cho công việc. Ví dụ như loại bỏ các yêu cầu được xử lý bởi các công cụ tìm kiếm tự động (Crawler, Spider, Robot) hoặc chỉ lấy các bản ghi chứa phương thương "GET".

Bên cạnh việc lấy file log trong server chính, các file log cũng có thể được kéo từ các trình thu thập log hoặc bộ tổng hợp riêng biệt như SIEM nơi file nguồn được xuất thường xuyên để theo dõi, phân tích cũng như lưu trữ. Bất kể nguồn nào, quản trị viên cũng phải xác định định dạng của file log server được trích xuất và xuất trong khi vẫn duy trì tính toàn vẹn của tất cả dữ liệu. Điều bắt buộc là các

nguồn tệp Log server và định dạng tệp Log server xuất ra phải được xác định trước, tốt nhất là trước khi xảy ra sự cố.

- **Preparation**

Đây là giai đoạn bắt đầu sau khi các tập tin log xác định đã được trích xuất và đưa vào một công cụ phân tích log. Quản trị viên cần đảm bảo nền tảng lựa chọn của họ có thể hỗ trợ điều tra kích thước dữ liệu lớn. Điều này bao gồm, ở mức tối thiểu, đủ bộ nhớ (RAM), dung lượng ổ đĩa, sức mạnh xử lý CPU hiệu quả và lựa chọn đúng ứng dụng phân tích log server.

Chính trong giai đoạn này thì cần kiểm tra tính toàn vẹn; quản trị viên thực hiện việc kiểm tra trực quan và lập trình để phân chia thành mỗi hàng (record) và cột (field). Đầu tiên, quản trị viên phải xác định cấu trúc của mỗi bản ghi Log server / hàng. Có nhiều cấu trúc khác nhau được sử dụng cho các tệp Log server (ví dụ: Syslog, LEEF, W3C Extended Log File format, NetFlow, IPFix, Microsoft Windows Event Viewer format, NCSA Common Log Format). Tiếp theo đó là chọn các loại dữ liệu cột được tối ưu hóa (ví dụ: Ngày / Giờ, Văn bản, Số nguyên). Cuối cùng, quản trị viên Log server phải thực hiện các quy trình dọn dẹp tệp Log server khác nhau và / hoặc dọn dẹp các thủ tục để đảm bảo chỉ phân tích thông tin liên quan.

Dưới đây là tóm tắt các bước mà quản trị viên cần thực hiện để phân tích Log server hiệu quả và hiệu quả:

- Lọc hoặc xóa bất kỳ hàng, cột và bảng không liên quan để giảm kích thước của tệp Log server.
- Chọn các loại dữ liệu phù hợp cho từng trường hoặc cột. Một số loại dữ liệu được tìm kiếm nhanh hơn những loại khác.
- Sử dụng tối ưu hóa các kỹ thuật chức năng lọc để tiến hành tìm kiếm mẫu (ví dụ: Regex, CIDR).

- **Log Modeling**

Đây giai đoạn đòi hỏi việc phân tích các thông tin chứa trong file log dựa vào loại sự kiện / sự cố liên quan đến cuộc điều tra. Do các mô hình điều tra khác nhau

(ví dụ, Threat Hunting) tồn tại, nên người phân tích phải có kiến thức về các loại khác nhau và chọn mô hình phù hợp nhất. Ví dụ: nếu quản trị viên đang thực hiện phân tích xâm nhập, nên sử dụng mô hình điều tra được thiết kế để xác định xâm nhập. Bảng dưới trình bày một mô tả cấp cao về các loại điều tra thích hợp để sử dụng trong một cuộc điều tra có thể bao gồm Log server hệ thống, ứng dụng, mạng và nội dung người dùng.

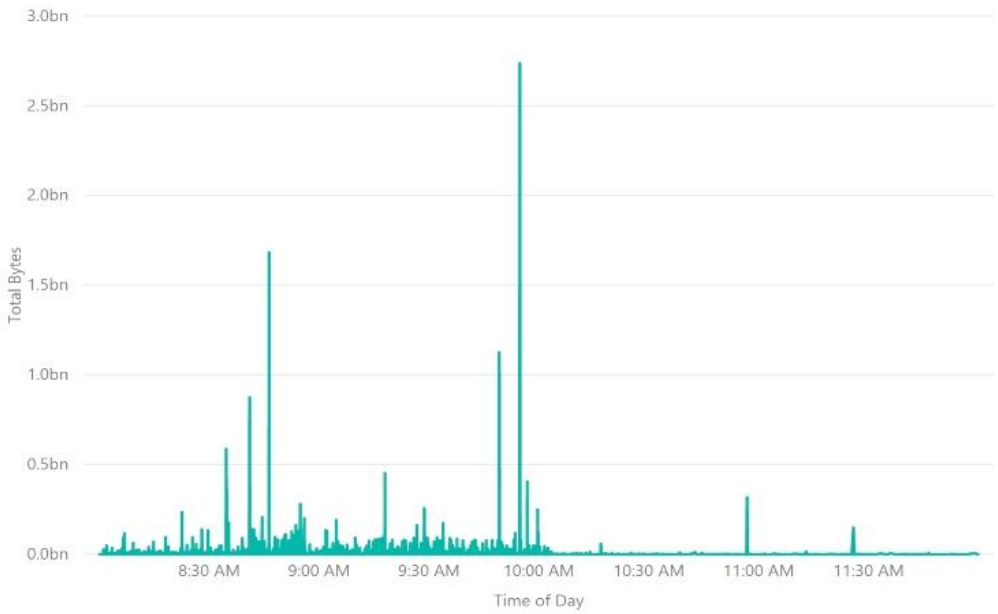
Các loại phân tích log		
SST	Tên loại phân tích	Mô tả
1	Temporal (Time)	Temporal analysis được sử dụng để phân tích dòng thời gian của sự cố. Nó cũng có thể được sử dụng để xác định thời gian trôi qua của một sự cố hoặc thời gian của vòng đời của một sự cố. Đối với sự cố lan truyền nhanh hơn, phân tích thời gian có thể chỉ ra việc sử dụng một công cụ tự động.
2	Frequency	Frequency analysis được sử dụng để xác định số lượng sự cố xảy ra trong một khoảng thời gian hoặc sự kiện cụ thể. ví dụ, bao nhiêu cuộc tấn công xảy ra trong ngày hôm nay.
3	Transition State	Transition State analysis được sử dụng để xác định sự thay đổi của một sự kiện từ trạng thái này sang trạng thái khác do sự cố.
4	Pre-occurrence	Pre-occurred analysis dùng để xác định các phát hiện ban đầu của một sự cố.
5	Historical	Historical analysis được sử dụng để xác định mức độ thường xuyên của một sự cố, hoặc sự cố tương tự đã được phát hiện trong quá khứ.
6	Traffic	Traffic analysis được sử dụng để xác định lượng lưu lượng được gửi và / hoặc nhận trên toàn mạng (ví dụ: Gói, Byte, Khung).

7	Behavior	Behavior analysis được sử dụng để xác định cách người dùng hoặc chức năng sự kiện độc hại (cách thức hoạt động).
8	Pattern	Pattern analysis được sử dụng để xác định chữ ký của một sự kiện bằng cách kiểm tra các hậu quả còn lại.
9	Port	Port analysis được sử dụng để xác định máy chủ hoặc cổng ứng dụng nào đang được quét hoặc tấn công trên hệ thống được nhắm mục tiêu.
10	Protocol	Protocol analysis được sử dụng để xác định giao thức mạng đang được sử dụng để quét hoặc tấn công một dịch vụ hoặc ứng dụng trên hệ thống được nhắm mục tiêu.
11	Statistical	Satistical analysis được sử dụng để cung cấp thu thập, phân tích, giải thích hoặc trình bày chi tiết về tỷ lệ sử dụng theo tỷ lệ phần trăm, trung bình, trung bình, sai lệch.
12	Payload	Payload analysis được sử dụng để xác định mức độ phá hủy hoặc tác động của tải trọng bằng cách kiểm tra các hậu quả còn lại.
13	Source Linkage	Source Linkage analysis được sử dụng để xác định mạng hoặc vị trí địa lý của cuộc tấn công cũng như tìm hiểu hồ sơ mạng của hệ thống tấn công.
14	Destination Linkage	Destination Linkage analysis được sử dụng để xác định mạng đích hoặc vị trí địa lý bị nhắm mục tiêu và phát triển cũng như phát triển hồ sơ mạng của hệ thống bị nhắm mục tiêu.
15	Artifact Uniqueness	Artifact Uniqueness analysis được sử dụng để xác định các đặc điểm duy nhất của từng phần (giá trị băm, nội dung)
16	Multivariate Correlation	Multivariate Correlation analysis được sử dụng để xác định mối quan hệ hoặc liên kết giữa hai hoặc nhiều loại danh mục.
17	Relationship	Relationship analysis để xác định mối quan hệ hoặc liên kết giữa source và hệ thống đích.

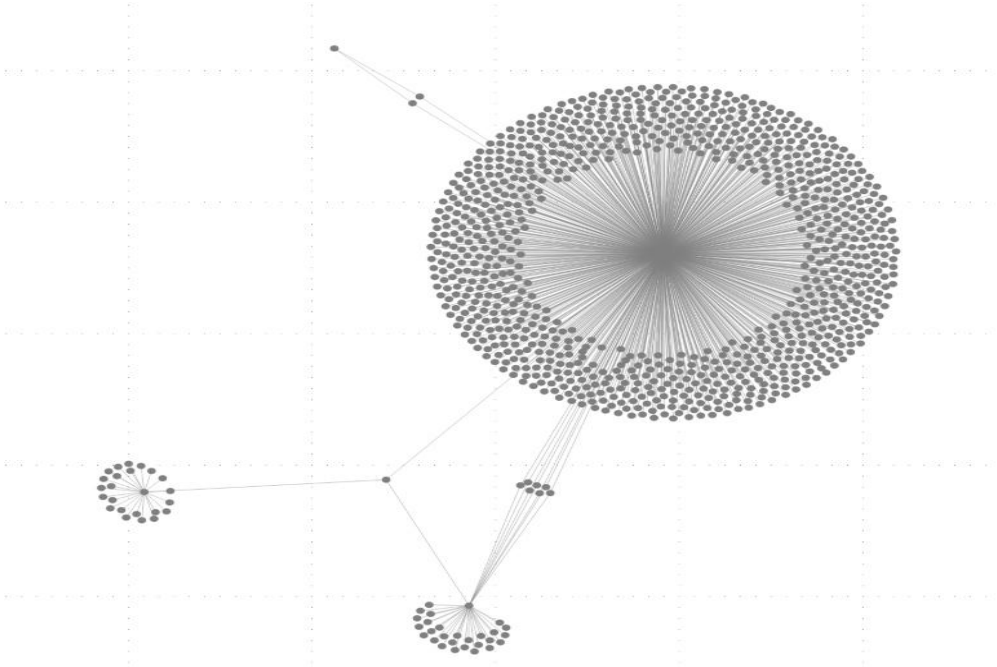
18	Content	Content analysis được sử dụng để phân tích các thành phần có trong sự kiện (ví dụ: email, tên người dùng, trang web, địa chỉ IP)
19	Context	Context analysis được sử dụng để phân tích môi trường xung quanh trong khi sự kiện xảy ra.
20	Size	Size analysis được dùng để xác định kích thước của sự kiện (ví dụ: kích thước của file download, tổng số packet)

Bảng 2.1. Các điều tra phân tích log

Mặc dù bảng trên trình bày các danh mục ở định dạng đơn biến và có thể được phân tích riêng lẻ, phân tích đa biến cũng có thể được thực hiện, như được xác định trong mục 16 bằng cách kết hợp 2 hoặc nhiều danh mục lại với nhau. Hình 2.2 trình bày một so sánh đa biến đơn giản bằng cách sử dụng Temporal Analysisvà Size Analysis. Trong khi hình 2.3, trình bày một so sánh đa biến phức tạp hơn bằng cách sử dụng Source Linkage (địa chỉ IP của máy khách) và Destination Linkage (tên miền email).



Hình 2.2. Phân tích đa biến Temporal và Size (nguồn: hello.global.ntt)



Hình 2.3. Phân tích đa biến Source và Destination Linkage (nguồn: hello.global.ntt)

- Report/Presentation

Đây là giai đoạn đòi hỏi phải trình bày những phát hiện hoặc kết quả thu được trong giai đoạn lập mô hình và truyền đạt kết luận trong một báo cáo dễ hiểu. Mặc dù có nhiều cách để trình bày kết quả (ví dụ: bảng, biểu đồ, sơ đồ, đồ thị), quản trị viên phải chọn kết hợp văn bản và trực quan hiệu quả nhất để hiển thị kết quả.

2.3 Các kỹ thuật phân tích log

Quá trình phân tích server bao gồm việc tổ chức dữ liệu được tìm thấy trên các mục nhật ký. Để làm điều đó, nó sử dụng các kỹ thuật như phân loại và gắn thẻ, nhận dạng mẫu, phân tích tương quan. Phân tích log server là một quá trình phức tạp cần tuân theo các chức năng sau:

- **Normalization**

Có rất nhiều cấu trúc khác nhau trong nhật ký từ các thiết bị mạng, vì chúng sử dụng các giao thức khác nhau hoặc chúng đến từ các nhà cung cấp khác nhau

(Cisco, Fortinet, SNMP, NetFlow, Nền tảng Linux hoặc Windows). Đó cũng là một xu hướng toàn cầu khi mà các công ty lựa chọn các sản phẩm bảo mật tốt nhất không nhất thiết phải từ cùng một nhà cung cấp.

Lựa chọn không đồng nhất này tạo cho các nhà sản xuất những ưu điểm riêng cho minh nhằm thu hút khách hàng. Tuy nhiên tất cả các thiết bị khác nhau này tạo ra các bản ghi không có chung thiết kế. Do vậy hệ thống có thể khó phát hiện các cuộc tấn công trong thời gian thực hoặc để thực hiện một phân tích. Do đó, các bản ghi cần phải được chuẩn hóa theo hướng mà công cụ phân tích log có thể tổng hợp và phản ánh dễ dàng nhất. Điều này gọi là normalization.

Việc chuẩn hóa và phân tích các sự kiện nhật ký đã được thực hiện trong một nhiều phương pháp và đã được tích hợp vào nhiều giải pháp phần mềm hiện có. Normalization có bốn phương pháp chuẩn hóa có thể được quan sát trên thị trường và cộng đồng nghiên cứu.

- **Rule Matching** (ví dụ, Regular Expressions) Việc chuẩn hóa từng loại log được mô tả trong một quy tắc xác định mức độ quan trọng của thông tin trích từ một sự kiện cụ thể. Một cách tiếp cận phổ biến trong danh mục này là các biểu thức chính quy, đặc biệt là Named-Group Regular Expressions (NGRE). Phương pháp này liên kết thông tin trong sự kiện với các trường sự kiện cụ thể, rất hữu ích cho normalization. Rule matching thường được quy định bởi các các phần mềm server.

- **Tokenization** Nội dung log được chia thành các token. Những token này có thể là những từ hoặc những cụm từ hoặc một số ký hiệu mà con người có thể đọc được trong log server. Cách tiếp cận phổ biến nhất cho tokenization là bằng từ, cho phép nhóm các log có chứa các từ tương tự. Tuy nhiên, phương pháp này phụ thuộc rất nhiều vào các từ tĩnh trong nhật ký. Một triển khai cụ thể cho tokenization là Apache Lucene tiền đề để phát triển ElasticSearch.

- **Natural Language Processing (NLP)** Một dòng log có thể bị phân tách bởi cấu trúc ngôn ngữ và biến thành các chủ đề, đối tượng, động từ và hơn thế nữa. Mỗi lần thông tin được trích xuất, người đọc log trích xuất riêng ra những từ có

nghĩa nằm trong dữ liệu log. Tuy nhiên, phương pháp dựa vào về khả năng đọc của con người đối với dữ liệu log. Ví dụ cho việc triển khai NLP là Thư viện StanfordN CoreNLP [19] hoặc khả năng phân tích văn bản của SAP HANA [20]. Việc sử dụng cụ thể kỹ thuật này để phân tích nhật ký đã được đề xuất bởi Kobayashi và cộng sự [11].

- **Custom Normalization** Phương pháp hiệu quả nhất nhưng cũng phức tạp nhất là sử dụng mã tùy chỉnh để chuẩn hóa từng định dạng nhật ký. Ví dụ một định dạng được đọc bằng trình phân tích cú pháp CSV, trong khi một định dạng khác được phân tích cú pháp bằng trình phân tích cú pháp Syslog đặc biệt và một trình phân tích cú pháp khác được xử lý với sự kết hợp của nhiều biểu thức chính quy đang được áp dụng theo thứ tự. Loại chuẩn hóa này có thể được quan sát một phần trong các công cụ phân tích nhật ký Logstash.

- **Classification and Tagging** là một phần của phân tích log server, quản trị viên cần nhóm các bản ghi log server cùng loại. Điều này tiện cho việc muốn theo dõi tất cả các lỗi của một loại nhất định trên các ứng dụng.

- **Pattern Recognition**

Đây là phần các kỹ thuật khác nhau bắt nguồn từ các lĩnh vực khác nhau như thống kê, machine learning ví dụ như Quy tắc kết hợp, khai thác dữ liệu, nhận dạng mẫu, v.v. áp dụng cho và dữ liệu có sẵn. Một số phương pháp và kỹ thuật đã được được phát triển cho bước này. Một số thường xuyên được sử dụng cho giải pháp statistical analysis, clustering và association rules [7].

- **Statistical analysis:** Statistical analysis là phương pháp phổ biến nhất để trích xuất dữ liệu về khách truy cập vào một server. Chúng ta có thể tính toán các loại thống kê mô tả các phép đo như (tần số, trung bình, v.v.) trên các biến như số lượt truy cập vào, hay các phần được truy cập vào nhiều nhất. Statistical analysis hữu ích để cải thiện hiệu năng hệ thống, tăng cường bảo mật hệ thống, hoặc tạo điều kiện khi sửa đổi dữ liệu trên server.

- C**lustering**: Phân cụm đã được sử dụng rộng rãi trong Khai thác sử dụng web để nhóm lại các phiên tương tự với số lượng lớn dữ liệu dựa trên ý tưởng chung về chức năng, khoảng cách trong đó tính toán sự tương đồng giữa các nhóm. Phân cụm có nghĩa là hành động phân vùng các tập dữ liệu không nhãn thành các nhóm đối tượng tương tự. Mỗi nhóm, được gọi là một cụm, bao gồm các đối tượng có nét tương đồng và không giống với các đối tượng của các nhóm khác. Phân cụm thường được sử dụng các thuật toán: K-means, Fuzzy C-means, k-Nearest Neighbor and Neural Network.
- Association rules: là một trong những kỹ thuật chính trong khai thác dữ liệu và nó là hình thức phổ biến nhất của khám phá localpotype trong các hệ thống học tập không giám sát. Nó phục vụ như một công cụ hữu ích để tìm mối tương quan giữa các mục trong big data.

- **Correlation Analysis**

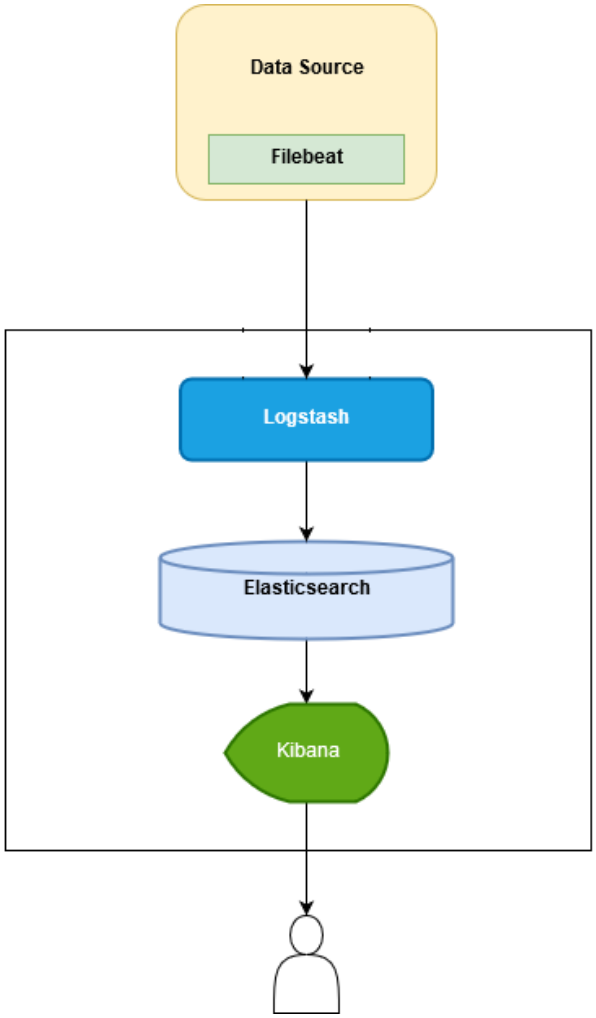
Tương quan của các sự kiện, so sánh và phân tích các bản ghi từ các nguồn khác nhau trong một khoảng thời gian, để xác định bất kỳ mô hình hoặc mối quan hệ phổ biến. Việc này giúp quản trị xác định sự cố bảo mật hoặc sự cố, phản ứng nhanh để giảm thiểu việc kinh doanh tiêu cực tác động và tổn thất. Chẳng hạn, nó có thể phát hiện một cuộc tấn công sắp xảy ra nếu có một vài lần đăng nhập cố gắng một thiết bị sử dụng một người dùng cụ thể, sau đó quét các cổng trong mạng LAN mạng sử dụng cùng một người dùng. SIEM có thể đánh dấu những sự kiện này rằng đang có nhiều lượt đăng nhập quá mức.

2.4 Xây dựng hệ thống phân tích log

ELK Stack là một giải pháp phân tích log server đầu cuối hoàn chỉnh giúp tìm kiếm sâu, phân tích và trực quan hóa nhật ký được tạo từ các máy khác nhau. ELK có thể cùng một lúc thu thập nhiều log server từ các server khác nhau giúp cho việc quản lý dữ liệu tập trung.

Trong luận văn này học viên sẽ lựa chọn xây dựng hệ thống phân tích log đơn giản với ELK stack, các bản cài đặt và thư mục file log sẽ nằm chung trên một Server có nhằm áp dụng vào mô hình thử nghiệm trong chương 3..

Tại đây học viên sẽ lựa chọn Filebeat để lấy dữ liệu log server thay vì dùng luôn Logstash để lấy dữ liệu như ý định ban đầu. Logstash tuy là một công cụ có thể đảm nhiệm hoàn toàn tốt việc lấy dữ liệu log nhưng với những dữ liệu log lớn thì Logstash yêu cầu rất nhiều về phần cứng để đảm bảo hoạt động thông suốt. Khi sử dụng Filebeat sẽ giúp phần giảm tải cho hệ thống khi thu thập dữ liệu log. Do vậy mô hình hệ thống phân tích file log server sẽ được biểu diễn như sau



Hình 2.4. Mô hình phân tích log server

Theo như mô hình trên, luồng dữ liệu sẽ xuất phát từ tệp dữ liệu log và đi theo luồng sau:

- Tệp dữ liệu log sẽ được Filebeat lấy dữ liệu ra và gửi sang Logstash theo tần xuất được cấu hình từ trước. Tại đây Filebeat đóng vai trò là một service lấy log server từ các thư mục được đặt sẵn và đưa sang Logstash xử lý.
- Dữ liệu sau khi được Filebeat chuyển sang Logstash sẽ được chuyển tiếp đến "Filter Plugin". Ở đây dữ liệu sẽ được chuẩn hoá và lọc theo yêu cầu của người quản trị để có thể đưa ra các dữ liệu cần thiết. Đầu ra trong bước Filter này là tài liệu dạng JSON chứa nội dung thông điệp của log.
- Log server sau khi được lọc sẽ được chuyển đến Elasticsearch.

- Dữ liệu JSON khi được đưa tới ElasticSearch sẽ được đánh chỉ mục nhằm phục vụ bài toán tìm kiếm, trực quan hóa dữ liệu, xây dựng báo cáo và phân tích dữ liệu log.
- Dữ liệu sau khi được đánh chỉ mục trong ElasticSearch sẽ được trực quan hóa, xây dựng báo cáo, xây dựng các màn hình giám sát, điều khiển trên Kibana.

2.5 Kết luận

Trong chương này, luận văn đã đề cập đến nền tảng phân tích file log server sẽ được áp dụng, ví dụ cho một mô hình phân tích log server cũng như các kỹ thuật phân tích. Luận văn cũng đã thiết kế một mô hình hệ thống phân tích log server có thể áp dụng được vào thực tiễn nhằm nâng cao khả năng phân tích log. Trong chương sau học viên sẽ tiến hành cài đặt và triển khai trên thực tế mô hình hệ thống phân tích log server và tự động báo lỗi đến quản trị viên.

Chương 3. ÁP DỤNG THỬ NGHIỆM HỆ THỐNG PHÂN TÍCH FILE LOG SERVER VÀO THỰC TIỄN

Trong chương 3 học viên sẽ xây dựng thử nghiệm hệ thống phân tích log ELK trên server chạy dịch vụ lưu trữ của công ty iNET dựa trên mô hình xử lý dữ liệu log trên hệ thống lẫn Logstash tự thiết kế, sau đó sẽ tiến hành phân tích các dữ liệu nhận được để đề xuất bảo mật cho server.

3.1 Cài đặt hệ thống phân tích log server

3.1.1 Giới thiệu về hệ thống máy chủ của công ty iNET

Công ty iNET hiện đang cung cấp các dịch vụ liên quan đến lưu trữ online như:

- Hosting
- Máy chủ ảo
- Email server

Là một trong những công ty cung cấp dịch vụ lưu trữ online hàng đầu tại Việt Nam, iNET hiện đang cung cấp dịch vụ cho hàng nghìn khách hàng khắp cả nước với hàng chục server hoạt động suốt ngày đêm. Với đặc thù dịch vụ phải hoạt động 24/7 thông suốt đòi hỏi các sự cố xảy ra trên hệ thống phải nằm ở mức thấp nhất có thể, thời gian gián đoạn ngắn nhất có thể.

- **Hiện trạng máy chủ**

Theo thống kê sơ bộ, hệ thống máy chủ server của công ty iNET hoạt động cung cấp dịch vụ cho khách hàng xấp xỉ 20 server.

Dịch vụ	Số lượng máy chủ
Hosting	12 máy chủ
Máy chủ ảo	5 máy chủ
Email server	3 máy chủ

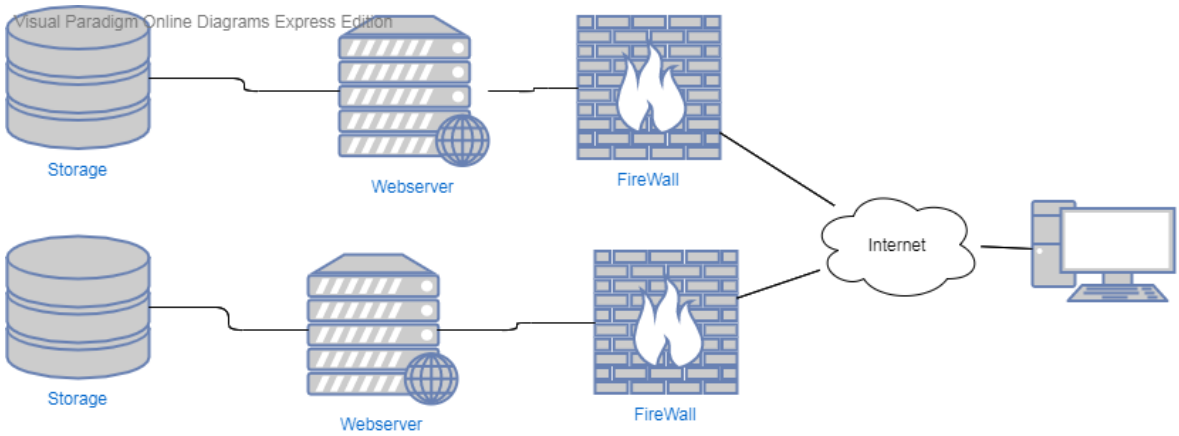
Bảng 3.1 Danh sách hệ thống máy chủ iNET

Do là công ty cung cấp dịch vụ lưu trữ cho nhiều khách hàng, do đó nền tảng hệ điều hành được cài đặt trên các máy chủ cũng đa dạng.

Windows	Linux
• Windows server 2012 • Windows server 2016	• CentOS 6 • CentOS 7 • CentOS 8 • Debian 8 • Debian 9 • Debian 10 • Ubuntu 14 • Ubuntu 16 • CloudLinux

Bảng 3.2 Danh sách hệ điều hành cài trên serever iNET

Mô hình hoạt động cơ bản của hệ thống dịch vụ.



Hình 3.1 Mô hình hoạt động của server iNET

Hệ thống server hoạt động theo mô hình nhiều server riêng biệt, mỗi server sẽ có 2 phần đó là gồm webserver xử lý các request đến từ client thông qua giao thức HTTP, sau đó tiến hành lấy dữ liệu từ Storage để hiển thị cho người dùng. Giữa các server sẽ được cài đặt firewall để đảm bảo an ninh, an toàn cho hệ thống.

- **Hiện trạng quản lý hệ thống**

Hiện tại iNET không chú trọng quá nhiều vào việc phân tích dữ liệu log server. Dữ liệu log sẽ được người quản trị hệ thống phân tích chỉ khi có lỗi gì xảy ra, nếu không các dữ liệu log sẽ được lưu lại và xóa đi theo định kỳ để giải phóng dung lượng ổ đĩa. Trong trường hợp xảy ra trục trặc hệ thống, quản trị viên sẽ tiến hành kiểm tra log trên server bị lỗi bằng cách truy cập thẳng vào thư mục chứa log của server này. Quá trình phân tích, lọc, đánh giá log sẽ phụ thuộc hết vào người đang phân tích và gửi kết quả sang những người chịu trách nhiệm sửa lỗi. Cách thức kiểm tra dữ liệu log của công ty iNET còn tồn tại các hạn chế sau:

- Quản trị viên phải truy xuất dữ liệu thủ công trên các máy chủ phân tán để phân tích và tìm lỗi trên hệ thống.
- Việc chờ đợi bị lỗi rồi mới phân tích log server khiến cho hệ thống nằm ở thế bị động có nguy cơ gặp nhiều rủi ro hơn so với việc phân tích log đều đặn hoặc có hệ thống giám sát.
- Phân tích thủ công đôi khi dẫn đến việc phân tích không đầy đủ mọi dữ liệu khiến bỏ sót thông tin.
- Dữ liệu log server chứa rất nhiều thông tin quan trọng về hệ thống, việc xóa định kỳ log sever tuy giải quyết được việc giải phóng dung lượng nhưng vô tình cũng làm mất đi rất nhiều dữ liệu cần thiết cho việc quản trị hệ thống sau này..
- Yêu cầu nhiều thời gian để tiến hành phân tích gây tốn nguồn lực con người trong khi hoàn toàn có thể giải quyết bằng việc tự động hóa.

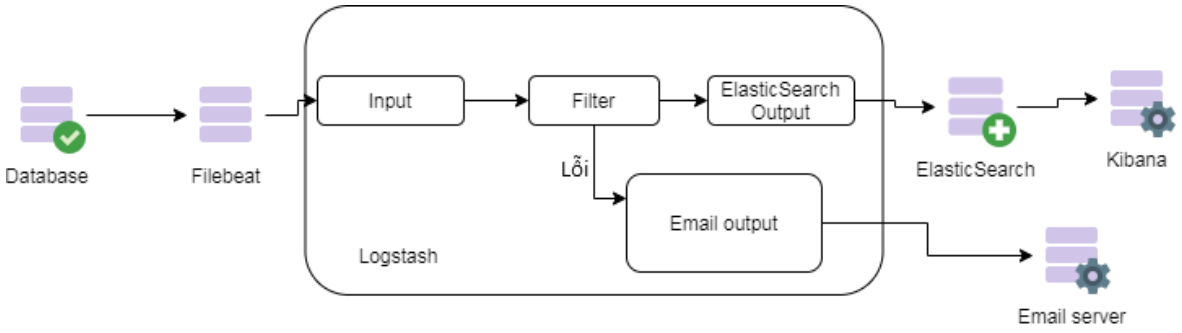
Vì những lý do trên, học viên sẽ ứng dụng cài đặt hệ thống phân tích log server trên trên server của công ty iNET dựa trên ELK stack nhằm nâng cao khả năng phân tích dữ liệu log, ứng dụng vào bảo mật hệ thống.

Do hệ thống server của công ty iNET lớn, đa dạng hóa về phần mềm, nền tảng, nên công đoạn triển khai sẽ được thực hiện thí điểm trên một server. Sau khi thí điểm sẽ dựa vào tính khả thi để áp dụng trên các server còn lại. Dựa trên dữ liệu

log, chúng tôi xây dựng màn hình giám sát, tìm kiếm, phát hiện lỗi tự động thay thế quá trình xử lý dữ liệu log thủ công trước đây. Quá trình thử nghiệm được thực hiện qua các công đoạn sau: Server được cài đặt thí điểm sử dụng Litespeed webserver, hệ điều hành CloudLinux bao gồm đầy đủ các thành phần máy chủ http, cơ sở dữ liệu,… được yêu cầu hoạt động 24/7 để đảm bảo dịch vụ không bị ngắt quãng.

3.1.2 Mô hình thử nghiệm

Mô hình cài đặt thử nghiệm hệ thống phân tích log cần được xây dựng theo hướng tối ưu cho việc lấy dữ liệu log lẫn thông báo cho quản trị viên hạn chế tốt nhất việc chồng chéo gây tải nặng lên hệ thống. Dựa theo mục 2.4 thiết kế hệ thống phân tích log, học viên sẽ ứng dụng vào để xây dựng một hệ thống phân tích log thử nghiệm.



Hình 3.2: Mô hình xây dựng hệ thống phân tích log

Dữ liệu log server sẽ được Filebeat trích xuất ra và gửi vào Logstash, sau đó Logstash sẽ lọc nội dung log rồi đưa vào đánh chỉ mục trên ElasticSearch hoặc đưa vào email server nếu gặp lỗi. Các dữ liệu được đánh chỉ mục trong ElasticSearch sẽ được hiển thị trên Kibana để quản trị viên tiện theo dõi.

3.1.3 Cài đặt hệ thống phân tích log bằng ELK stack

- **Cấu hình filebeat**

Filebeat sẽ có nhiệm vụ truyền dữ liệu log từ thư mục chứa sang Logstash. Tại đây filebeat sẽ nhận tất cả các file log nằm trong thư mục và đẩy sang port 5044 để Logstash có thể nhận được dữ liệu log. Filebeat hoạt động như một service giúp

giảm tải cho CPU nhưng vẫn đảm bảo dữ liệu được chuyển sang một cách nhanh nhất. Mọi thay đổi cấu hình filebeat sẽ nằm trong phần filebeat.yml.

```
#Cấu hình đầu vào filebeat

- type: log

  paths:

    - \filebeat\logaccess\*.log

  fields:

    type: apache

    fields: access

  fields_under_root: true

  encoding: utf-8

- type: log

  paths:

    - \filebeat\logerror\*

  fields:

    type: error log

    fields: error

  fields_under_root: true
```

```
  encoding: utf-8#

Cấu hình đầu ra filebeat

output.logstash:

hosts: ["localhost:5044"]

#đầu ra tại đây để cổng mặc định của Logstash, dữ liệu sau khi lấy sẽ được chuyển đến port 5044
```

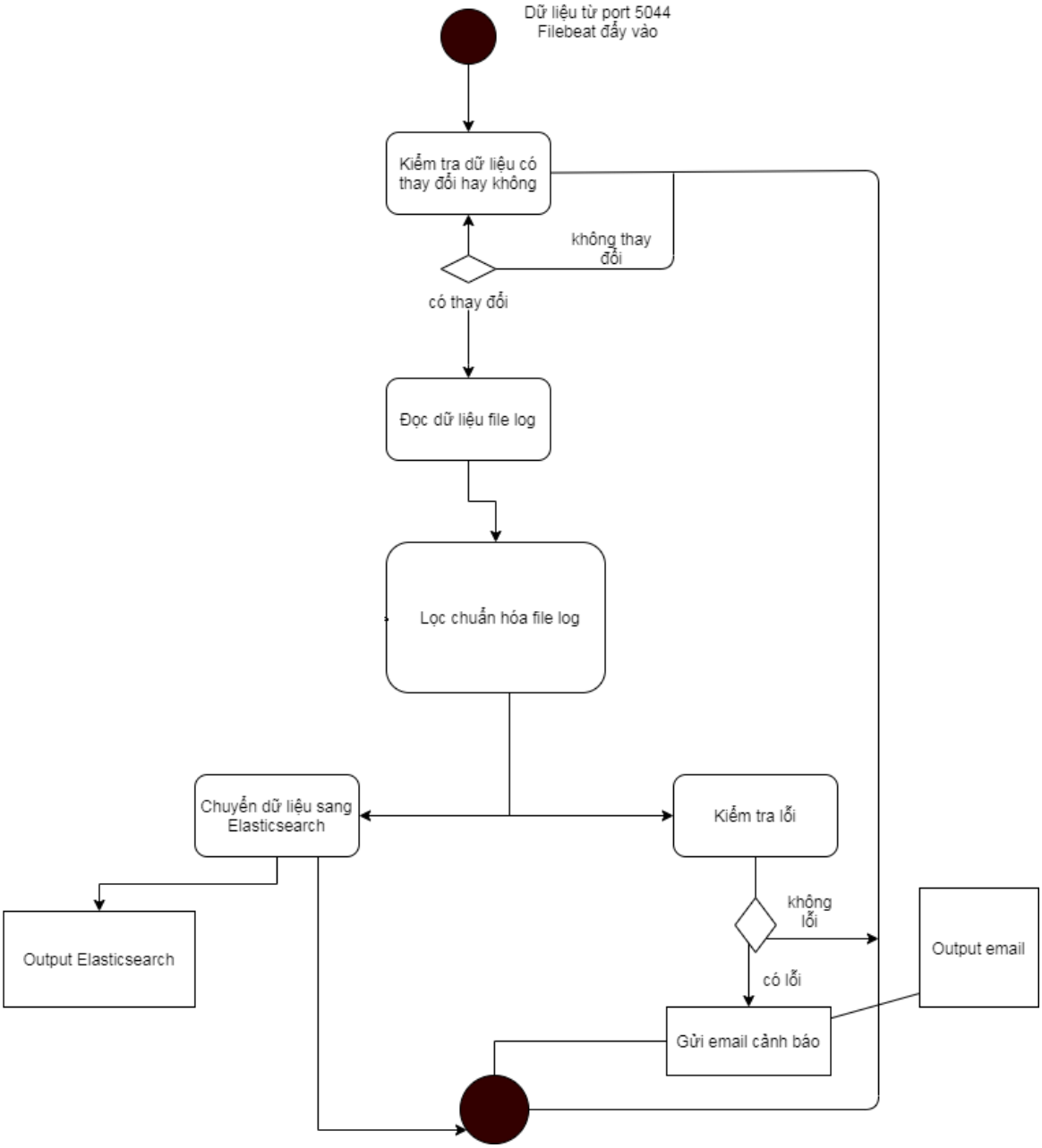
- **Cấu hình Logstash**

Logstash khi khởi chạy sẽ cần có một file config để có thể hướng dẫn hoạt động. Do vậy mà sẽ cần đến một file config chứa toàn bộ input, filter và output. Nếu như không có file config để hướng dẫn Logstash sẽ dẫn đến việc Logstash không thể hoạt động.

Học viên sẽ tạo một file config có tên là mypipeline.conf hướng dẫn Logstash chạy.

```
bin/logstash -f mypipeline.conf
```

Do vậy phần dưới học viên sẽ tạo một file config cho Logstash hoạt động theo như mô hình dưới:



Hình 3.3 Mô hình xử lý dữ liệu file config logstash

Sau khi khởi chạy Filebeat như một service xong, dữ liệu sẽ được đẩy đến port output được cài đặt khi cấu hình filebeat . Để dữ liệu log server được gửi đến Logstash thì cần cấu hình đầu vào cho Logtash.

```
#Cấu hình đầu vào Logstash

#Lấy dữ liệu từ filebeat được chuyển qua port 5044

input {

   beats {

      port => "5044"

   }

}
```

Sau khi lấy được dữ liệu logserver từ filebeat thì cần cấu hình filter để dữ liệu log nhất quán. Vì có nhiều loại log được đưa vào Logstash do vậy sẽ có nhiều dữ liệu log với các cấu trúc khác nhau. Tại đây học viên sẽ chuyển các nội dung log thành một dòng với các biểu thức chính quy nhằm giúp file log dễ dàng sử dụng hơn.

```
#Cấu hình file filter

filter {

# cấu hình theo dạng mặc định của file apache

filter {

  if [fields] =="access" {

grok {

          match => { "message" => "%{COMBINEDAPACHELOG}" }
```

```
            overwrite => "message"

        }           overwrite => "message"

        }

  }

if [fields] =="error"  {

    grok {

            match => { "message" => "%{APACHE_ERROR_LOG}" }

        }

}

 mutate {

     remove_field => "host"

}

      date {

    match => [ "timestamp" , "dd/MMM/yyyy:HH:mm:ss Z" ]

      }

  geoip {

      source => "clientip"

    }
```

```
}
```

Do trên server có nhiều loại log khác nhau nên học viên sẽ áp dụng tương ứng các bộ log cho từng loại log riêng, ở trên dữ liệu log được lọc theo 2 loại đó là access log và error log.

Dữ liệu log được sau khi Logstash trích xuất sẽ tổng hợp dữ liệu sau đó được đưa vào 2 đầu ra là ElasticSearch để đánh chỉ mục và Email Server để gửi email cảnh báo tự động nếu lỗi được phát hiện trong dữ liệu log. Sau đó dữ liệu sau khi được đánh chỉ mục trong ElasticSearch thì sẽ được đưa lên Kibana để người dùng, các nhà phân tích dữ liệu xây dựng các biểu đồ, trừu tượng hóa dữ liệu để phân tích chúng

```
output {

#đầu ra cho elasticsearch

  elasticsearch {

    hosts => ["localhost:9200"]

          index => "%{[@metadata][filebeat]}-%{+YYYY.MM.dd}"

  }

   stdout { codec => rubydebug }

#đầu ra cho email báo lỗi

   if "access denied" in [message] {

email {
```

```
port => 587

address => "smtp.gmail.com"

username => "sin****gum16@gmail.com"

password => "Tung*****"

authentication => "plain"

use_tls => true

from => " sin****gum16@gmail.com "

subject => " Cảnh Báo: Phát hiện có truy cập bị chặn"

to => "htv.sky.1994@gmail.com"

via => "smtp"

body => "%{message}"

}
```

- **Cấu hình Elasticsearch và Kibana**

Sau khi thiết lập các cài đặt trên Filebeat và Logstash hoàn thành, dữ liệu sẽ được chuyển sang lập chỉ mục tại Elasticsearch rồi hiển thị thông qua Kibana. Với mô hình cơ bản được sử dụng trong luận văn này, thì các cài đặt cơ bản có sẵn trong Elasticsearch là đã đủ để đáp ứng nhu cầu cho việc phân tích log trên một server.

Để có thể hiển thị các dữ liệu được lập chỉ mục trong ElasticSearch, học viên sẽ cấu hình port chạy Kibana lẫn port của ElasticSearch:

```
# Kibana is served by a back end server. This setting specifies the port to use.
```

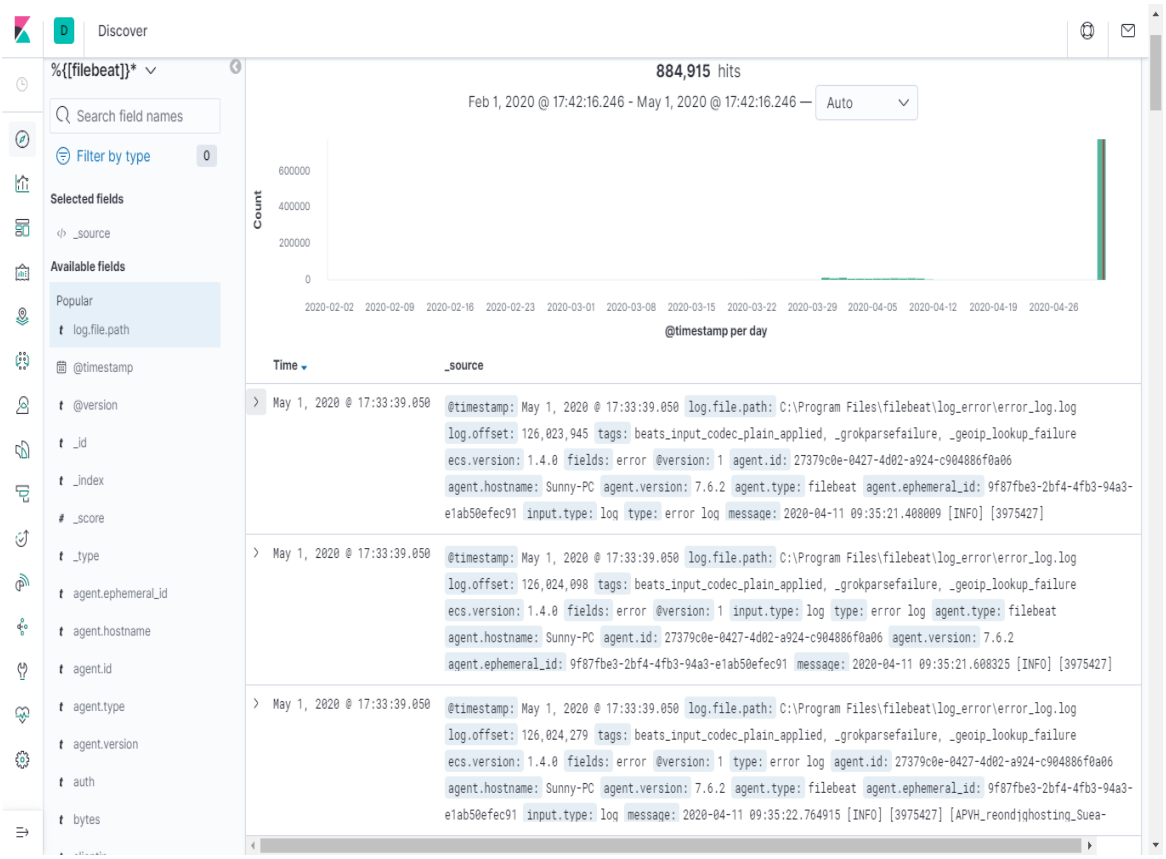
```
server.port: 5601

# The URLs of the Elasticsearch instances to use for all your queries.

elasticsearch.hosts: ["http://localhost:9200"]
```

3.2 Vận hành và thử nghiệm

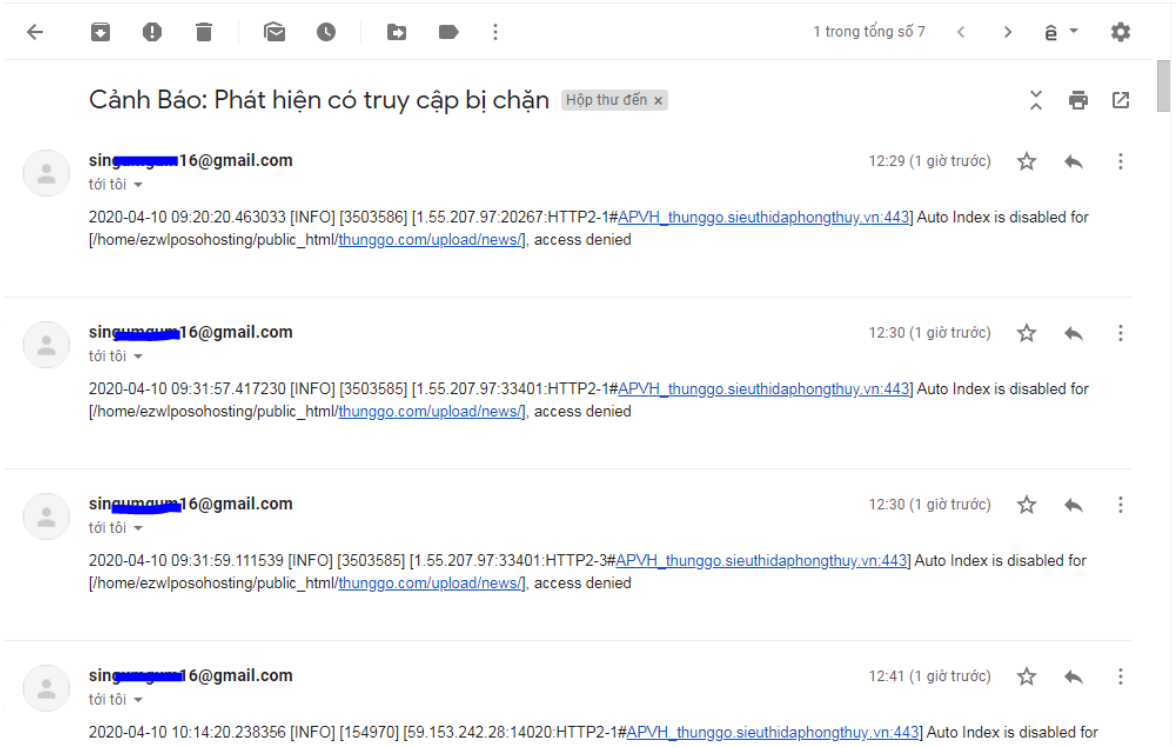
Sau khi cài đặt toàn hệ thống phân tích ELK trên server, toàn bộ dữ liệu log thu thập được sẽ được hiển thị trên Kibana. Kibana cung cấp một giao diện trực quan dễ nhìn và dễ dàng sử dụng tạo thuận lợi cho việc quản trị log server. Dữ liệu log sẽ được chia ra các trường như hình ảnh dưới kèm theo một filter ở bên trái cho phép lựa chọn các trường dữ liệu nào cần thiết cho việc phân tích.



Hình 3.4: Giao diện quản lý dữ liệu log tổng quan trên Kibana

Khi đưa giải pháp ELK vào xây dựng hệ thống quản lý log, dữ liệu log sẽ được thu thập theo thời gian thực (real-time). Ngay khi nào hệ thống nhận ra các bất thường trên hệ thống, lập tức sẽ tự động gửi email cảnh báo cho quản trị viên biết để có các thức xử lý. Quản trị viên thông qua việc nhận mail báo lỗi thể biết được vấn đề mà hệ thống gặp phải ngay lập tức. Điều này khi so với việc giám sát dữ liệu thủ công thường thấy sẽ là một bước đi đúng đắn, nó giúp quản trị viên có thể có những phương án thích hợp cho server khi server bị tấn công.

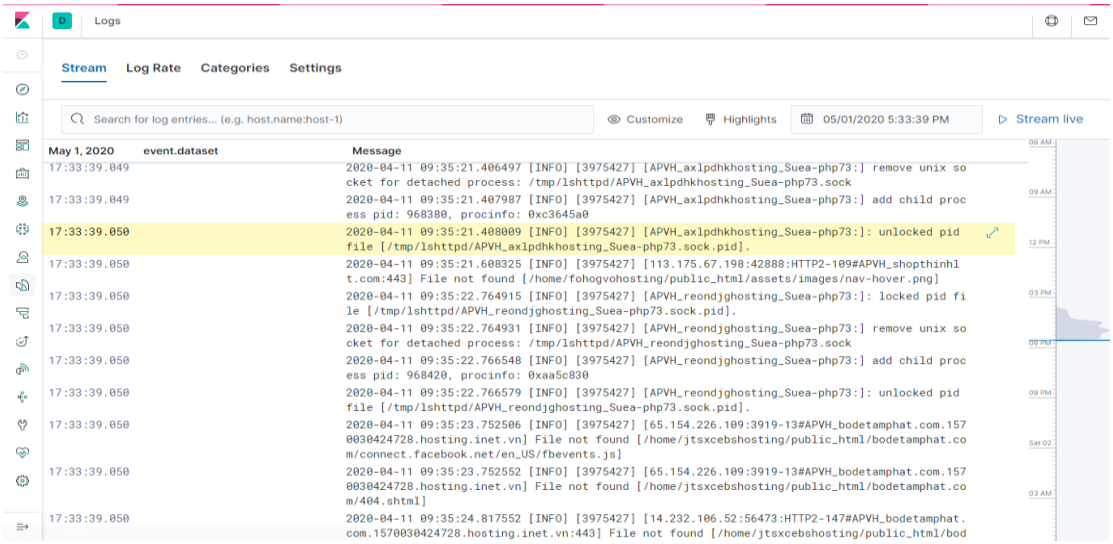
Trong phần cấu hình Logstash đã được cài đặt khi phát hiện ra các truy cập bị chặn lại hệ thống sẽ tiến hành gửi mail ngay lập tức cho quản trị viên.



Hình 3.5 Hệ thống gửi mail báo lỗi cho quản trị viên

Dữ liệu log khi theo dõi thủ công gặp nhiều vấn đề như hạn chế việc dõi tiến trình sinh ra log hay theo dõi nhiều file log cùng một lúc gây mất thời gian. Để cải thiện vấn đề này hệ thống phân tích log server ELK cung cấp biểu đồ thông qua Kibana cho phép quản trị viên theo dõi các dữ liệu log đang được thu thập theo thời gian thực. Từ biểu đồ, quản trị viên hoàn toàn có thể nắm được tại một thời điểm

đang có bao nhiêu dòng log được sinh ra và có cách thức xử lý hiệu quả khi log server thu được quá nhiều.



Hình 3.6 Giao diện hiển thị theo thời gian thực việc lấy dữ liệu log

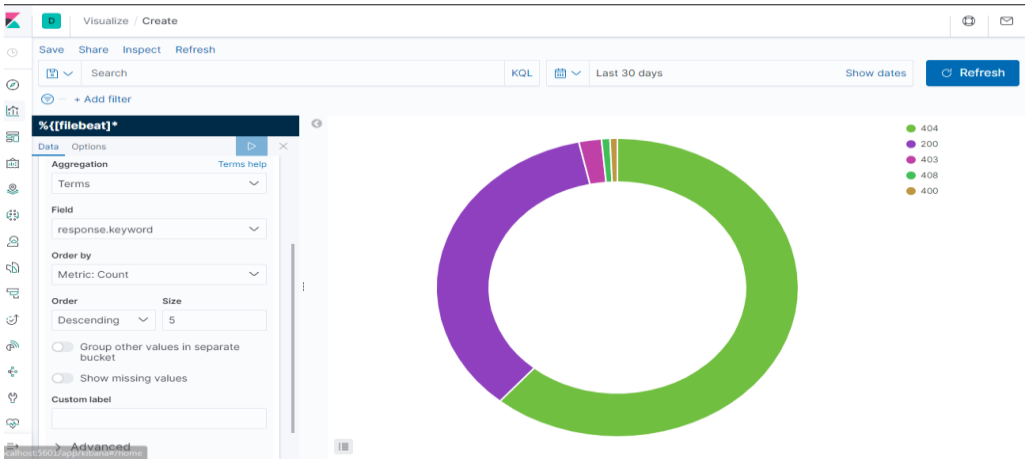
3.3 Phân tích các dữ liệu thu được từ log Server

Từ dữ liệu log thu được sau khi cài đặt hệ thống phân tích log server ELK stack, học viên có thể dựa vào những dữ liệu này để kiểm tra xem hệ thống có đang gặp vấn đề gì hay không.

- **Phân tích tấn công DDoS dựa trên dữ liệu log**

DDoS là phương thức tấn công tuy không lạ nhưng rất khó để có thể ngăn chặn hoàn toàn. Trong đó có cách thức tấn công phổ biến dựa trên việc khiến server bị hết băng thông hay gửi quá nhiều các yêu cầu dữ liệu sai khiến phần cứng không đáp ứng đủ. Cách thức tấn công trên đa phần dựa trên công cụ dẫn đến các đường dẫn, thư mục phổ biến được dùng sẽ là mục tiêu được quét đầu tiên, tuy nhiên về việc cài đặt không phải web server nào cũng như nhau nên các phản hồi 404 nhiều có thể đang phản ánh vấn đề gì đó với hệ thống. Do đó việc theo dõi các phản hồi 404 nhiều bất thường có thể giúp ích trong việc phân tích tấn công DDoS.

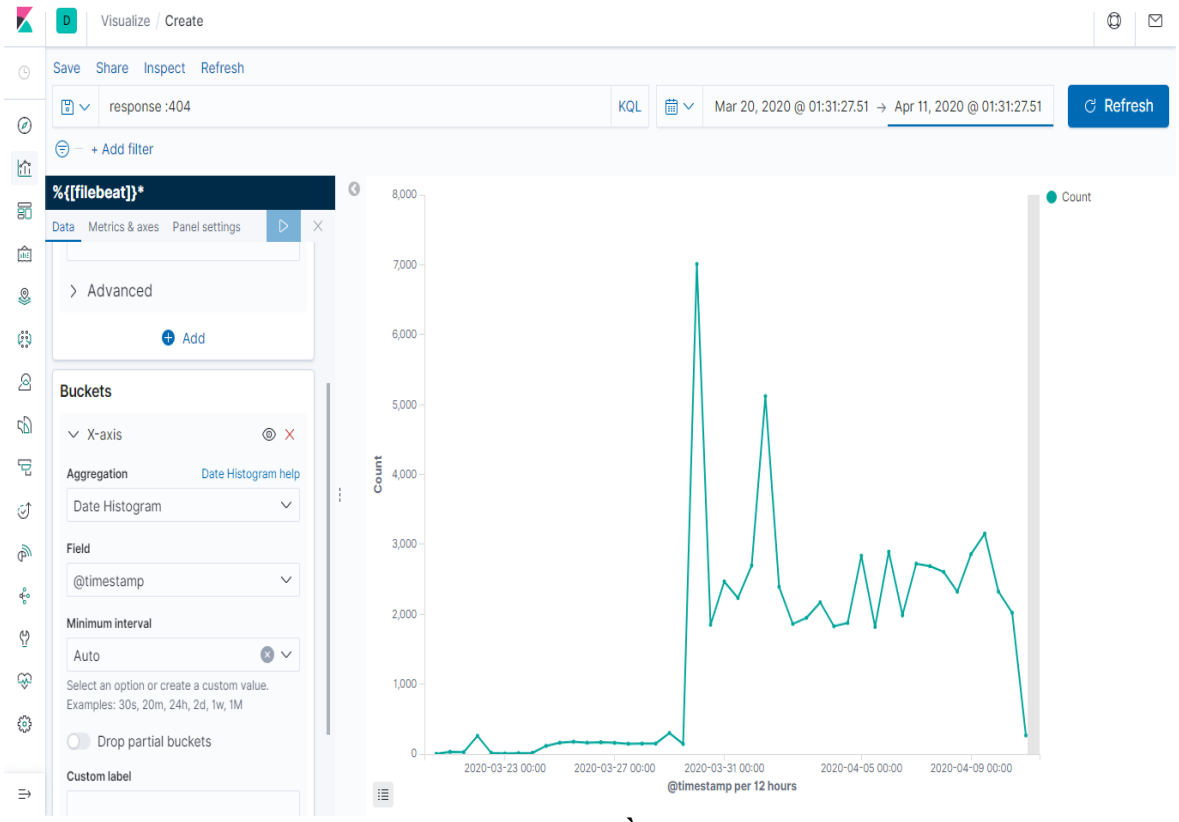
Do đó học viên sẽ tiến hành lập biểu đồ kiểm tra các phản hồi từ server về client.



Hình 3.7 Biểu đồ các phản hồi từ server về client

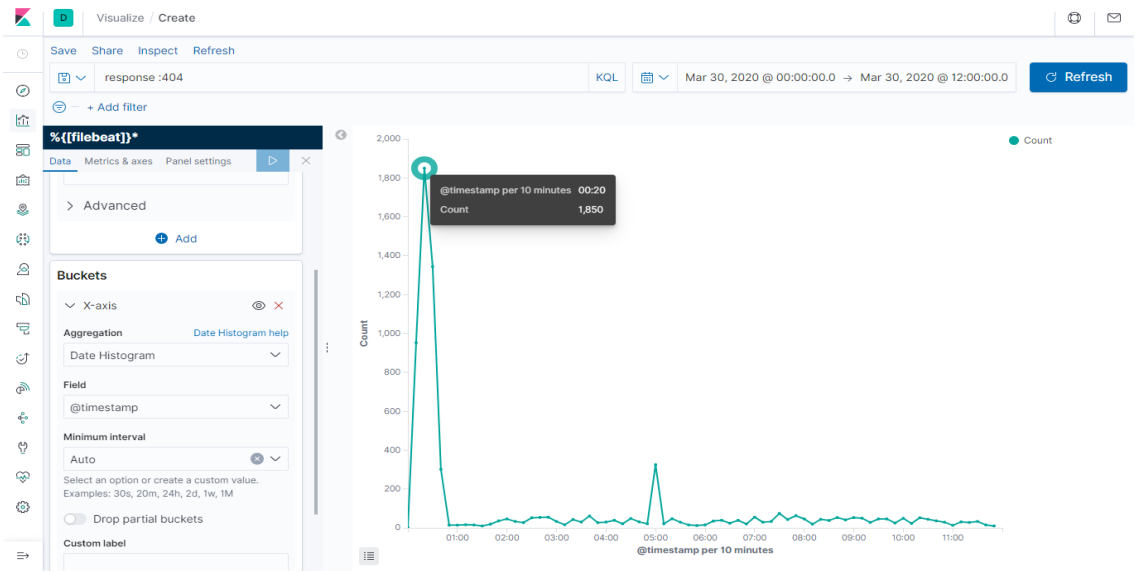
Biểu đồ dựa vào dữ liệu log như hình trên cho thấy số lượng các phản hồi request đến từ server, các phản hồi 404 có số lượng nhiều hơn rất nhiều so với các phản hồi khác. Đây có thể là một lỗ hổng khi có quá nhiều lượt truy cập vào server nhưng lại đến một trang không nằm trên hệ thống.

Quản trị viên cần phải khai thác xem thời gian nào hệ thống gặp nhiều request 404 và bởi các IP nào. Dựa trên số liệu log đã thu được, quản trị viên hoàn toàn có thể biết được thời điểm nào hệ thống đang có số lượng yêu cầu 404 gia tăng.



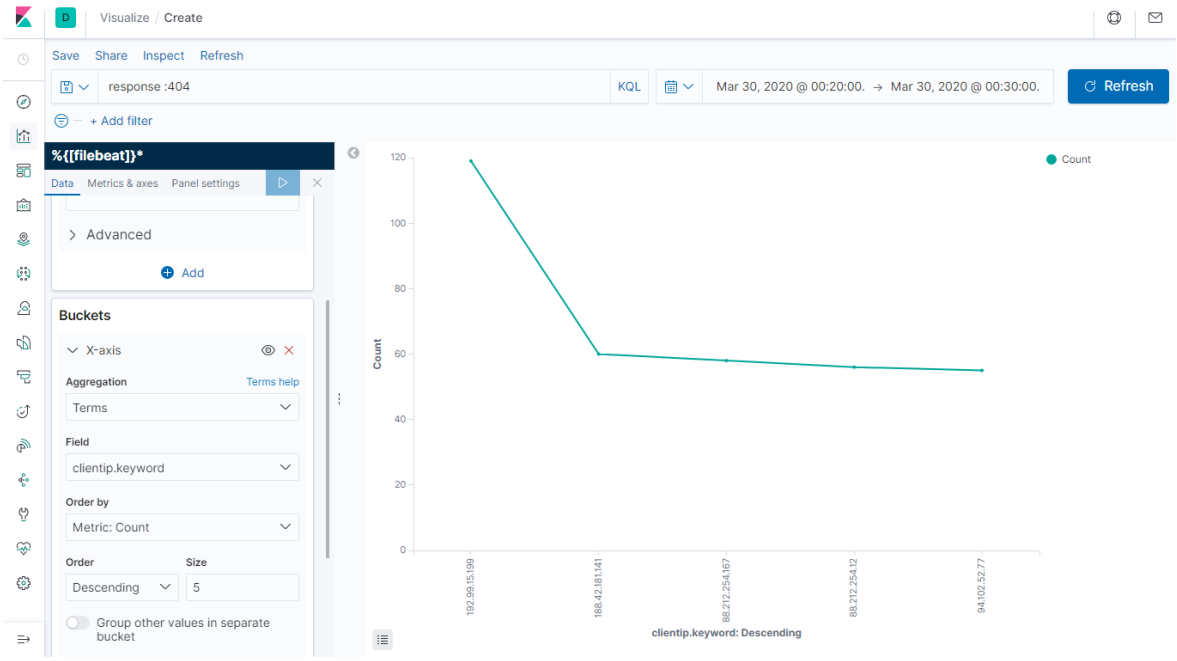
Hình 3.8 Các phản hồi 404 từ Server

Qua biểu đồ lọc các phản hồi 404 theo thời gian, có thể nhận thấy vào ngày 30 tháng 3 số lượng phản hồi 404 tăng cao bất thường, tiếp tục nhấp vào ngày 30 để có thêm các thông tin vào khoảng thời gian này.



Hình 3.9 Chi tiết lượng phản hồi 404 trong ngày 30/3

Sau đó tiến hành lọc các IP gửi nhiều request đến server trong thời điểm đang truy xét.



Hình 3.10 Thống kê các IP nhận phản hồi 404 cao trong ngày 30/3

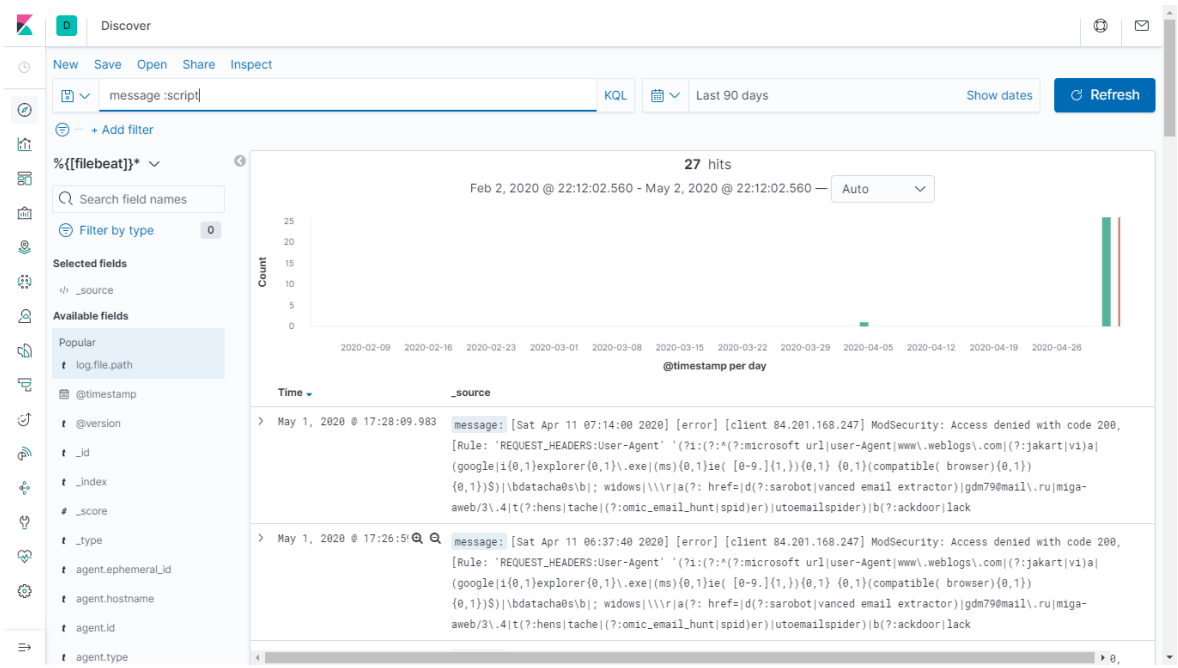
Thông qua biểu đồ trên có thể nhận thấy IP 192.99.15.199 đang có lượng phản hồi 404 cao cụ thể trong 10 phút đã tiến hành gửi 120 lần request đến server,

do vậy IP này có thể được đưa vào danh sách nghi vấn và chặn nếu lượt phản hồi cao tiếp diễn đều đặn.

Trong thời gian cài đặt thử nghiệm hệ thống không ghi nhận các cuộc tấn công, nên trường hợp trên chỉ hoạt động như một cách thức tìm ra IP tấn công sever. Tuy nhiên việc theo dõi được các lượng request lỗi cao hoàn toàn có thể giúp quản trị viên chủ động trong việc phòng tránh bị tấn công DDoS.

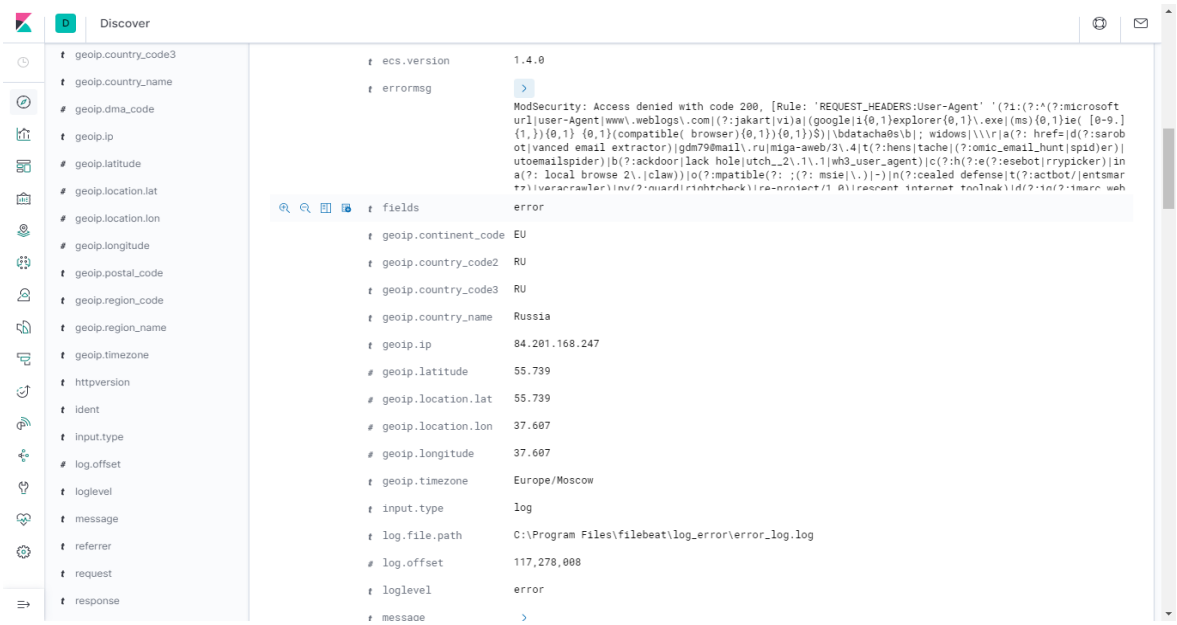
- **Phát hiện tấn công XSS với hệ thống phân tích log**

Như đã đề cập ở chương 1, các tấn công XSS cơ bản có thể phát hiện được thông qua việc tìm các thẻ như script, frame,… Thông qua dữ liệu log tiến hành tìm kiếm các thẻ có khả năng là do bị chèn mã độc.



Hình 3.11 Tìm kiếm script trong message

Qua việc tìm kiếm, có thể nhận ra có một số lượng script đã bị chèn trên server tuy nhiên đã bị chặn bởi Firewall ModSecurity. Nhấp chi tiết vào quản trị viên có thể xem được chi tiết về dữ liệu của script này.



Hình 3.12 Nội dung message chi tiết có chứa mã script

Dựa trên thông tin thu được, có thể đây là một srcipt được chèn vào nhằm crawl dữ liệu trên server. Với script crawl dữ liệu như vậy có thể dẫn đến tình trạng ăn tài nguyên của hệ thống hoặc website hoạt động không đúng như cài đặt ban đầu. Tuy nhiên script đã được chặn lại bởi tường lửa nên điều này không ảnh hưởng gì đến hệ thống.

3.4. Đánh giá, đề xuất bảo mật cho server

Sau khi thực hiện cấu hình thử nghiệm hệ thống phân tích log trên server đã được trình bày ở trên, học viên đánh giá kết quả đạt được có thể giải quyết được một số vấn đề được đặt ra trong bài toán đề cập tại chương 1 mục giới thiệu bài toán như:

Xây dựng được một hệ thống phân tích log có khả năng cập nhật dữ liệu theo thời gian thực, giúp cho việc phân tích log server dễ dàng và trực quan hơn với Kibana.

- Hệ thống phân tích log server giúp cho việc bảo mật server được dễ dàng hơn khi phát hiện và cảnh báo ngay lập tức khi hệ thống gặp lỗi, dấu hiệu bất thường giúp quản trị viên có thể nhận được thông báo lỗi một cách sớm và chính

xác nhất. So với việc phân tích thủ công hoặc đi theo sự cố thì đây là một lựa chọn hoàn toàn hợp lý.

Việc giải quyết được hai vấn đề trên cũng đồng nghĩa với việc giải quyết được phần nào những hạn chế khi sử dụng cũng như quản lý giám sát dữ liệu log trên server tại công ty iNET như đã đề cập ở đầu chương.

- **Các đề xuất bảo mật cho Server**

Qua việc triển khai hệ thống phân tích log trên server của công ty iNET, học viên nhận thấy những vấn đề dưới đây cần được thực hiện để có thể đảm bảo hệ thống được an toàn nhất:

- Phân tích dữ liệu log có ý nghĩa vô cùng quan trọng trong bảo mật của server, nhưng chỉ sử dụng riêng một hệ thống phân tích log dựa trên ELK stack là chưa đủ để có thể đảm bảo an toàn bảo mật cho server. Việc kết hợp với những công cụ giám sát khác như giám sát mạng, giám sát hệ thống để nâng cao khả năng truy tìm, chặn các lỗ hổng cơ bản hay thông báo lỗi đến quản trị viên là rất cần thiết.
- Tường lửa, các công cụ antivirus vẫn là những phần mềm quan trọng trong việc bảo mật hệ thống server. Các phần mềm này giúp cho hệ thống ngăn chặn từ đầu các tác nhân gây hại giúp giảm thiểu khối lượng công việc cần làm cho quản trị viên. Điều này đồng nghĩa với việc hệ thống server cần được cập nhật các phần mềm tường lửa, antivirus nhanh và sớm nhất.
- Là một hệ thống server dịch vụ lưu trữ online nên dữ liệu là thứ thiết yếu nhất, nên hệ thống sẽ cần một ổ để sao lưu lại toàn bộ các dữ liệu được đưa lên server dịch vụ. Trong những trường hợp xấu nhất, việc sao lưu dữ liệu sẽ giúp giải quyết được vấn đề của khách hàng trước khi có những động thái tìm kiếm lỗ hổng trên server.
- Kiểm tra các dữ liệu log để nắm được các thông tin về hệ thống định kỳ ngay cả khi đã xây dựng tự động báo lỗi để tránh việc để sót thông tin quan trọng.

3.5 Kết luận

Chương 3 luận văn đã trình bày chi tiết về việc triển khai hệ thống phân tích log trên server của công ty iNET dựa trên công nghệ ELK stack cho phép quản trị viên có thể quản lý log tập trung, nhận thông báo lỗi từ sớm giúp sớm khắc phục các lỗi hệ thống. Bên cạnh đó mô hình vận hành hệ thống phân tích và luồng lấy dữ liệu log được học viên thiết kế cũng hoạt động theo ý muốn giúp dữ liệu log được tập trung toàn bộ trên hệ thống phục vụ cho việc phân tích dễ dàng hơn.

Việc cài đặt thành công hệ thống phân tích log trên server của công ty iNET sẽ tạo tiền đề cho việc triển khai cho những server khác nâng cao khả năng phòng ngự bảo mật. Một số thử nghiệm đánh giá về dữ liệu log thu được trên server công ty iNET cũng đã được học viên thực hiện. Với việc chọn các yếu tố quan trọng, loại bỏ các yếu tố dư thừa khi phân tích dữ liệu log; học viên đã có thể xác định được nguy cơ xảy ra của các cuộc tấn công nhằm vào server. Bên cạnh đó chương 3 tổng hợp một số đề xuất cho server nhằm nâng cao khả năng chống tấn công cho sever, hạn chế mức thấp nhất những tác động gây hại giảm tải việc phải giám sát hệ thống quá nhiều.

KẾT LUẬN

1. Những đóng góp của luận văn:

Với mục tiêu nghiên cứu bài toán tìm hiểu file log và ứng dụng file log vào bảo mật server. Luận văn đã đi sâu vào nghiên cứu về các vấn đề xung quanh file log, và những ứng dụng của file log đối với server.

Những kết quả đã đạt được trong luận văn:

- Tìm hiểu tổng quan về file log server, những ứng dụng của file log server trong việc vận hành lẫn bảo mật hệ thống, những cách thức phân tích và cách nhận biết tấn công thông qua file log.
- Tìm hiểu về công nghệ ELK stack và dựa vào công nghệ học viên đã xây dựng được một mô hình tổng quát hệ thống phân tích log server.
- Tìm hiểu về hệ thống máy chủ công ty iNET, sau đó tiến hành xây dựng mô hình hệ thống phân tích log cho những máy chủ trên. Thông qua mô hình hệ thống phân tích log, học viên đã cấu hình chạy thử nghiệm thành công hệ thống phân tích dữ liệu log và cảnh báo lỗi đến quản trị viên.

2. Hướng phát triển luận văn

Tuy đạt được một số kết quả đã nêu ở trên, nhưng luận văn còn có những hạn chế do điều kiện về mặt thời gian lẫn trình độ của học viên. Vì vậy, hướng nghiên cứu tiếp theo của học viên đó là:

- Áp dụng thêm các công cụ khác vào hệ thống phân tích file log nhằm nâng cấp tính năng, nâng cao khả năng quản trị server lẫn dữ liệu log.
- Những hình thức tấn công vào hệ thống server đang phát triển không ngừng theo thời gian. Tin tặc hiện đang áp dụng những hình thức tấn công đa dạng hơn, khó để phát hiện và ngăn chặn hơn đang là bài toán khó trong bảo mật server đối với các quản trị viên. Luận văn sẽ phát triển theo hướng tìm hiểu về các hình thức tấn công mới nhất, thông qua tìm hiểu cách thức để áp dụng vào việc phân tích log.

- Không chỉ dùng dữ liệ trong phân tích log vào bảo mật server, luận văn có thể phát triển theo hướng ứng dụng vào những vấn đề liên quan đến người dùng nhằm cải thiện chất lượng của hệ thống.
- Ứng dụng của file log server vào bảo mật server là vô cùng lớn. Quản trị viên hoàn toàn có thể phát triển hệ thống SIEM (hệ thống giám sát an ninh mạng) bằng những dữ liệu log trên server để quản lý server một cách toàn diện nhất.

DANH MỤC CÁC TÀI LIỆU THAM KHẢO

Tiếng Việt

[1] Kim Thanh (2019), *Đang có một chiến dịch tấn công có chủ đích nhằm vào các Server Public của Việt Nam*

https://www.sggp.org.vn/dang-co-mot-chien-dich-tan-cong-co-chu-dich-nham-vao-cac-server-public-cua-viet-nam-575735.html

[2] Kiến Văn (2019), *Wikipedia xác nhận sự cố ngừng hoạt động do bị tấn công DDoS*

https://thanhnien.vn/cong-nghe/wikipedia-xac-nhan-su-co-ngung-hoat-dong-do-bi-tan-cong-ddos-1123957.html

Tiếng Anh

[3] Arvind K. Sharma1, P.C. Gupta (2013), "Analysis of Web Server Log Files to Increase the Effectiveness of the Website Using Web Mining Tool", *International Journal of Advanced Computer and Mathematical Sciences*, vol. 4, issue 1, pp1-8.

[4] BulletProof (2019), *BulletProof Annual Cyber Security Report 2019*

https://www.bulletproof.co.uk/industry-reports/2019.pdf

[5] Krishnamoorthi R., K. R. Suneetha (2009), "Identifying User Behavior by Analyzing Web Server Access Log File" *International Journal of Computer Science and Network Security*, vol. 9, no. 4.

[6] Karen Kent, Murugiah Souppaya (2006), *Guide to Computer Security Log Management*, National Institute of Standards and Technology, Gaithersburg.

[7] Mohammed Hamed Ahmed Elhiber and Ajith Abraham (2013), "Access Patterns in Web Log Data: A Review" *Journal of Network and Innovative Computing*, ISSN 2160-2174, pp. 348-355.

[8] Merve Bas ß SeyyarFerhat, Ozgur Catak, Ensar Gul (2017), "Detection of attack-targeted scans from the Apache HTTP Server access logs" *Applied Computing and Informatics*, Volume 14, Issue 1, January 2018, Pages 28-36.

[9] Neha Goel, C.K. Jha (2013), "Analyzing Users Behavior from Web Access Logs using Automated Log Analyzer Tool" *International Journal of Computer Applications*, vol 62, no. 2.

[10] Roger Meyer (2008), *Detecting Attacks on Web Applications from Log Files*
https://www.sans.org/reading-room/whitepapers/logging/detecting-attacks-web-applications-log-files-2074.

[11] Satoru Kobayashi, Kensuke Fukuda, Hiroshi Esaki (2014), *Towards an NLP-based log template generation algorithm for system log analysis*

https://dl.acm.org/doi/pdf/10.1145/2619287.2619290.

[12] https://www.elastic.co/what-is/elasticsearch, truy nhập ngày 10/02/2020

[13] https://www.elastic.co/logstash, truy nhập ngày 14/02/2020

[14] https://www.elastic.co/what-is/kibana, truy nhập ngày 20/02/2020

[15] https://hello.global.ntt/, truy nhập ngày 30/03/2020

[16] https://www.splunk.com/ truy nhập ngày 15/01/2020

[17] https://www.graylog.org/ truy nhập ngày 15/01/2020

[18] https://www.elastic.co/what-is/elk-stack truy nhập ngày 16/01/2020

[19] https://stanfordnlp.github.io/CoreNLP/ truy nhập ngày 10/02/2020

[20] https://en.wikipedia.org/wiki/SAP_HANA truy nhập ngày 10/02/2020