# LỜI MỞ ĐẦU

## LÝ DO CHỌN ĐỀ TÀI

Ngày nay do tốc độ phát triển như vũ bão của các ngành khoa học kỹ thuật, đặc biệt là sự bùng nổ trong lĩnh vực công nghệ thông tin làm cho số lượng tri thức nhân loại tăng lên một cách “chóng mặt”, cùng với việc Việt Nam chính thức là thành viên của tổ chức thương mại thế giới WTO. Nền kinh tế tài chính ngày càng phát triển khởi sắc, đồng nghĩa với việc dữ liệu thông tin vô cùng quan trọng, quyết định đến sự sống còn của doanh nghiệp. Chính vì thế quan niệm về bảo mật an ninh mạng ngày được quan tâm hơn. Giám sát an ninh mạng chính là phương thức giúp chúng ta có thể thực hiện việc này một cách tối ưu nhất.

Trong vai trò là người quản trị hệ thống hay là một chuyên gia bảo mật thông tin thì công tác giám sát luôn là việc cần thiết. Chúng ta cần phải biết những gì đang xảy ra trên hệ thống của mình vào mọi lúc mọi nơi, bao gồm thời gian thực. Nắm bắt mọi thông tin lịch sử về sử dụng, hiệu suất và tình trạng của tất cả các ứng dụng, thiết bị và tất cả dữ liệu trên hệ thống mạng. Chính vì vậy việc giám sát hệ thống là một công việc vô cùng quan trọng và cấp thiết đối với mọi tổ chức, doanh nghiệp, cơ quan.

Có một số phần mềm thương mại cũng như phần mềm mã nguồn mở để giám sát mạng rất mạnh cùng với những công cụ hỗ trợ như là Splunk, Logstach, Loglly, Papertrail…. Graylog cũng thuộc nhóm những công cụ này, mặc dù là một công cụ còn mới tuy nhiên nó vẫn thể hiện là một công cụ khá tốt dùng để triển khai trong một hệ thống giám sát an ninh mạng.

## MỤC TIÊU CỦA ĐỀ TÀI

Biết được tầm quan trọng của việc giám sát an ninh mạng và xây dựng một hệ thống giám sát log tập trung, biết được các công cụ cần thiết để triển khai hệ thống và thấy được cơ chế hoạt động của các công cụ trong hệ thống đã triển khai.

## PHẠM VI NGHIÊN CỨU

* Các lý thuyết về giám sát an ninh mạng
* Các lý thuyết về công cụ nguồn mở Graylog2 và các công cụ hỗ trợ.
* Các mô hình trong thực tế.

## Ý NGHĨA KHOA HỌC VÀ THỰC TIỄN

* **Ý nghĩa khoa học:**
* Cung cấp các lý thuyết về giám sát an ninh mạng cũng như công cụ Graylog2 và các thành phần hỗ trợ.
* **Ý nghĩa thực tiễn:**
* Có thêm giải pháp để triển khai hệ thống quản lý log tập trung.

## NỘI DUNG CHÍNH CỦA KHÓA LUẬN

Trong khóa luận này em sẽ đưa ra các lý thuyết tổng quan về giám sát an ninh mạng sau đó sẽ đi sâu vào lý thuyết tổng quát, các giải pháp log tập trung và cuối cùng là thực hiện triển khai cài đặt graylog2 trên linux rồi rút ra thực nghiệm. Nội dung bao gồm 3 chương chính như sau:

* Chương 1: Tổng Quan Về Giám Sát An Ninh Mạng
* Chương 2: Tìm Hiểu Về Graylog2
* Chương 3: Triển Khai Hệ Thống Graylog2 Trên Linux

# CHƯƠNG 1: TỔNG QUAN VỀ GIÁM SÁT AN NINH MẠNG

## Khái niệm

### 1.1.1 Giới thiệu chung

Internet phát triển, sự kết nối trên toàn thế giới đang mang lại thuận tiện cho tất cả mọi người. nhưng bên cạnh đó nó cũng tiềm ẩn những nguy cơ đe dọa đến mọi mặt của đời sống xã hội. việc đánh cắp thông tin, truy cập hệ thống trái phép, tấn công từ chối dịch vụ... là nguy cơ mà người dùng Internet phải đương đầu.

Rất nhiều các giải pháp an ninh mạng đã được đưa ra và cũng đã có những đóng góp to lớn trong việc đảm bảo an toàn thông tin, ví dụ như: Firewall ngăn chặn những kết nối không đáng tin cậy, mã hóa làm tăng độ an toàn cho việc truyền dữ liệu, các chương trình diệt virus với cơ sở dữ liệu được cập nhật thường xuyên…

Tuy nhiên thực tế cho thấy chúng ta vẫn luôn thụ động trước các cuộc tấn

công đặc biệt là các tấn công kiểu mới vì vậy yêu cầu đặt ra là cần có một hệ thống phát hiện và cảnh báo sớm trước các cuộc tấn công. Hệ thống phát hiện xâm nhập được xem như là một lựa chọn tối ưu.

### 1.1.2 Một số khái niệm cơ bản

#### 1.1.2.1 Các thành phần trong hệ thống mạng

Để một hệ thống mạng hoạt động tốt nó bao gồm rất nhiều thành phần, hoạt động trên các nền tảng và mỗi trường khác nhau

* Các máy trạm
* Các máy chủ
* Các thiết bị hạ tầng mạng: Router, switch, Hub…
* Các thiết bị, hệ thống phát hiện và phòng chống xâm nhập: IDS/IPS, Snort, FireWall…

#### 1.1.2.2 Log hệ thống

Là một thành phần quan trọng của hệ thống mạng. Nó lưu lại một cách chính xác mọi hoạt động của hệ thống, tình trạng hoạt động của hệ thống, các ứng dụng, các thiết bị đã và đang hoạt động trong hệ thống.

Các loại log chính trong hệ thống

Log Access: Là log ghi lại toàn bộ thông tin truy cập của người dùng tới hệ thống,

truy cập của các ứng dụng tới cơ sở dữ liệu…

* Log Event: là log ghi lại chi tiết những sự kiện mà hệ thống đã thực hiện. Log ứng dụng, log của hệ điều hành…
* Log Device: là log ghi lại tình trạng hoạt động của các thiết bị phần cứng và phần mềm đang được sử dụng: Router, Switch, IDS, IPS…
* Log là một thành phần cực kỳ hữu hiệu cho việc giám sát cũng như khắc phục các sự cố trong hệ thống mạng. Tuy nhiên, với những hệ thống lớn, chạy nhiều ứng dụng, lượng truy cập cao thì công việc phân tích Log thực sự là một điều vô cùng khó khăn.

## 1.2 Giám sát mạng

### 1.2.1 Khái niệm

Giám sát mạng là việc giám sát, theo dõi và ghi nhận những luồng dữ liệu mạng, từ đó sử dụng làm tư liệu để phân tích mỗi khi có sự cố xảy ra. Trong các hệ thống thông tin, việc khắc phục các sự cố thường tốn một chi phí rất lớn. vì vậy, giải pháp giám sát mạng để phát hiện sớm các sự cố là một sự lựa chọn được nhiều người ưa thích nhằm mang lại hiệu quả cao với chi phí vừa phải.

Một hệ thống giám sát mạng thường có các thành phần sau:

* Máy trính sát (Sensor): là những máy trạm làm nhiệm vụ trinh sát. Thành phần này sẽ tiếp cận, tương tác với các hệ thống và dịch vụ cần giám sát để nhận biết trạng thái của những dịch vụ đó. Trong quá trình triển khai hệ thống, thành phần này sẽ được phân tán nằm rải rác nhiều nơi trên mạng để thu thập thông tin từ những nguồn khác nhau như Tường lửa, Bộ định tuyến, file nhật ký…
* Máy thu thập (Collector): Một điều đáng chú ý trong hệ thống giám sát mạng là các hệ thống, các dịch vụ cần giám sát có thể khác nhau. Điều này đồng nghĩa với việc thông tin thu được cũng có nhiều dạng khác nhau. Để có được thông tin một cách đồng nhất nhằm mục đích xử lý và thống kê, cần có một thành phần làm nhiệm vụ chuẩn hóa thông tin. Máy thu thập sẽ đọc những thông tin thu được từ các máy trinh sát và chuẩn hóa thông tin dựa trên những quy tắc chuẩn hóa biết trước. Thông tin đầu ra sẽ có định dạng giống nhau và được lưu vào cơ sở dữ liệu trung tâm.
* Cơ sở dữ liệu trung tâm: là nơi lưu trữ dữ liệu của toàn bộ hệ thống giám sát. Các dữ liệu ở đây đã được chuẩn hóa nên có thể sử dụng để tính toán các số liệu thống kê trên toàn hệ thống
* Công cụ phân tích (Analysis tool): Thành phần này sẽ đọc các dữ liệu từ cơ sở dữ liệu trung tâm và tính toán để tạo ra bản báo cáo số liệu thống kê trên toàn hệ thống.

### 1.2.2 Cách thức hoạt động và mục đích ứng dụng

Mỗi máy trinh sát sẽ có một danh sách những đối tượng mà máy trinh sát đó cần giám sát. Những đối tượng này có thể là file nhật ký hoạt động trên một máy tính, có thể là một dịch vụ trên hệ thống khác, cũng có thể là thành phần báo cáo trạng thái của Tường lửa/Bộ định tuyến… Dựa vào bản danh sách này, Máy trinh sát sẽ gửi truy vấn đến đối tượng để truy vấn thông tin. Thông tin thu thập được sẽ gửi đến Máy thu thập để chuẩn hóa trước khi lưu trữ vào cơ sở dữ liệu trung tâm.

Tùy theo thiết kế của hệ thống, nếu những thông tin mà Máy trinh sát thu thập được có định dạng giống nhau thì sẽ không cần đến thành phần Máy thu thập.

Trong một số trường hợp khác, các Máy trinh sát cũng có thể kiêm luôn vai trò của Máy thu thập thực hiện việc chuẩn hóa dữ liệu trước khi lưu trữ. Tại cơ sở dữ liệu trung tâm, mọi dữ liệu thu dược đã có định dạng rõ ràng. Bộ phân tích sẽ đọc thông tin tại đây để tính toán và đưa ra những số liệu thống kê tạo thành một bản báo cáo hoàn chỉnh. Báo cáo này sẽ được gửi tới người quản trị. Trong một số hệ thống giám sát, để nâng cao mức độ tự động hóa, Bộ phân tích có thể có thêm chức năng phát hiện dấu hiệu xác định trước để phát ra cảnh báo. Ví dụ, sau khi lấy thông tin từ file nhật ký ghi nhận lại những lần đăng nhập không thành công vào hệ thống, nếu phát hiện thấy có 3 lần đăng nhập không thành công liên tiếp trong vòng 5 phút thì Bộ phân tích phát ra cảnh báo tới người quản trị. Cảnh báo này có thể là thư điện tử, tin nhắn SMS gửi tới điện thoại di động…

#### 1.2.2.1 Thu thập dữ liệu

Việc thu thập dữ liệu ở đây chính là việc lấy các thông tin liên quan đến tình trạng hoạt động của các thiết bị trong hệ thống mạng. Tuy nhiên, trong những hệ thống mạng lớn thì các dịch vụ hay các thiết bị không đặt tại trên máy, một địa điểm mà nằm trên các máy chủ, các hệ thống con riêng biệt nhau. Các thành phần hệ thống cũng hoạt động trên những nền tảng hoàn toàn khác nhau. Mô hình Log tập trung được đưa ra để giải quyết vấn đề này. Cụ thể, là tất cả Log sẽ được chuyển về một trung tâm để phân tích và xử lý.

Với mỗi thiết bị có những đặc điểm riêng và các loại log cũng khác nhau. Như log của các thiết bị mạng như: Router, Swich. Log của các thiết bị phát hiện xâm nhập: IDS, IPS, Snort … Log của các Web Server, Application Server, Log Event, Log Registry của các Server Windows, Unix/Linux.

#### 1.2.2.2 Phân tích dữ liệu

Khi đã thu thập được những thông tin về hệ thống thì công việc tiếp theo là phân tích thông tin, cụ thể là việc thực hiện chỉ mục hóa dữ liệu, phát hiện những điều bất thường, những mối đe dọa của hệ thống. Dựa trên những thông tin về lưu lượng truy cập, trạng thái truy cập, định dạng request… Ví dụ như lưu lượng truy cập bỗng dưng tăng vọt tại một thời điểm. Đây có lẽ là nơi thích hợp để Big data lên tiếng.

* **Phương pháp đẩy**

Các sự kiện từ các thiết bị, Các máy trạm, Server sẽ được tự động chuyển về các Collector theo thời gian thực hoặc sau mỗi khoảng thời gian phụ thuộc vào việc cấu hình trên các thiết bị tương ứng. Các Collector của Log Server sẽ thực hiện việc nghe và nhận các sự kiện khi chúng xảy ra. Ví dụ như: NetFlow, Syslog-ng Message (Syslog-ng gồm 2 thành phần là Syslog-Agent và Syslog-Server), Access-list (ACL) logs …

* **Phương pháp kéo**

Các Collector thu tập các sự kiện được phát sinh và lưu trữ trên chính các thiết bị và sẽ được lấy về bởi các bộ Collector. Hai giao thức phổ biến để thu thập được các sự kiện là Security Device Event Exchange (SDEE – Gồm các thiết bị nằm trong hệ thống các thiết bị phát hiện xâm nhập được phát triển bởi ICSA) và SNMP (Simple Network Management Protocol – Giao thức hỗ trợ việc quản lý các thiết bị từ xa).

#### 1.2.2.3 Phát hiện và phản ứng

Phát hiện và phản ứng là hai thành phần quan trọng trong các yếu tố của tiến trình. Sau khi bức tường phòng ngự cuối cùng bị phá vỡ, các tổ chức cần nhanh trong phát hiện ra cách thức xâm nhập của kẻ tấn công và chúng sẽ làm gì tiếp theo. Quá trình này được gọi là phạm vi ứng phó sự cố. Bởi xâm nhập không có nghĩa là có quyền root. Một kẻ xâm nhập có thể leo thang đặc quyền của mình để thực hiện những âm mưu sau đó.

Bất kỳ ai khi thực hiện công việc ứng phó sự cố thường xuyên sẽ hiểu được công việc nào nên làm trước vì giám đốc, CEO, hay những nhân viên cấp cao không quan tâm đến việc kẻ xâm nhập làm thế nào mà chỉ quan tâm đến những vấn đề sau:

* Những kẻ tấn công đã làm gì
* Khi nào
* Chúng ta ngăn chặn được chưa
* Đã có thiệt hại như thế nào

Mặc dù các nhà lãnh đạo không quan tâm đến cách thức xâm nhập của kẻ tấn công, nhưng đó luôn là công việc hàng đầu để có thể phản ứng hiệu quả với các cuộc xâm nhập. Chỉ có cách xác định phương thức xâm nhập của kẻ tấn công và ngăn chặn chúng thì việc phục hồi mới có thể diễn ra trọn vẹn được.

#### 1.2.2.4 Cảnh báo

Sau khi đã thực hiện việc phân tích dữ liệu từ các thông tin thu thập được việc tiếp theo là thực hiện việc đánh giá, đưa thông tin cảnh báo tới người quản trị và thực hiện những công tác nhằm chống lại những mỗi đe dọa, khắc phục các sự cố có thể sảy ra.

Cảnh báo có thể thông qua email, SMS, hoặc thực thi các mã script nhằm hạn chế hậu quả của sự cố. Khi xảy ra sự cố, hệ thống sẽ tự động gửi email, sms cho người quản trị và cũng có thể chạy script để thêm một địa chỉ IP có biểu hiện tấn công và danh sách đen của Firewall. Việc này đòi hỏi người lập trình phải có hiểu biết sâu và kinh nghiệm về hệ thống.

## Hệ thống phát hiện xâm nhập

### 1.3.1 Giới thiệu chung

Nếu như hiểu Firewall là một hệ thống “khóa” chốt chặn ở cửa ngõ mạng, thì hệ thống IDS có thể được coi như các “cảm ứng giám sát” được đặt khắp nơi trong mạng để cảnh báo về các cuộc tấn công đã “qua mặt” được Firewall hoặc xuất phát từ bên trong mạng. Một IDS có nhiệm vụ phân tích các gói tin mà Firewall cho phép đi qua, tìm kiếm các dấu hiệu tấn công từ các dấu hiệu đã biết hoặc thông qua việc phân tích các sự kiện bất thường, từ đó ngăn chặn các cuộc tấn công trước khi nó có thể gây ra những hậu quả xấu với tổ chức.

Hệ thống IDS hoạt động dựa trên 3 thành phần chính là Cảm ứng (Sensor), Giao diện (Console) và Bộ phân tích (Engine). Hoạt động của một hệ thống IDS được chia làm 5 giai đoạn chính là: Giám sát, Phân tích, Liên lạc, Cảnh báo và Phản ứng.

* Có hai cách tiếp cận cơ bản đối với việc phát hiện và phòng chống xâm nhập là:
* **Xâm nhập**: Để hiểu được như thế nào là “ phát hiện xâm nhập” trước hết chúng ta cần xem xét “xâm nhập” là gì. “Xâm nhập” trong thuật ngữ mạng máy tính, là sự truy nhập vào hệ thống một cách trái phép. Một ví dụ đơn giản là một người dùng truy cập vào hệ thống và thao tác với quyền quản trị khi không được phép.
* **Phát hiện xâm nhập**: là việc sử dụng một tập hợp những ký thuật và phương thức để phát hiện những dấu hiệu bị xâm nhập cả ở cấp độ mạng lẫn máy tính.

### 1.3.2 Nguyên lý hoạt động

Nguyên lý hoạt động của một hệ thống phòng chống xâm nhập được chia làm 5 giai đoạn chính: Giám sát mạng, phân tích lưu thông, Liên lạc giữa các thành phần, Cảnh báo về các hành vi xâm nhập và cuối cùng có thể tiến hành phản ứng lại tùy theo chức năng của từng IDS.

#### 1.3.2.1 Giám sát mạng (monotoring)

Giám sát mạng là quá trình thu thập thông tin về lưu thông trên mạng. Việc này thông thường được thực hiện bằng các Sensor. Yêu cầu đòi hỏi đối với giai đoạn này là có được thông tin đầy đủ và toàn vẹn về tình hình mạng. Đây cũng là một vấn đề khó khăn, bởi vì nếu theo dõi toàn bộ thông tin thì sẽ tốn khá nhiều tài nguyên, đồng thời gây ra nguy cơ tắc nghẽn mạng. Nên cần thiết phải cân nhắc để không làm ảnh hưởng đến toàn bộ hệ thống. Có thể sử dụng phương án là thu thập liên tục trong khoảng thời gian dài hoặc thu thập theo từng chu kì. Tuy nhiên khi đó những hành vi bắt được chỉ là những hành vi trong khoảng thời gian giám sát. Hoặc có thể theo vết những lưu thông TCP theo gói hoặc theo liên kết. Bằng cách này sẽ thấy được những dòng dữ liệu vào ra được phép. Nhưng nếu chỉ theo dõi những liên kết thành công sẽ có thể bỏ qua những thông tin có giá trị về những liên kết không thành công mà đây lại thường là những phần quan tâm trong một hệ thống IDS, ví dụ như hành động quét cổng.

***1.3.2.2 Phân tích lưu thông (Analyzing)***

Khi đã thu thập được những thông tin cần thiết từ những điểm trên mạng. IDS tiến hành phân tích những dữ liệu thu thập được. Mỗi hệ thống cần có một sự phân tích khác nhau vì không phải môi trường nào cũng giống nhau. Thông thường ở giai đoạn này, hệ thống IDS sẽ dò tìm trong dòng traffic mang những dấu hiệu đáng nghi ngờ dựa trên kỹ thuật đối sánh mẫu hoặc phân tích hành vi bất thường.

#### 1.3.2.3 Liên lạc

Giai đoạn này giữ một vai trò quan trọng trong hệ thống IDS. Việc liên lạc diễn ra khi Sensor phát hiện ra dấu hiệu tấn công hoặc Bộ xử lý thực hiện thay đổi cấu hình, điều khiển Sensor. Thông thường các hệ thống IDS sử dụng các bộ giao thức đặc biệt để trao đổi thông tin giữa các thành phần. Các giao thức này phải đảm bảo tính tin cậy, bí mật và chịu lỗi tốt, ví dụ: SSH, HTTPS, SNMPv3…Chẳng hạn hệ thống IDS của hãng Cisco thường sử dụng giao thức PostOffice định nghĩa một tập các thông điệp để giao tiếp giữa các thành phần.

#### 1.3.2.4 Cảnh báo

Sau khi đã phân tích xong dữ liệu, hệ thống IDS cần phải đưa ra được những cảnh báo. Ví dụ như:

* Cảnh báo địa chỉ không hợp lệ.
* Cảnh báo khi máy cố gắng kết nối đến những máy nằm trong danh sách cần theo dõi ở trong hay ngoài mạng.

#### 1.3.2.5 Phản ứng (Response)

Trong một số hệ thống IDS tiên tiến hiện nay, sau khi các giai đoạn trên phát hiện được dấu hiệu tấn công, hệ thống không những cảnh báo cho người quản trị mà còn đưa ra các hành vi phòng vệ ngăn chặn hành vi tấn công đó. Điều này giúp tăng cường khả năng tự vệ của Mạng, vì nếu chỉ cần cảnh báo cho người quản trị thì đôi khi cuộc tấn công sẽ tiếp tục xảy ra gây ra các tác hại xấu. Một hệ thống IDS có thể phản ứng lại trước những tấn công phải được cấu hình để có quyền can thiệp vào hoạt động của Firewall, Switch và Router. Các hành động mà IDS có thể đưa ra như:

* Ngắt dịch vụ.
* Gián đoạn phiên.
* Cấm địa chỉ IP tấn công.
* Tạo log.

## 1.4 Một số công cụ hỗ trợ

### 1.4.1 SPlunk

Splunk là một phần mềm giám sát mạng dựa trên sức mạnh của việc phân tích Log. Splunk thực hiện các công việc tìm kiếm, giám sát và phân tích các dữ liệu lớn được sinh ra từ các ứng dụng, các hệ thống và các thiết bị hạ tầng mạng. Nó có thể thao tác tốt với nhiều loại dịnh dạng dữ liệu khác nhau (Syslog, csv, apache-log, access\_combined…). Splunk được xây dựng dựa trên nền tảng Lucene and MongoDB với một giao diện web hết sức trực quan.

**CHÍNH SÁCH BẢN QUYỀN:**

Splunk cung cấp 2 bộ miễn phí và trả phí cho người dùng

* Sản phẩm trả phí: Có tất cả các chức năng của Splunk, không hạn chế kích thước dữ liệu.
* Sản phẩm miễn phí: Hạn chế một số chức năng, hạn chế khối lượng dữ liệu mỗi ngày là 500MB. Bao gồm các chức năng: Đánh chỉ mục dữ liệu, tìm kiếm trong thời gian thực, thống kế và kết xuất báo cáo.

**TÍNH NĂNG:**

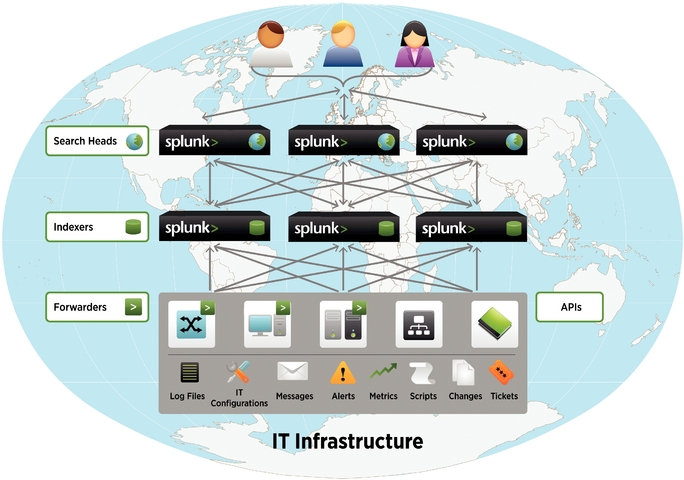
* Định dạng Log: Hỗ trợ hầu như tất cả các loại log của hệ thống, thiết bị hạ tầng mạng, phần mềm, Firewall, IDS/IPS, Log Event, Register của các máy trạm ….
* Các hình thức thu thập dữ liệu: Splunk có thể thực hiện việc thu thập log từ rất nhiều nguồn khác nhau. Từ một file hoặc thư mục (kể cả file nén) trên server, qua các kết nối UDP, TCP từ các Splunk Server khác trong mô hình Splunk phân tán, từ các Event Logs, Registry của Windows …Splunk kết hợp rất tốt với các công cụ thu thập log khác.
* Cập nhật dữ liệu: Splunk cập nhật dữ liệu liên tục khi có thay đổi trong thời gian thực. Giúp cho việc phát hiện và cảnh báo trong thời gian thực.
* Đánh chỉ mục dữ liệu: Splunk có thể đánh chỉ mục dữ liệu với một khối lượng dữ liệu rất lớn trong một khoảng thời gian ngắn. Giúp việc tìm kiếm diễn ra nhanh chóng và thuận tiện.
* Tìm kiếm thông tin: Splunk làm việc rất tốt với dữ liệu lớn và cập nhật liên tục. Nó cung cấp cơ chế tìm kiếm với một “Splunk Language” cực kỳ thông minh bao gồm các từ khóa, các hàm và cấu trúc tìm kiếm giúp người sử dụng có thể truy xuất mọi thứ, theo rất nhiều tiêu chí từ tập dữ liệu rất lớn. Những nhà quản trị mạng cao cấp và chuyên nghiệp thường gọi Splunk với cái tên “Splunk toàn năng” hay “Splunk as Google for Log files” để nói lên sức mạnh của Splunk.
* Giám sát và cảnh báo: Splunk cung cấp cho người dùng một cơ chế cảnh báo dựa trên việc tìm kiếm các thông tin do chính người sử dụng đặt ra. Khi có vấn đề liên quan tới hệ thống phù hợp với các tiêu chí mà người dùng đã đặt ra thì hệ thống sẽ cảnh báo ngay tới người dùng (cảnh bảo trực tiếp qua giao diện, giử Email).
* Khắc phục sự cố: Splunk còn cung câp một cơ chế tự động khắc phục với các vấn đề xảy ra bằng việc tự động chạy các file Script mà người dùng tự tạo (Ví dụ như: Chặn IP, đòng Port …) khi có các cảnh báo xảy ra.
* Hiển thị thông tin: Splunk cung cấp một cơ chế hiển thị rất trực quan giúp người sử dụng có thể dễ dàng hình dung về tình trạng của hệ thống, đưa ra các đánh giá về hệ thống. Splunk còn từ động kết xuất ra các báo cáo với nhiều loại định dạng một cách rất chuyên nghiệp.
* Phát triển: Cũng cung cấp các API hỗ trợ việc tạo các ứng dụng trên Splunk của người dùng. Một số bộ API điển hình như Splunk SDK (Cung cấp các SDK trên nền tảng Python, Java, JS, PHP), Shep (Splunk Hadoop Intergration – Đây là sự kết hợp giữa Splunk và Hadoop), Shuttl (Là một sản phẩm hỗ trợ việc sao lưu dữ liệu trong Splunk), Splunkgit (Giúp bạn hình dung dữ liệu tốt hơn), Splunk power shell resource Kit (Bộ công cụ hỗ trợ việc mở rộng và quản lý hệ thống).

**LƯU Ý:**

* Splunk mạnh về khả năng phân tích và cảnh báo tuy nhiên nó lại không mạnh và không đảm bảo về việc thu thập và truyền tải log. Cụ thể là nó chưa có cơ chế bảo mật trên đường truyền, không phù hợp với những hệ thống đòi hỏi bảo mật cao.
* Để phát huy hết được sức mạnh của Splunk cần có thời gian tìm hiểu và sử dụng. Nó chưa có cơ chế giúp tự động phát hiện ra các tấn công hay các vấn đề từ bên ngoài. Nhưng điều này phụ thuộc vào kinh nghiệm sử dụng và vốn hiểu biết của người quản trị.
* Đề triển khai được một hệ thống sử dụng Splunk hiệu quả chúng ta cũng cần có một hệ thống riêng, đây cũng là một trở ngại không nhỏ với các hệ thống có quy mô trung bình và nhỏ.

**TRIỂN KHAI:**

* Với một hệ thống lớn để triển khai được Splunk chúng ta cần phải có một Server riêng để tập trung Log. Tuy nhiên Splunk làm không tốt việc tập trung Log từ các Server hay thiết bị khác. Vì thế chúng ta cần sử dụng một số công cụ khác để thực hiện việc tập trung Log và về Splunk Server. Cụ thể là ta có thể kết hợp với syslog, Snare (for Windows), sử dụng qua Heroku ….
* Triển khai hệ thống phân tán: Splunk hỗ trợ người dùng thiết lập một hệ thống phân tán khi lượng
* Dữ liệu là quá lớn vượt qua khả năng lưu trữ và xử lý của một máy.



##### Hình 1.1: Sơ đồ Splunk

### 1.4.2 Loggly

Là một dịch vụ quản lý Log trực tuyến dựa trên mô hình điện toán đám mây. Nó được phát triển bời chính các nhân viên đã từng làm việc với Splunk. Là một dịch vụ giúp người dùng dễ dàng để triển khai một hệ thống giám sát an ninh mạng. Cụ thể là mọi dữ liệu về Log sẽ được chuyển đến Loggly Server quan các client như snare hoặc một công cụ do Loggly cung cấp. Dữ liệu được chuyển về server của Loggly sẽ được xử lý, phân tích và đưa ra các cảnh báo tới các nhà qsssuản trị.

**CHÍNH SÁCH BẢN QUYỀN:**

* Loggly là một dịch vụ trả phí. Người sử dụng cần phải trả phí để sử dụng (Có miễn phí dùng thử 30 ngày).

**TÍNH NĂNG:**

Loggy cung cấp các chức năng gần giống với Splunk với việc tìm kiếm thông tin trên Log, hiển thị dưới dạng biểu đồ một cách trực quan, cảnh báo tới người sử dụng khi hệ thống có vấn đề.

* Hỗ trợ định dạng: Khác với Splunk, Loggly chỉ hoạt động tốt với Syslog. Muốn xử dụng các loại Log khác chúng ta cần phải convert chúng về Syslog để sử dụng.
* Thu thập thông tin: Loggly xây dựng trên mô hình SaaS (Software as a Service). Nên để thu thập Log chúng ta chỉ cần cấu hình hệ thống với một client được cung cấp để gửi Syslog tới và xử lý tại Server của Splunk.
* Đánh chỉ mục dữ liệu: Loggly có thể thực hiện việc đánh chỉ mục dữ liệu một cách nhanh chóng và tối ưu.
* Tìm kiếm thông tin: Loggly cung cấp một cơ chế tìm kiếm thông minh và nhanh chóng có thể xác định sự cố và có thể thiết lập các cảnh báo. Tuy không đầy đủ các tính năng như tìm kiếm với Splunk nhưng cơ chế Loggly cũng giúp người quản trị có thể tìm kiếm và thiết lập các vấn đề mà người quản trị mong muốn.
* Cảnh báo và giám sát: Loggly cung cấp một giao diện hiển thị tình trạng, kết quả tìm kiếm bằng biểu đồ rất trực quan giúp người quản trị dễ dàng hình dung hệ thống. Cũng giống như Splunk cơ chế cảnh báo của Loggly cũng hoạt động dựa trên việc thiết lập tìm kiếm thông tin trên Log, cảnh báo tới người dùng khi có một (nhiều) các bản ghi phù hợp với cú pháp tìm kiếm theo những cấu hình của người sử dụng.

**NHƯỢC ĐIỂM:**

* Do hoạt động trên mô hình Service nên hiệu suất của Loggly không thực sự thích hợp với những hệ thống ở mức độ doanh nghiệp lớn.
* Việc truyền tải một lượng dữ liệu lớn cũng làm giảm hiệu suất trong việc phân tích và cảnh báo. Đặc biệt là với những mạng có tốc độ truy cập thấp.
* Loggly không hỗ trợ các tính năng tự động khắc phục lỗi. (tự động chạy script đã cấu hình sẵn để khắc phục lỗi).

**TRIỂN KHAI:**

* Triển khai hệ thống với Loggly rất đơn giản bằng việc chúng ta chỉ cần đăng ký một tài khoản, thực hiện cấu hình để gửi Syslog tới Loggly Server là chúng ta đã có thể sử dụng đầy đủ các tính năng của Loggly.
* Rất thích hợp với những hệ thống ở quy mô nhỏ và vừa.

### 1.4.3 Syslog-Ng

Syslog-ng là một công cụ thu thập Log rất hiệu quả và linh hoạt là sự lựa chọn của rất nhiều nhà quản trị mạng trong việc xây dựng một hệ thống log tập trung. Syslog-ng được xây dựng dựa trên chuẩn syslog trên nền tảng Unix và các hệ điều hành tương tự. Gồm xây dựng với 2 thành phần Syslog-ng client và Syslog-ng Server. Các Client thực hiện việc thu thập log quan trọng gửi tới máy chủ tập trung và lưu trữ.

**CHÍNH SÁCH BẢN QUYỀN:** Syslog-ng là một phần mềm mã nguồn mở được phát triển trên nền tảng của Syslogd. Hiện nay nó có hai phiên bản và được phát triên bởi Balabit IT Security Ltd

* Phiên bản miễn phí: Syslog-ng Open Source Edition (OSE).
* Phiên bản trả phí độc quyền: Premium Edition (PE).

**TÍNH NĂNG:**

* Thu thập dữ liệu: Syslog-client thực hiện việc tập trung log từ các host và gửi về Syslog server. Syslog-ng thực hiện việc thu thập log từ các server khác nhau dựa trên giao thức TCP, đảm bảo không bị mất mát thông tin trên đường truyền Synlog-ng cung cấp một số cơ chế truy xuất log an toàn dựa trên SSL/TLS
* Định dạng log: Theo mặc định Syslog-ng chỉ hỗ trợ chuẩn, Syslog trong Unix. Theo mặc đinh Windows không hỗ trợ Syslog.Tuy nhiên chúng ta có thể sử dụng một số biện pháo để chuyển các loại log về dạng Syslog. Syslog-ng cũng hoạt động rất tốt trên những môi trường (hệ điều hành, phần cứng) khác nhau: Linux, BSD, Sun Solaris, HP-UX, AIX và Unix khác.
* Lưu trữ: Với Syslog-ng, ta có thể lưu trữ dữ liệu vào cơ sở dữ liệu cho phép tìm kiếm và truy vấn dễ dàng. Syslog-ng hỗ trợ các hệ CSDL: MSSQL, MYSQL, Oracle và PostgreSQL.
* Lọc và phân loại: Syslog-ng cung cấp cơ chế lọc nhằm phân loại các Log message và cũng hạn chế lượng dữ liệu đổ về server log từ các client. Cơ chế lọc của Syslog-ng dựa trên các thông số khác nhau như source host, ứng dụng, sự ưu tiên trong Log message.
* Cơ chế thu thập Log: Syslog-ng client được đặt trên các các client sẽ thực hiệc việc thu tập các loại Log trên client đó. Sau đó dữ liệu sẽ được đi qua bộ phận lọc của syslog-ng (gồm những luật đã được cấu hình trước). Sau đó mới được gửi đến các Server log hoặc chuyển đến một Relay server rồi mới chuyển tới Log Server.

**NHƯỢC ĐIỂM:**

* Syslog-ng không phải là 1 phần mềm phân tích cho nên syslog-ng chỉ có thể lọc những log message phù hợp với 1 số tiêu chí định trước. Syslog-ng không thể làm tốt nhiệm vụ phân tích và cảnh báo các nguy cơ đến người quản trị.

**TRIỂN KHAI:**

* Để triển khai một hệ thống syslog-ng ta cần có 2 thành phần là một server được cài đặt syslog-ng server và các client được cài đặt trên các client để thu thập log. Một điểm đáng chú ý là Syslog không hỗ trợ windows.

### 1.4.4 Logzilla (Php Syslog-Ng)

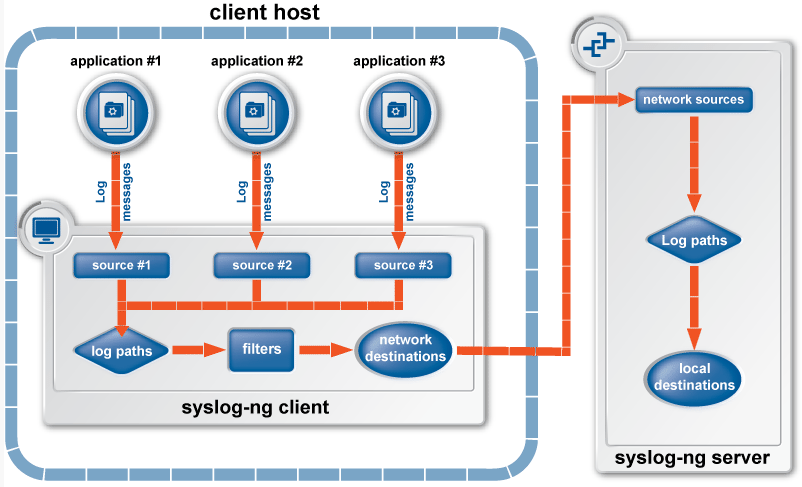
Là phần mềm mã nguồn mở hỗ trợ việc quản lý Log tập trung được phát triển dự trên PHP-Syslog-ng. Logzilla có thể quản lý với hàng triệu thông điệp Log, hàng ngàn thiết bị cùng lúc. Được xây dựng trên nền web với một giao diện quản lý trực quan và thuận tiện cho người dùng. Là sự lựa chọn của nhiều nhà quản lý và giám sát anh ninh mạng.

**TÍNH NĂNG:**

* Thu thập dữ liệu: LogZilla mặc định không hỗ trợ việc thu thập Log từ các thiết bị hay các Server khác nó tập trung vào việc thực hiện trên Log đã có dựa trên việc thu thập Log của Syslog-ng.
* Hỗ trợ định dạng Log: Theo mặc định LogZilla chỉ hỗ trợ Syslog chuẩn Syslog giống như Syslog-ng. Tuy nhiên, nó có thể hỗ trợ quản lý các sự kiện trong Windows.
* Tìm kiếm thông tin: LogZilla cung cấp một giao diện tìm kiêm theo từ khóa và theo một số thuộc tính khá trực quan và thông minh. Tuy không được đánh giá cao như SPLUNK nhưng LogZilla cũng được các nhà quản trị mạng đánh giá khá cao về chức năng tìm kiếm các thông tin trong Log.
* Cảnh báo và giám sát mạng: LogZilla hỗ trợ việc phát hiện các sự kiện một cách nhanh chóng trong thời gian thực. Có thể nhanh chóng phát hiện các điểm suy thoái của các thiết bị và máy chủ. LogZilla cũng hỗ trợ việc cảnh báo qua Email.
* Trích xuất thông tin: LogZilla cũng hỗ trợ việc tạo kết xuất ra các báo cáo theo các định dạng: Excel và CSV. LogZilla còn hỗ trợ việc hiển thị dưới một số dạng biểu đồ giúp người quản trị dễ dàng hình dung hệ thống một cách trực quan.

**TRIỂN KHAI:**

* LogZilla nhìn chung chỉ thực hiện việc tìm kiếm và quản lý các thông tin đã có trên Log một cách nhanh chóng. Ta cần thiết phải kết hợp với các công cụ khác để tập trung và xử lý Log hiệu quả hơn (Syslog-ng).
* Để triển khai LogZilla ta cần kết hợp với một hệ thống thu thập Log khác thực hiện công việc thu thập thông tin từ các máy chủ và thiết bị khác trên mạng.
* Triển khai một hệ thống LogZilla hoạt động tương đối dễ dàng ta chỉ cần thực hiện cấu hình trên Syslog-ng Server để đọc các dữ liệu mà Syslog-ng đã lưu trữ.



##### Hình 1.2. Mô hình Logzilla

### 1.4.5 HP ArcSight Logger

HP ArcSight Logger là một sản phẩm trong bộ sản phẩm ArcSight của Hp. Nó cung cấp một giải pháp hiệu quả về trong việc quản lý log. Nó có khả năng thu thập, phân tích và lưu trữ với một khối lượng Log lớn với nhiều loại định dạng khác nhau. Nó hỗ trợ việc triển khai hệ thống dưới nhiều hình thức như thiết bị, phần mềm, máy ảo hoặc các dịch vụ đám mây.

**CHÍNH SÁCH BẢN QUYỀN:**

HP ArcSight Logger cung cấp 2 phiên bản dùng thử và trả phí.

* Với phiên bản trả phí, có đầy đủ các tính năng của HP ArcSight Logger và không giới hạn về khối lượng dữ liệu.
* Phiên bản dùng thử được hỗ trợ xử lý với dữ liệu 750 MB/ngày. Hạn chế một số chức năng: Hỗ trợ triển khai hệ thống phân tán và Support từ các chuyên gia của hệ thống của HP.

**TÍNH NĂNG:**

* Hỗ trợ nhiều loại định dạng Log: Syslog, Eventlog, Device Log …
* Phân tích toàn diện dữ liệu.
* Cảnh báo và giám sát hệ thống trong thời gian thực.
* Đánh chỉ mục dữ liệu, tìm kiếm và kết xuất báo cáo.
* Đi sâu vào việc phân tích ngữ cảnh từ các thông tin nhận được.

**TRIỂN KHAI:**

* Ta có thể dễ dàng triển khai HP ArcSight Logger như một thiết bị, một phần mềm hay một dịch vụ đám mây.
* HP ArcSight Logger hỗ trợ việc triển khai hệ thống phân tán một cách dễ dàng.

### 1.4.6 Nagios

Nagios là một hệ thống dùng để giám sát một hệ thống mạng. Nagios thực hiện việc theo dõi và đưa ra các cảnh báo về trạng thái các host và các dịch vụ. Nó được xây dựng trên nền Linux và đã hỗ trợ hầu hết các hệ điều hành tương tự Linux. Một điểm khác so với các công cụ khác là Nagios giám sát dựa tình trạng hoạt động của các máy trạm và dịch vụ. Nó sử dụng các Plug-in được cài đặt trên các máy trạm, thực hiện việc kiểm tra các máy trạm và dịch vụ theo định kỳ và gửi thông tin trạng thái về Nagios Server sau đó thông tin sẽ được đưa lên với một giao diện Web (Sử dụng Nagvis) và có thể gửi thông tin về trạng thái tới nhà quản trị qua email, SMS… khi có sự cố xảy ra. Việc theo dõi có thể được cấu hình một cách chủ động hoặc bị động dựa trên mục đích sử dụng của người quản trị.

**CHÍNH SÁCH BẢN QUYỀN:**

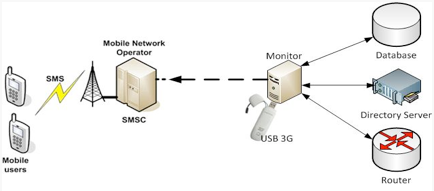
Cung cấp 2 phiên bản miễn phí và trả phí hỗ trợ các hệ thống nhỏ và cả các hệ thống doanh nghiệp.

**TÍNH NĂNG:**

* Giám sát các dịch vụ mạng (SNMP, POP3, HTTP, NNTP, PING…) và các tài nguyên của các máy trạm (processor load, disk usage…).
* Dễ dàng phát triển các plug-in riêng. Chophép người sử dụng dễ dàng phát triển các dịch vụ giám sát nhu cầu sử dụng bằng việc sử dụng các ngôn ngữ shell script, C ++, Perl, Ruby, Python, PHP, C# ….).
* Việc giám sát các dịch vụ là song song.
* Có khả năng phát hiện và phân biệt được host nào là down và host nào là unreachable.
* Thông tin cảnh báo (khi host và các dịch vụ xảy ra xự cố) bằngemail, SMS sử dụng 3G, …
* Sử dụng giao diện Web để theo dõi trạng thái của mạng, xem lịch sử các cảnh báo và các sự cố xảy ra.

**TRIỂN KHAI:**

* Nagios chỉ hoạt động trên các máy chủ chạy hệ điều hành họ Unix/Linux.
* Để triển khai một hệ thống Nagios ta cần một Server chạy Nagios được cấu hình để có thể thực hiện việc nhận thông báo trạng thái từ các Plug-in đã được cài đặt trên các máy trạm.
* Nagios có thể hoạt động tốt với Splunk. Hỗ trợ việc tìm kiếm và cảnh báo hiệu quả hơn.
* Nagios cũng hỗ trợ việc xây dựng một hệ thống phân tán giúp cân bằng tải và hoạt động ổn định hơn trong các hệ thống lớn.
* Một số Plug-in điển hình: NRPE(giám sát thông tin từ xa), NSCA(hỗ trợ việc giám sát chủ động), NDOUtils (Hỗ trợ việc lưu trữ dữ liệu), PNP4Nagios (Hỗ trợ việc phân tích dữ liệu), Nagvis(Hỗ trợ việc hiển thị và biểu diễn trạng thái) …



##### Hình 1.3. Mô hình Nagios

### 1.4.7 Một số sản phẩm khác

Trên đây là một trong những công cụ chính đảm báo việc quản trị mạng ngoài ra còn một số công cụ cũng có tính năng tương tự:

* **Graylog2:** là một hệ thống quản lý Log mã nguồn mở, nó được xây dựng bằng Rubu trên Rails và MongoDB. Dựa trên một định dạng Log riêng dựa trên JSON-based được gọi là GELF (Graylog Extended Log Format).
* **CSMARS (Cisco Security Monitoring, Analysis, and Response System):** Là một thiết bị mạng nằm trong hệ thống các thiết bị quản lý bảo mật của Cisco được triển khai trên các hệ thống mạng nhằm nhận biết và giảm thiểu các mỗi đe dọa đến hệ thống. Thiết bị được tích hợp mạng lưới tình báo hiện đại, thực hiện khi có dị thường trong mạng hoặc sự kiện vi phạm an ninh hệ thống. Nó giúp người quản trị hình dung hệ thống, giám sát hệ thống, … Dễ dàng triển khai với chi phí thấp

### 1.4.8 KẾT LUẬN

Hiện nay, có rất nhiều công cụ hỗ trợ việc giám sát an ninh mạng. Tuy nhiên, mỗi công cụ, sản phẩm có những điểm mạnh, điểm yếu riêng đòi hỏi người quản trị cần có kinh nghiệm trong việc sử dụng sản phẩm. Lựa chọn công cụ, sản phẩm dựa trên các yêu cầu về quy mô của hệ thống, mức độ an toàn và nhiệm vụ của hệ thống cũng như kinh phí trong việc phát triển hệ thống.

Trong một số trường hợp chỉ sử dụng một sản phẩm là chưa tối ưu, chúng ta cần sự kết hợp của các công cụ, sản phẩm với nhau để thu được kết quả tốt nhất.

# CHƯƠNG 2: TÌM HIỂU VỀ GRAYLOG2

## 2.1 Một kiến thức về Log trong Linux

### 2.1.1 Khái niệm về log

Trước hết Bạn là người quản trị mạng của một doanh nghiệp, trong hệ thống mạng của bạn có một máy chủ chứa dữ liệu rất quan trọng. Một buổi tối bạn để máy chủ đó chạy suốt đêm nhưng khi về đến nhà bạn truy cập vào máy chủ thì báo lỗi từ chối dịch vụ do không thể kết nối, buổi sáng bạn vội vã đến xem xét tình hình thì thấy một số dữ liệu đã bị mất và vấn đề lúc này là xem ai đã gây ra vấn đề trên. Vậy phải làm thế nào để điều tra xử lý, hay đơn giản là tìm nguyên nhân để khắc phục hậu quả vừa xảy ra. Log sẽ giúp bạn làm việc này.

Vậy nên tác dụng của log là:

* Log ghi lại liên tục các thông báo về hoạt động của cả hệ thống hoặc của các dịch vụ được triển khai trên hệ thống và file tương ứng. Log file thường là các file văn bản thông thường dưới dạng “clear text” tức là bạn có thể dễ dàng đọc được nó, vì thế có thể sử dụng các trình soạn thảo văn bản (vi, vim, nano...) hoặc các trình xem văn bản thông thường (cat, tailf, head...) là có thể xem được file log.
* Các file log có thể nói cho bạn bất cứ thứ gì bạn cần biết, để giải quyết các rắc rối mà bạn gặp phải miễn là bạn biết ứng dụng nào, tiến trình nào được ghi vào log nào cụ thể.
* Trong hầu hết hệ thống Linux thì **/var/log** là nơi lưu lại tất cả các log.

Như đã nói ở trên, 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.

**Tóm lại:**

* Log = Thời điểm + Dữ liệu.
* Log ghi lại những hoạt động của hệ thống.



##### Hình 2.1: Vòng đời của chung của Log

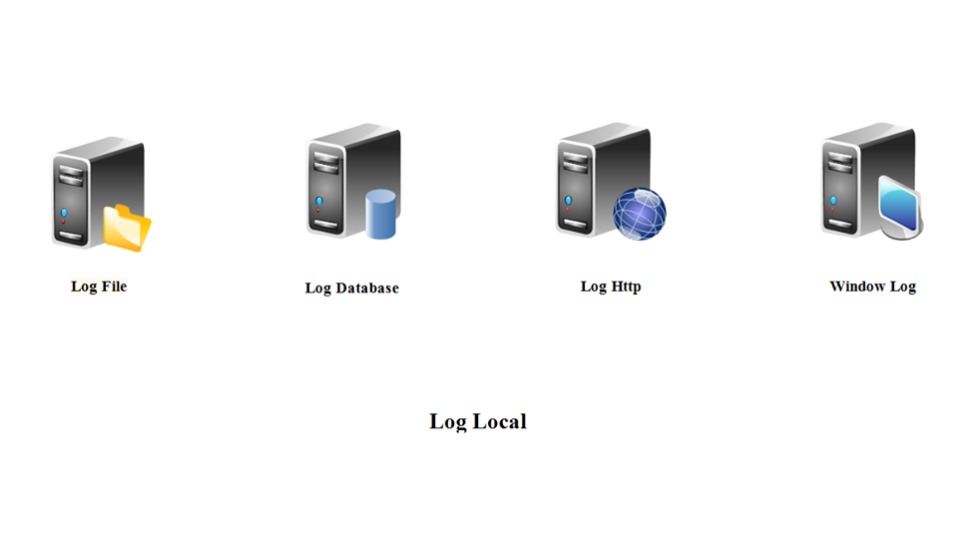
Một vòng đời của Log bao gồm 5 bước chính được minh họa trong hình 3.1 cụ thể là:

Đầu tiên log sẽ được ghi lại tại chính máy local sau đó nó sẽ được vận chuyển sang máy chủ quản lý log. Người quản trị mạng sẽ từ những bản ghi đó mà tiến hành phân tích, từ đó có thể giám sát được hoạt động của các máy client. Qua bước phân tích này mà người quản trị có thể phát hiện các hoạt động, hành vi xâm nhập không được phép. Sau khi phân tích, dữ liệu log sẽ được lưu trữ lại để sử dụng lại nếu cần. Bước cuối cùng là xóa, thường những tập tin log không cần thiết có thể được xóa bởi người quản trị nhằm giảm bớt lượng thông tin log không cần thiết.

### 2.1.2 Mô hình chứa Log

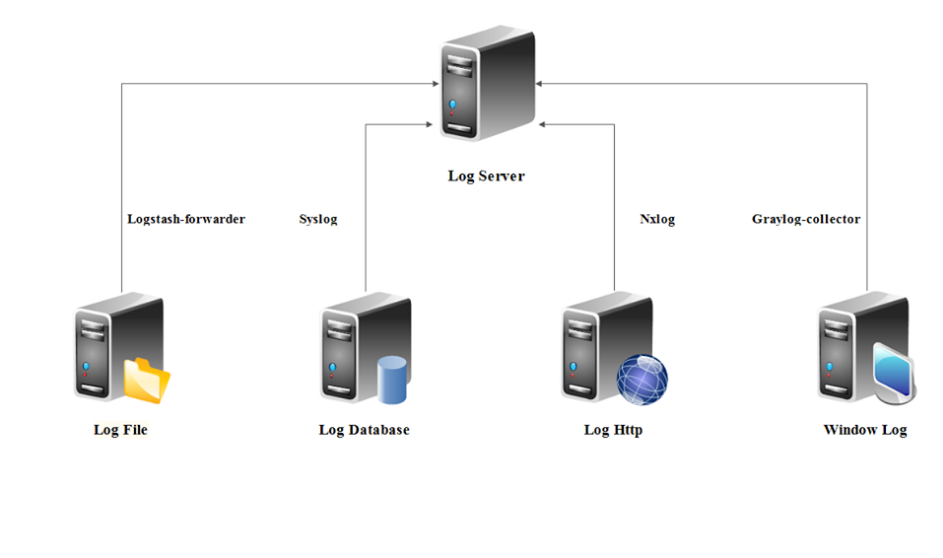
**Mô hình Log:**

* **Log local:**
* Chỉ lưu lại bản thân Server
* Dùng command find, tail… để xem log.



##### Hình 2.2. Mô hình Log local

* **Log tập trung:**
* Log máy local đẩy về máy Log Server.
* Mỗi ứng dụng có giao thức đẩy Log khác nhau.



##### Hình 2.3. Mô hình Log tập trung

### 2.1.3 Một số công dụng của Log

* Phân tích nguyên nhân khi có sự cố xảy ra.
* Giúp cho việc khắc phục sự cố nhanh hơn khi hệ thống gặp vấn đề.
* Giúp cho việc phát hiện, dự đoán một vấn đề có thể xảy ra đối với hệ thống.

### 2.1.4 Syslog và Rsyslog

#### 2.1.4.1 Giới thiệu về Syslog

Syslog là một giao thức client/server là giao thức dùng để chuyển log và thông điệp đến máy nhận log. Máy nhận log thường được gọi là syslogd, syslog daemon hoặc syslog server. Syslog có thể gửi qua UDP hoặc TCP. Các dữ liệu được gửi dạng cleartext. Syslog dùng cổng 514.

Syslog được phát triển năm 1980 bởi Eric Allman, nó là một phần của dự án Sendmail, và ban đầu chỉ được sử dụng duy nhất cho Sendmail. Nó đã thể hiện giá trị của mình và các ứng dụng khác cũng bắt đầu sử dụng nó. Syslog hiện nay trở thành giải pháp khai thác log tiêu chuẩn trên Unix-Linux cũng như trên hàng loạt các hệ điều hành khác và thường được tìm thấy trong các thiết bị mạng như router Trong năm 2009, Internet Engineering Task Forec (IETF) đưa ra chuẩn syslog trong RFC 5424

Syslog ban đầu sử dụng UDP, điều này là không đảm bảo cho việc truyền tin. Tuy nhiên sau đó IETF đã ban hành RFC 3195 Reliable Delivery for syslog. Đảm bảo tin cậy cho syslog và RFC 6587 Transmission of Syslog Messages over TCP.

Truyền tải thông báo syslog qua TCP. Điều này có nghĩa là ngoài UDP thì giờ đây syslog cũng đã sử dụng TCP để đảm bảo an toàn cho quá trình truyền tin.

Trong chuẩn syslog, mỗi thông báo đều được dán nhãn và được gán các mức độ nghiêm trọng khác nhau. Các loại phần mềm sau có thể sinh ra thông báo: auth , authPriv , daemon , cron , ftp , dhcp , kern , mail, syslog, user, ... Với các mức độ nghiêm trọng từ cao nhất trở xuống Emergency, Alert, Critical, Error, Warning, Notice, Info, and Debug.

**Nguồn sinh ra log:**

|  |  |  |
| --- | --- | --- |
| **Facility Number** | **Nguồn tạo log** | **Ý nghĩa** |
|  |  |  |
| **0** | kernel | Những log mà do kernel sinh ra |
| **1** | user | Log ghi lại cấp độ người dùng |
| **2** | mail | Log của hệ thống mail |
| **3** | daemon | Log của các tiến trình trên hệ thống |
| **4** | auth | Log từ quá trình đăng nhập hệ hoặc xác thực hệ thống |
| **5** | syslog | Log từ chương trình syslogd |
| **6** | lpr | Log từ quá trình in ấn |
| **7** | news | Thông tin từ hệ thống |
| **8** | uucp | Log UUCP subsystem |
| **9** |  | Clock deamon |
| **10** | authpriv | Quá trình đăng nhập hoặc xác thực hệ thống |
| **11** | ftp | Log của FTP deamon |
| **12** |  | Log từ dịch vụ NTP của các subserver |
| **13** |  | Kiểm tra đăng nhập |
| **14** |  | Log cảnh báo hệ thống |
| **15** | cron | Log từ clock daemon |
| **16 – 23** | local 0 -local 7 | Log dự trữ cho sử dụng nội bộ |

**Mức độ cảnh bảo:**

|  |  |  |
| --- | --- | --- |
| **Code** | **Mức cảnh báo** | **Ý nghĩa** |
| **0** | emerg | Thông báo tình trạng khẩn cấp |
| **1** | alert | Hệ thống cần can thiệp ngay |
| **2** | crit | Tình trạng nguy kịch |
| **3** | error | Thông báo lỗi đối với hệ thống |
| **4** | warn | Mức cảnh báo đối với hệ thống |
| **5** | notice | Chú ý đối với hệ thống |
| **6** | info | Thông tin của hệ thống |
| **7** | debug | Quá trình kiểm tra hệ thống |

**Định dạng chung của một gói tin syslog:**

Định dạng hoàn chỉnh của một thông báo syslog gồm có 3 phần chính như sau:

**<PRI> <HEADER> <MSG>**

Độ dài một thông báo không được vượt quá 1024 bytes

* PRI

Phần PRI là một số được đặt trong ngoặc nhọn, thể hiện cơ sở sinh ra log hoặc mức độ nghiêm trọng. là 1 số 8bit. 3 bit đầu tiên thể hiện cho tính nghiêm trọng của thông báo.5bit còn lại đại diện cho sơ sở sinh ra thông báo.

Giá trị Priority được tính như sau: Cơ sở sinh ra log x 8 + Mức độ nghiêm trọng. Ví dụ, thông báo từ kernel (Facility = 0) với mức độ nghiêm trọng (Severity =0) thì giá trị Priority = 0x8 +0 = 0. Trường hợp khác,với "local use 4" (Facility =20) mức độ nghiêm trọng (Severity =5) thì số Priority là 20 x 8 + 5 = 165.

Vậy biết một số Priority thì làm thế nào để biết nguồn sinh log và mức độ nghiêm trọng của nó. Ta xét 1 ví dụ sau:

Priority = 191 Lấy 191:8 = 23.875 -> Facility = 23 ("local 7") -> Severity = 191 - (23 \* 8 ) = 7 (debug)

* HEADER

Phần Header thì gồm các phần chính sau

* Time stamp -- Thời gian mà thông báo được tạo ra. Thời gian này được lấy từ thời gian hệ thống ( Chú ý nếu như thời gian của server và thời gian của client khác nhau thì thông báo ghi trên log được gửi lên server là thời gian của máy client)
* Hostname hoặc IP
* Message

Phần MSG chứa một số thông tin về quá trình tạo ra thông điệp đó. Gồm 2 phần chính

* Tag field
* Content field

Tag field là tên chương trình tạo ra thông báo. Content field chứa các chi tiết của thông báo

#### 2.1.4.2 Rsyslog

Rsyslog - "The rocket-fast system for log processing" được bắt đầu phát triển từ năm 2004 bởi Rainer Gerhards rsyslog là một phần mềm mã nguồn mở sử dụng trên Linux dùng để chuyển tiếp các log message đến một địa chỉ trên mạng (log receiver, log server) Nó thực hiện giao thức syslog cơ bản, đặc biệt là sử dụng TCP cho việc truyền tải log từ client tới server. Hiện nay rsyslog là phần mềm được cài đặt sẵn trên hầu hết hệ thống Unix và các bản phân phối của Linux như : Fedora, openSUSE, Debian, Ubuntu, Red Hat Enterprise Linux, FreeBSD…

### 2.1.5 Log tập trung

Tác dụng của log là vô cùng to lớn vậy làm thế nào để quản lý log tốt hơn?

Để quản lý log một cách tốt hơn, xu thế hiện nay sẽ sử dụng log tập trung. Vậy log tập trung là gì? Tác dụng của nó thế nào?

Hiểu một cách đơn giản : Log tâp trung là quá trình tập trung, thu thập, phân tích... các log cần thiết từ nhiều nguồn khác nhau về một nơi an toàn để thuận lợi cho việc phân tích, theo dõi hệ thống.

Tại sao lại phải sử dụng log tập trung?

* Do có nhiều nguồn sinh log
* Có nhiều nguồn sinh ra log, log nằm trên nhiều máy chủ khác nhau nên khó quản lý.
* Nội dung log không đồng nhất (Giả sử log từ nguồn 1 có có ghi thông tin về ip mà không ghi thông tin về user name đăng nhập mà log từ nguồn 2 lại có) -> khó khăn trong việc kết hợp các log với nhau để xử lý vấn đề gặp phải.
* Định dạng log cũng không đồng nhất -> khó khăn trong việc chuẩn hóa.
* Đảm bảo tính toàn vẹn, bí mật, sẵn sàng của log.
* Do có nhiều các rootkit được thiết kế để xóa bỏ logs.
* Do log mới được ghi đè lên log cũ. -> Log phải được lưu trữ ở một nơi an toàn và phải có kênh truyền đủ đảm bảo tính an toàn và sẵn sàng sử dụng để phân tích hệ thống.

Do đó lợi ích của log tập trung đem lại là:

* Giúp quản trị viên có cái nhìn chi tiết về hệ thống -> có định hướng tốt hơn về hướng giải quyết.
* Mọi hoạt động của hệ thống được ghi lại và lưu trữ ở một nơi an toàn (log server) -> đảm bảo tính toàn vẹn phục vụ cho quá trình phân tích điều tra các cuộc tấn công vào hệ thống.
* Log tập trung kết hợp với các ứng dụng thu thập và phân tích log khác nữa giúp cho việc phân tích log trở nên thuận lợi hơn -> giảm thiểu nguồn nhân lực.

## 2.2 Giới thiệu chung về GRAYLOG

* Phần mềm mã nguồn mở quản lý log tập trung.
* Ra đời 2010 bởi Lennart Koopman với tên Graylog2.
* Tháng 2/2014, phát hành Graylog2 V0.20.0 Final.
* Tháng 1/2015, phát hành V1.x Beta, đổi tên thành Graylog, Graylog Inc được thành lập.
* Từ V1.0, đã trải qua 5 phiên bản, version mới nhất là Graylog 2.0 Alpha 5, stable version là 1.3

### 2.2.1 Đặc điểm

* Cơ chế nhận log rất linh hoạt: nhận log từ nhiều nguồn khác nhau.
* Sử dụng Elasticsearch => việc tìm kiếm chính xác, nhanh chóng và linh hoạt.
* Phân tích dữ liệu thành dạng số liệu thống kê, biểu đồ.
* Cảnh báo qua Email, Slack
* Mở rộng với REST API.
* Web-interface
* Graylog Collector
* Stream Alerts
* LDAP
* Output

### 2.2.**2** Elasticsearch

#### 2.2.2.1 Giới thiệu ElasticSearch:

Đầu tiên ta cần hiểu ElasticSearch là một công cụ tìm kiếm cấp doanh nghiệp (enterprise-level search engine). Mục tiêu của nó là tạo ra một công cụ, nền tảng hay kỹ thuật tìm kiếm và phân tích trong thời gian thực (ý nói ở đây là nhanh chóng và chính xác), cũng như cách để nó có thể áp dụng hay triển khai một cách dễ dàng vào nguồn dữ liệu (data sources) khác nhau.

Nguồn dữ liệu nói ở trên trên bao gồm các cơ sở dữ liệu nổi tiếng như MS SQL, PostgreSQL, MySQL, ... mà nó có thể văn bản (text), thư điện tử (email), pdf, ... nói chung tất tần tật mọi thứ liên quan tới dữ liệu có văn bản. Vấn đề này tôi sẽ giải thích rõ hơn trong phần bên dưới.

ElasticSearch được phát triển bởi Shay Banon và dựa trên Apache Lucene, ElasticSearch là một bản phân phối mã nguồn mở cho việc tìm kiếm dữ liệu trên máy chủ. Đây là một giải pháp mở rộng, hỗ trợ tìm kiếm thời gian thực mà không cần có một cấu hình đặc biệt. Nó đã được áp dụng bởi một số công ty, bao gồm cả StumbleUpon và Mozilla. ElasticSearch được phát hành theo Giấy phép Apache 2.0.

#### Một số thông tin về ElasticSearch:

* Elasticsearch là một search engine.
* Elasticsearch được xây dựng để hoạt động như một server cloud theo cơ chế của RESTful.
* Kế thừa và phát triển từ Lucene Apache.
* Phát triển bằng ngôn ngữ Java.
* Là phần mềm open-source được phát hành theo giất phép của Apache License.
* Tương tự như Solr (Apache).
* ELASTIC-SEARCH có thể tích hợp được với tất cả các ứng dụng sử dụng các loại ngôn ngữ sau.
* Java
* JavaScript
* Groovy
* .NET
* PHP
* Perl
* Python
* Ruby

Những ai đã dùng Elasticsearch:

* Mozilla
* Quora
* SoundCloud
* GitHub
* Stack Exchange
* Center for Open Science
* Reverb
* Netflix.

#### 2.2.2.3 Cơ chế hoạt động của ElasticSearch:

Chắc các bạn hẳn sẽ để ý tới từ "real time" thời gian thực mà tôi thường hay nói tới ở phía trên, đây cũng là tiêu chí phát triển của của Elasticsearch, sở dĩ Elasticsearch được gọi là "search & analyze in real time" là vì nó có khả năng trả về kết quả tìm kiếm một cách nhanh chóng và chính xác trong một nguồn dữ liệu lớn (big data source).

Vậy nó làm như thế nào? Để giải thích điều này ta quay trở về nguồn dữ liệu (data source) mà tôi đã nói ở trên nó không chỉ bao gồm các nguồn lưu trữ cơ sở dữ liệu nổi tiếng như MySQL, MS SQL, PostgreSQL, mà nó có thể là văn bản (text), pdf, doc, ...

Theo như cách thông thường tìm kiếm trong cơ sở dữ liệu database đều biết thì có hai cách là:

* Cách 1: Lật từng trang để tìm kiếm (No index)
* Cách 2: Lật tới phần mục lục để tìm kiếm.

Về cơ bản thì ElasticSearch cũng áp dụng giải pháp giống index. Tuy nhiên về mặt cơ chế xử lý và tìm kiếm thì có sự khác biệt, Index trong ElasticSearch được gọi là inverted index.

Inverted Index là gì?

Kỹ thuật thay vì index theo từng đơn vị row(document) giống như mysql thì chúng ta sẽ biến thành index theo đơn vị term. Cụ thể hơn, Inverted index là một cấu trúc dữ liệu, nhằm mục đích map giữa term và các document chứa term đó.

Ta sẽ xém xét ví dụ sau để hiểu rõ hơn về inverted index.

Ta có 3 document là D1, D2, D3

* D1 = "hom nay troi mua"
* D2 = "hom nay troi nang"
* D3 = "mot ngay dep troi"

Theo đó ta sẽ có inverted index của 3 document trên là

* "hom" = {D1, D2}
* "nay" = {D1, D2}
* "troi" = {D1, D2, D3}
* "mua" = {D1}
* "nang" = {D2}
* "mot" = {D3}
* "ngay" = {D3}
* "dep" = {D3}

Nhìn vào kết quả của inverted index trên ta có thể thấy việc search full text sẽ diễn ra rất dễ dàng và nhanh chóng. nó chỉ là việc tính toán của các term. Ví dụ khi muốn query từ “hom nay” (tùy theo việc là query theo kiểu and hay or thì phép tính toán sẽ khác đi. ở đây mình sẽ lấy ví dụ là query theo kiểu and.) thì phép toán là

{D1, D2} ∩ {D1, D2} = {D1, D2}

Kết quả thu được chính là document 1 và 2(D1 and D2).

Qua ví dụ trên các bạn đã hiểu cơ bản về inverted index.

**Chú thích:**

Nếu ở đây nếu bạn thiết lập một phương thức tìm kiếm chính xác thì sẽ không có kết quả nào được trả về.

Tất nhiên đây chỉ là một ví dụ đơn giản để mô tả về **ElasticSearch**, về cơ bản nó cũng cho phép bạn tạo ra các "**Inverted Index**" theo nhưng mục đích cụ thể hoặc phù hợp với câu query để đạt chuẩn tốc độ cao hơn và chính xác hơn.

Như tôi đã trình bày bạn thấy rất rõ ElasticSearch rất thích hợp áp dụng cho các Website hơn so với applicaiton, nhưng với sự vượt trội về hiệu suất (performance) mà nó mang lại nên một số applicaiton hoặc application services cũng đã áp dụng kỹ thuật này.

Bài viết xin được dừng tai đây, nếu có gì thiếu xót hoặc chưa chính xác rất mong nhận được ý kiến đóng góp của các bạn.

### MongoDB

#### 2.2.3.1 Giới thiệu MongoDB

MongoDB là một mã nguồn mở và là một tập tài liệu dùng cơ chế NoSQL để truy vấn, nó được viết bởi ngôn ngữ C++. Chính vì được viết bởi C++ nên nó có khả năng tính toán với tốc độ cao chứ không giống như các hệ quản trị CSDL hiện nay.

Cấu trúc của MongoDB bao gồm các cặp "thuộc tính" - "giá tri", MongoDB document tương tự như đối tượng JSON. Giá trị của thuộc tính bao gồm các thuộc tính giá trị cụ thể (số, chữ, ...), document, mảng, mảng các đối tượng.

Các tính năng chính:

* Hiệu suất cao: Hỗ trợ nhúng dữ liệu dạng mô hình dữ liệu giúp giảm thiểu hoạt động của server, truy vấn dữ liệu sử dụng chỉ mục giupps tối ưu tốc độ truy vấn.
* Dễ dàng tăng tính mở rộng: chế độ sharding tự động (tự động phân dùng dữ liệu trên máy chủ, đồng bộ hóa dữ liệu tốt).

MongoDB chạy trên hầu hết các nền tảng và hỗ trợ kiến trúc 64-bit để sử dụng sản xuất và cả hai 64-bit và 32-bit kiến trúc để thử nghiệm.

MongoDB được sử dụng bởi nhiều hang lớn như MTV Networks, Adobe, Google, Cisco, Ebay, Facebook, …

**Các hệ điều hành hỗ trợ:**

|  |  |  |
| --- | --- | --- |
|  | MongoDB | MongoDB Enterprise |
| Amazon Linux | Hỗ trợ | Hỗ trợ |
| Debian 7.1 | Hỗ trợ | Hỗ trợ |
| RedHat/Centos 6.2+ | Hỗ trợ | Hỗ trợ |
| SUSE 11 | Hỗ trợ | Hỗ trợ |
| Ubuntu LTS 12.04 | Hỗ trợ | Hỗ trợ |
| Ubuntu LTS 14.04 | Hỗ trợ | Hỗ trợ |
| Windows Server 2012 | Hỗ trợ | Hỗ trợ |
| Mac OSX 10.6+ | Hỗ trợ |  |
| RedHat/Centos 5.5+ | Hỗ trợ |  |
| RedHat/Centos 5.7+ | Hỗ trợ | Hỗ trợ |
| Windows Server 2008 R2 | Hỗ trợ | Hỗ trợ |

#### 2.2.3.2 Các khái niệm

MongoDB cũng có cấu trúc lưu trữ tương tự JSON, chính vì thế nó có hiệu suất cao, tương tác nhanh và khả năng mở rộng tốt, nó hoạt động trên khái niệm **Collection** và **Document**.

**Database** là một ô chứa dữ liệu ở mức vật lý (physical), mỗi database sẽ có nhiều collection và được thiết lập lưu trữ ở một nơi trong máy chủ máy tính. Một máy chủ MongoDB thường có thể tạo nhiều cơ sở dữ liệu.

**Metadata:** là nơi chứa toàn bộ thông tin tổng quát của database:

* Namespaces: danh sách tất cả các namespace.
* Index: danh sách các chỉ mục
* Profile: thông tin profile.
* User: danh sách các user được truy cập vào database.

**Collection** trong MongoDB bản chất được gọi là nhóm các document. Một collection chứa tập các document, Ta có thể xem nó tương tự như các table(bảng) trong cơ sở dữ liệu quan hệ. Tên của collection nên được bắt đầu bởi các chữ cái hoặc dấu gạch dưới và có thể chứa các kiểu số. các collection có thể được cấu tạo trong các namespace (là tên của một nhóm các collection).

Bản chất của **Collection**: tập hợp các văn bản BSON (BSON documents). Nhưng trong MongDB khác biệt là các document không tuân theo cấu trúc tức là các document trong 1 collection không có cấu trúc cố định như nhau (không cần lưu các cột giống nhau), như vậy ta không cần phải định nghĩa thành phần các cột trong 1 collection như trong cơ sở dữ liệu quan hệ.

**Document** trong MongoDB có cấu trúc tương tự như kiểu dữ liệu JSON, nghĩa là sẽ có các cặp (key => giá trị) nên nó có tính năng động rất lớn. Document ta có thể hiểu nó giống như các record dữ liệu trong MYSQL, tuy nhiên nó có sự khác biệt là các cặp (key => value) có thể không giống nhau ở mỗi document.

**BSON** là dãy tuần tự nhị phân được mã hóa như tài liệu JSON. BSON giống như JSON, hỗ trợ nhúng các đối tượng và mảng trong các đối tượng và mảng khác. MongoDB dùng BSON khi lưu trữ dữ liệu và là định dạng các document khi chúng được di chuyển trên network.

Để hiểu rõ hơn sau đây là bảng so sánh giữa CSDL thông thường với MongoDB và một ví dụ về cách lưu trữ của MongoDB.

|  |  |
| --- | --- |
| **CSDL thông thường** | **MongoDB** |
| Database | Database |
| Table | Collection |
| Tuple/Row | Document |
| Column | Field |
| Table Join | Embedded Documents |
| Primary Key | Primary Key (mặc định là \_id) |

Ví dụ:

{

\_id: ObjectId(7df78ad8902c)

title: **'Tìm hiểu mongodb'**,

description: **'MongoDB sử dụng khái niệm NO-SQL'**,

comments: [

{

user:**'user1'**,

message: **'First comment'**,

dateCreated: new Date(2011,1,20,2,15),

like: 0

},

{

user:**'user2'**,

message: **'Second comment'**,

dateCreated: new Date(2011,1,25,7,45),

like: 5

}

]

}

Trong đó ta thấy **key\_id** là do hệ thống tự tạo ra một field như vậy làm khóa chính. Đây là cấu trúc của một mẫu document trong MongoDB, như vậy khi có thêm comment thì ta sẽ thêm nó vào phần comment của tin.

#### 2.2.3.3 Khi nào sử dụng MongoDB

MongoDB thật sự rất hot nhưng không phải lúc nào ta sử dụng nó cũng tốt, có những trường hợp không nên sử dụng và nên sử dụng, chi tiết thế nào thì chúng ta cùng thảo luận nhé.

* Nếu website của bạn có tính chất INSERT cao, bởi vì mặc định MongoDB có sẵn cơ chế ghi với tốc độ cao và an toàn.
* Website của bạn ở dạng thời gian thực nhiều, nghĩa là nhiều người thao tác với ứng dung. Nếu trong quá trình load bị lỗi tại một điểm nào đó thì nó sẽ bỏ qua phần đó nên sẽ an toàn.
* Website bạn có nhiều dữ liệu quá, giả sử web bạn có đến 10 triệu records thì đó là cơn ác mộng với MYSQL. Bởi vì MongoDB có khả năng tìm kiến thông tin liên quan cũng khá nhanh nên trường hợp này nên dùng nó.
* Máy chủ không có hệ quản trị CSDL, trường hợp này thường bạn sẽ sử dụng SQLITE hoặc là MongoDB.

### 2.2.**4** Web Interface

Khi đối tượng/cụm Graylog của bạn đang bật và chạy, điều tiếp theo bạn thường muốn làm là kiểm tra giao diện web của chúng tôi, trong đó cung cấp cho bạn khả năng tuyệt vời để tìm kiếm và phân tích các dữ liệu được lập chỉ mục của bạn và cấu hình môi trường Graylog của bạn. Mỗi mặc định, bạn có thể truy cập nó bằng cách sử trình duyệt của bạn **http://<graylog-server>:9000/api/**.

#### 2.2.4.1 Tổng quan

Giao diện web Graylog được viết lại bằng JavaScript cho 2.0 là một trang duy nhất ứng dụng trình duyệt phía client. Điều này có nghĩa là mã của nó đang chạy chỉ trong trình duyệt của bạn, lấy tất cả dữ liệu thông qua HTTP (S) từ API REST của máy chủ Graylog của bạn.

**Chú thích:**

Cả hai giao diện web URI (**web\_listen\_uri**) và REST API (**rest\_listen\_uri & rest\_transport\_uri**) phải được truy cập bởi tất cả mọi người sử dụng giao diện web. Điều này có nghĩa rằng Graylog phải lắng nghe trên một giao diện mạng công cộng hoặc được tiếp xúc với một cách sử dụng một proxy hoặc NAT!

#### Đơn hay riêng biệt người nghe cho giao diện web và REST API?

Kể từ Graylog 2.1 bạn có hai lựa chọn khi nói đến lộ giao diện web của nó:

* Chạy cả trên cùng một cổng, sử dụng đường dẫn khác nhau (mặc định để **http://localhost:9000/api/** cho các API REST và **http://localhost:9000/** cho giao diện web), đây là mặc định từ 2.1 và được giả định cho hầu hết các bộ phận của tài liệu.
* Chạy trên hai cổng khác nhau (ví dụ **http://localhost:12900/** cho các API REST và **http://localhost:9000/** cho giao diện web)

**Chú thích:**

Khi bạn đang sử dụng các tùy chọn đầu tiên và bạn muốn chạy các API REST và các giao diện web trên cùng một máy chủ và cổng, một phần con đường của cả hai URI (**rest\_listen\_uri & web\_listen\_uri**) phải khác nhau và một phần con đường **web\_listen\_uri** phải không trống và khác nhau hơn.

#### 2.2.4.3 Tùy chọn cấu hình

Nếu cài đặt mặc định của chúng tôi không làm việc cho bạn, có một số tùy chọn trong tập tin cấu hình máy chủ Graylog mà bạn có thể thay đổi ảnh hưởng đến hành vi của nó:

|  |  |  |
| --- | --- | --- |
| **Cài đặt** | **Mặc định** | **Giải trình** |
| **web\_enable** | thật | Xác định điểm cuối giao diện web được bắt đầu hay không. |
| **web\_listen\_uri** | **http://127.0.0.1:9000/** | Mặc định giải quyết cho người nghe giao diện web liên kết với. |
| **web\_endpoint\_uri** | Nếu không được thiết lập, **rest\_transport\_uri** sẽ được sử dụng. | Đây là địa chỉ bên ngoài của REST API của máy chủ Graylog. giao diện Web khách hàng cần để có thể kết nối này cho giao diện web để làm việc. |
| **web\_enable\_cors** | sai | Hỗ trợ Cross-Origin Resource Sharing cho các tài sản giao diện web. Không cần thiết, bởi vì không có cuộc gọi REST của được làm cho người nghe này. Thiết lập này được bỏ qua, nếu các máy chủ và cổng phụ của **web\_listen\_uri & rest\_listen\_uri** là giống hệt nhau. |
| **web\_enable\_gzip** | thật | Phục vụ tài sản giao diện web sử dụng nén. Thiết lập này được bỏ qua, nếu các máy chủ và cổng phụ của **web\_listen\_uri & rest\_listen\_uri** là giống hệt nhau. |
| **web\_enable\_tls** | sai | Nên giao diện web phục vụ tài sản sử dụng mã hóa hay không. Thiết lập này được bỏ qua, nếu các máy chủ và cổng phụ của **web\_listen\_uri & rest\_listen\_uri** là giống hệt nhau. |
| **web\_tls\_cert\_file** | (Không mặc định) | Đường dẫn đến tệp TLS, nếu TLS được kích hoạt. Thiết lập này được bỏ qua, nếu các máy chủ và cổng phụ của **web\_listen\_uri & rest\_listen\_uri** là giống hệt nhau. |
| **web\_tls\_key\_file** | (Không mặc định) | Đường dẫn đến khóa riêng cho giấy chứng nhận, được sử dụng nếu TLS được kích hoạt. Thiết lập này được bỏ qua, nếu các máy chủ và cổng phụ của**web\_listen\_uri & rest\_listen\_uri** là giống hệt nhau. |
| **web\_tls\_key\_password** | (Không mặc định) | Mật khẩu cho khóa TLS (nếu nó được mã hóa). Thiết lập này được bỏ qua, nếu các máy chủ và cổng phụ của **web\_listen\_uri & rest\_listen\_uri** là giống hệt nhau. |
| **web\_thread\_pool\_size** | 16 | Số chủ đề được sử dụng cho giao diện web nghe. Thiết lập này được bỏ qua, nếu các máy chủ và cổng phụ của **web\_listen\_uri & rest\_listen\_uri** là giống hệt nhau. |

#### Làm thế nào để giao diện web kết nối đến máy chủ Graylog?

Giao diện web được lấy tất cả các thông tin nó là hiển thị từ API REST của máy chủ Graylog. Vì vậy nó cần phải kết nối với nó bằng cách sử HTTP (S). Có một số cách làm thế nào bạn có thể xác định cách mà các giao diện web kết nối đến máy chủ Graylog. URI được sử dụng bởi giao diện web được xác định theo thứ tự chính xác:

* Nếu HTTP (S) của khách hàng để đưa về cảng giao diện web sẽ gửi một **X-Graylog-Server-URL** tiêu đề, trong đó có chứa một URL hợp lệ, sau này là trọng mọi thứ khác.
* Nếu **web\_endpoint\_uri** được xác định trong tập tin cấu hình Graylog, điều này được sử dụng nếu các tiêu đề nói trên không được thiết lập.
* Nếu cả hai đều không được xác định, **rest\_transport\_uri** được sử dụng.

#### Khả năng tương thích trình duyệt

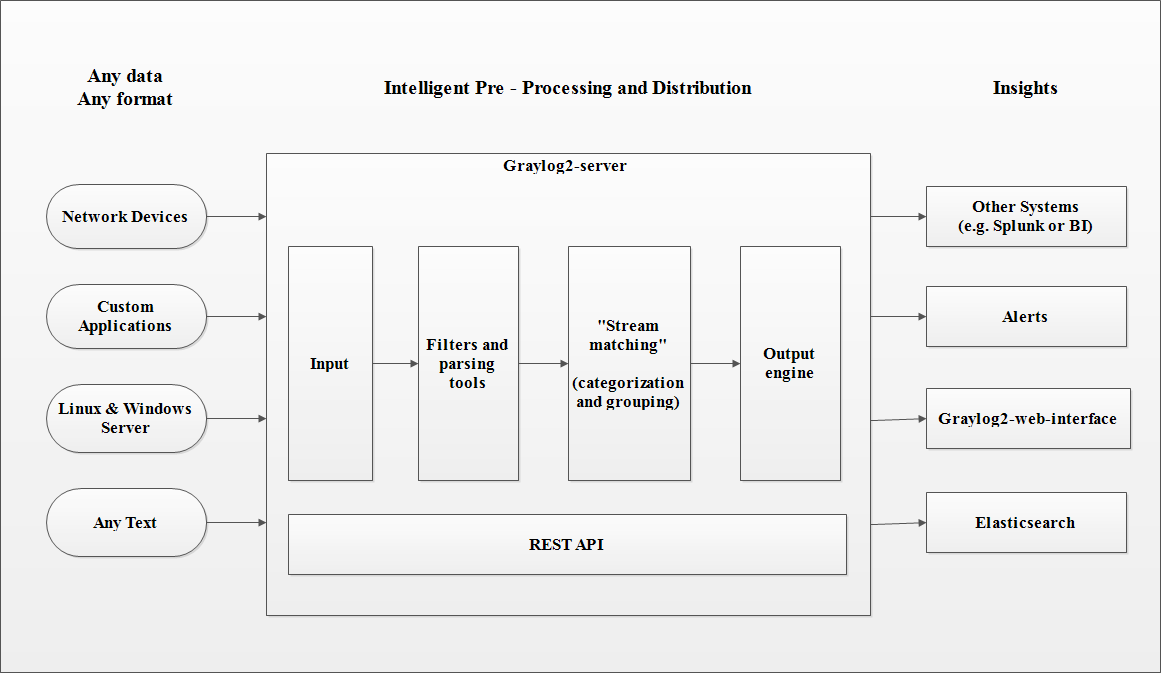
Viết giao diện web như là một ứng dụng duy nhất trang là một nhiệm vụ đầy thử thách. Chúng tôi muốn cung cấp các kinh nghiệm tốt nhất có thể để tất cả mọi người, trong đó thường có nghĩa là sử dụng công nghệ web hiện đại chỉ có sẵn trong các trình duyệt gần đây, trong khi vẫn giữ một khả năng tương thích hợp lý với các trình duyệt cũ và ít có khả năng. Các trình duyệt được hỗ trợ chính thức trong Graylog 2.0:

|  |  |  |
| --- | --- | --- |
| **Trình duyệt** | **Hệ điều hành** | **Phiên bản tối thiểu** |
| Chrome | Windows, OS X, Linux | 50 |
| Firefox | Windows, OS X, Linux | 45 / 38 ESR |
| Internet Explorer | Windows | 11 |
| Microsoft Edge | Windows | 25 |
| Safari | OS X | 9 |

### Các thành phần trong Graylog2

* **Graylog Server:** Nhận, xử lý các bản tin và truyền thông với các thành phần khác – **Cần CPU.**
* **Elasticsearch:** Công cụ lưu trữ, tìm kiếm dữ liệu – tất cả phụ thuộc vào tốc độ **I/O, Cần Ram**.
* **MongoDB:** Lưu trữ metadata (file cấu hình…). Chỉ cần cấu hình thấp.
* **Web Interface:** Cung cấp giao diện cho người dung

### 2.2.6 Nguyên lý hoạt động trong Graylog2



##### Hình 2.4 Nguyên lý hoạt động Graylog2

Tất cả các định dạng, các ứng dụng trên các nền tảng, mọi loại dữ liệu đều có thể nhận được bởi Graylog

Các dữ liệu vào thông qua SYSLOG (Linux) hoặc NXLOG (Window)

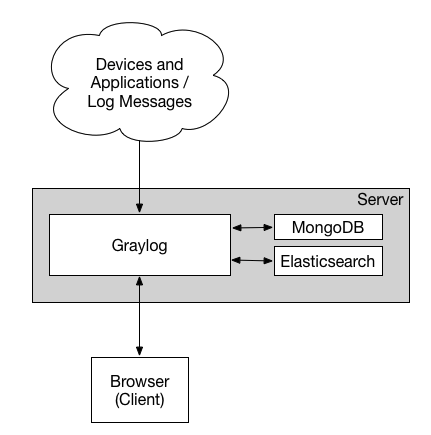
Tất cả các dữ liệu vào (dữ liệu thô) thông qua quá trình xử lý của Graylog Server đi ra ngoài thông qua cơ chế REST API

Dữ liệu ra ngoài có thể được lưu trữ trong Elasticseach, hiển trị trên web interface hoặc cũng có thể kết hợp với các ứng dụng của các nền tảng khác

### 2.2.7 Các mô hình triển khai của Graylog

#### 2.2.7.1 Mô hình Minimum setup

Đây là một thiết lập Graylog tối thiểu có thể được sử dụng cho các thiết lập nhỏ hơn, không quan trọng, hoặc thử nghiệm. Không ai trong số các thành phần là không cần thiết nhưng nó rất dễ dàng và nhanh chóng để thiết lập.

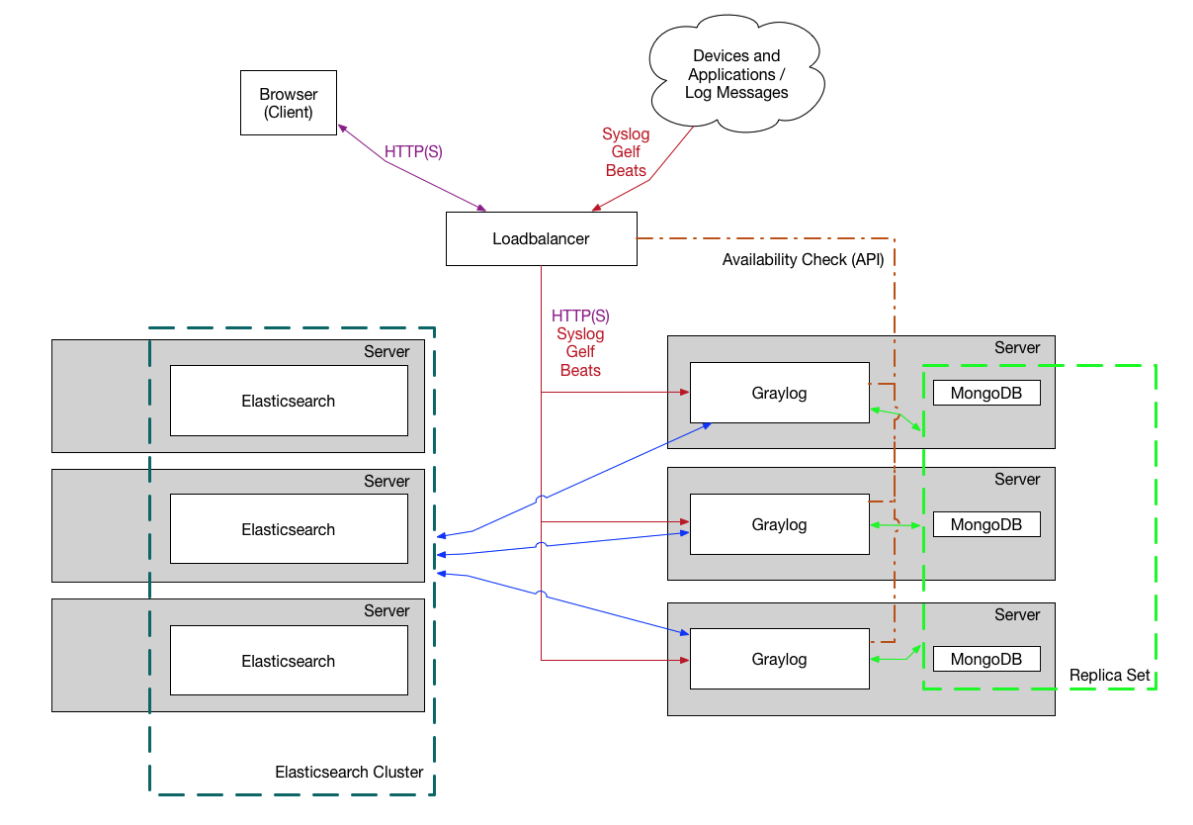


##### Hình 2.5. Mô hình Minimum setup

#### Mô hình Bigger production setup

Đây là một thiết lập cho các môi trường sản xuất lớn hơn. Nó có một số nút Graylog đằng sau một cân bằng tải phân phối tải trọng xử lý.

Bộ cân bằng tải có thể ping các nút Graylog qua HTTP trên Graylog REST API để kiểm tra nếu họ còn sống và có các nút chết ra khỏi cluster.



##### Hình 2.6. Mô hình Bigger production setup

#### Mô hình Highly Available setup with Graylog2 Radio

Graylog2-Radio là một add-on cho Graylog2-Server giúp bạn gửi các bản ghi của bạn đến một chủ đề AMQP (Advanced Message Queuing Protocol) để trao đổi.

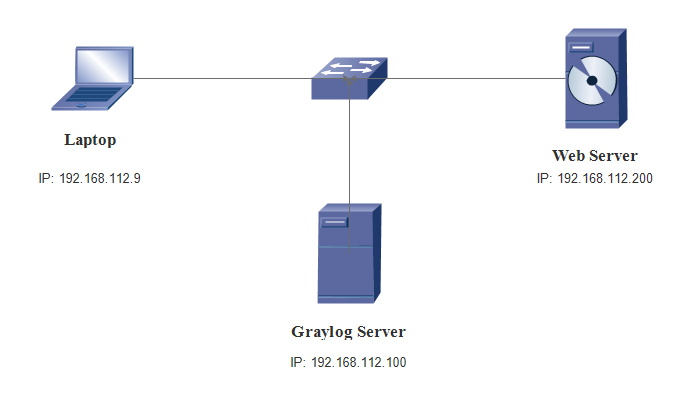
Với Graylog2-Radio bạn có thể tránh được các vấn đề như bộ đệm đầy đủ và bác bỏ các thông điệp và cũng có thể bạn có thể dừng graylog2-server của bạn mà không bị mất bất kỳ tin nhắn bởi vì họ đang nhận được bằng văn bản cho một máy chủ AMQP.

Có gì graylog2 radio làm là lắng nghe một tcp / udp port cho thông điệp syslog hoặc gelf và xuất bản chúng trong một cuộc trao đổi AMQP, sau đó graylog2-server có thể được cấu hình để tiêu thụ các thư này và lưu trữ chúng trong elasticsearch.

##### *Hình 2.7.3. Mô hình Highly Available setup with Graylog2 Radio*

# CHƯƠNG 3: TRIỂN KHAI HỆ THỐNG

## 3.1 Mô hình Graylog2



##### Hình 3.1: Mô hình Graylog2

## 3.2 Cài đặt và sử dụng Graylog2

### 3.2.1 Chuẩn bị

Một máy chủ đóng vài trò là Graylog-Server. Trong phần Demo này do điều kiện hạn chế nên em sẽ chỉ thiết lập Graylog rất cơ bản, tất cả các thành phần đều sẽ được cài đặt trên cùng một máy chủ.

* OS: Centos 7.2.1511 (Core) đã được update đầy đủ
* NIC: eno16777736, IP: 192.168.112.100
* Services: Graylog 2.x, Mongodb 3.2, Elasticsearch 2.3, Java 1.8.0\_73

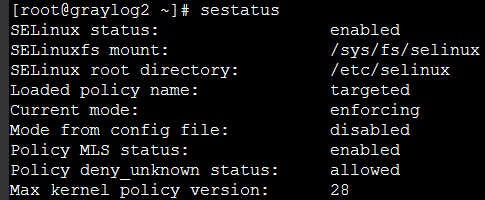
Một vài máy chủ đóng vai trò là máy chủ ứng dụng sẽ gửi log về cho máy chủ Graylog-Server. Trong phần trình bày này em sẽ sử dụng một WebServer và một DNSServer.

* Đầu tiên chúng ta sẽ tắt tường lửa Centos và kiểm tra trạng thái:

*# systemctl disable firewalld*

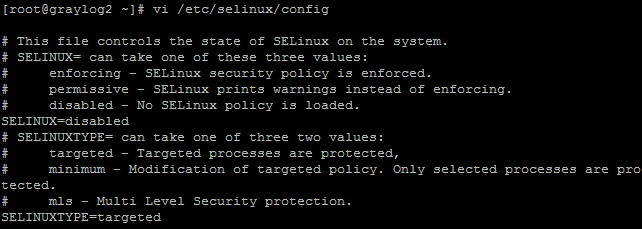
*# systemctl status firewalld*

* Kiểm tra trạng thái SELinux *(Security-Enhanced Linux)* bằng lệnh: *# sestatus*



##### Hình 3.2: Kiểm tra trạng thái SELinux

* Nếu trạng thái đang được bật thì chúng ta tắt SELinux bằng cách thay đổi file /etc/selinux/config như sau:



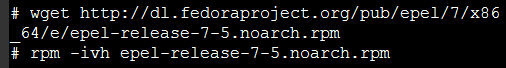
##### Hình 3.3: Disable SELinux

* Thay đổi SELINUX=enforcing thành SELINUX=disabled sau đó khởi động lại hệ thống. Kiểm tra trạng thái một lần nữa sau khi khởi động.



##### Hình 3.4: Kiểm tra trạng thái SELinux

* Kích hoạt chức năng EPEL *(Extra Packages for Enterprise Linux)*  Repository



##### Hình 3.5: Kích hoạt chức năng EPEL

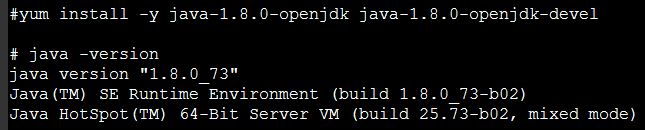
### 3.2.2 Các bước cài đặt

Trong phần này, sẽ cài đặt 4 dịch vụ là Java, Elasticsearch, Mongodb, Graylog-Server.

(Lưu ý ở phần cài đặt này, các hệ thống Graylog phiên bản 1.x sẽ chỉ chạy được trên các phiên bản Elasticsearch dưới 2.x, ngược lại các phiên bản Graylog 2.x sẽ chỉ chạy được trên các phiên bản Elasticsearch 2.x trở lên. Đồng thời, để có thể tập trung được log ta còn phải cài đặt và cấu hình Rsyslog để các máy chủ ứng dụng có thể đẩy log về cho máy chủ log tập trung, mà máy chủ ứng dụng ở đây em sẽ dùng WebServer Nginx đóng vai trò là Rsyslog Client, và máy chủ log tập trung là máy Graylog-Server đóng vai trò là Rsyslog Server).

#### Cài đặt Java

Để cài đặt được Elasticsearch, đầu tiên cần phải cài đặt Java, ở đây cụ thể phiên bản Java ít nhất phải là Java 7, và tốt nhất là sử dụng các phiên bản Oracle JDK 1.8.0\_73.



##### Hình 3.6: Cài đặt Java

#### Cài đặt và cấu hình Elasticsearch

* Đầu tiên import GPR-key

*#rpm --import* [*https://packages.elastic.co/GPG-KEY-elasticsearch*](https://packages.elastic.co/GPG-KEY-elasticsearch)

* Thêm Elasticsearch vào file repositoriy bằng cách mở file elasticsearch.repo

*#vi /etc/yum.repos.d/elasticsearch.repo*

* Sau đó thêm các dòng sau vào file

*[elasticsearch-2.x]*

*name=Elasticsearch repository for 2.x packages*

*baseurl=https://packages.elastic.co/elasticsearch/2.x/centos*

*gpgcheck=1*

*gpgkey=https://packages.elastic.co/GPG-KEY-elasticsearch*

*enabled=1*

* Tiến hành cài đặt

*#yum install elasticsearch*

* Cấu hình Elasticsearch khởi động cùng với hệ thống

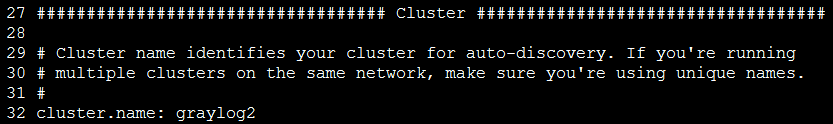


##### Hình 3.7: Cấu hình Elasticsearch khởi động cùng hệ thống

* Vào file elasticsearch.yml để cấu hình cho nó hoạt động với graylog2

*#vi /etc/elasticsearch/elasticsearch.yml*

* Xuống dòng 32 trong file và đặt cluster name là graylog2



##### Hình 3.8: Rename cluster.name

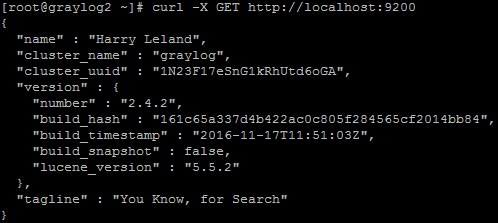
(Cũng cần chú ý rằng cluster.name trong Elasticsearch phải giống với cluster.name trong tập tin cấu hình Graylog).

* Khởi động Elasticsearch và đặt ở chế độ luôn bật:



##### Hình 3.9: Khởi động và enable Elasticsearch

* Có thể dùng lệnh để kiểm tra Elasticsearch đã chạy đúng chưa:



##### Hì*nh 3.10: Kiểm tra Elasticsearch*

#### 3.2.2.3 Cài đặt MongoDB

* Tạo một mongodb-stable repository file và thêm các cấu hình như ở dưới:

*#vi /etc/yum.repos.d/mongodb-org-3.2.repo*

* Thêm các dòng sau vào file và lưu

[mongodb-org-3.2]

name=MongoDB Repository

baseurl=https://repo.mongodb.org/yum/redhat/$releasever/mongodb-org/3.2/x86\_64/

gpgcheck=1

enabled=1

gpgkey=https://www.mongodb.org/static/pgp/server-3.2.asc

* Cài đặt MongoDB bằng lệnh

#yum install -y mongodb-org

* Khởi động Mongodb:



##### Hình 3.11: Khởi động MongoDB

#### 3.2.2.4 Cài đặt và cấu hình Graylog2

* Tải về kho lưu trữ mới nhất của Graylog

*#sudo rpm -Uvh https://packages.graylog2.org/repo/packages/graylog-2.0 repository\_latest.rpm*

* Tiếp theo cài đặt Graylog2 server bằng lệnh

*#yum install graylog-server*

* Khởi động graylog-server

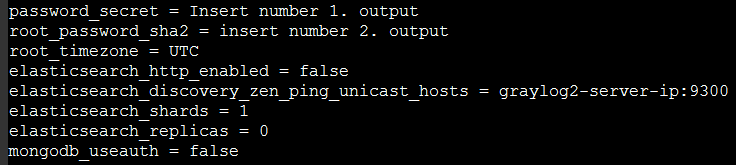


##### Hình 3.12: Khởi động graylog-server

* Chúng ta mở tệp cấu hình Graylog2



##### Hình 3.13: Cấu hình graylog2



##### Hình 3.14: File cấu hình mặc định của graylog server

* Tạo password\_secret với pwgen bằng lệnh sau

*# pwgen -N 1 -s 96*

* Sử dụng chuỗi vừa tạo chèn vào file

*password\_secret=Hz63qI3vRiJezJ6U44PmgtNV5eh0ixiBlRImqPwJAOeUDaWb5wopeSXavxFdzHlAP6BZtJBol8fdHqDjrOCoUq2fczBGhud6*

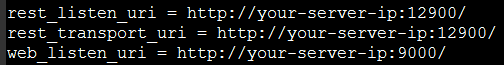
Bây giờ chúng ta cần phải thiết lập một mật khẩu băm cho người dùng root. Chúng ta sẽ sử dụng mật khẩu này để đăng nhập vào giao diện web bằng cách lấy mã sha265 của mật khẩu bằng lệnh:

*# echo -n yourpassword | sha256sum*

* Sử dụng chuỗi vừa tạo chèn vào file

*root\_password\_sha2=e3c652f0ba0b4801205814f8b6bc49672c4c74e25b497770bb89b22cdeb4e951 -*

Bây giờ chúng ta cần phải cấu hình các API REST và các giao diện Web Graylog để cho phép chúng kết nối với Graylog Server



##### Hình 3.15: Cấu hình API REST

* Khởi động lại hệ thống Graylog để nhận cấu hình mới



##### Hình 3.16: Khởi động hệ thống Graylog

### 3.2.3 Cấu hình Web Interface và sử dụng

* Chúng ta sẽ cài đặt web interface cho Graylog2 như sau:

*#yum – y install graylog-web*

* Vào file cấu hình web:

*# vi /etc/graylog/web/web.conf*

* Sửa những dòng sau trong tệp cấu hình và lưu lại:



##### Hình 3.17: Edit trong tệp cấu hình

* Để khởi động Graylog-web cùng hệ thống, chúng ta dùng lệnh sau:

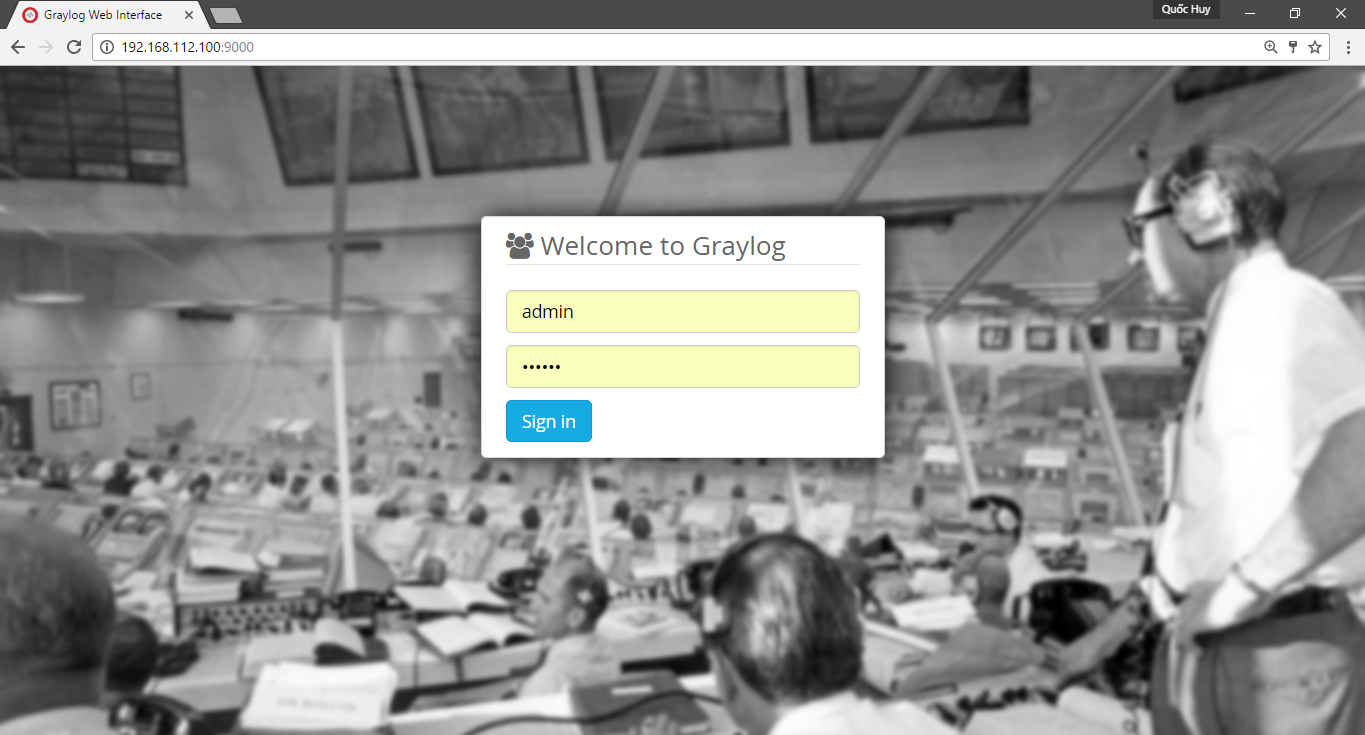


##### Hình 3.18: Khởi động Graylog-web

## 3.3 Kiểm thử

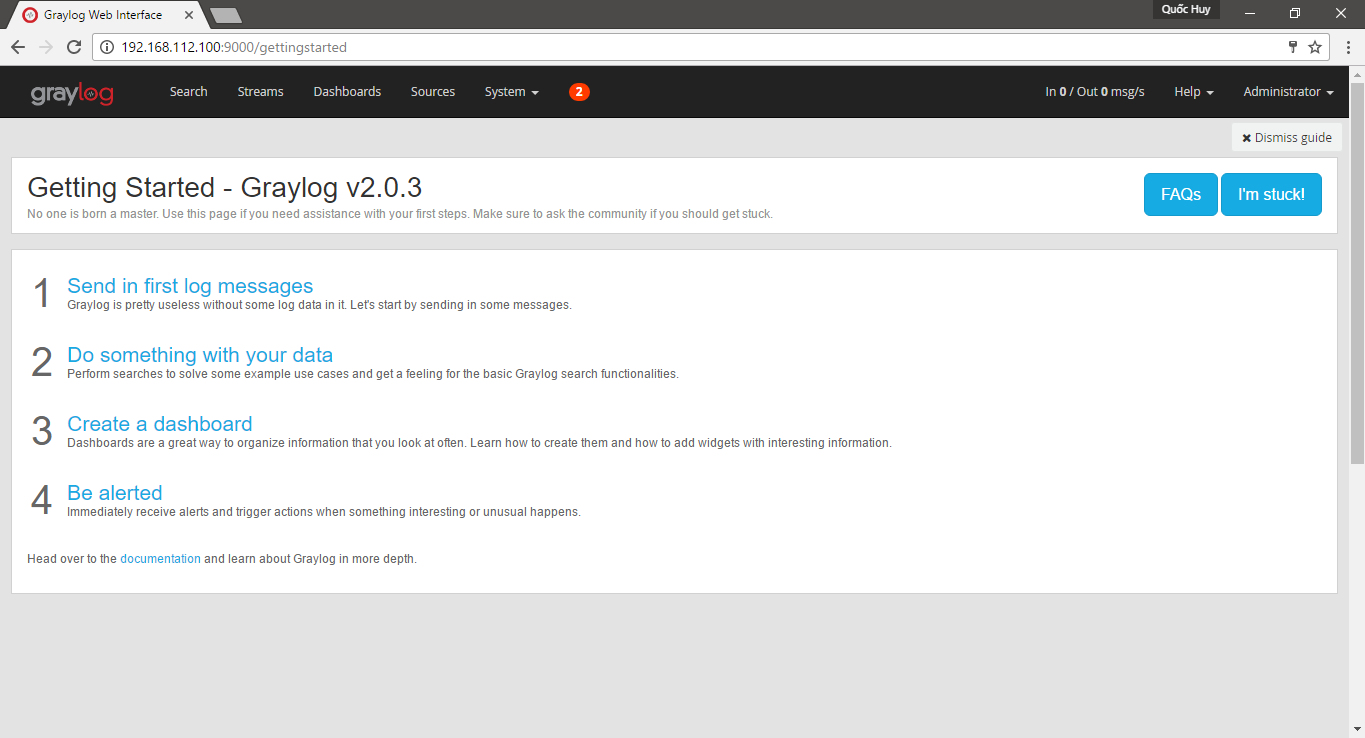
### 3.3.1 Đăng nhập

* Tiếp tục truy cập http://192.168.112.100:9000 để và trang chủ Graylog vừa thiết lập.



##### Hình 3.19: Màn hình đăng nhập của Graylog

* Sau khi đăng nhập sẽ được như sau:

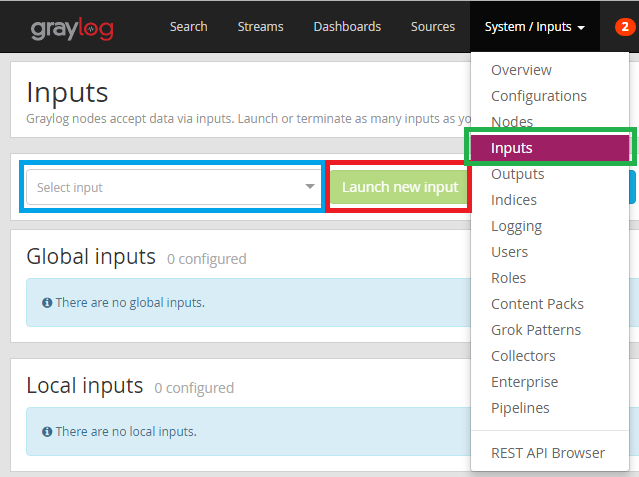


##### Hình 3.20: Trang chủ của Web Interface Graylog vừa tạo.

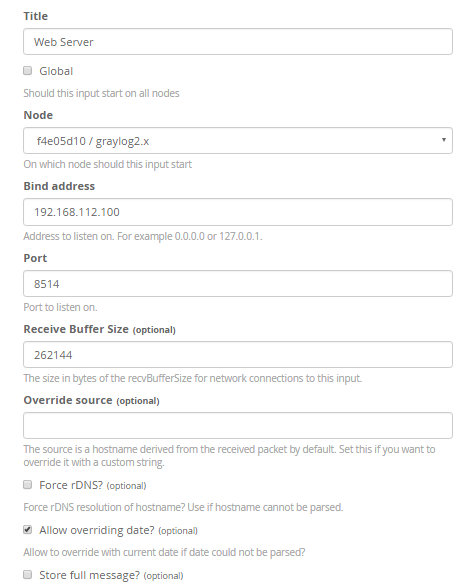
### 3.3.2 Bắt đầu kiểm thử

Tại máy chủ Graylog em sẽ thiết lập 1 đầu vào để có thể nhận log từ WebServer đẩy về.

Ở đây em sẽ sử dụng phương thức chuyển UDP và port 8514 cho việc đẩy log từ WebServer về máy chủ Graylog.



##### Hình 3.21: Tạo Input cho Client, ở đây là WebServer, màu xanh lam là chọn giao thức cho Input, màu đỏ là bắt đầu một Input.

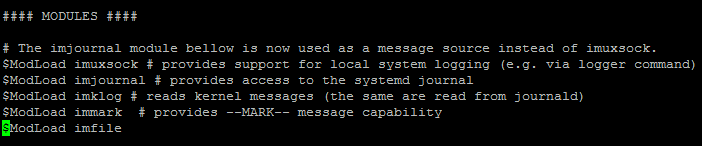


##### Hình 3.22: Giao diện thiết lập Input cho WebServer

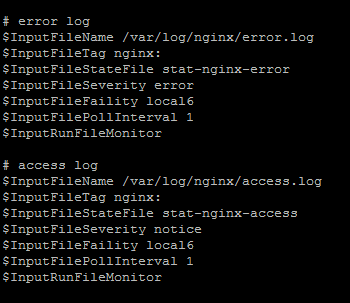
Trong phần kiểm thử này, em sẽ sử dụng một máy chủ ứng dụng là WebServer để đẩy log về server tập trung để tiến hành thu thập cũng như phân tích log.

Để có thể gửi log được về server tập trung em sẽ cấu hình Rsyslog trên máy chủ Web như sau:

* Vào file /etc/rsyslog.conf thêm các dòng vào như bên dưới:

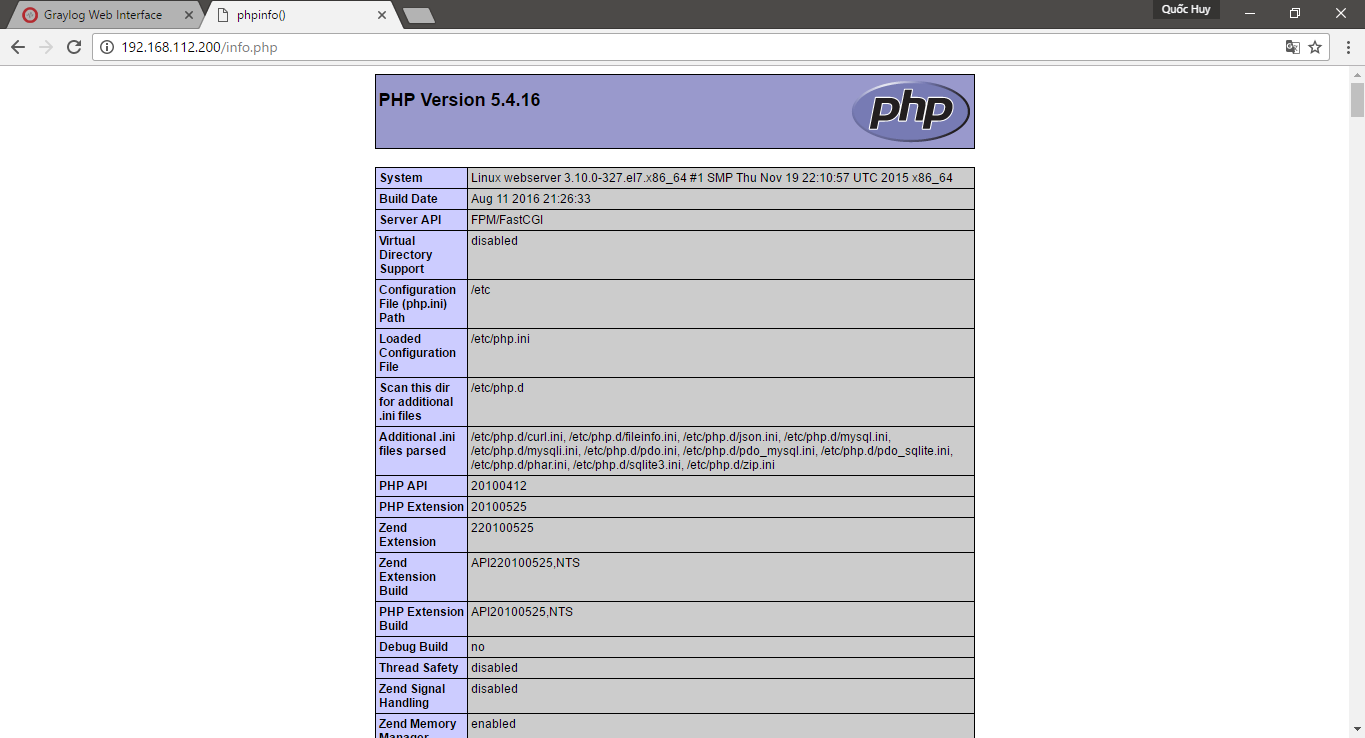


##### Hình 3.23: Cấu hình MODULE Rsyslog

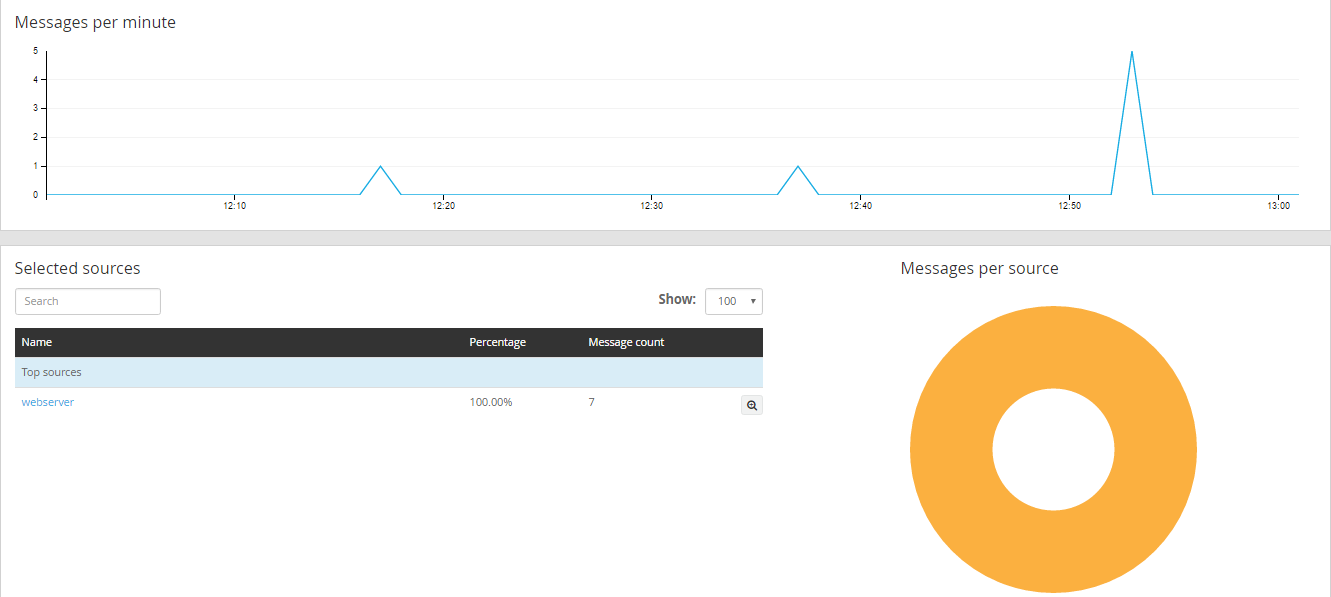


##### Hình 3.24: Phương thức để gửi log Nginx về máy chủ Rsyslog

* Trên WebServer này em đã cài đặt sẵn một Web là 192.168.112.200/info.php để demo. Em sẽ tiến hành truy cập trang này.

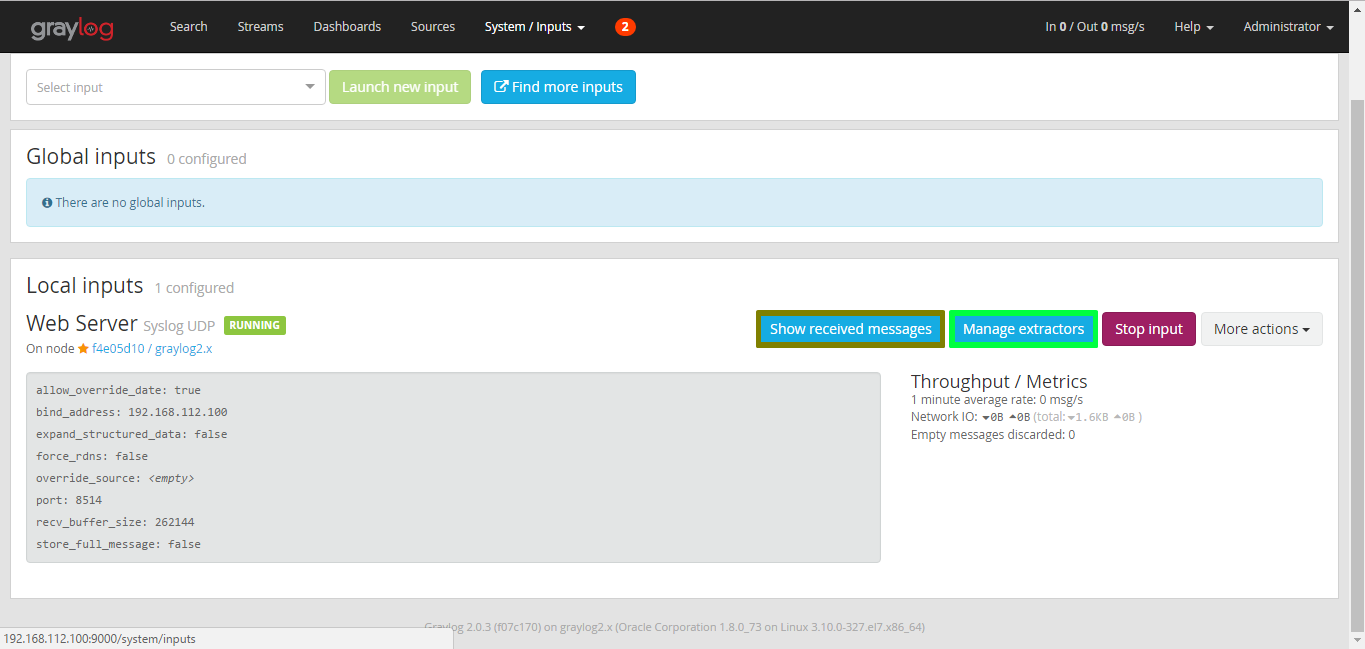


##### Hình 3.25: Truy cập vào máy chủ WebServer bằng 192.168.112.200/info.php



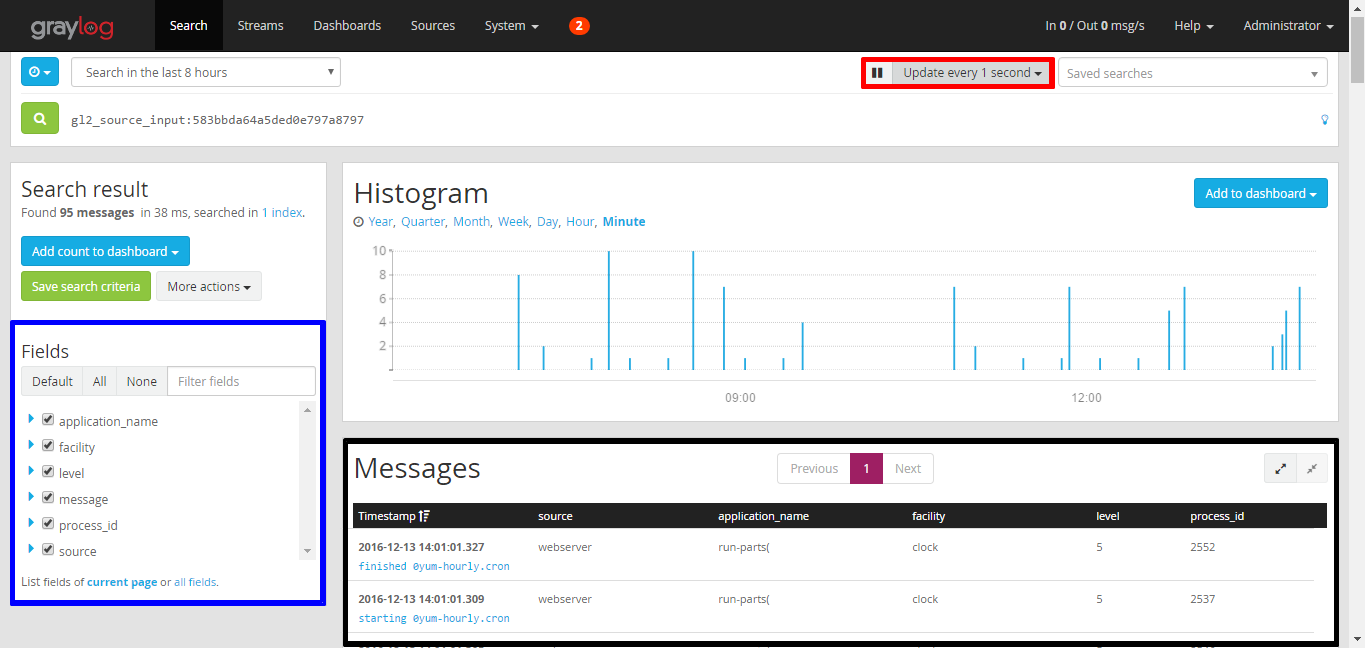
##### Hình 3.26: Giao diện hiển thị số Input được thiết lập, ở đây chỉ có 1 Input là từ máy chỉ WebServer

* Để tiến hành đọc log cũng như phân tích ta sẽ chọn như hình dưới:



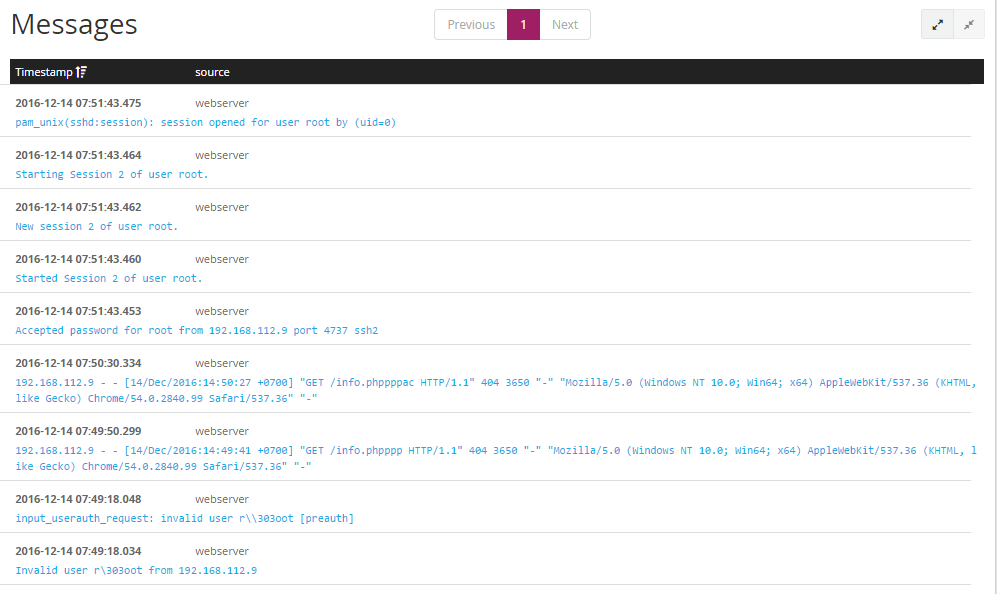
##### Hình 3.27: Thiết lập trong Inputs

Trong hình này, để hiển thị toàn bộ log ta sẽ chọn “Show received messages” ở ô màu đỏ, hoặc để có thể thiết lập thêm các phương thức phân tích log ta sẽ chọn “Manager extractors”



##### Hình 3.28: Toàn bộ giao diện khi chọn “Show received messages”

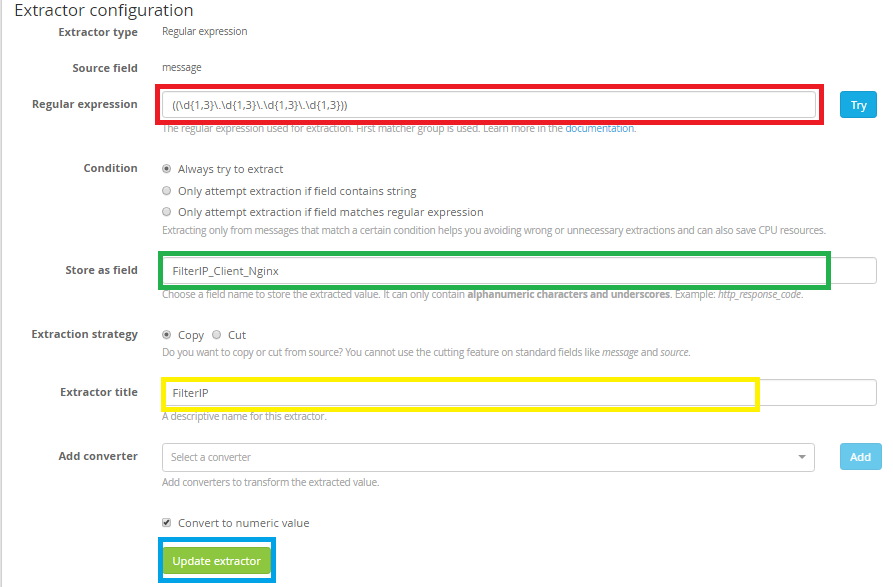
Trong hình này, ô màu đỏ được chọn sẽ hiển thị messages log theo thời gian thực, ở đây em thiết lập sẽ update 1giây/1 lần, ô màu xanh dương sẽ là các field sẽ hiển thị của messages, ô màu đen lá sẽ là toàn bộ nội dung messages log, ở góc trên phía trái là nơi chọn mốc thời gian hiển thị messages log và tìm kiếm.



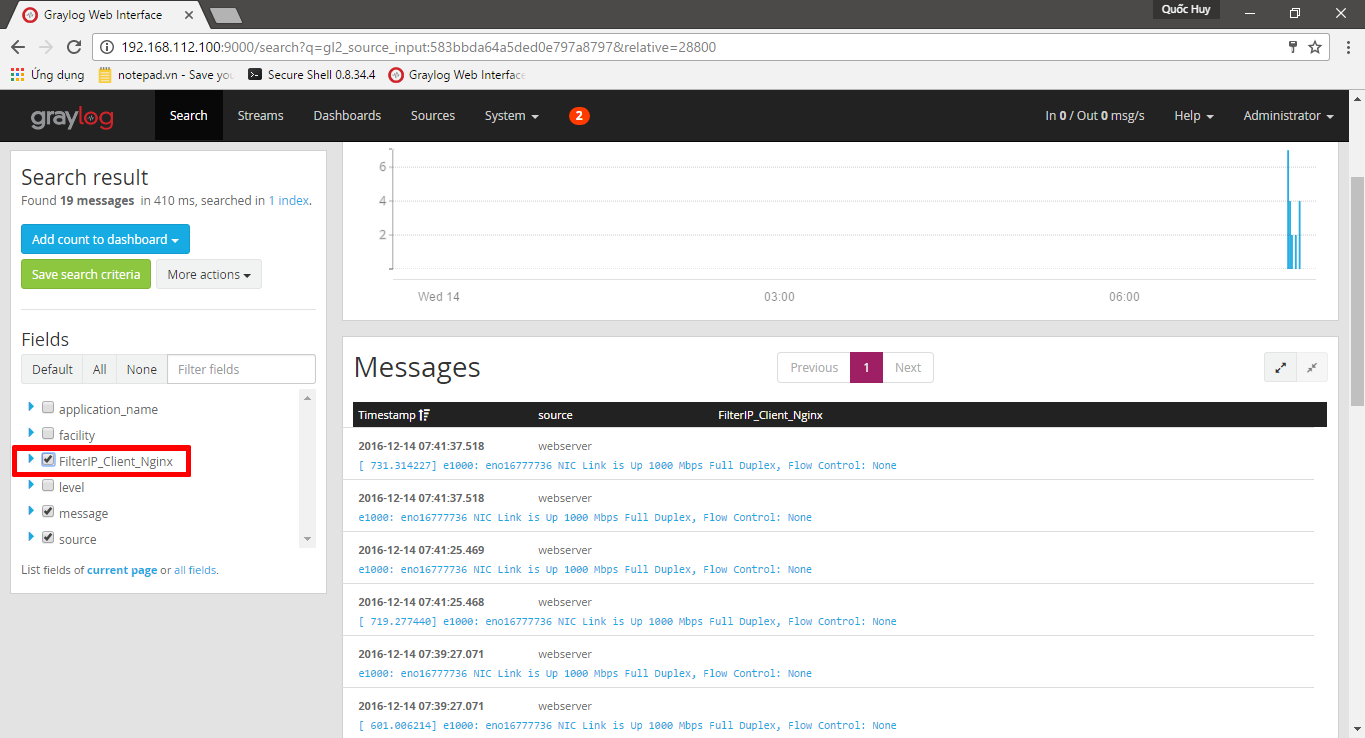
##### Hình 3.29: Thông tin các Messages chứa log được gửi về Graylog

Để tiến hành phân tích một messages log, em sẽ chọn bất kỳ một log và chọn phương thức phân tích. Ví dụ ở đây em sẽ chọn phương thức lọc IP để xác định những IP nào đã truy cập WebServer và sử dụng dịch vụ web ở đây, cụ thể là Nginx.

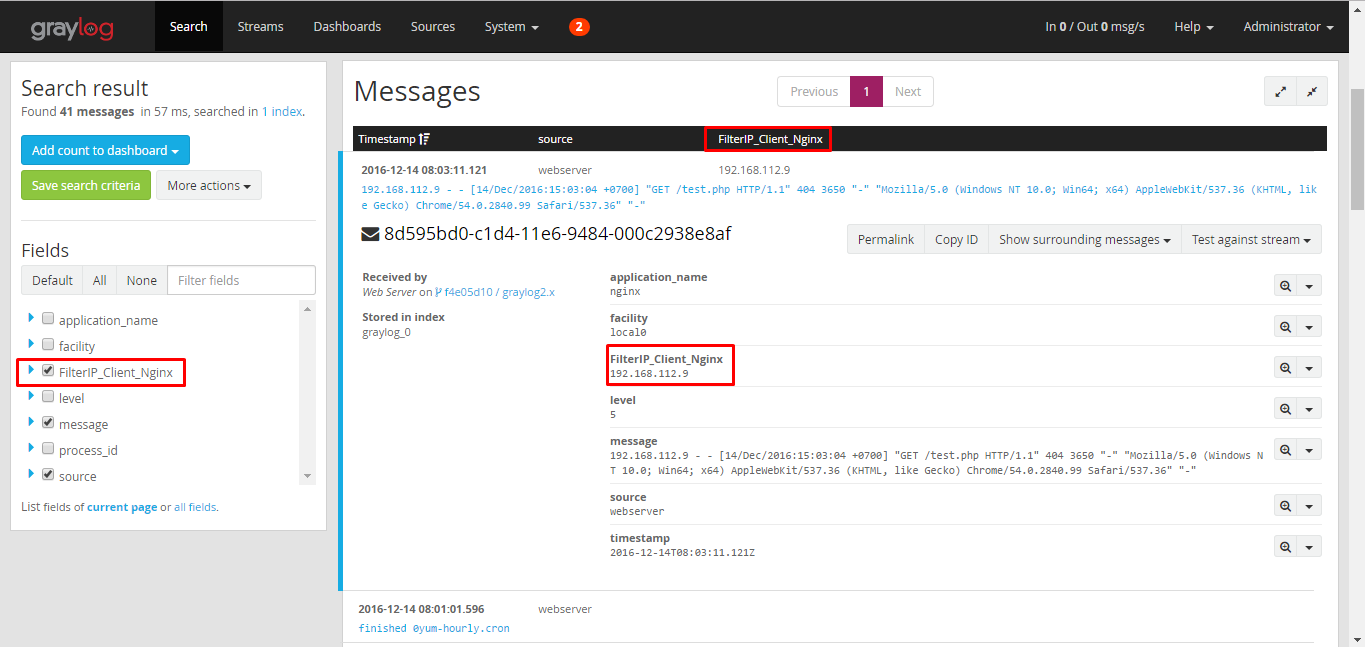
Ở hình này, em chọn phương thức Extrator là Regular expression. Trong phần Rengular Expression em chọn cách lọc IP bằng lệnh “((\d{1,3}\.\d{1,3}\.\d{1,3}))”, phần Store as field là tên Field sẽ hiển thị ở ô “Field” trong hình phái trên, phần Extrator title là tên field sẽ hiển thị ở Manager extractor. Cuối cùng là ấn “Create Extractor”. Ở phần này của em, do em đã tạo sẵn nên sẽ hiển thị là “Update extractor”. Kết quả:



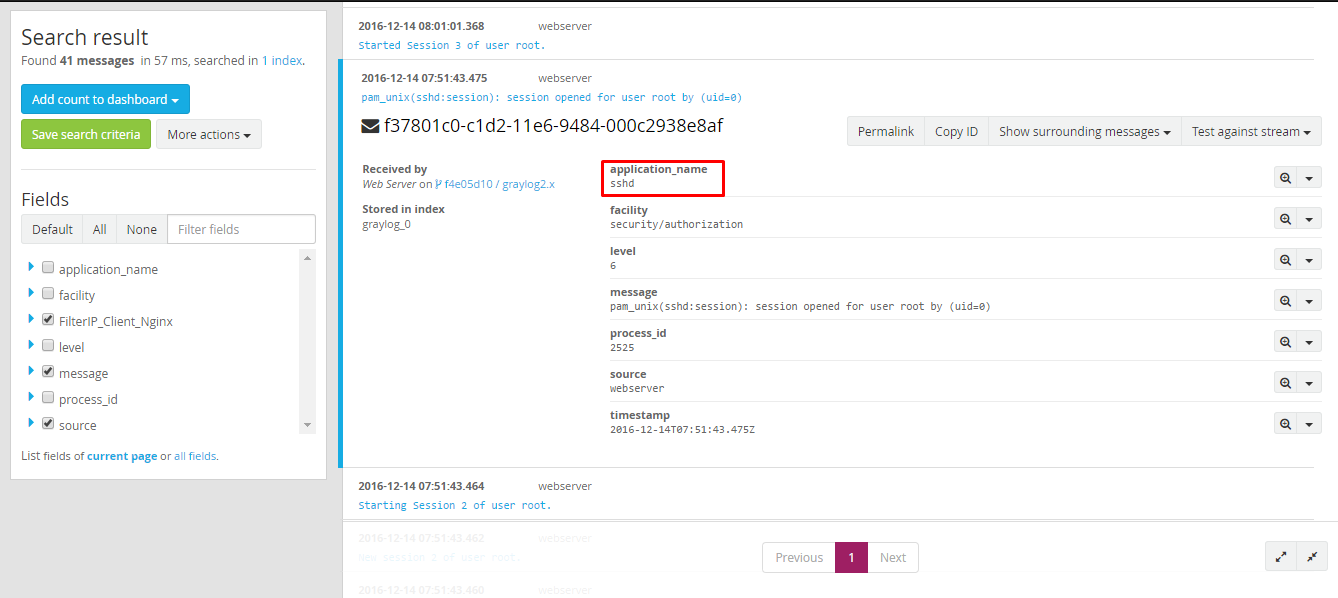
##### Hình 3.30: Lọc IP truy cập WebServer sử dụng dịch vụ web.



##### *Hình 3.31: Fields mới đã hiển thị ở phần Fields trên giao diện chính (ô màu đỏ)*



##### *Hình 3.32: Hiển thị IP truy cập vào Web*



##### Hình 3.33: Hiển thị một người dùng mới bắt đầu một phiên SSH

Tương tự có nhiều cách để phân tích một messages log. Ví dụ như máy chủ ứng dụng cài đặt hoặc update ứng dụng gì, vào thời điểm nào, ai ssh vào máy chủ, vào thời điểm nào, đúng hay sai user, user tên gì, đúng hay sai mật khẩu, người dùng sử dụng dịch vụ gì trên máy chủ ứng dụng đó, vào một thời điểm có bao nhiêu người truy cập vào máy chủ ứng dụng đó, ứng dụng gì đang chạy trên đó …

# KẾT LUẬN

* **KẾT QUẢ ĐẠT ĐƯỢC**
* Về lý thuyết
* Trình bày các lý thuyết về Graylog2.
* Nguyên tắc hoạt động của Graylog2 cũng như các mô hình Graylog2
* Nắm bắt được việc triển khai và quản lý sao cho hiệu suất hoạt động của hệ thống nâng cao, dễ dàng và tập trung.
* Về triển khai thực tế:
* Triển khai ứng dụng Graylog2 cho hệ thống mạng.
* **HẠN CHẾ CỦA ĐỀ TÀI**
* Do còn nhiều hạn chế về thiết bị và thời gian cho nên trong báo cáo này còn thiếu những khía cạnh chưa thể nói lên đầy đủ các chức năng của Graylog2. Một số hạn chế và một số điều chưa thực hiện được.
* Việc triển khai ứng dụng Graylog2 cho hệ thống mạng chỉ nằm ở cấp độ Local.
* Các thiết bị sử dụng trong đề tài chưa được nhiều, mới chỉ nằm trong các máy client ảo.
* **HƯỚNG PHÁT TRIỂN**

Trong phạm vi một khóa luận tốt nghiệp, báo cáo cơ bản đã đạt được các yêu cầu đặt ra. Tuy nhiên, các kết quả còn khá khiêm tốn do hạn chế về thời gian. Trong thời gian tới, nếu có điều kiện em sẽ cố gắng phát triển thêm những nội dung sau:

* Nghiên cứu xây dựng mạng Campus áp dụng vào thực tế. Áp dụng được các thiết bị vật lý có sẵn như router, firewall,switch,…

Xây dựng mô hình ứng dụng đầy đủ hơn và có thể phát triển cho doanh nghiệp.