**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG**

---

ISO 9001:2015

# ĐỒ ÁN TỐT NGHIỆP

**NGÀNH: CÔNG NGHỆ THÔNG TIN**

**Sinh viên : Hoàng Văn Cận**

**Giảng viên hướng dẫn: ThS. Phùng Anh Tuấn**

**HẢI PHÒNG - 2019**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG**

---

# XÂY DỰNG HỆ THỐNG GIÁM SÁT MẠNG DỰA TRÊN PHẦN MỀM NGUỒN MỞ ZABBIX

**ĐỒ ÁN TỐT NGHIỆP ĐẠI HỌC HỆ CHÍNH QUY**

**NGÀNH: CÔNG NGHỆ THÔNG TIN**

**Sinh viên : Hoàng Văn Cận**

**Giảng viên hướng dẫn: ThS. Phùng Anh Tuấn**

**HẢI PHÒNG - 2019**

**BỘ GIÁO DỤC VÀ ĐÀO TẠO**

**TRƯỜNG ĐẠI HỌC DÂN LẬP HẢI PHÒNG**

---

# NHIỆM VỤ ĐỀ TÀI TỐT NGHIỆP

Sinh viên: Hoàng Văn Cận Mã SV: 1412101028

Lớp: CT1802 Ngành: Công nghệ thông tin

Tên đề tài: Xây dựng hệ thống giám sát mạng dựa trên phần mềm nguồn mở Zabbix

# MỤC LỤC

# LỜI CẢM ƠN

Đề tài "Xây dựng ứng dụng hệ thống giám sát mạng dựa trên phần mềm nguồn mở Zabbix" là nội dung Em chọn để nghiên cứu và làm đồ án tốt nghiệp sau bốn năm học chương trình đại học ngành công nghệ thông tin tại trường Đại Học Dân Lập Hải Phòng.

Để hoàn thành quá trình nghiên cứu và hoàn thiện đồ án tốt nghiệp này, lời đầu tiên Em xin gửi lời cảm ơn chân thành cảm ơn tới toàn thể quý Thầy Cô, bạn bè của Trường Đại Học Dân Lập Hải Phòng.

Bày tỏ lòng biết ơn sâu sắc nhất thầy cô trong khoa công nghệ thông tin đã dìu dắt, chia sẻ những kiến thức quý báu trong suốt quá trình học tập tại trường. Đặc biệt là thầy ThS.Phùng Anh Tuấn cùng với tri thức và tâm huyết của Thầy đã tạo điều kiện em hoàn thành đồ án tốt nghiệp tại trường. Nếu không có Thầy đồ án tốt nghiệp của Em khó có thể hoàn thành được.

Cuối cùng, Em xin cảm ơn những người thân, bạn bè đã luôn bên Em, động viên, sẻ chia, giúp đỡ, cổ vũ tinh thần… Đó là nguồn động lực giúp Em hoàn thành chương trình học và đồ án tốt nghiệp này.

*Hải Phòng, ngày 28 tháng 03 năm 2019*

**Sinh viên**

Hoàng Văn Cận

# LỜI MỞ ĐẦU

Cùng với sự phát triển của công nghệ thông tin, sự đầu tư cho hạ tầng mạng trong mỗi doanh nghiệp ngày càng tăng cao, dẫn đến việc quản trị sự cố một hệ thống mạng gặp rất nhiều khó khăn. Đi cùng với những lợi ích khi phát triển hạ tầng mạng như băng thông cao, khối lượng dữ liệu trong mạng lớn, đáp ứng được nhu cầu của người dùng, hệ thống mạng phải đối đầu với rất nhiều thách thức như các cuộc tấn công bên ngoài, tính sẵn sàng của thiết bị, tài nguyên của hệ thống,…

Một trong những giải pháp hữu hiệu nhất để giải quyết vấn đề này là thực hiện việc giải pháp giám sát mạng, dựa trên những thông tin thu thập được thông qua quá trình giám sát, các nhân viên quản trị mạng có thể phân tích, đưa ra những đánh giá, dự báo, giải pháp nhằm giải quyết những vấn đề trên. Để thực hiện giám sát mạng có hiệu quả, một chương trình giám sát phải đáp ứng được các yêu cầu sau: phải đảm bảo chương trình luôn hoạt động, tính linh hoạt, chức năng hiệu quả, đơn giản trong triển khai, chi phí thấp. Hiện nay, có khá nhiều phần mềm hỗ trợ việc giám sát mạng có hiệu quả như Nagios, Zabbix, Zenoss, Cacti,…

Vì vậy, Em đã chọn đề tài **"Xây dựng hệ thống giám sát mạng dựa trên phần mềm nguồn mở Zabbix"**, một phần mềm mã nguồn mở với nhiều chức năng mạnh mẽ cho phép quản lý các thiết bị, dịch vụ trong hệ thống mạng. Với mục tiêu nghiên cứu, tìm hiểu về giải pháp giúp cho mọi người có cái nhìn tổng quan về một hệ thống giám sát mạng hoàn chỉnh, đồng thời đưa ra một giải pháp cụ thể đối với một hệ thống mạng dành cho doanh nghiệp.

# CHƯƠNG 1. TỔNG QUAN HỆ THỐNG GIÁM SÁT MẠNG

## 1.1. Giám sát mạng

### 1.1.1. 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.

Khi phụ trách hệ thống mạng máy tính, để giảm thiểu tối đa các sự cố làm gián đoạn hoạt động của hệ thống mạng, người quản trị hệ thống mạng cần phải nắm được tình hình "sức khỏe" các thiết bị, dịch vụ được triển khai để có những quyết định xử lý kịp thời và hợp lý nhất. Ngoài ra, việc hiểu rõ tình trạng hoạt động của các thiết bị, các kết nối mạng… cũng giúp cho người quản trị tối ưu được hiệu năng hoạt động của hệ thống mạng để đảm bảo được các yêu cầu sử dụng của người dùng. Việc giám sát hoạt động của các thiết bị mạng, ứng dụng và dịch vụ trong môi trường mạng, với hàng chục hay hàng trăm thiết bị, mà người quản trị thực hiện thủ công sẽ không mang lại hiệu quả. Vì thế, cần phải có một phần mềm thực hiện việc giám sát một cách tự động và cung cấp các thông tin cần thiết để người quản trị nắm được hoạt động của hệ thống mạng, đó là hệ thống giám sát mạng.

Hệ thống giám sát mạng (Network Monitoring System) là một phần mềm thực hiện việc giám sát hoạt động của hệ thống và các dịch vụ, ứng dụng bên trong hệ thống mạng đó. Nó thực hiện việc thu thập thông tin của các thiết bị mạng, các kết nối, các ứng dụng và dịch vụ bên trong hệ thống mạng để phân tích và đưa ra các thông tin hỗ trợ người quản trị mạng có cái nhìn tổng quan, chi tiết về môi trường mạng. Dựa trên những thông tin thu thập được, hệ thống giám sát mạng có thể tổng hợp thành các báo cáo, gửi các cảnh báo cho người quản trị để có hướng xử lý phù hợp nhằm giảm thiểu sự cố và nâng cao hiệu suất mạng. Với những thông tin nhận được từ hệ thống giám sát mạng, người quản trị có thể xử lý các sự cố và đưa ra các hướng nâng cấp thiết bị, dịch vụ để đảm bảo hệ thống mạng hoạt động thông suốt.

### 1.1.2. Các yếu tố cơ bản trong giám sát mạng

Để việc giám sát mạng đạt hiệu quả cao nhất, cần xác định các yếu tố cốt lõi của giám sát mạng như:

- Các đơn vị, hệ thống, thiết bị, dịch vụ cần giám sát.
- Các trang thiết bị, giải pháp, phần mềm thương mại phục vụ giám sát.
- Xác định các phần mềm nội bộ và phần mềm mã nguồn mở phục vụ giám sát.

Ngoài ra, yếu tố con người, đặc biệt là quy trình phục vụ giám sát là vô cùng quan trọng.

### 1.1.3. Chức năng của giám sát mạng

- Cảnh báo qua Web, Email và SMS khi phát hiện tấn công vào hệ thống mạng.
- Báo động bằng âm thanh và SMS khi một host (Server, Router, Switch…) hoặc một dịch vụ mạng ngưng hoạt động.
- Giám sát lưu lượng mạng qua các cổng giao tiếp trên Router, Switch, Server… hiển thị qua các đồ thị trực quan, thời gian thực. Giám sát lưu lượng giữa các thiết bị kết nối với nhau một cách trực quan

### 1.1.4. Cần giám sát những gì và tại sao?

Đối với hệ thống mạng, điều quan trọng nhất là nắm được các thông tin chính xác nhất vào mọi thời điểm. Tầm quan trọng chính là nắm bắt thông tin trạng thái của thiết bị vào thời điểm hiện tại, cũng như biết được thông tin về các dịch vụ, ứng dụng của hệ thống.

Thông tin sau đây chứa một vài nội dung trạng thái hệ thống mà ta phải biết và lý do tại sao:

| Cần giám sát gì | Tại sao |
|---|---|
| Tính sẵn sàng của thiết bị (Router, Switch, Server…). | Đây là những thành phần chủ chốt giữ cho mạng hoạt động. |
| Các dịch vụ trong hệ thống (DNS, FTP, HTTP…) | Những dịch vụ này đóng vai trò quan trọng trong một cơ quan, tổ chức, nếu các dịch vụ này không được đảm bảo hoạt động bình thường và liên tục, nó sẽ ảnh hưởng nghiêm trọng đến cơ quan tổ chức đó. |

| | |
|---|---|
| Tài nguyên hệ thống | Các ứng dụng đều đòi hỏi tài nguyên hệ thống, việc giám sát tài nguyên sẽ đảm bảo cho chúng ta có những can thiệp kịp thời, tránh ảnh hưởng đến hệ thống. |
| Lưu lượng trong mạng | Nhằm đưa ra những giải pháp, ngăn ngừa hiện tượng quá tải trong mạng. |
| Các chức năng về bảo mật | Nhằm đảm bảo an ninh trong hệ thống. |
| Lượng dữ liệu vào và ra của router. | Cần xác định chính xác thông tin lượng dữ liệu để tránh quá tải hệ thống. |
| Các sự kiện được viết ra log như WinEvent or Syslog. | Có thể thu được thông tin chính xác các hiện tượng xảy ra trong hệ thống. |
| Nhiệt độ, thông tin về máy chủ, máy in | Ta có thể biết được thông tin về máy in bị hư hỏng hay cần thay mực trước khi được người dùng báo cũng như đảm bảo máy chủ không bị quá nóng. |

*Hình 1.1.4.1: Thông tin các thiết bị và lý do cần giám sát*

Khi một hệ thống mạng được triển khai và đưa vào vận hành, vấn đề giám sát hoạt động của toàn bộ hệ thống có vai trò quan trọng. Các bất thường liên quan đến thiết bị, dịch vụ, tấn công mạng, hay tài nguyên hệ thống... cần được phát hiện nhanh chóng để có giải pháp sửa chữa, thay thế, phản ứng kịp thời giúp hệ thống mạng hoạt động ổn định, thông suốt.

Trong các hệ thống mạng lớn và phức tạp như hiện nay, các thiết bị, kết nối, dịch vụ, ứng dụng đều được thiết kế mang tính dự phòng cao để sẵn sàng giải quyết khi có sự cố xảy ra. Việc phát hiện kịp thời các thiết bị, các kết nối hư hỏng để tiến hành sửa chữa, thay thế lại càng cấp thiết. Vì khi sự hư hỏng xảy ra một phần, thành phần dự phòng vẫn hoạt động. Nếu thành phần hư hỏng không được phát hiện, xử lý kịp thời sẽ có nguy cơ cao cho hoạt động của hệ thống. Nếu không có công cụ hỗ trợ, người quản trị sẽ bị động trước các tình huống bất thường xảy ra.

10 lý do hàng đầu cho việc cần thiết phải sử dụng hệ thống giám sát mạng:

- Biết được những gì đang xảy ra trên hệ thống: giải pháp giám sát hệ thống cho phép được thông báo tình trạng hoạt động cũng như tài nguyên của hệ thống. Nếu không có những chức năng này ta phải đợi đến khi người dùng thông báo.
- Lên kế hoạch cho việc nâng cấp, sửa chữa: nếu một thiết bị ngưng hoạt động một cách thường xuyên hay băng thông mạng gần chạm tới ngưỡng thì lúc này cần phải có sự thay đổi trong hệ thống. Hệ thống giám sát mạng cho phép ta biết được những thông tin này để có thể có những thay đổi khi cần thiết.
- Chẩn đoán các vấn đề một cách nhanh chóng: giả sử máy chủ của ta không thể kết nối tới được. Nếu không có hệ thống giám sát ta không thể biết được nguyên nhân từ đâu, máy chủ hay router hay cũng có thể là switch. Nếu biết được chính xác vấn đề ta có thể giải quyết một cách nhanh chóng.
- Xem xét những gì đang hoạt động: các báo cáo bằng đồ họa có thể giải thích tình trạng hoạt động của hệ thống. Đó là những công cụ rất tiện lợi phục vụ cho quá trình giám sát.
- Biết được khi nào cần áp dụng các giải pháp sao lưu phục hồi: với đủ các cảnh báo cần thiết ta nên sao lưu dữ liệu của hệ thống phòng trường hợp hệ thống có thể bị hư hại bất kì lúc nào. Nếu không có hệ thống giám sát ta không thể biết có vấn đề xảy ra khi đã quá trễ.
- Đảm bảo hệ thống bảo mật hoạt động tốt: các tổ chức tốn rất nhiều tiền cho hệ thống bảo mật. Nếu không có hệ thống giám sát ta không thể biết hệ thống bảo mật của ta có hoạt động như mong đợi hay không.
- Theo dõi hoạt động của các tài nguyên dịch vụ trên hệ thống: hệ thống giám sát có thể cung cấp thông tin tình trạng các dịch vụ trện hệ thống, đảm bảo người dùng có thể kết nối đến nguồn dữ liệu.
- Được thông báo về tình trạng của hệ thống ở khắp mọi nơi: rất nhiều các úng dụng giám sát cung cấp khả năng giám sát và thông báo từ xa chỉ cần có kết nối Internet.

- Đảm bảo hệ thống hoạt động liên tục: nếu tổ chức của ta phụ thuộc nhiều vào hệ thống mạng, thì tốt nhất là người quản trị cần phải biết và xử lý các vấn đề trước khi sự cố nghiêm trọng xảy ra.
- Tiết kiệm tiền: với tất cả các lý do ở trên, ta có thể giảm thiểu tối đa thời gian hệ thống ngưng hoạt động, làm ảnh hưởng tới lợi nhuận của tổ chức và tiết kiệm tiền cho việc điều tra khi có sự cố xảy ra.

### 1.1.5. Tầm quan trọng của giám sát mạng

Giám sát mạng thực sự là một việc rất cần thiết trong công việc. Không chỉ bởi tính an toàn và bảo mật dữ liệu, giám sát mạng có thể giúp doanh nghiệp tiết kiệm chi phí sửa chữa, giảm thiểu thời gian chết của hệ thống khi gặp sự cố, đảm bảo tính thông suốt trong toàn hệ thống. Những tiêu chí dưới đây sẽ giải thích rõ hơn vì sao giám sát mạng lại là một phần quan trọng đối với các doanh nghiệp:

- Tính bảo mật: Đảm bảo các thông tin không bị lộ ra ngoài. Là một trong những phần quan trọng của giám sát mạng, tính năng này sẽ theo dõi những biến động trong hệ thống mạng và cảnh báo cho quản trị viên biết khi có sự cố xảy ra kịp thời. Thông qua màn hình giám sát, người quản trị có thể xác định được vấn đề khả nghi và tìm cách giải quyết phù hợp nhất cho vấn đề đó.
- Khả năng xử lý sự cố: Khả năng này là một trong các lợi thế của giám sát mạng. Tiết kiệm thời gian chẩn đoán sai lệch trong mạng, giám sát viên có thể biết chính xác thiết bị nào đang có vấn đề và xử lý nó một cách nhanh nhất trước khi người dùng mạng phát hiện.
- Tiết kiệm thời gian và tiền bạc: Nếu không có phần mềm giám sát thì sẽ mất nhiều thời gian để tìm kiếm và sửa lỗi hệ thống mà lẽ ra chỉ mất vài giây để sửa lỗi đó. Điều này không chỉ tốn thêm chi phí mà còn làm giảm năng suất lao động. Ngược lại, nhờ có phần mềm giám sát, vấn đề sẽ nhanh chóng được tìm ra và xử lý hiệu quả, có thể tập trung nhiều hơn vào công việc khác, lợi nhuận công ty cũng gia tăng.

- Lập kế hoạch thay đổi: Với giám sát mạng, giám sát viên có thể theo dõi được thiết bị nào sắp hỏng và cần phải thay mới. Giám sát mạng cho người giám sát khả năng lên kế hoạch sẵn và dễ dàng tạo ra thay đổi cần thiết cho hệ thống mạng.

## 1.2. Những lợi ích của việc xây dựng hệ thống giám sát mạng

- Phát hiện sự cố, kết nối thất bại của hệ thống, dịch vụ hay thiết bị mạng 24/7 và gửi các thông tin tới người quản trị
- Thay thế thiết bị quá tải trước khi nó ảnh hưởng đến hệ thống
- Xác định các điểm thắt cổ chai trong hệ thống
- Tìm ra bất thường trong mạng có thể dẫn đến mối đe dọa an ninh

## 1.3. Ba bài toán của giám sát mạng cần giải quyết

### 1.3.1. Bài toán thứ nhất

Giám sát tài nguyên máy chủ:

- Chúng ta cần giám sát tài nguyên của tất cả máy chủ hàng ngày, hàng giờ để kịp thời phát hiện các máy chủ sắp bị quá tải và đưa ra phương thức giải quyết phù hợp và kịp thời.
- Giám sát tài nguyên máy chủ nghĩa là theo dõi tỷ lệ chiếm dụng CPU, dung lượng còn lại của ổ cứng, tỷ lệ sử dụng bộ nhớ RAM, ....
- Chúng ta không thể kết nối vào từng máy để xem vì số lượng máy nhiều và vì các HĐH khác nhau có cách thức kiểm tra khác nhau.

### 1.3.2. Bài toán thứ hai

Giám sát lưu lượng trên các port của switch, router, giám sát các thiết bị (end devices, switch, router …):

- Chúng ta có hàng ngàn thiết bị mạng (network devices) của nhiều hãng khác nhau, mỗi thiết bị có nhiều port. Chúng cần được giám sát lưu lượng đang truyền qua tất cả các port của các thiết bị suốt 24/24, kịp thời phát hiện các port sắp quá tải.
- Chúng ta cũng không thể kết nối vào từng thiết bị để gõ lệnh lấy thông tin vì thiết bị của các hãng khác nhau có lệnh khác nhau.

### 1.3.3. Bài toán thứ ba

Hệ thống tự động cảnh báo sự cố tức thời. Bạn có hàng ngàn thiết bị mạng và chúng có thể gặp nhiều vấn đề trong quá trình hoạt động như:

Một host hay 1 services nào đó bị mất tín hiệu, có ai đó đã cố kết nối (login) vào thiết bị nhưng nhập sai username và password, thiết bị vừa mới bị khởi động lại (restart) .... Hệ thống cần thông báo sự kiện để người quản trị biết được sự kiện khi nó vừa mới xảy ra.

Để giải quyết các vấn đề trên bạn có thể dùng một ứng dụng phần mềm giám sát được máy chủ, nó sẽ lấy được thông tin từ các máy chủ.

## 1.4. Các quy tắc khi thiết kế hệ thống giám sát mạng

### 1.3.1. Mô hình FCAPS (Fault Configuration Accounting Performance Security)

Một trong những quy tắc khi thiết kế hệ thống giám sát là tuân theo mô hình FCAPS. "Theo tiêu chuẩn của ISO (International Standard Organization), mô hình được phân loại thành 5 chức năng chính, đó là chức năng quản lý lỗi (Fault management), quản lý cấu hình (Configuration management), quản lý kế toán (Accounting management), quản lý hiệu năng (Performance management) và quản lý bảo mật (Security management)" [1].

- Quản lý lỗi: Hạng mục này có thể thực hiện quá trình ghi nhận, cô lập và xử lý lỗi xảy ra trên mạng. Việc xác định những vấn đề tiềm ẩn trong mạng cũng do hạng mục này đảm nhiệm.
- Quản lý cấu hình: Giúp thu thập và lưu trữ các cấu hình của vô số thiết bị, bao gồm việc lần ra những thay đổi cấu hình trên thiết bị, góp phần quan trọng trong việc chủ động quản trị và giám sát mạng.
- Quản lý kế toán: Thường áp dụng cho các nhà cung cấp dịch vụ mạng. Trong hệ thống mạng, công việc này được thay bằng việc quản lý người dùng mạng, nói cách khác, quản trị viên sẽ cấp cho người dùng mật khẩu, quyền để vào mạng.
- Quản lý hiệu năng: Quản lý toàn bộ hiệu năng của mạng, tốc độ truyền, thông lượng truyền, những gói tin bị mất, thời gian phản hồi, v.v. và thường sử dụng bằng giao thức SNMP.

- Quản lý bảo mật: Là một hoạt động rất quan trọng trong quản trị mạng. Quản lý bảo mật trong FCAPS bao gồm quá trình kiểm soát truy cập tài nguyên trên mạng, kèm theo các dữ liệu, cấu hình và bảo vệ thông tin người dùng.

### 1.3.2. Báo cáo và cảnh báo

Công việc của giám sát mạng là thu thập dữ liệu từ các thành phần mạng và xử lý, trình bày chúng dưới dạng mà quản trị viên có thể hiểu - tiến trình này được gọi là báo cáo. Báo cáo giúp quản trị viên biết được hiệu suất của các nút mạng, trạng thái mạng hiện tại. Với các dữ liệu từ bản báo cáo, quản trị viên có thể đưa ra quyết định về việc quản lý dung lượng, bảo trì mạng, xử lý sự cố hay bảo mật mạng.

Tuy nhiên, việc làm này không giúp quản trị viên bảo trì mạng ở hiệu suất cao. Vì thế, việc tạo các cảnh báo dựa trên ngưỡng cùng các điểm kích hoạt sẽ là nhân tố bổ sung giúp các nhà quản trị xác định các vấn đề có thể xảy ra trước khi nó gây sụp đổ toàn hệ thống.

### 1.3.3. Tích hợp lưu trữ dữ liệu

Hệ thống giám sát thu thập và dùng dữ liệu từ các thành phần mạng cho các chức năng liên quan. Trong khi đó, mạng vẫn tiếp tục giám sát để đảm bảo vấn đề sẽ được phát hiện trước khi mạng bị sập. Việc tiếp tục công việc như vậy sẽ tích lũy một lượng lớn dữ liệu và nó có thể làm chậm hiệu suất, tác động đến không gian lưu trữ dữ liệu hay làm chậm việc xử lý sự cố, giám sát hệ thống sử dụng dữ liệu tích hợp là để tránh những việc như vậy xảy ra. Tích hợp dữ liệu là một quá trình thu thập dữ liệu theo thời gian đã được tổng hợp và gói gọn để dữ liệu trở thành dạng chi tiết. Mức độ chi tiết của bản báo cáo được tạo ra bởi dữ liệu tích hợp sẽ phụ thuộc vào mô hình mà hệ thống được tích hợp. Dữ liệu sẽ được lấy trung bình theo thời gian và đưa vào bảng dữ liệu chi tiết, điều này giúp hệ thống giám sát tạo ra các bản báo cáo về các nút có thể kéo dài khoảng thời gian trong mạng mà không gây ra các vấn đề về hiệu suất hay không gian lưu trữ.

## 1.5. Các giải pháp và công cụ giám sát mạng phổ biến

Hệ thống giám sát mạng có thể được xây dựng theo một trong ba giải pháp sau:

- Giải pháp quản lý thông tin an ninh: tập trung thu thập, lưu trữ và biểu diễn nhật ký.
- Giải pháp quản lý sự kiện an ninh: tập trung xử lý, phân tích các nhật ký đã được thu thập để đưa ra cảnh báo cho người dùng.
- Giải pháp quản lý và phân tích sự kiện an ninh: là sự kết hợp của hai giải pháp trên nhằm khắc phục những hạn chế vốn có.

Mô hình của giải pháp quản lý và phân tích sự kiện an ninh gồm 3 thành phần chính [2]:

a) Thu thập nhật ký an toàn mạng bao gồm các giao diện thu thập nhật ký trực tiếp từ các thiết bị, ứng dụng và dịch vụ. Thành phần này có tính năng:

- Thu thập toàn bộ dữ liệu toàn bộ nhật ký từ các nguồn thiết bị, ứng dụng.
- Kiểm soát băng thông và không gian lưu trữ thông qua khả năng lưu giữ và chọn lọc dữ liệu nhật ký.
- Phân tách từng sự kiện và chuẩn hóa các sự kiện vào một lược đồ chung.
- Tích hợp các sự kiện để giảm thiểu số lượng các sự kiện gửi về thành phần phân tích và lưu trữ.
- Chuyển toàn bộ các sự kiện đã thu thập về thành phần phân tích và lưu trữ.

b) Thành phần phân tích và lưu trữ bao gồm các thiết bị lưu trữ dung lượng lớn, cung cấp khả năng tổng hợp và phân tích tự động. Tính năng:

- Kết nối với các thành phần thu thập nhật ký để tập hợp nhật ký tập trung và tiến hành phân tích, so sánh tương quan.
- Module phân tích sẽ được hỗ trợ bởi các luật (định nghĩa trước) cũng như khả năng tùy biến, nhằm đưa ra kết quả phân tích chính xác nhất.
- Các nhật ký an toàn mạng được tiến hành phân tích, so sánh tương quan theo thời gian thực nhằm đưa ra cảnh báo tức thời cho người quản trị.
- Hỗ trợ kết nối đến các hệ thống lưu trữ dữ liệu.

c) Thành phần quản trị mạng tập trung:

- Cung cấp giao diện quản trị tập trung cho toàn bộ hệ thống giám sát an toàn mạng.
- Hỗ trợ sẵn hàng nghìn mẫu báo cáo, các giao diện theo dõi, điều kiện lọc.
- Hỗ trợ các công cụ cho việc xử lý các sự kiện an toàn mạng xảy ra trong hệ thống.

d) Các thành phần khác:

Gồm các thành phần cảnh báo, hệ thống Dashboard theo dõi thông tin, các báo cáo đáp ứng tiêu chuẩn quản lý hoặc thành phần lưu trữ dữ liệu lâu dài.

### 1.6. Chi phí sử dụng

Tùy theo chính sách và trang thiết bị hạ tầng, hệ thống mạng thực tế của từng doanh nghiệp mà người người quản trị sẽ quyết định sử dụng phần mềm phù hợp với hệ thống giám sát của mình.

***Đối với các doanh nghiệp lớn:*** đã xây dựng nền tảng hạ tầng sử dụng các thiết bị của các hãng lớn như Cisco, HP thì nên ưu tiên sử dụng các giải pháp phần mềm giám sát của các hãng này như HP Network Node Manager, Cisco Works… để nhận được sự hỗ trợ tốt nhất từ các chuyên gia của hãng.

***Đối với các doanh nghiệp vừa và nhỏ***: với khoản kinh phí ít hơn, thì việc ưu tiên sử dụng các phần mềm giám sát mã nguồn mở là điều cần thiết. Các phần mềm này được nhiều tổ chức cộng đồng mã nguồn mở phát triển với tính năng giám sát mạnh, nhận diện các vấn đề trước khi phát sinh, khả năng tùy biến cao và được cung cấp hoàn toàn miễn phí.

# CHƯƠNG 2. GIAO THỨC HỖ TRỢ VÀ CÁC PHẦN MỀM GIÁM SÁT MẠNG

## 2.1. Giao thức giám sát mạng SNMP

### 2.2.1. Khái niệm

SNMP – Simple Network Management Protocol (Giao thức quản lý mạng đơn giản). Về bản chất SNMP là một tập các thao tác cho phép người quản trị hệ thống có thể thay đổi trạng thái của các thiết bị (có hỗ trợ SNMP). Ví dụ, ta có thể sử dụng SNMP để tắt một interface nào đó trên router của mình, theo dõi hoạt động của card Ethernet, hoặc kiểm soát nhiệt độ trên switch và cảnh báo khi nhiệt độ quá cao.

Một thiết bị hiểu được và hoạt động theo giao thức SNMP được gọi là "có hỗ trợ SNMP" (SNMP supported) hoặc "tương thích SNMP" (SNMP compartible).

SNMP dùng để quản lý nghĩa là: có thể theo dõi, lấy thông tin, được thông báo, và có thể tác động để hệ thống hoạt động như ý muốn. Ví dụ một số khả năng của phần mềm SNMP:

+ Theo dõi tốc độ đường truyền của một router, biết được tổng số byte đã truyền/nhận.
+ Lấy thông tin máy chủ đang có bao nhiêu ổ cứng, mỗi ổ cứng còn trống bao nhiêu.
+ Tự động nhận cảnh báo khi switch có một port bị down.
+ Điều khiển tắt (shutdown) các port trên switch.

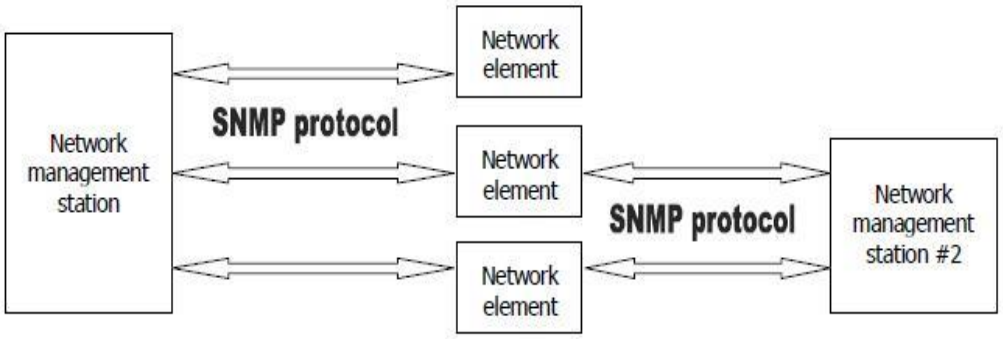
SNMP được thiết kế để chạy trên nền TCP/IP và quản lý các thiết bị có nối mạng TCP/IP. Các thiết bị mạng không nhất thiết phải là máy tính mà có thể là switch, router, firewall, ADSL gateway, và cả một số phần mềm cho phép quản trị bằng SNMP.

SNMP là giao thức đơn giản, do nó được thiết kế đơn giản trong cấu trúc bản tin và thủ tục hoạt động, và còn đơn giản trong bảo mật (ngoại trừ SNMP version 3). Sử dụng phần mềm SNMP, người quản trị mạng có thể quản lý, giám sát tập trung từ xa toàn mạng của mình [7].

### 2.2.2. Các thành phần trong SNMP

Theo RFC1157, kiến trúc của SNMP bao gồm 2 thành phần: các trạm quản lý mạng (network management station) và các thành tố mạng (network element).

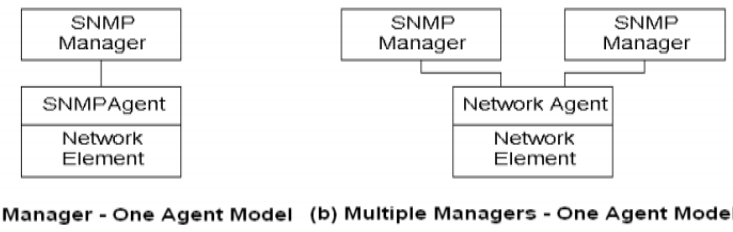
Network management station thường là một máy tính chạy phần mềm quản lý SNMP (SNMP management application), dùng để giám sát và điều khiển tập trung các network element.




*Hình 2.2.2.1: Kiến trúc của SNMP*

Network element là các thiết bị, máy tính, hoặc phần mềm tương thích SNMP và được quản lý bởi network management station. Như vậy element bao gồm device, host và application.

Một management station có thể quản lý nhiều element, một element cũng có thể được quản lý bởi nhiều management station. Vậy nếu một element được quản lý bởi 2 station thì điều gì sẽ xảy ra? Nếu station lấy thông tin từ element thì cả 2 station sẽ có thông tin giống nhau. Nếu 2 station tác động đến cùng một element thì element sẽ đáp ứng cả 2 tác động theo thứ tự cái nào đến trước.




*Hình 2.2.2.2: Quan hệ giữa Network management station và Network Element*

Khái niệm SNMP agent: SNMP agent là một tiến trình (process) chạy trên network element, có nhiệm vụ cung cấp thông tin của element cho station, nhờ đó station có thể quản lý được element. Nói cách khác, Application chạy trên station và agent chạy trên element là 2 tiến trình SNMP trực tiếp liên hệ với nhau. Các ví dụ minh họa sau đây sẽ làm rõ hơn các khái niệm này [3]:

+ Để dùng một máy chủ (= station) quản lý các máy con (= element) chạy HĐH Windows thông qua SNMP thì bạn phải: cài đặt một phần mềm quản lý SNMP (= application) trên máy chủ, bật SNMP service (= agent) trên máy con.

+ Để dùng một máy chủ (= station) giám sát lưu lượng của một router (= element) thì bạn phải : cài phần mềm quản lý SNMP (= application) trên máy chủ, bật tính năng SNMP (= agent) trên router.

**a. Object ID**

Một thiết bị hỗ trợ SNMP có thể cung cấp nhiều thông tin khác nhau, mỗi thông tin đó gọi là một object. Ví dụ:

+ Máy tính có thể cung cấp các thông tin : tổng số ổ cứng, tổng số port nối mạng, tổng số byte đã truyền/nhận, tên máy tính, tên các process đang chạy, ….

+ Router có thể cung cấp các thông tin : tổng số card, tổng số port, tổng số byte đã truyền/nhận, tên router, tình trạng các port của router, ….

Mỗi object có một tên gọi và một mã số để nhận dạng object đó, mã số gọi là Object ID (OID).

**Ví dụ:**

+ Tên thiết bị được gọi là sysName, OID là 1.3.6.1.2.1.1.5
+ Tổng số port giao tiếp (interface) được gọi là ifNumber, OID là 1.3.6.1.2.1.2.1.
+ Địa chỉ Mac Address của một port được gọi là ifPhysAddress, OID là 1.3.6.1.2.1.2.2.1.6.
+ Số byte đã nhận trên một port được gọi là ifInOctets, OID là 1.3.6.1.2.1.2.2.1.10.

Một object có thể có nhiều giá trị cùng loại. Chẳng hạn một thiết bị có thể có nhiều tên, có nhiều Mac address. Một object chỉ có một OID, vì vậy để chỉ ra các giá trị khác nhau của cùng một object thì ta dùng thêm một phân cấp nữa: sub-id. Ví dụ:

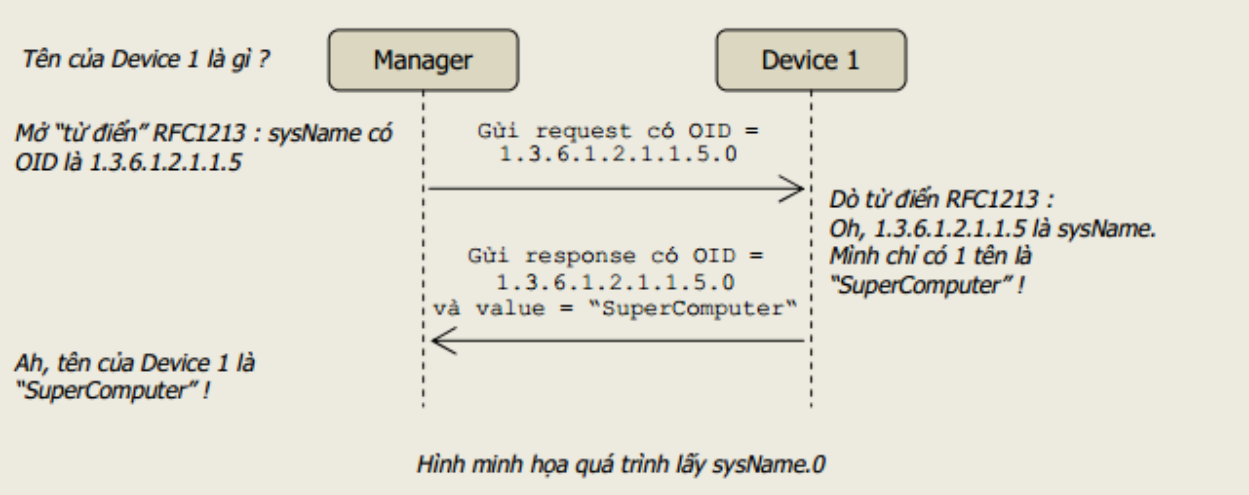
+ Tên thiết bị được gọi là sysName, OID là 1.3.6.1.2.1.1.5; nếu thiết bị có 2 tên thì chúng sẽ được gọi là sysName.0 & sysName.1 và có OID lần lượt là 1.3.6.1.2.1.1.5.0 & 1.3.6.1.2.1.1.5.1.
+ Địa chỉ Mac address được gọi là ifPhysAddress, OID là 1.3.6.1.2.1.2.2.1.6; nếu thiết bị có 2 mac address thì chúng sẽ được gọi là ifPhysAddress.0 & ifPhysAddress.1 và có OID lần lượt là 1.3.6.1.2.1.2.2.1.6.0 & 1.3.6.1.2.1.2.2.1.6.1.
+ Tổng số port được gọi là ifNumber, giá trị này chỉ có 1 (duy nhất) nên OID của nó không có phân cấp con và vẫn là 1.3.6.1.2.1.2.1.

Các object có thể có nhiều giá trị hoặc 1 giá trị thì luôn luôn được viết dưới dạng có phân cấp con sub-id. Ví dụ một thiết bị dù chỉ có 1 tên thì nó vẫn phải viết là sysName.0 hay 1.3.6.1.2.1.1.5.0.

Đối với các object có nhiều giá trị thì các chỉ số của phân cấp con không nhất thiết phải liên tục hay bắt đầu từ 0. Ví dụ một thiết bị có 2 mac address thì có thể chúng được gọi là ifPhysAddress.23 và ifPhysAddress.125645.

OID của các object phổ biến có thể được chuẩn hóa, hoặc tự định nghĩa. Để lấy một thông tin có OID đã chuẩn hóa thì SNMP application phải gửi một bản tin SNMP có chứa OID của object đó cho SNMP agent, SNMP agent khi nhận được thì nó phải trả lời bằng thông tin ứng với OID đó.

**Ví dụ:** Muốn lấy tên của một PC chạy Windows, tên của một PC chạy Linux hoặc tên của một router thì SNMP application chỉ cần gửi bản tin có chứa OID là 1.3.6.1.2.1.1.5.0. Khi SNMP agent chạy trên PC Windows, PC Linux hay router nhận được bản tin có chứa OID 1.3.6.1.2.1.1.5.0, agent lập tức hiểu rằng đây là bản tin hỏi sysName.0, và agent sẽ trả lời bằng tên của hệ thống. Nếu SNMP agent nhận được một OID mà nó không hiểu (không hỗ trợ) thì nó sẽ không trả lời.




*Hình 2.2.2.3: Hình minh hoạ quá trình trao đổi*

Một trong các ưu điểm của SNMP là nó được thiết kế để chạy độc lập với các thiết bị khác nhau. Chính nhờ việc chuẩn hóa OID mà ta có thể dùng một SNMP application để lấy thông tin các loại device của các hãng khác nhau.

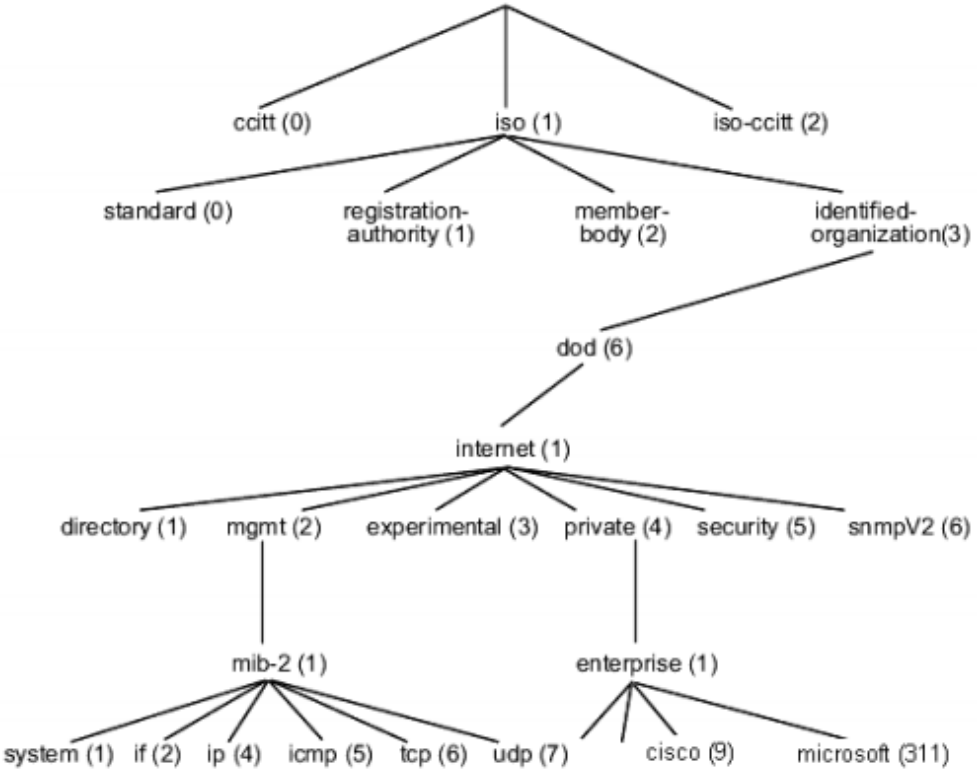
**b. Object Access**

Mỗi object có quyền truy cập là READ_ONLY hoặc READ_WRITE. Mọi object đều có thể đọc được nhưng chỉ những object có quyền READ_WRITE mới có thể thay đổi được giá trị.

**Ví dụ:** Tên của một thiết bị (sysName) là READ_WRITE, có thể thay đổi tên của thiết bị thông qua giao thức SNMP. Tổng số port của thiết bị (ifNumber) là READ_ONLY và không thể thay đổi số port của nó.

**c. Management Information Base**

MIB (cơ sở thông tin quản lý) là một cấu trúc dữ liệu gồm các đối tượng được quản lý (managed object), được dùng cho việc quản lý các thiết bị chạy trên nền TCP/IP. MIB là kiến trúc chung mà các giao thức quản lý trên TCP/IP nên tuân theo, trong đó có SNMP. MIB được thể hiện thành 1 file (MIB file), và có thể biểu diễn thành 1 cây (MIB tree). MIB có thể được chuẩn hóa hoặc tự tạo [3].




*Hình 2.2.2.4: Minh họa MIB tree*

***<u>Một node trong cây là một object, có thể được gọi bằng tên hoặc id. Ví dụ:</u>***

- Node iso.org.dod.internet.mgmt.mib-2.system có OID là 1.3.6.1.2.1.1, chứa tất cả các object liên quan đến thông tin của một hệ hống như tên của thiết bị (iso.org.dod.internet.mgmt.mib-2.system.sysName hay 1.3.6.1.2.1.1.5).
- Các OID của các hãng tự thiết kế nằm dưới iso.org.dod.internet.private.enterprise.

**Ví dụ:** Cisco nằm dưới iso.org.dod.internet.private.enterprise.cisco hay 1.3.6.1.4.1.9, Microsoft nằm dưới iso.org.dod.internet.private.enterprise.microsoft hay 1.3.6.1.4.1.311. Số 9 (Cisco) hay 311 (Microsoft) là số dành riêng cho các công ty do IANA cấp. Nếu Cisco hay Microsoft chế tạo ra một thiết bị nào đó, thì thiết bị này có thể hỗ trợ các MIB chuẩn đã được định nghĩa sẵn (như mib-2) hay hỗ trợ MIB được thiết kế riêng. Các MIB được công ty nào thiết kế riêng thì phải nằm bên dưới OID của công ty đó.

Các objectID trong MIB được sắp xếp thứ tự nhưng không phải là liên tục, khi biết một OID thì không chắc chắn có thể xác định được OID tiếp theo trong MIB. Ví dụ trong chuẩn mib-2 thì object ifSpecific và object atIfIndex nằm kề nhau nhưng OID lần lượt là 1.3.6.1.2.1.2.2.1.22 và 1.3.6.1.2.1.3.1.1.1.

Muốn hiểu được một OID nào đó thì bạn cần có file MIB mô tả OID đó. Một MIB file không nhất thiết phải chứa toàn bộ cây ở trên mà có thể chỉ chứa mô tả cho một nhánh con. Bất cứ nhánh con nào và tất cả lá của nó đều có thể gọi là một MIB.

Một manager có thể quản lý được một device chỉ khi ứng dụng SNMP manager và ứng dụng SNMP agent cùng hỗ trợ một MIB. Các ứng dụng này cũng có thể hỗ trợ cùng lúc nhiều MIB

**d. Các phương thức của SNMP**

Giao thức SNMP có 5 phương thức hoạt động cơ bản, tương ứng với 5 loại bản tin như sau:

| Bản tin/phương thức | Mô tả tác dụng |
|---|---|
| GetRequest | Manager gửi GetRequest cho agent để yêu cầu agent cung cấp thông tin nào đó dựa vào ObjectID (trong GetRequest có chứa OID) |
| GetNextRequest | Manager gửi GetNextRequest có chứa một ObjectID cho agent để yêu cầu cung cấp thông tin nằm kế tiếp ObjectID đó trong MIB. |
| SetRequest | Manager gửi SetRequest cho agent để đặt giá trị cho đối tượng của agent dựa vào ObjectID. |
| GetResponse | Agent gửi GetResponse cho Manager để trả lời khi nhận được GetRequest/GetNextRequest |
| Trap | Agent tự động gửi Trap cho Manager khi có một sự kiện xảy ra đối với một object nào đó trong agent. |

*Hình 2.2.2.5: Bảng các phương thức cơ bản của SNMP*

Mỗi bản tin đều có chứa OID để cho biết object mang trong nó là gì. OID trong GetRequest cho biết nó muốn lấy thông tin của object nào. OID trong GetResponse cho biết nó mang giá trị của object nào. OID trong SetRequest chỉ ra nó muốn thiết lập giá trị cho object nào. OID trong Trap chỉ ra nó thông báo sự kiện xảy ra đối với object nào.

- **GetRequest**

Bản tin GetRequest được manager gửi đến agent để lấy một thông tin nào đó. Trong GetRequest có chứa OID của object muốn lấy. VD: Muốn lấy thông tin tên của Device 1 thì manager gửi bản tin GetRequest OID=1.3.6.1.2.1.1.5 đến Device 1, tiến trình SNMP agent trên Device 1 sẽ nhận được bản tin và tạo bản tin trả lời. Trong một bản tin GetRequest có thể chứa nhiều OID, nghĩa là dùng một GetRequest có thể lấy về cùng lúc nhiều thông tin.

- **GetNextRequest**

Bản tin GetNextRequest cũng dùng để lấy thông tin và cũng có chứa OID, tuy nhiên nó dùng để lấy thông tin của object nằm kế tiếp object được chỉ ra trong bản tin.

Tại sao phải có phương thức GetNextRequest? Như bạn đã biết khi đọc qua những phần trên: một MIB bao gồm nhiều OID được sắp xếp thứ tự nhưng không liên tục, nếu biết một OID thì không xác định được OID kế tiếp. Do đó ta cần GetNextRequest để lấy về giá trị của OID kế tiếp. Nếu thực hiện GetNextRequest liên tục thì ta sẽ lấy được toàn bộ thông tin của agent.

- **SetRequest**

Bản tin SetRequest được manager gửi cho agent để thiết lập giá trị cho một object nào đó. Ví dụ:

+ Có thể đặt lại tên của một máy tính hay router bằng phần mềm SNMP manager, bằng cách gửi bản tin SetRequest có OID là 1.3.6.1.2.1.1.5.0 (sysName.0) và có giá trị là tên mới cần đặt.

+ Có thể shutdown một port trên switch bằng phần mềm SNMP manager, bằng cách gửi bản tin có OID là 1.3.6.1.2.1.2.2.1.7 (ifAdminStatus) và có giá trị là $2^7$.

Chỉ những object có quyền READ_WRITE mới có thể thay đổi được giá trị.

- **GetResponse**

Mỗi khi SNMP agent nhận được các bản tin GetRequest, GetNextRequest hay SetRequest thì nó sẽ gửi lại bản tin GetResponse để trả lời. Trong bản tin GetResponse có chứa OID của object được request và giá trị của object đó.

- **Trap**

Bản tin Trap được agent tự động gửi cho manager mỗi khi có sự kiện xảy ra bên trong agent, các sự kiện này không phải là các hoạt động thường xuyên của agent mà là các sự

kiện mang tính biến cố. Ví dụ: Khi có một port down, khi có một người dùng login không thành công, hoặc khi thiết bị khởi động lại, agent sẽ gửi trap cho manager.

Tuy nhiên không phải mọi biến cố đều được agent gửi trap, cũng không phải mọi agent đều gửi trap khi xảy ra cùng một biến cố. Việc agent gửi hay không gửi trap cho biến cố nào là do hãng sản xuất device/agent quy định.

Phương thức trap là độc lập với các phương thức request/response. SNMP request/response dùng để quản lý còn SNMP trap dùng để cảnh báo. Nguồn gửi trap gọi là Trap Sender và nơi nhận trap gọi là Trap Receiver. Một trap sender có thể được cấu hình để gửi trap đến nhiều trap receiver cùng lúc.

Có 2 loại trap: trap phổ biến (generic trap) và trap đặc thù (specific trap). Generic trap được quy định trong các chuẩn SNMP, còn specific trap do người dùng tự định nghĩa (người dùng ở đây là hãng sản xuất SNMP device).

### 2.2.3. Các cơ chế bảo mật cho SNMP

Một SNMP management station có thể quản lý/giám sát nhiều SNMP element, thông qua hoạt động gửi request và nhận trap. Tuy nhiên một SNMP element có thể được cấu hình để chỉ cho phép các SNMP management station nào đó được phép quản lý/giám sát mình. Các cơ chế bảo mật đơn giản này gồm có: community string, view và SNMP access control list.

**a. Community string**

Community string là một chuỗi ký tự được cài đặt giống nhau trên cả SNMP manager và SNMP agent, đóng vai trò như "mật khẩu" giữa 2 bên khi trao đổi dữ liệu. Community string có 3 loại: Read-community, Write-Community và Trap-Community.

Khi manager gửi GetRequest, GetNextRequest đến agent thì trong bản tin gửi đi có chứa Read- Community. Khi agent nhận được bản tin request thì nó sẽ so sánh Read-community do manager gửi và Read-community mà nó được cài đặt. Nếu 2 chuỗi này giống nhau, agent sẽ trả lời; nếu 2 chuỗi này khác nhau, agent sẽ không trả lời.

Write-Community được dùng trong bản tin SetRequest. Agent chỉ chấp nhận thay đổi dữ liệu khi writecommunity 2 bên giống nhau.

Trap-community nằm trong bản tin trap của trap sender gửi cho trap receiver. Trap receiver chỉ nhận và lưu trữ bản tin trap chỉ khi trap-community 2 bên giống nhau, tuy nhiên cũng có nhiều trap receiver được cấu hình nhận tất cả bản tin trap mà không quan tâm đến trap-community.

Community string có 3 loại như trên nhưng cùng một loại có thể có nhiều string khác nhau. Nghĩa là một agent có thể khai báo nhiều read-community, nhiều write-community.

Trên hầu hết hệ thống, read-community mặc định là "public", write-community mặc định là "private" và trap-community mặc định là "public".

Community string chỉ là chuỗi ký tự dạng cleartext, do đó hoàn toàn có thể bị nghe lén khi truyền trên mạng. Hơn nữa, các community mặc định thường là "public" và "private" nên nếu người quản trị không thay đổi thì chúng có thể dễ dàng bị dò ra. Khi community string trong mạng bị lộ, một người dùng bình thường tại một máy tính nào đó trong mạng có thể quản lý/giám sát toàn bộ các device có cùng community mà không được sự cho phép của người quản trị.

**b. View**

Khi manager có read-community thì nó có thể đọc toàn bộ OID của agent. Tuy nhiên agent có thể quy định chỉ cho phép đọc một số OID có liên quan nhau, tức là chỉ đọc được một phần của MIB. Tập con của MIB này gọi là view, trên agent có thể định nghĩa nhiều view. Ví dụ : agent có thể định nghĩa view interfaceView bao gồm các OID liên quan đến interface, storageView bao gồm các OID liên quan đến lưu trữ, hay AllView bao gồm tất cả các OID.

Một view phải gắn liền với một community string. Tùy vào community string nhận được là gì mà agent xử lý trên view tương ứng. Ví dụ : agent định nghĩa read-community "inf" trên view interfaceView, và "sto" trên storageView; khi manager gửi request lấy OID ifNumber với community là "inf" thì sẽ được đáp ứng do ifNumber nằm trong interfaceView; nếu manager request OID hrStorageSize với community "inf" thì agent sẽ không trả lời do hrStorageSize không nằm trong interfaceView; nhưng nếu manager

request hrStorageSize với community "sto" thì sẽ được trả lời do hrStorageSize nằm trong storageView.

Việc định nghĩa các view như thế nào tùy thuộc vào từng SNMP agent khác nhau. Có nhiều hệ thống không hỗ trợ tính năng view.

**c. SNMP access control list.**

Khi manager gửi không đúng community hoặc khi OID cần lấy lại không nằm trong view cho phép thì agent sẽ không trả lời. Tuy nhiên khi community bị lộ thì một manager nào đó vẫn request được thông tin. Để ngăn chặn hoàn toàn các SNMP manager không được phép, người quản trị có thể dùng đến SNMP access control list (ACL).

SNMP ACL là một danh sách các địa chỉ IP được phép quản lý/giám sát agent, nó chỉ áp dụng riêng cho giao thức SNMP và được cài trên agent. Nếu một manager có IP không được phép trong ACL gửi request thì agent sẽ không xử lý, dù request có community string là đúng.

## 2.2. Hai phương thức giám sát cơ bản Poll và Alert

Trước khi giới thiệu về các phần mềm, chúng ta cần tìm hiểu về hai phương thức giám sát "Poll" và "Alert". Đây là 2 phương thức cơ bản của các kỹ thuật giám sát hệ thống, nhiều phần mềm và giao thức được xây dựng dựa trên 2 phương thức này, trong đó có SNMP, Zabbix … Việc hiểu rõ hoạt động của Poll & Alert và ưu nhược điểm của chúng sẽ giúp bạn dễ dàng tìm hiểu nguyên tắc hoạt động của các giao thức hay phần mềm giám sát khác [8].

### 2.2.1. Phương thức Poll

**Nguyên tắc hoạt động:** Trung tâm giám sát (manager) sẽ thường xuyên hỏi thông tin của thiết bị cần giám sát (device). Nếu Manager không hỏi thì Device không trả lời, nếu Manager hỏi thì Device phải trả lời. Bằng cách hỏi thường xuyên, Manager sẽ luôn cập nhật được thông tin mới nhất từ Device.

**Ví dụ:** Người quản lý cần theo dõi khi nào thợ làm xong việc. Anh ta cứ thường xuyên hỏi người thợ "Anh đã làm xong chưa?", và người thợ sẽ trả lời "Xong" hoặc "Chưa".

### 2.2.2. Phương thức Alert

**Nguyên tắc hoạt động:** Mỗi khi trong Device xảy ra một sự kiện (event) nào đó thì Device sẽ tự động gửi thông báo cho Manager, gọi là Alert. Manager không hỏi thông tin định kỳ từ Device.

**Ví dụ:** 1 máy chủ bị down do mất nguồn, thì lập tức sẽ có 1 thông báo về cho server giám sát để người quản trị có thể xử lý kịp thời.

## 2.3. Các phần mềm giám sát hệ thống mạng

### 2.3.1. Phần mềm giám sát mạng Cacti

Cacti là một phần mềm mã nguồn mở, giám sát mạng và công cụ đồ họa viết bằng ngôn ngữ PHP/MySQL. Phần mềm giám sát hệ thống bằng đồ thị dựa trên bộ công cụ RRDTool. Cacti cung cấp cho người quản trị các mẫu đồ thị, các phương thức tổng hợp dữ liệu và công cụ quản lý. Phần mềm giám sát các thiết bị như ổ cứng, tốc độ quạt, điện năng theo thời gian thực. Điều đó sẽ giúp ích rất nhiều cho việc quản trị hệ thống. Hơn nữa, phần mềm còn cho phép quản lý phân quyền người dùng đối với dữ liệu đang giám sát, đưa ra các cảnh bảo khi hệ thống gặp sự cố bằng việc gửi thư điện tử, tin nhắn và rất nhiều tính năng khác.

Phần mềm Cacti cài đặt dễ dàng và hỗ trợ các hệ điều hành Linux (Centos, Fedora, Red Hat, OpenSUSE, Ubuntu…) và hệ điều hành Windows (Windows XP, Windows Server 2003, Windows Server 2008, Windows 7, Windows 8…)

***Chính sách bản quyền:*** Phần mềm cung cấp phiên bản miễn phí, hỗ trợ các hệ thống nhỏ và cả các hệ thống doanh nghiệp.

***Ưu điểm:*** Phần mềm được cung cấp miễn phí, hỗ trợ tính năng hiển thị thông tin bằng đô thị. Phần mềm cài đặt dễ dàng và hỗ trợ nhiều hệ điều hành. Giao diện thân thiện, dễ sử dụng cho người dùng lần đầu tiên.

***Nhược điểm:*** Phần mềm cung cấp ít tùy chọn quản trị hơn so với các phần mềm giám sát khác.

### 2.3.2. Phần mềm giám sát mạng Icinga

Phần mềm Icinga là một hệ thống mã nguồn mở có chức năng giám sát hệ thống mạng, các máy chủ, các dịch vụ, thông báo tới người dùng khi hệ thống có sự cố và đưa ra các báo cáo kịp thời. Phần mềm Icinga được xây dựng dựa trên mã nguồn được phát triển từ hệ thống giám sát Nagios. Thừa hưởng các tính năng quan trọng của "Người tiền nhiệm" Nagios, vì vậy nó tương thích hoàn toàn với các phần mềm hỗ trợ của Nagios. Đồng thời, phần mềm cũng cung cấp rất nhiều tính năng tùy biến mới, trong đó phải kể đến như giao diện người dùng Web 2.0, hỗ trợ các hệ quản trị cơ sở dữ liệu phổ biến như MySQL, Oracle và PorgreSQL. Phần mềm chạy trên nhiều phiên bản của Linux (Bao gồm Fedora, Ubuntu và OpenSuSE) cũng như một số các nền tảng của Unix (Solaris và HP).

***Chính sách bản quyền:*** Phần mềm cung cấp phiên bản miễn phí, hỗ trợ các hệ thống nhỏ và cả các hệ thống doanh nghiệp.

***Ưu điểm:*** Phần mềm được cung cấp miễn phí, hỗ trợ nhiều tùy chọn giao diện quản trị Web. Phần mềm cài đặt dễ dàng, hỗ trợ tốt hệ điều hành Linux. Giao diện quản trị Web thân thiện, dễ sử dụng cho người dùng lần đầu. Tương thích với các phần mềm hỗ trợ của Nagios.

***Nhược điểm:*** Phần mềm không cung cấp nhiều tùy chọn hiển thị thông tin giám sát bằng đồ thị.

### 2.3.3. Phần mềm giám sát mạng Nagios

Nagios là một phần mềm mã nguồn mở giám sát hệ thống mạng. Phần mềm thực hiện theo dõi và đưa ra các cảnh báo về trạng thái các máy chủ và các dịch vụ. Phần mềm được xây dựng trên nền tảng Linux nên hỗ trợ hầu hết các hệ điều hành của Linux. Một điểm khác so với các phần mềm giám sát là Nagios giám sát dựa trên tình

trạng hoạt động của các máy trạm và các dịch vụ. Nagios sử dụng các phần mềm hỗ trợ được cài đặt trên máy trạm, thực hiện kiểm tra các máy trạm và dịch vụ định kỳ. Tiếp đo, các thông tin của các máy trạm và dịch vụ sẽ được gửi về máy chủ Nagios và được hiển thị trên giao diện web. Đông thời, trong trường hợp hệ thống gặp sự cố, Nagios sẽ gửi các thông tin trạng thái hệ thống tới người quản trị thông qua thư điện tử, tin nhắn… Việc theo dõi có thể được cấu hình chủ động hoặc bị động dựa trên mục đích sử dụng của người quản trị.

***Chính sách bản quyền:***Phần mềm cung cấp 02 phiên bản miễn phí và trả phí, hỗ trợ các hệ thống nhỏ và cả các hệ thống doanh nghiệp.

***Ưu điểm:*** Phần mềm cung cấp phiên bản miễn phí, hỗ trợ rất nhiều chức năng hữu ích cho người quản trị. Các phần mềm hỗ trợ nhiều và được cung cấp miễn phí.

***Nhược điểm:*** Việc cài đặt, cấu hình phần mềm khá phức tạp và yêu cầu kiến thức về hệ điều hành Linux cũng như sự hỗ trợ của các tài liệu cài đặt. Giao diện sử dụng khá phức tạp, kho tiếp cận với người sử dụng lần đầu.

# CHƯƠNG 3. GIỚI THIỆU PHẦN MỀM NGUỒN MỞ GIÁM SÁT MẠNG ZABBIX

## 3.1. Giới thiệu phần mềm zabbix

### 3.3.1. Khái niệm

Zabbix được sáng lập bởi Alexei Vladishev, hiện tại được hỗ trợ và phát triển bởi Zabbix SIA.

Zabbix là công cụ mã nguồn mở giải quyết vấn đề giám sát. Zabbix là phần mềm liệt kê các tham số của một mạng, tình trạng và tính toàn vẹn của server, router, switch... Zabbix sử dụng một cơ chế thông báo linh hoạt các thông tin của các thành phần mạng cho phép người dùng cấu hình email cảnh báo cho sự kiện bất kỳ. Điều này cho phép giải quyết nhanh các vấn đề của hạ tầng mạng. Zabbix cung cấp báo cáo và dữ liệu chính xác dựa trên cơ sở dữ liệu. Điều này làm cho Zabbix trở nên lý tưởng hơn.

Tất cả các báo cáo, thống kê cũng như các thông số cấu hình của Zabbix được truy cập thông qua giao diện web. Giao diện giúp ta theo dõi được tình trạng hệ mạng và server. Zabbix đóng một vai trò quan trọng trong việc theo dõi cơ sở hạ tầng công nghệ thông tin. Điều này phù hợp cho các tổ chức nhỏ có một server và các công ty lớn với nhiều server.

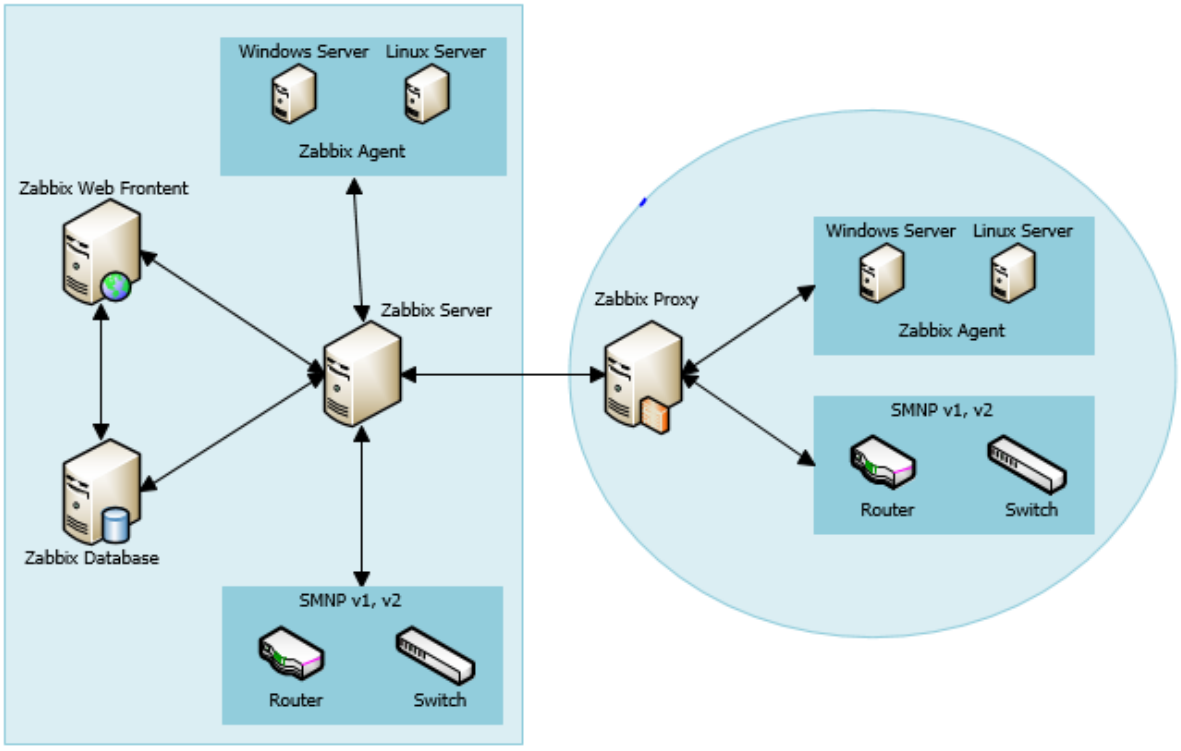
Zabbix được viết và phát hành với General Public License GPL phiên bản 2 [4].

### 3.3.2. Ưu điểm

- Phần mềm miễn phí.
- Hỗ trợ SNMP (Simple Network Management Protocol – Dùng để trao đổi thông tin quản lý giữa các thiết bị mạng).
- Tự động phát hiện server và thiết bị mạng.
- Giám sát Server, router, switch và thiết bị mạng khác.
- Dễ dàng thao tác và cấu hình.
- Hỗ trợ máy chủ Windows, Linux, Solaris, FreeBSD …
- Đáng tin cậy trong việc chứng thực người dùng.
- Linh hoạt trong việc phân quyền người dùng.
- Quản lý trên giao diện web thân thiện, dễ sử dụng.

- Thông báo sự cố qua email và SMS.
- Biểu đồ theo dõi và báo cáo.
- Mã nguồn mở và chi phí thấp.
- Kiểm soát theo dõi việc truy xuất.
- Tất cả thông tin (cấu hình, hiệu suất) được lưu trong cơ sở dữ liệu.
- Cài đặt đơn giản, dễ dàng.

### 3.3.3. Kiến trúc của hệ thống giám sát Zabbix




*Hình 3.3.3.1: Kiến trúc của Zabbix*

Kiến trúc của Zabbix bao gồm 4 thành phần cơ bản: Zabbix server, Zabbix proxy, Zabbix agent, Zabbix Web frontend [4].

#### a. Zabbix Server

Đây là thành phần trung tâm của phần mềm Zabbix. Server có thể kiểm tra các dịch vụ mạng từ xa (web server và mail server ). Agent sẽ báo cáo toàn bộ thông tin và số lượng thống kê cho server. Server sẽ lưu trữ tất cả cấu hình và dữ liệu thống kê.

### b. Zabbix Proxy

Proxy là phần tuỳ chọn của Zabbix. Proxy sẽ thu nhận dữ liệu , lưu trong bộ nhớ đệm và được chuyển đến Zabbix server.

Zabbix Proxy là một giải pháp lý tưởng cho một giám sát tập trung của địa điểm từ xa, chi nhánh, mạng lưới không có các quản trị viên địa phương.

Zabbix proxy cũng có thể được sử dụng để phân phối tải của một đơn Zabbix Server.. Zabbix proxy sẽ giúp giảm tải cho Zabbix Server nhờ vào việc thu thập dữ liệu và chuyển về cho zabbix server.

Zabbix Proxy có thể được dùng để:

- Giám sát các host từ những nơi khác
- Giám sát các host từ những nơi có kết nối không ổn định
- Giảm tải cho Zabbix server khi phải giám sát nhiều thiết bị
- Đơn giản hóa cho việc bảo trì và giám sát

Zabbix Proxy chỉ cần một kết nối TCP đến Zabbix server vì vậy phải cho phép kết nối này khi có tường lửa giữa Zabbix Proxy và Zabbix Server.

### c. Zabbix Agent

Agent là thành phần được cài đặt trên máy chủ, các thiết bị mạng cần giám sát. Agent sẽ thu thập thông tin hoạt động (ổ cứng, bộ nhớ, bộ xử lý số liệu thống kê, …) từ hệ thống mà nó đang chạy và báo cáo dữ liệu này đến Zabbix server để xử lý tiếp. Trong trường hợp lỗi (ổ cứng đầy hoặc một tiến trình chết…), Zabbix server sẽ gửi cảnh báo cho người quản trị về các sự cố này.

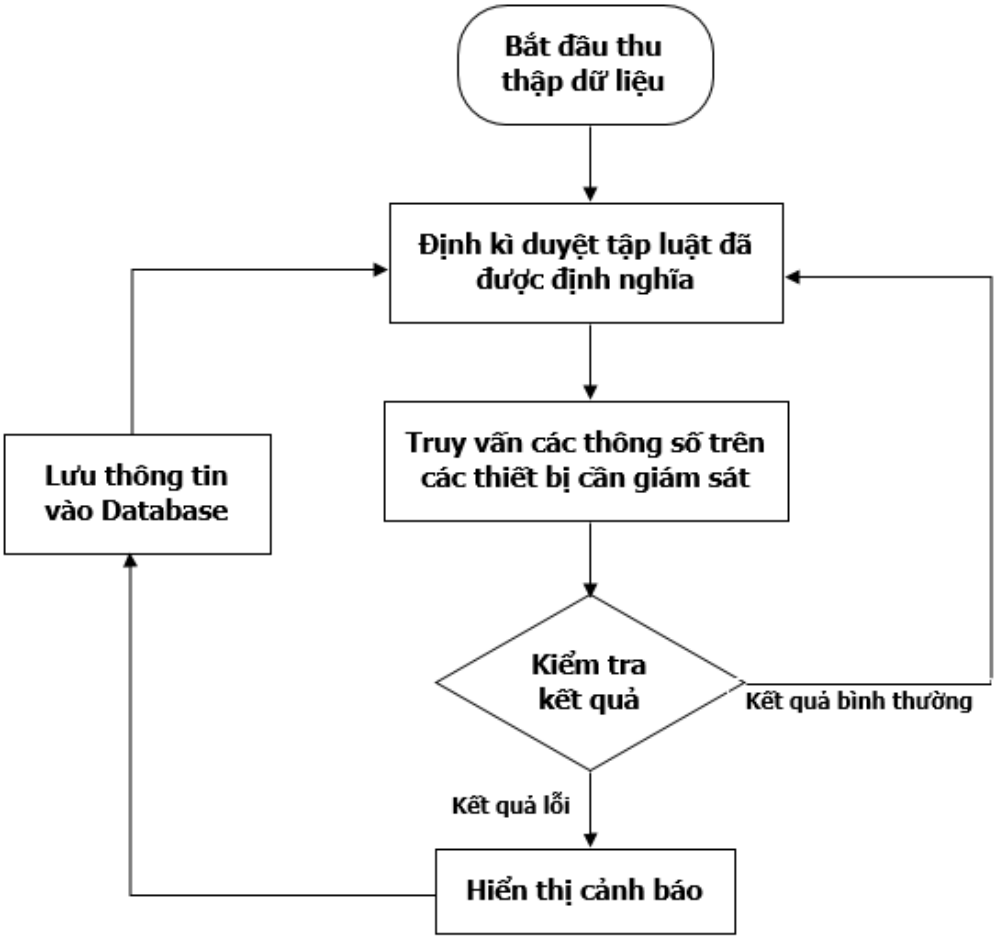
### d. Zabbix Web frontend

Là một giao diện web được viết bằng ngôn ngữ lập trình PHP, cho phép người quản trị hệ thống có thể cấu hình, giám sát, xem các dữ liệu thu thập được trên một giao diện web duy nhất

### 3.3.4. Cơ chế hoạt động

Zabbix giám sát các thiết bị qua Zabbix agent hoặc qua các giao thức SNMP…

Các Agent (thiết bị mạng hay server được cài đặt snmp hay zabbix agent) sẽ gửi các event liên quan đến thiết bị mạng, máy chủ tới Zabbix server, Zabbix server làm nhiệm vụ phân tích số liệu thu thập được và dựa vào các trigger do người quản trị hệ thống thiết lập mà quyết định đưa ra các cảnh báo theo mức độ khác nhau (Critical, warning, hoặc infomation…) Hình thức cảnh báo là gửi SMS, email…




*Hình 3.3.4.1: Cơ chế hoạt động Zabbix*

### 3.3.5. Tính năng của Zabbix

Các chức năng của Zabbix rất linh hoạt, nó có thể được cấu hình để theo dõi, giám sát thiết bị mạng, máy chủ theo cách ta muốn. Nó cũng có một cơ chế để tự động phản ứng với các vấn đề, và một hệ thống cảnh báo mạnh. Tất cả điều này được dựa trên một hệ thống định nghĩa các đối tượng rõ ràng.

- Khả năng giám sát: Zabbix có cấu hình tập trung, các thông tin giám sát được tập trung vào một cơ sở dữ liệu. Zabbix có khả năng sử dụng các proxy với số lượng không giới hạn, số nút đó có thể lên tới hàng ngàn.
- Khả năng mở rộng: Các thí nghiệm cho thấy nó có khả năng xử lý quản trị tới 100.000 thiết bị và máy chủ. Số lượng thông tin, dịch vụ giám sát có thể lên tới 1.000.000
- Hỗ trợ giám sát thời gian thực: Zabbix có thể cảnh báo ngay tới người quản trị viên khi hệ thống được giám sát có sự cố gì thông qua mail, SMS… Hơn nữa Zabbix còn có hồ sơ về các thông tin giám sát
- Khả năng hiển thị kết quả bằng đồ thị, biểu đồ giúp người dùng có thể dễ dàng giám sát.
- Khả năng nhập và xuất cơ sở dữ liệu thông qua XML.
- Khả năng tự động phát hiện: Người dùng có thể tạo ra các luật dựa trên nó Zabbix có thể tự động phát hiện ra các địa chỉ IP, các dịch vụ hoặc các thiết bị SNMP để thực hiện việc giám sát.
- Tính linh hoạt: Zabbix hỗ trợ cả IPv4 và IPv6, các Zabbix agent có khả năng cài đặt trên nhiều nền tảng khác nhau.
- Khả năng giám sát các thiết bị không hỗ trợ cài đặt Zabbix agent: Zabbix có khả năng giám sát các thiết bị hỗ trợ IPMI, SNMP v1,2,3,4.
- Khả năng bảo mật: Zabbix hỗ trợ người dùng một các linh hoạt, nó cung cấp khả năng chứng thực của địa chỉ IP.
- Quản trị các chức năng: Ta có thể chạy lệnh ping, traceroute trên một chuỗi các máy chủ, các thiết bị được quản trị.

### 3.3.6. Cấu trúc thư mục

- docs: Thư mục chứa file hướng dẫn pdf
- src: Thư mục chứa tất cả source cho các tiến trình Zabbix.
  + src/zabbix_server: Thư mục chứa file tạo và source cho zabbix_server.
  + src/zabbix_agent: Thư mục chứa file tạo và source cho zabbix_agent và zabbix_agentd.
  + src/zabbix_get: Thư mục chứa file tạo và source cho zabbix_get.
  + src/zabbix_sender: Thư mục chứa file tạo và source cho zabbix_sender.
- include: Thư mục chứa các thư viện Zabbix.
- misc
  + misc/init.d: Thư mục chứa các tập lệnh khởi động trên các nền khác nhau.
- frontends
  + frontends/php: Thư mục chứa các file PHP.
- create: Thư mục chứa các tập lệnh SQL để tạo cơ sở dữ liệu ban đầu.
  + create/schema: Thư mục tạo biểu đồ cơ sở dữ liệu.
  + create/data: Thư mục chứa dữ liệu cho việc tạo cơ sở dữ liệu ban đầu.
- upgrades: thư mục chứa các thủ tục nâng cấp cho phiên bản khác nhau của Zabbix.

### 3.3.7. Các mô hình triển khai hệ thống Zabbix

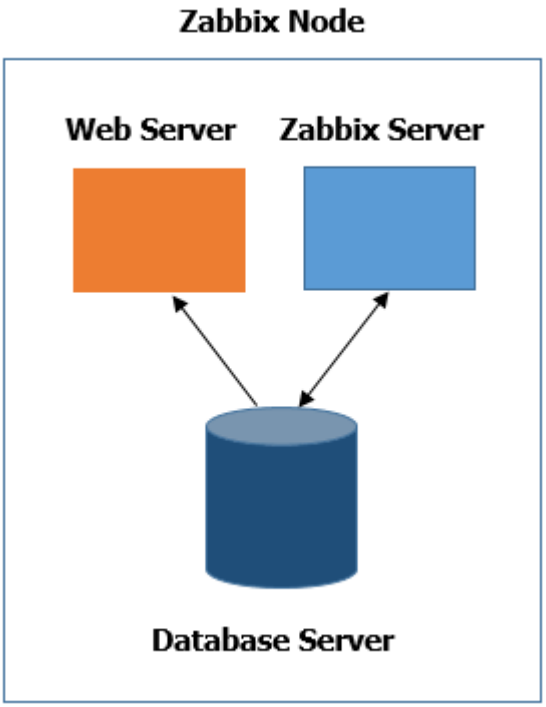
Nhìn chung, kiến trúc Zabbix cho các hệ thống lớn bao gồm 3 thành phần như sau: Web server, Zabbix server, Database server.

Ngoài ra còn có thêm 2 thành phần khác nữa bao gồm Zabbix Agent và Zabbix proxy. Dựa vào đặc điểm kiến trúc mà Zabbix thông thường được triển khai theo 2 mô hình dưới đây [5]:

#### a. Mô hình tập trung

Mô hình cài đặt trên một máy chủ 01 máy chủ không được khuyến cáo trên các hệ thống giám sát lớn, tuy nhiên đây là một mô hình cơ bản và phù hợp với doanh nghiệp

nhỏ có số lượng thiết bị cần giám sát ít. Một Node cài tất cả các thành phần Zabbix server, Zabbix Database, Zabbix web fondend




*Hình 3.3.7.1: Mô hình triển khai Zabbix trên một node*

Đối với các doanh nghiệp vừa và nhỏ, chúng ta chỉ cần triển khai trên một server vật lý để tiết kiệm chi phí đầu tư phần cứng.

Đối với các doanh nghiệp lớn, cần đảm bảo tính dự phòng của hệ thống thì có thể cài đặt thêm các node dự phòng, share tải và HA (High Availability) cho hệ thống.

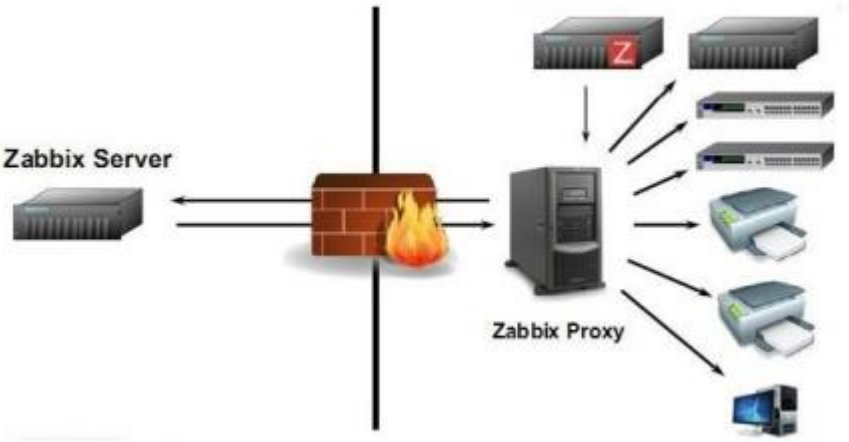
**b. Mô hình phân tán**

Zabbix là một ứng dụng giám sát có thể monitor hàng trăm đối tượng trong một mô hình cài đặt single server. Tuy nhiên khi hệ thống mà có đến hàng nghìn máy chủ, thiết bị mạng, với mô hình mạng phức tạp, các máy chủ có thể đặt ở nhiều vị trí địa lý khác nhau thì đây lại là một điểm giới hạn của việc cài đặt single server.

Trong phần lớn trường hợp, trong doanh nghiệp có thể đã có một hệ thống Zabbix đã được cài đặt. Zabbix có ưu điểm về tính linh hoạt, cho phép nâng cấp mô hình cài đặt lên một bước, theo mô hình: proxy-based monitoring (hay mô hình phân tán).

Proxy-based monitoring được triển khai với 01 Zabbix server và nhiều Zabbix proxies, mỗi proxy có thể ở tại một chi nhánh hoặc 01 data center. Cấu hình này dễ dàng duy trì và có nhiều ưu điểm triển khai, đặc biệt trong việc giám sát tập trung. Loại kiến thức triển khai này phù hợp với các giám sát có mô hình mạng phức tạp.

Kiến trúc giám sát phân tán là mô hình cài đặt phức tạp nhất trong việc triển khai Zabbix. Hình ảnh bên dưới mô tả một hệ thống giám sát phân tán kết hợp proxy-based để giám sát.




*Hình 3.3.7.2: Mô hình Zabbix phân tán*

### 3.3.8. Các phần tử cơ bản trong Zabbix

- **Host**: Các thiết bị mà cần giám sát, có thể là server, máy trạm, thiết bị mạng, thiết bị Firewall, UPS… Tạo một host chính là việc đầu tiên trong cấu hình giám sát Zabbix.
- **Item**: Chính là các đối tượng, dữ liệu cần thu thập trong một host. Có nhiều kiểu item khác nhau, nó phụ thuộc vào các đối tượng giám sát khác nhau.
- **Triggers**: Là một điều kiện khi thỏa mãm điều kiện của Triggers mà người lập trình đặt ra thì sẽ thực hiện một hành động nào đó tiếp theo.

Ví dụ: Bạn đang giám sát Ram và bạn tạo một Triggers cho việc giám sát Ram nếu Ram sử dụng trên 90% thì thông báo đến người quản trị hoặc gửi mail cho người quản trị.

- **Template**: Là một mẫu chuẩn, đã được định nghĩa sẵn các item, triggers, graph, screen… Nó vô cùng thuận tiện khi triển khai giám sát nhiều host có những thành phần cần giám sát giống nhau. Vì vậy chỉ cần tạo 1 template là có thể áp dụng cho nhiều host khác nhau.

## 3.2. Cài đặt phần mềm zabbix

### 3.3.1. Yêu cầu hệ thống

- Server: CPU duo core, RAM >= 1GB, HDD >=50 GB tùy theo nhu cầu lưu log, cách lưu log.
- OS: Windows, Linux, Unix. Ở đây ta chọn CentOS 7.2 64bit, vừa nhẹ, không quan tâm lisence và tận dụng cho các công tác hệ thống khác.
- Zabbix: Hiện tại đã có bản zabbix 4.0, nhưng ta dùng version 3.4 cho server vì tính ổn định. Phía client sử dụng Zabbix Agent version >= 3.4
- Database: dùng mysql hoặc mariaDB bản open source mới nhất.
- Web: dùng Apache2 hoặc httpd
- PHP, net-snmp: bản mới nhất

### 3.3.2. Cài đặt Zabbix Server

#### a. Chuẩn bị môi trường cài đặt

- Sử dụng hệ điều hành CentOS 7
- Sử dụng quyền root

Bước 1. Disable SELINUX

```
# vi /etc/selinux/config
[…]
SELINUX=enforcing
[…]
```

Bước 2. Khởi động lại Server để xác nhận thay đổi

```
# reboot
```

Bước 3. Cài đặt, khởi động service httpd

```
# yum -y install httpd
# systemctl start httpd
# systemctl enable httpd
```

Bước 4. Mở port firewall

```
# firewall-cmd --add-service=http --permanent
# firewall-cmd --reload
```

Bước 5. Cài đặt, cấu hình PHP

```
# yum -y install php php-mbstring php-pear
# vi /etc/php.ini
[…]
date.timezone = "Asia/Ho_Chi_Minh" (uncomment)
[…]
# systemctl restart httpd
```

Bước 6. Cài đặt, khởi động service database

```
# yum -y install mariadb-server
# systemctl start mariadb
# systemctl enable mariadb
```

- Thiết lập password root cho database

```
# mysql_secure_installation
[…]
Enter current password for root (enter for none):
OK, successfully used password, moving on...

Setting the root password ensures that nobody can log into the
MariaDB
root user without the proper authorisation.

Set root password? [Y/n] Y
New password:
Re-enter new password:
Password updated successfully!
Reloading privilege tables…
 ... Success!
[…]
```

**b. Tải file cài đặt**

Có 2 cách để cài đặt Zabbix:

- Tải source code mới nhất của Zabbix về và tiến hành compiling
- Cài đặt từ packages

Ngoài ra còn có cách khác là tải các bản Appliance đã cài đặt sẵn Zabbix về và chạy, tuy nhiên cách này không được khuyến cáo vì quá trình thực hiện bằng tay sẽ giúp kiểm soát các bước cài đặt tốt hơn.

Cách cài đặt từ package có 2 ưu điểm chính:

- Quá trình cài và nâng cấp sẽ nhanh và dễ dàng hơn
- Các gói phần mềm phụ thuộc (dependencies) sẽ tự động được cài đặt

Cách cài đặt từ source code compilation cũng có một số ưu điểm:

- Chỉ compile các tính năng cần thiết đối với hệ thống, vì vậy sẽ tối ưu hơn
- Có thể xây dựng và đóng gói, sau đó có thể cài đặt trên 01 hệ thống Linux khác
- Hoàn toàn kiểm soát việc nâng cấp.

Copy gói cài đặt dành cho Centos 7:
https://www.zabbix.com/documentation/3.4/manual/installation/install_from_packages/rhel_centos


## c. Cài đặt, cấu hình Zabbix Server

Bước 1. Cài đặt một số package khác và download Zabbix repository

```
# yum -y install php-mysql php-gd php-xml php-bcmath

# rpm -ivh http://repo.zabbix.com/zabbix/3.4/rhel/7/x86_64/zabbix-release-3.4-2.el7.noarch.rpm
```

Bước 2. Cài đặt Zabbix Server

```
# yum -y install zabbix-get zabbix-server-mysql zabbix-web-mysql zabbix-agent
```

Bước 3. Tạo Database cho Zabbix

- Login database

```
# mysql -u root -p
```

- Tạo database có tên zabbix cho Zabbix Server

```
MariaDB [(none)]> create database zabbix character set utf8 collate
utf8_bin;
Query OK, 1 row affected (0.00 sec)
```

- Gán quyền cho user zabbix với mật khẩu là password cho database zabbix.

```
MariaDB [(none)]> grant all privileges on zabbix.* to
zabbix@'localhost' identified by 'password' ;

MariaDB [(none)]> grant all privileges on zabbix.* to zabbix@'%'
identified by 'password';
```

- Áp dụng thay đổi và thoát khỏi Database.

```
MariaDB [(none)]> flush privileges;
MariaDB [(none)]> exit;
```

Bước 4. Import Database Zabbix

```
# cd /usr/share/doc/zabbix-server-mysql-3.4.15/
# gunzip create.sql.gz
# mysql -u root -p zabbix < create.sql
```

Bước 5. Cấu hình và khởi động Zabbix Server

```
# vi /etc/zabbix/zabbix_server.conf

[…]
DBHost=localhost (uncomment)
DBPassword=password (uncomment)
[…]

# systemctl start zabbix-server
# systemctl enable zabbix-server
```

Bước 6. Mở port Firewall

```
# firewall-cmd --add-port={10051/tcp,10050/tcp} --permanent
# firewall-cmd --reload
```

Bước 7. Cấu hình và khởi động Zabbix Agent để tự giám sát Zabbix Server

```
# vi /etc/zabbix/zabbix_agentd.conf

[…]
Server=127.0.0.1
ServerActive=127.0.0.1
Hostname=zabbixsrv.local
[…]

# systemctl start zabbix-agent
# systemctl enable zabbix-agent
```

Bước 8. Thay đổi cài đặt và restart httpd

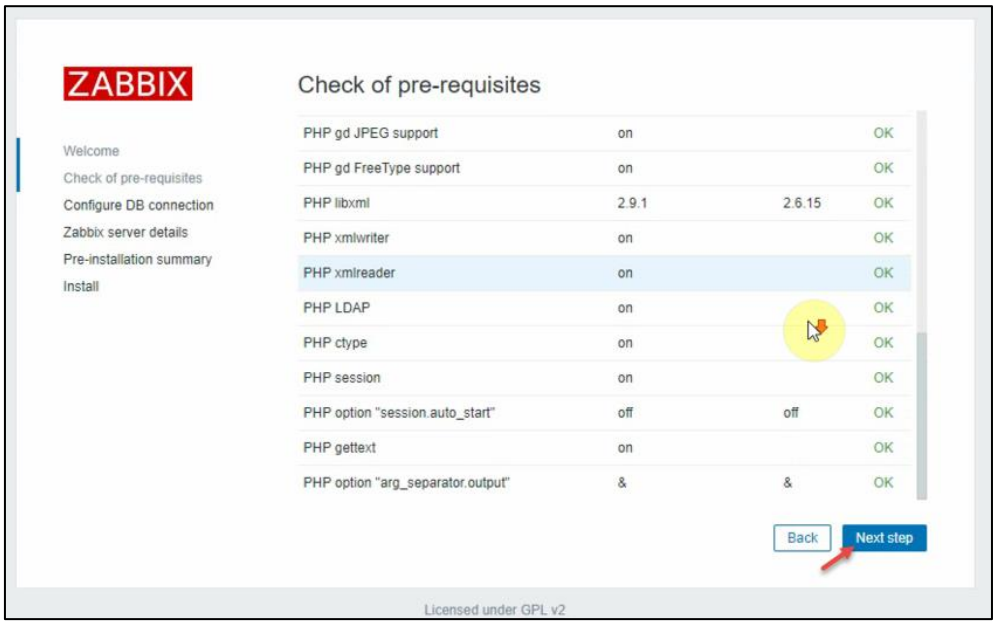
```
# vi /etc/httpd/conf.d/zabbix.conf
[…]
php_value date.timezone Asia/Ho_Chi_Minh (uncomment)
[…]

# systemctl restart httpd
```

### 3.3.3. Cài đặt giao diện Zabbix Web frontend

Bước 1. Truy cập vào http://ipserver/zabbix . Sau đó, trang bắt đầu Zabbix được hiển thị, Nhấp vào **Next step** để tiếp tục

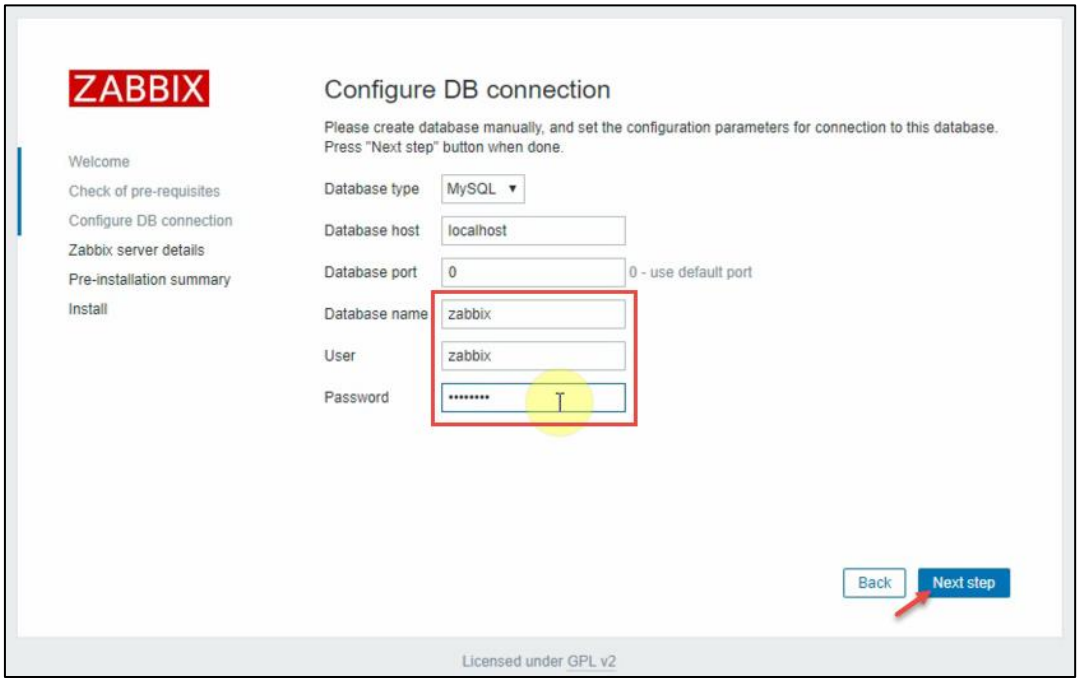

*Hình 3.3.3.1: Giao diện cài đặt Zabbix Web*

Bước 2. Thông số config php, đảm bảo tất cả các mục là [OK]. Sau đó nhấn **Next step** để tiếp tục



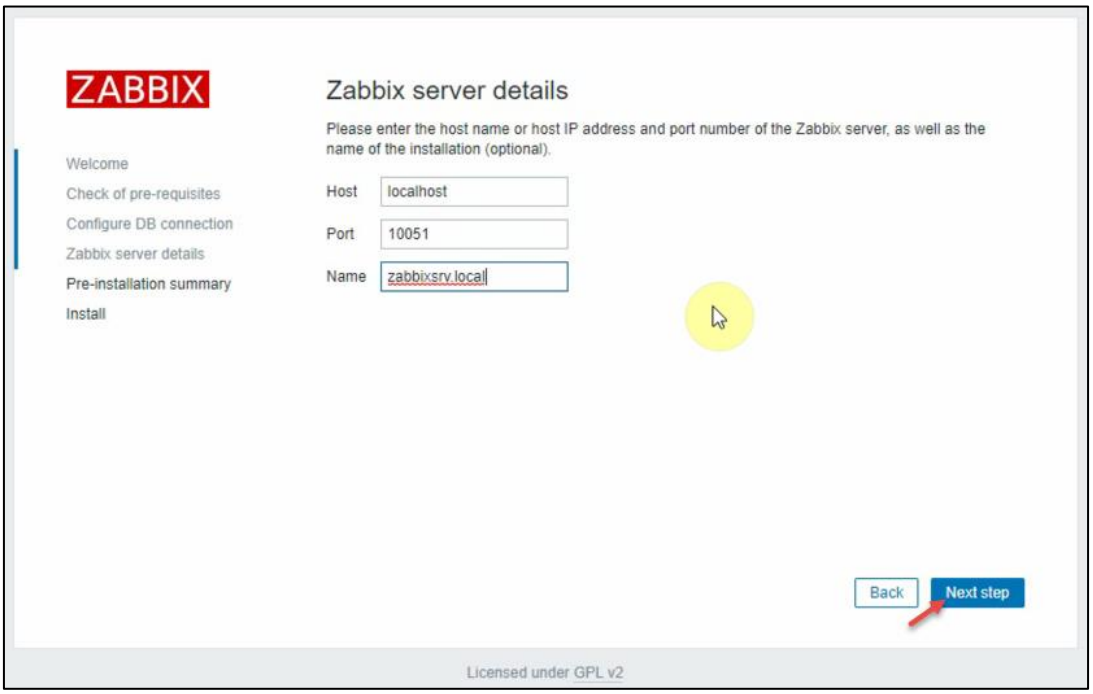

*Hình 3.3.3.2: Kiểm tra thông số config php*

Bước 3. Nhập các thông tin về Database Zabbix đã được tạo. Nhấn **Next step**



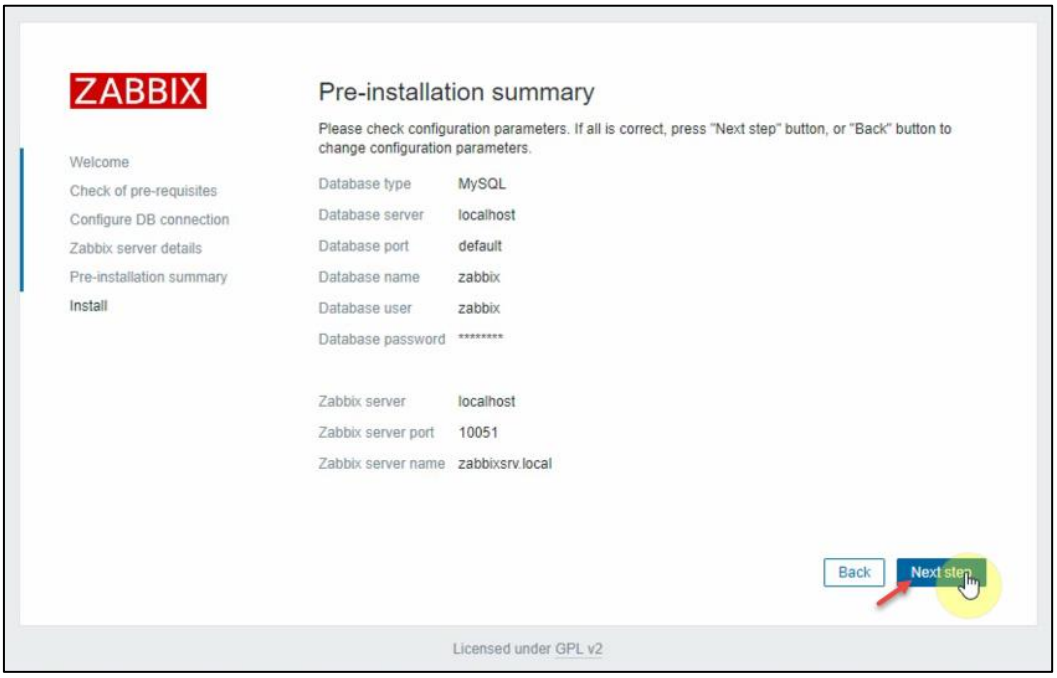

*Hình 3.3.3.4: Thông tin Database Zabbix*

Bước 4. Cài đặt kết nối đến Zabbix Server. Thay đổi thông tin **Name** (nếu cần). Nhấn **Next step**



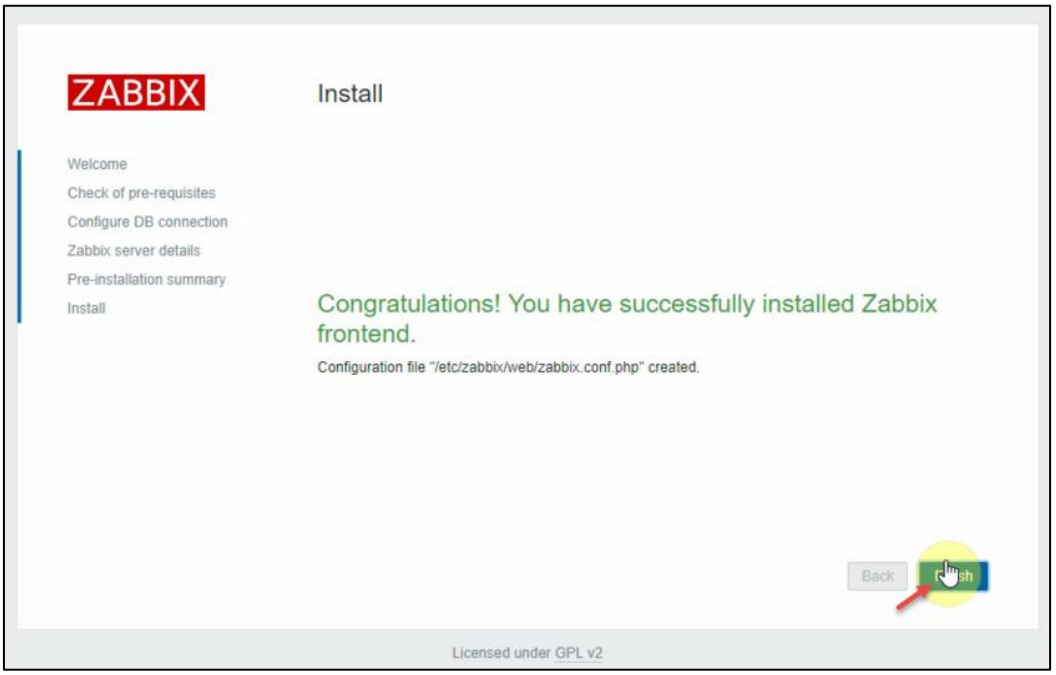

*Hình 3.3.3.5: Thiết lập kết nối đến Zabbix Server*

Bước 5. Tóm tắt các thông tin đã cài đặt. Nhấn **Next step** để tiếp tục



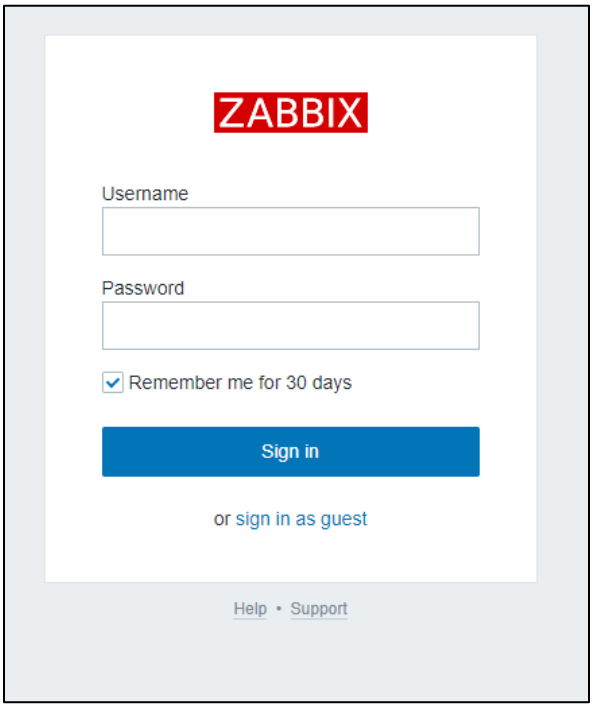

*Hình 3.3.3.6: Thông tin đã cài đặt*

Bước 6. Kết thúc quá trình cài đặt. Chọn **Finish**




*Hình 3.3.3.7: Hoàn thành quá trình cài đặt*

Bước 7. Đăng nhập sử dụng. Sử dụng tài khoản default để login zabbix server **admin\zabbix**




*Hình 3.3.3.8: Giao diện đăng nhập*

### 3.3.4. Cài đặt Zabbix Agent

### a. Cài đặt Zabbix Agent trên Windows Server

**Cách 1.** Tải Zabbix agent từ trang chủ https://www.zabbix.com/download

Bước 1. Các bạn chọn mục Zabbix pre-compiled agents và tải bản cho windows (all)


*Hình 3.3.4.1: Tải zabbix-agent*

Bước 2. Giải nén và khởi tạo thư mục zabbix tại ổ C và copy zabbix_agentd.conf and zabbix_agentd.win.exe vào thư mục zabbix

Bước 3. Thiết lập file zabbix_agentd.conf thay đổi các thông số sau:

```
LogFile=C:\Zabbix\Zabbix_agentd.log
Server=<ip address of your zabbix server>
ListenPort=10050 (uncomment)
ServerActive=<ip address of your zabbix server>
Hostname=<hostname của máy đang cài đặt Zabbix Agent>
```

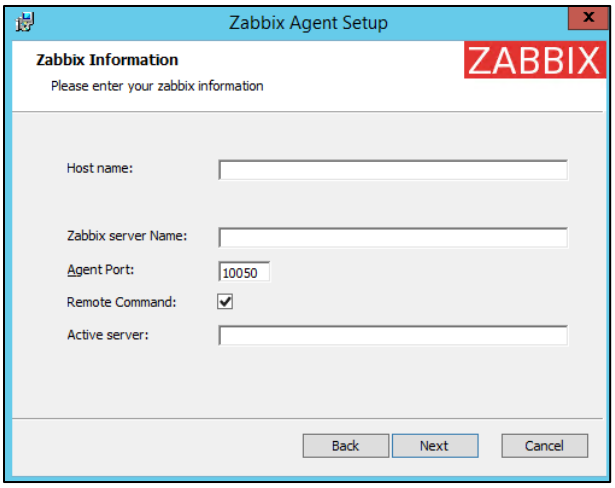
Bước 4. Cài đặt và khởi động service

```
zabbix_agentd.exe -c C:\zabbix\zabbix_agentd.win.conf -i (To install
the agent)
zabbix_agentd.exe -c C:\zabbix\zabbix_agentd.win.conf -s (To start
the service)
```

**Cách 2.** Tải phần mềm được đóng gói dưới dạng file đóng gói sẵn (.msi)

Bước 1. Tải file cài đặt Zabbix Agent https://www.suiviperf.com/zabbix/old/

Bước 2. Cài đặt ứng dụng




*Hình 3.3.4.2: Thông tin cấu hình zabbix agent*

Tại hộp thoại Zabbix Information nhập các thông tin:

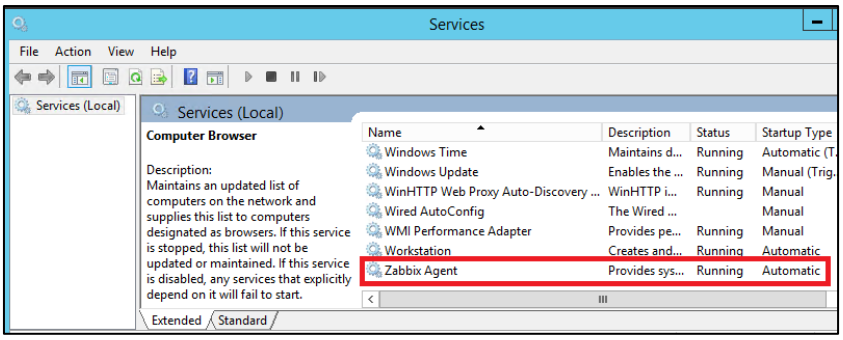
Host name: Tên máy tính cài đặt Zabbix Agent

Zabbix server Name: IP của Zabbix server

Agent Port: Mặc định 10050

Active server: IP của Zabbix server

**Chú ý:** Sau khi cài đặt (bằng cả 2 cách), vào services.msc restart hoặc start dịch vụ




*Hình 3.3.4.3: Thiết lập khởi động zabbix-agent*

Tiến hành mở port 10050 cho dịch vụ Zabbix-agent trên Windows Firewall with Advanced Security để cho phép Zabbix server và Zabbix-agent có thể trao đổi thông tin.

## b. Cài đặt Zabbix Agent trên Linux Server

Bước 1. Chuẩn bị môi trường

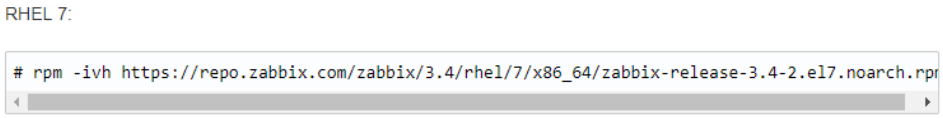
- Kiểm tra phiên bản hệ điều hành cần cài đặt Zabbix Agent

```
#cat /etc/redhat-release
```

- Kiểm tra disable SELINUX

```
cat /etc/selinux/config
[…]
SELINUX=disabled
[…]
```

- Tải packages repo tương ứng version hệ thống Zabbix đang chạy và OS tương ứng: https://www.zabbix.com/documentation/3.4/manual/installation/install_from_packages/rhel_centos




Bước 2. Cài đặt Zabbix repository

```
# rpm -ivh https://repo.zabbix.com/zabbix/3.4/rhel/7/x86_64/zabbix-release-3.4-2.el7.noarch.rpm
```

Bước 3. Cài đặt zabbix agent

```
# yum install zabbix-agent -y
```

Bước 4. Cấu hình dịch vụ Zabbix agent

```
# vi /etc/zabbix/zabbix_agentd.conf
[…]
Server=<zabbix-server-ip>
ServerActive=<zabbix-server-ip>
[…]
```

Bước 5. Mở port firewall

```
# firewall-cmd --add-port=10050/tcp –permanent
# firewall-cmd --reload
# setenforce 0
```

Bước 6. Start service Zabbix agent

```
# systemctl start zabbix-agent
# systemctl enable zabbix-agent
```

# CHƯƠNG 4: ỨNG DỤNG THỰC NGHIỆM

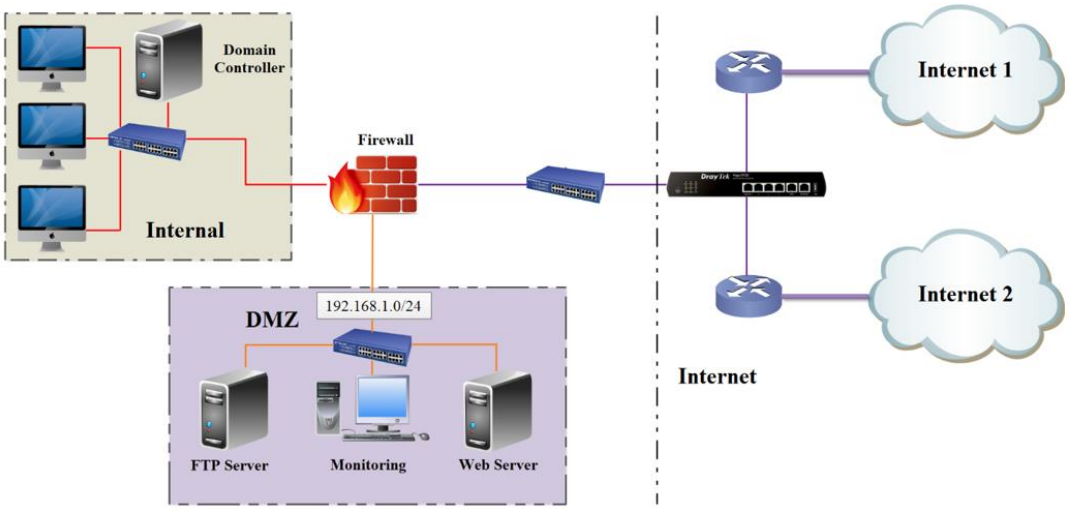
## 4.1. Phát biểu bài toán

Công ty A là công ty hoạt động trong lĩnh vực viễn thông – công nghệ thông tin. Do đó công ty có các máy chủ chứa nhiều thông tin quan trọng về khách hàng cũng như các số liệu tài chính. Cùng với sự đòi hỏi lớn hơn về băng thông mạng các thiết bị, máy chủ chứa các dữ liệu và tài nguyên này yêu cầu cần được giám sát tình trạng sử dụng một cách chặt chẽ và cần thiết.

Ngoài sự hỗ trợ của hệ thống bảo vệ mạng như tường lửa thì vai trò của người quản trị viên cũng hết sức quan trọng. Tuy nhiên không phải lúc nào người quản trị cũng có thể nắm bắt được hết tình của hệ thống, có khi hệ thống bị sự cố rồi thì quản trị viên mới bắt đầu dò tìm nguyên nhân và khắc phục sự cố. Hoặc khi cấp trên yêu cầu báo cáo tình trạng hệ thống hằng ngày, hằng tuần thì công việc đó cũng làm người quản trị mất rất nhiều công sức để thực hiện. Nhưng hiệu quả đem lại thực sự chưa cao, thông tin hệ thống đáp ứng chưa đầy đủ. Hiện nay có rất nhiều phần mềm quản lý hệ thống tài nguyên mạng sử dụng các thiết bị phần cứng đắt tiền. Tuy nhiên một số phần mềm mã nguồn mở cũng đáp ứng một cách toàn diện với nhiều tính năng linh hoạt vượt trội.

Xuất phát từ những nhu cầu trên, vậy nên xây dựng một hệ thống giám sát mạng là điều hết sức cần thiết.

## 4.2. Mô hình triển khai thực nghiệm




*Hình 4.2.1: Mô hình hệ thống giám sát sử dụng Zabbix*

### 4.2.1. Giới thiệu mô hình

Trong sơ đồ này sẽ xác định 3 vùng riêng biệt gồm có: Internal, DMZ và Internet

- Vùng Internal: là vùng chứa Domain Controller và các máy Client
- Vùng DMZ: là vùng chứa các Server của công ty như FTP Server, Web Server... Ngoài ra để giám sát các máy Server trong vùng DMZ sử dụng một hệ thống giám sát Zabbix
- Vùng Internet: là vùng bên ngoài cung ty, cung cấp dịch vụ Internet cho công ty
- Ngoài ra để kết nối 3 vùng này với nhau thì sử dụng Firewall pfSense.

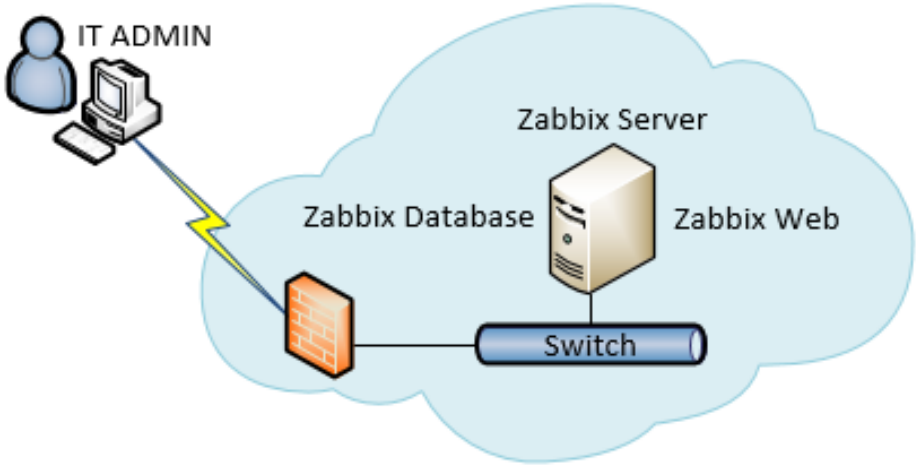
### 4.2.2. Giải thích mô hình

- Trong vùng Internal: ở trong vùng Internal chứa các máy Client sẽ đặt luôn Domain Controller (vì lí do Domain Controller là nơi xây dựng các User – Group được sử dụng trong công ty, hơn thế nữa nơi đây cũng là nơi triển khai toàn bộ các chính sách của công ty xuống các phòng ban, các User và Group, và các máy Client sẽ phải Join vào Domain để lấy User sử dụng trong công ty cũng như sẽ chấp hành theo những chính sách người quản trị đưa xuống để làm việc)
- Trong vùng DMZ: vùng này sẽ chứa toàn bộ những server sử dụng cho công ty như Web Server (Website của công ty), FTP Server (lưu giữ các file tài liệu), hay Mail Server (mail riêng của nội bộ công ty)… Vùng này cũng sẽ là nơi được public các dịch vụ ra ngoài internet để người sử dụng cũng có thể làm việc, như việc public Website của công ty để người sử dụng truy cập bằng tên miền vào website . . . Trong vùng DMZ sẽ có một hệ thống giám sát hoạt động của các Server, để người quản trị có thể kịp thời phát hiện sự cố - hỏng thiết bị và khắc phục dễ dàng hơn, ở đây sẽ sử dụng giám sát hệ thống bằng Zabbix. Với phần mềm Zabbix giúp giải quyết được toàn bộ những khó khăn của doanh nghiệp trong việc quản lý tài nguyên, cho phép quản lý toàn sự cố, quản lý topo mạng và cấu hình thiết bị mạng. Tạo nên một hệ thống mạng chủ động.
- Vùng internet: là nơi cung cấp dịch vụ Internet cho người sử dụng internet trong công ty.
- Firewall pfSense: Trước khi Internet từ nhà mạng qua router đến công ty sẽ đi qua một hệ thống Firewall. Là nơi kết nối các khu vực với nhau bằng các Rule sao cho hệ

thống có thể hoạt động tốt nhưng cũng đảm bảo độ an toàn cho các khu vực riêng biệt, tránh sự tấn công xâm nhập từ bên ngoài thông qua các Rule quản lý.

### 4.3. Triển khai hệ thống thực nghiệm

Triển khai hệ thống theo mô hình tập trung, cài đặt các thành phần của Zabbix trên cùng một server phục vụ cho việc giám sát hệ thống mạng. Thực hiện cài đặt theo các bước hướng dẫn "*Mục 3.2. Cài đặt phần mềm Zabbix*".




*Hình 4.3.1: Mô hình triển khai Zabbix tập trung*

Các thành phần của hệ thống mạng bao gồm:

| STT | Host name | IP | OS | Note |
|---|---|---|---|---|
| 1. | zabbixsrv | 192.168.1.120 | Centos 7.2 | Máy chủ giám sát thiết bị mạng, chương trình ứng dụng, tài nguyên các máy server khác |
| 2. | ad-sota | 192.168.1.6 | Windows Server 2012 | Máy chủ quản lý domain và cấp DNS cho mạng 192.168.1.0 |
| 3. | 11g | 192.168.1.33 | Redhat 7.2 | Máy chủ CSDL |
| 4. | filesrv-sota | 192.168.1.10 | Windows Server 2012 | Máy chủ lưu giữ các file tài liệu |
| 5. | ftp | 192.168.1.133 | Centos 7.2 | Máy chủ dịch vụ FPT |
| 6. | AD02 | 192.168.1.138 | Windows Server 2012 | Máy chủ phục vụ hệ thống LAP |

### 4.3.1. Kịch bản giám sát hệ thống mạng

Xây dựng kịch bản giám sát phần hệ thống (hosts, end devices) trong hệ thống mạng bao gồm: 03 máy chủ chạy hệ điều hành windows server 2012, 02 máy chủ chạy hệ điều hành linux server (Centos 7.2 và Redhat 7.2).

**a. Giám sát trạng thái trên host**

- Trạng thái hoạt động của hosts
- Trạng thái hoạt động của services
- Trạng thái của các interfaces trên hosts

**b. Giám sát việc sử dụng các tài nguyên**

- CPU số lượng processes trong hàng đợi hay theo % sử dụng CPU của hosts
- Ram: Cho biết dung lượng tổng, số lượng dung lượng sử dụng hay còn trống
- Disk: Cho biết dung lượng tổng, còn trống, và đã sử dụng.

**c. Giám sát lưu lượng mạng vào ra trên các host**

- Giám sát lưu lượng vào ra trên các interface của host: tổng lưu lượng vào ra.

**d. Thông tin, quản lý dữ liệu giám sát của các host**

- Lưu trữ các thông số trong 7->30 ngày có thể xem lại và phục vụ cho việc phân tích và nâng cấp hệ thống sau này sau này
- Biển diễn theo danh sách hoặc biểu đồ trực quan về tình hình sử dụng tài nguyên và thông số trên các host

**e. Cảnh báo**

- Cảnh báo trạng thái: ví dụ host bị down bất thường, hay sự cố ngoài ý muốn
- Cảnh báo dịch vụ: Services bị tắt hay thay đổi trạng thái

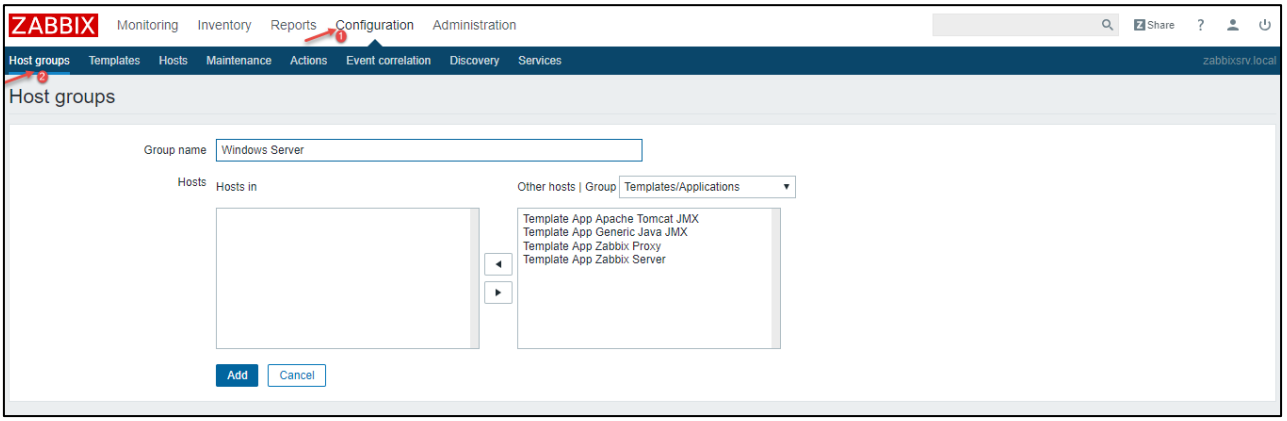
**Ví dụ:** Máy chủ bất ngờ shutdown

- Cảnh báo thời gian máy chủ shutdown

### 4.3.2. Giám sát hệ thống mạng

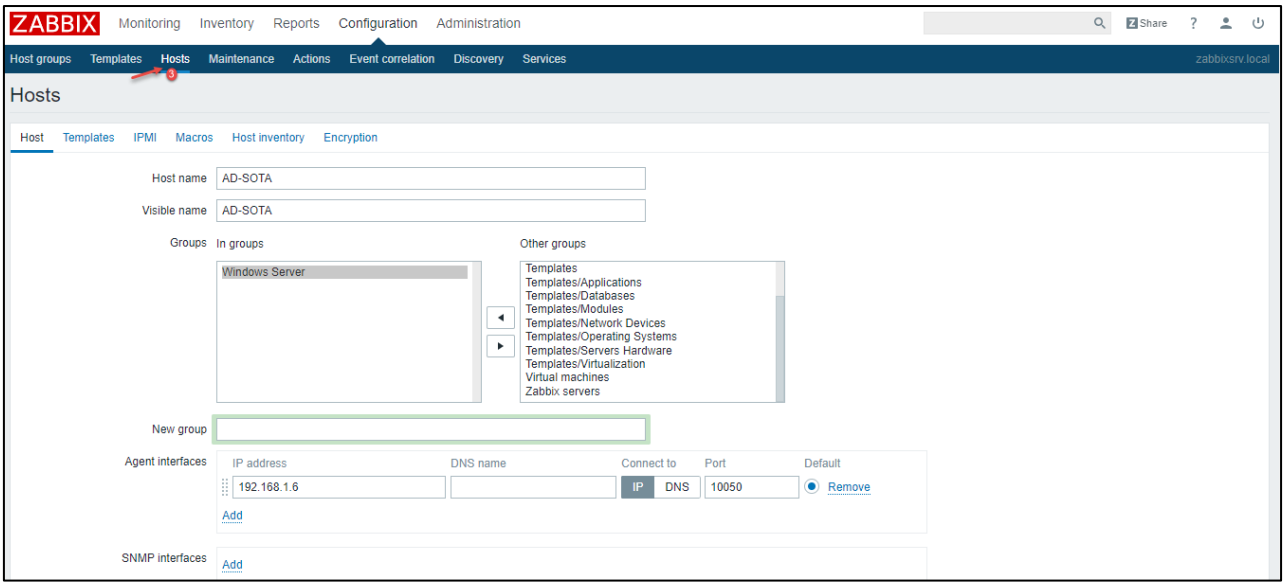
Khi hệ thống hoạt động, việc theo dõi trạng thái các máy chủ đang trong tình trạng như thế nào, các dịch vụ quan trọng có được chạy hay không? Hay trạng thái các interfaces cần thiết trên các thiết bị đầu cuối ấy là 1 điều bắt buộc cần giám sát. Thiếp lập từng bước giám sát hệ thống mạng: tạo host, host group và template cho các host windows server và linux server trong hệ thống.

Bước 1. Tạo host group và host




*Hình 4.3.2.1: Tạo host groups*

Tiếp theo, tạo host và thêm vào host group, ở đây host AD-SOTA được add vào host group Windows Server vừa tạo




*Hình 4.3.2.2: Tạo host*

Điền các thông số cấu hình:

Name: AD-SOTA

Groups: Windows Server

IP address: 192.168.1.6

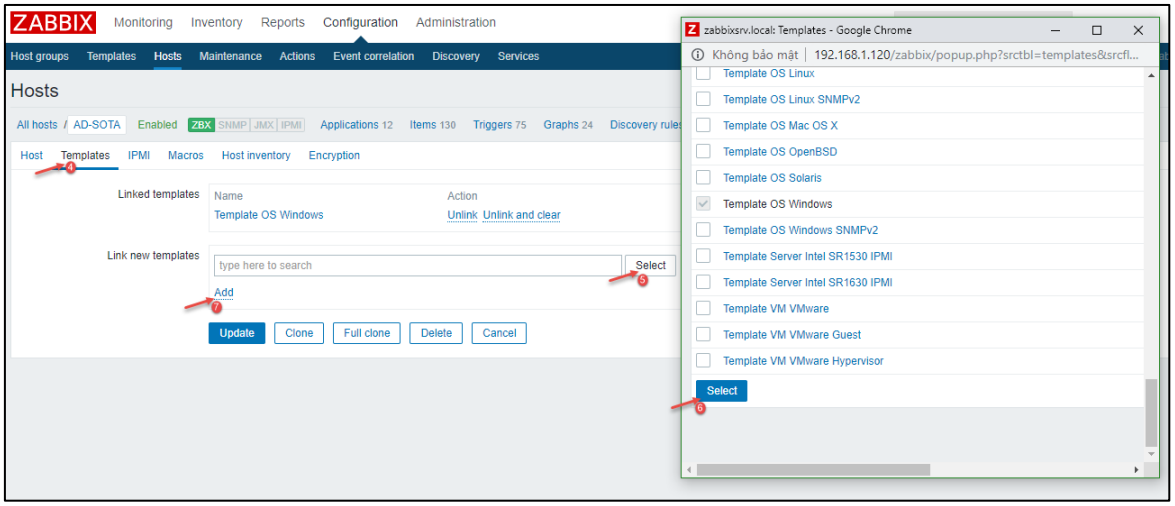
Conect to: IP address

Port: 10050

Monitored by proxy: No proxy

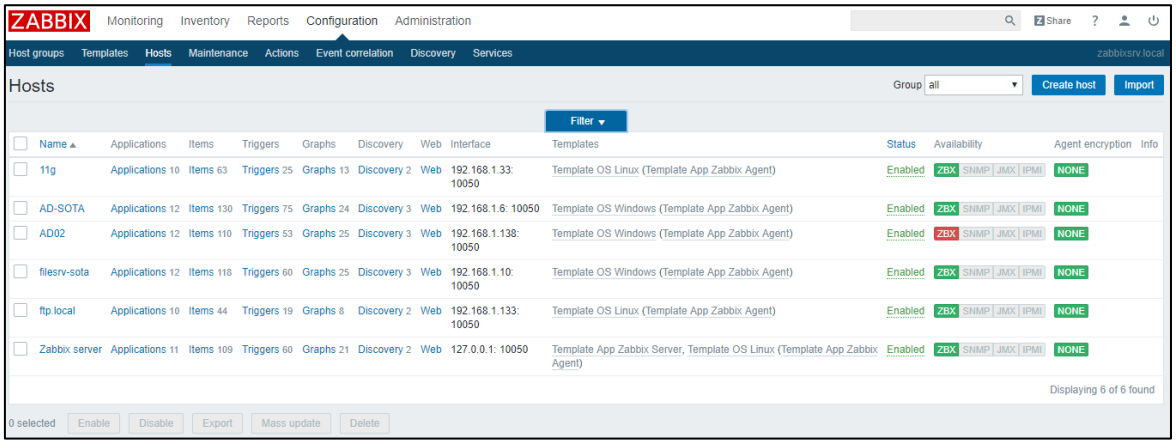
Status: Monitored

Bước 2. Sau đó chúng ta thêm các dịch vụ cần giám sát, chọn select và tích và các template cần thiết cho server.




*Hình 4.3.2.3: Lựa chọn và thêm các template cho host*

Bước 3. Thực hiện tương tự đối với các host còn lại trong hệ thống mạng



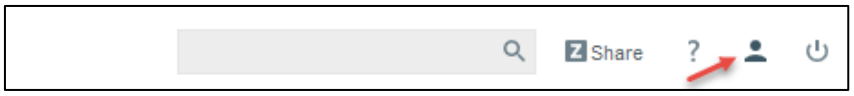

*Hình 4.3.2.4: Thêm các host trong hệ thống mạng thành công*

### 4.3.3. Thiết lập cảnh báo

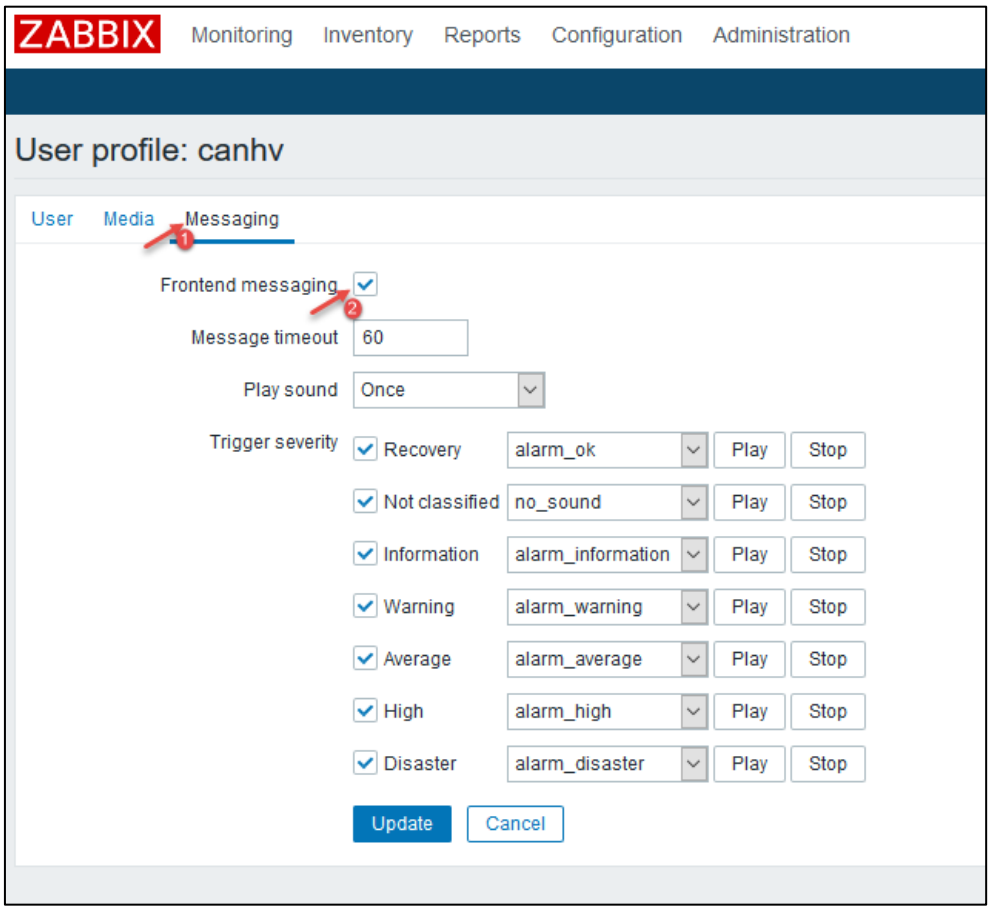
**a. Cảnh báo tại máy:**

Hình ảnh và âm thanh trực quan trên Web của Zabbix. Hay nói cách khác khi sự cố xảy, hệ thống sẽ phát ra tín hiệu cảnh báo như đổ chuông hay hiển thị các hình ảnh cảnh báo. Và nhờ đó người quản trị có thể đưa ra hành động kịp thời cho từng trường hợp cụ thể.

Bước 1. Vào Profile của user mà bạn dùng để đăng nhập vào zabbix frontend và sử dụng user đó để nghe thông báo cũng như âm thanh cảnh báo từ Zabbix Server




Bước 2. Chọn Messaging → Tích chọn Frontend messaging




*Hình 4.3.3.1. Thiết lập thông tin hiển thị chuông cảnh báo*

Thiết lập các thông tin cho phù hợp với yêu cầu:

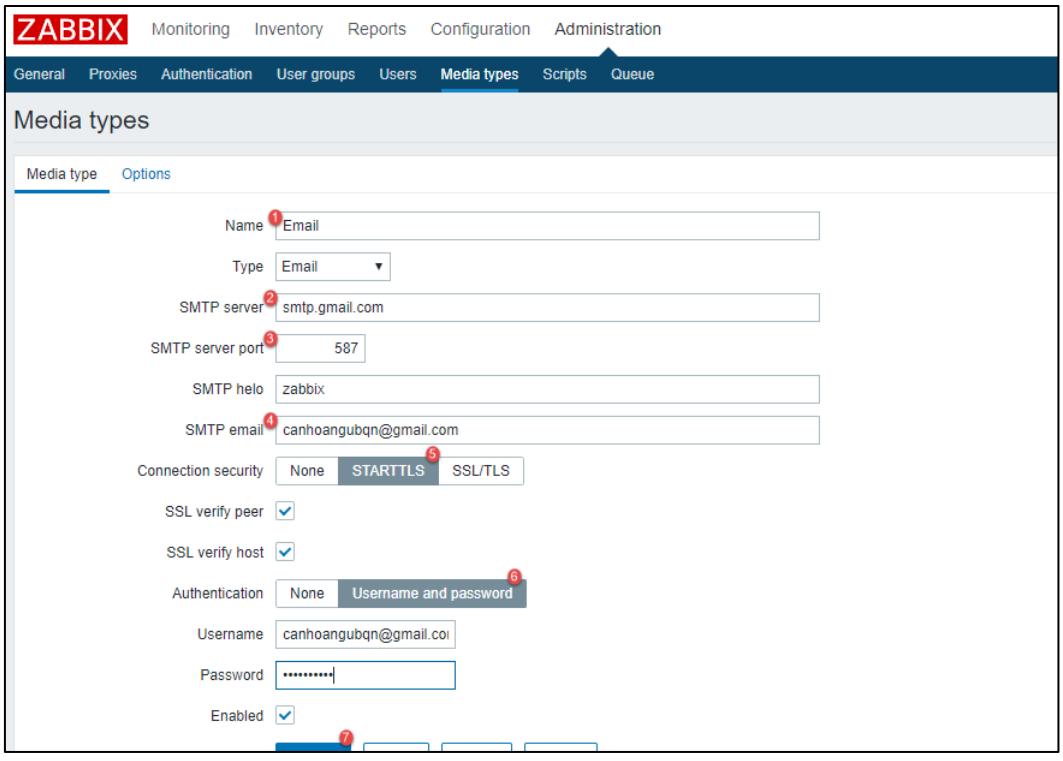
- Frontend messaging: Kích hoạt global notification

- Message timeout: Thời gian hiển thị thông báo tin nhắn trigger. Mặc định 60s
- Play sound: Chỉ định theo 3 mức sau
    + Once: Kêu 1 lần và đầy đủ lượng file âm thanh
    + 10 seconds: Âm thanh sẽ lặp đi lặp lại trong vòng 10s
    + Message timeout: Âm thanh lặp đi lặp lại cho từng thông báo
- Trigger severity: Có thể chỉ định mức trạng thái trigger sẽ được Zabbix thông báo **global noti**. Giả sử chỉ cần **High** và **Disaster** thì chỉ 2 trigger này kích hoạt mới có thông báo âm thanh và khung hiển thị.

**b. Cảnh báo online:**

Gửi mail tới gmail cá nhân người quản trị. Do người quản trị không thể ngồi 24/24 trong phòng quản trị được, hay người quản trị đi vắng. Khi đó chúng ta cần thiết lập 1 hệ thống cảnh báo online, từ đó người quản trị có thể truy cập từ xa và quản trị hệ thống được kịp thời, chính xác hay giảm tối đa thiệt hại khi bị xảy ra sự cố ngoài ý muốn.

Bước 1. Tiến hành cấu hình cho Zabbix server gửi mail cảnh báo vào gmail. Đầu tiên, trên giao diện Zabbix, vào mục Adminstration → Media types → Email:

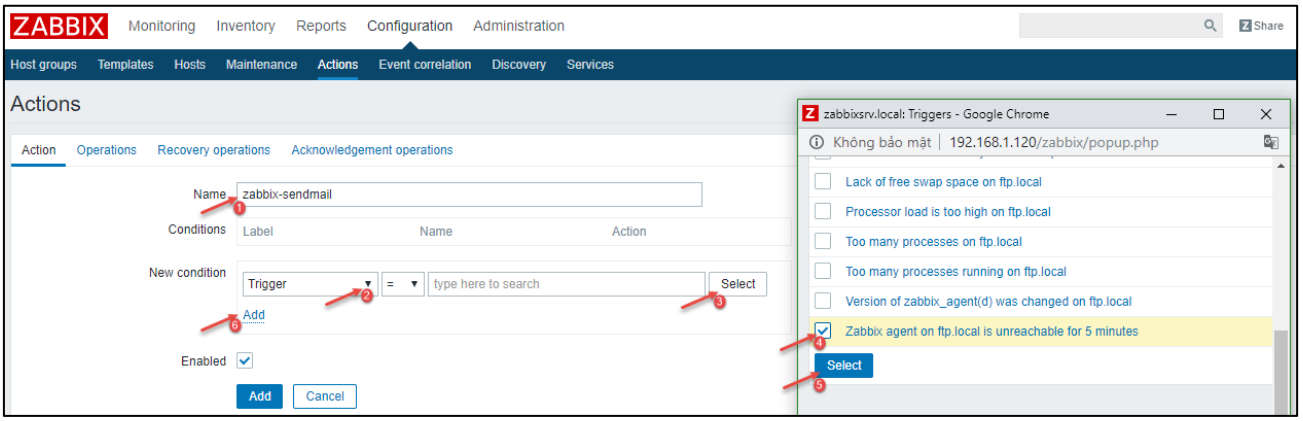



*Hình 4.3.3.2: Thiết lập gửi mail cảnh báo*

Hoàn thành các thông tin khai báo email:

- Name: Đặt tên dịch vụ là Mail hoặc Alert Email…
- SMTP server: Địa chỉ SMTP server (ví dụ: smtp.gmail.com)
- SMTP server port: Port SMTP (ví dụ :465 cho gmail)
- SMTP email: Email trong trường này đóng vai trò Email gửi lỗi "From address"
- Connection security: Giao thức bảo mật tùy theo mail serverUsername/Password: Nhập Email (email này đóng vai trò là email gửi thông báo)

Bước 2. Để hệ thống thực hiện việc gửi email, cần phải tạo các Triggers. Chọn Configuration → Actions → Create action

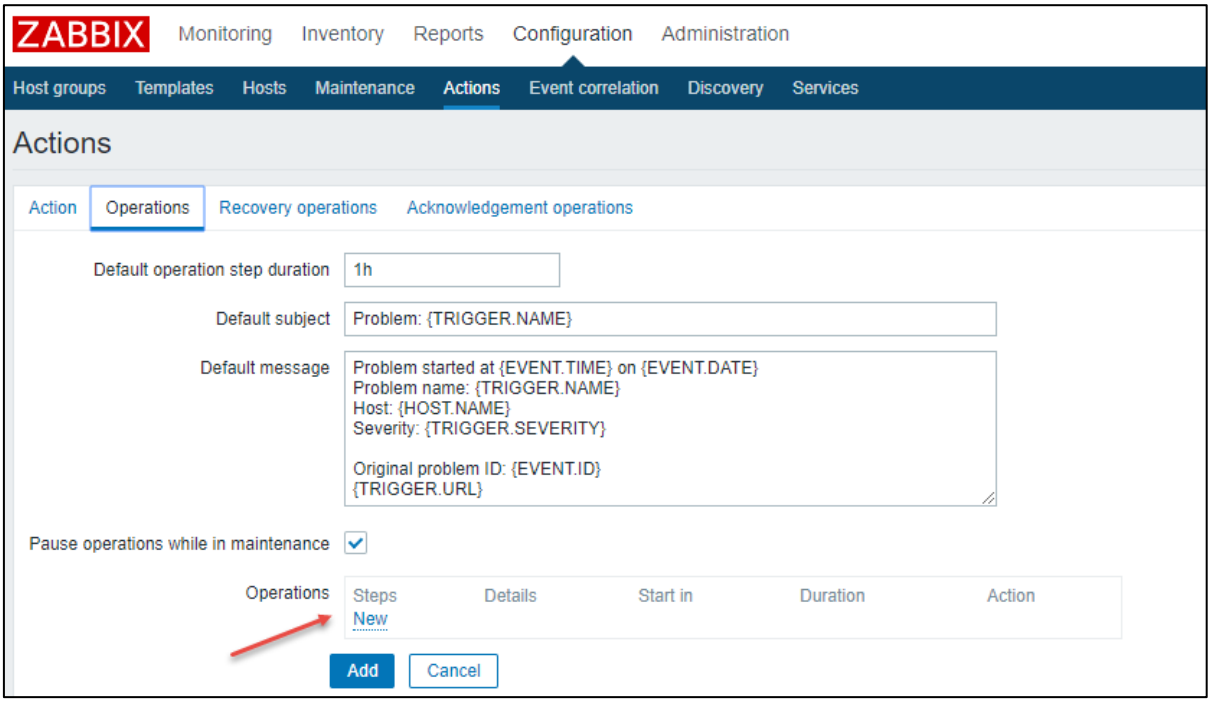



*Hình 4.3.3.3: Thiết lập các Triggers để gửi email thông báo*

Thiết lập thông tin theo dõi

- Name: Tên dịch vụ
- New Condition: Chọn giá trị muốn theo dõi qua mail
- Lựa chọn các điệu kiện muốn theo dõi

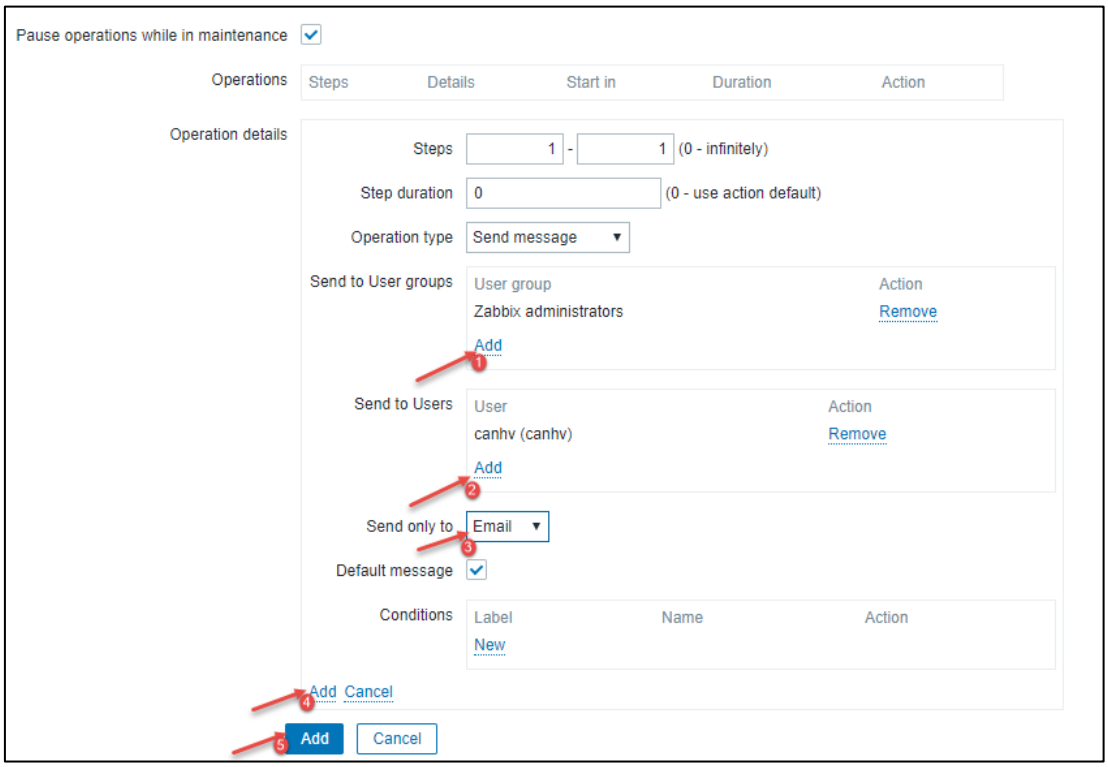
Bước 3. Trong tab **Operations** bao gồm thông tin địa mà hệ thống sẽ gửi về email, các thông tin này sẽ được chèn vào trong vị trí code. Trong mục **Operations** chọn **New**



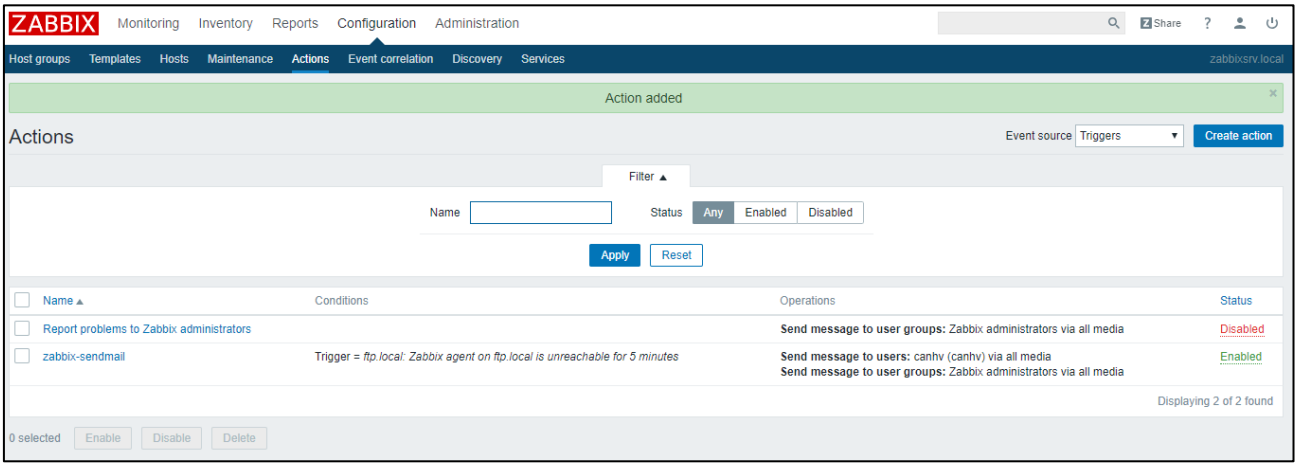

*Hình 4.3.3.4: Thiết lập các hoạt động cảnh báo*

Thiết lập thông tin User Group hoặc nhận thông tin thông báo của hệ thống

- Send to User groups: Nhận thông tin thông báo của hệ thống
- Send to User: Nhận thông tin thông báo của hệ thống
- Send only to: Hình thức gửi




*Hình 4.3.3.5: Thiết lập mail nhận thông báo*




*Hình 4.3.3.6: Hoàn thành thiết lập cảnh báo qua mail*

## 4.4. Kết quả giám sát hệ thống mạng

Xây dựng thành công hệ thống giám sát sử dụng phần mềm Zabbix giải quyết 3 bài toán:

- Giám sát tài nguyên hệ thống dưới dạng biểu đồ trực quan
- Giám sát trạng thái các thiết bị trợ giúp cho việc quản trị của người dùng
- Hệ thống cảnh báo sự cố của Zabbix hoạt động chính xác và kịp thời. Hỗ trợ việc kiểm tra hệ thống và sử dụng các tài nguyên của máy chủ và các thiết bị trong hệ thống.

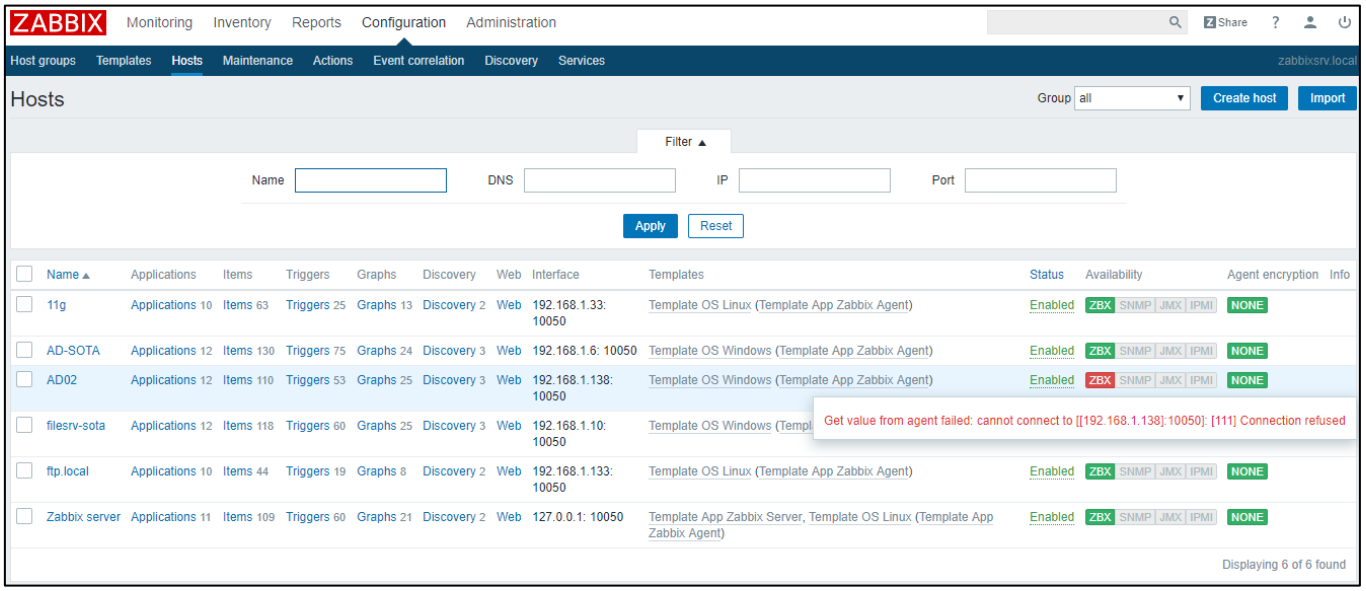
Zabbix là 1 hệ thống mở và hoạt động rất ổn định. Phát hiện chính xác các thay đổi trong hệ thống mạng. Ngoài ra, người dùng có thể tự viết script thực thi việc giám sát các dịch vụ theo ý người quản trị. Tuy nhiên việc cài đặt, cấu hình Zabbix còn rườm rà và mất khá nhiều thời gian và công sức. Cần người quản trị phải hiểu rất rõ về hệ thống cũng như là hiểu về hệ thống giám sát.

### 4.4.1. Giám sát các trạng thái của hosts

#### a. Giám sát trạng thái của host

Theo dõi trạng thái các hosts ở phần Dashboard của Monitoring như hình 4.4.1.1 và hình 4.4.1.1 là hình ảnh hiển thị trạng thái của hosts. Màu xanh báo hiện host đang hoạt động bình thường (up). Còn màu đỏ là hosts đang gặp sự cố mất kết nối hay bị down. Trong hình ảnh các bạn có thể thấy, việc giám sát tôi đã sử dụng zabbix agent (port 10050-10051).

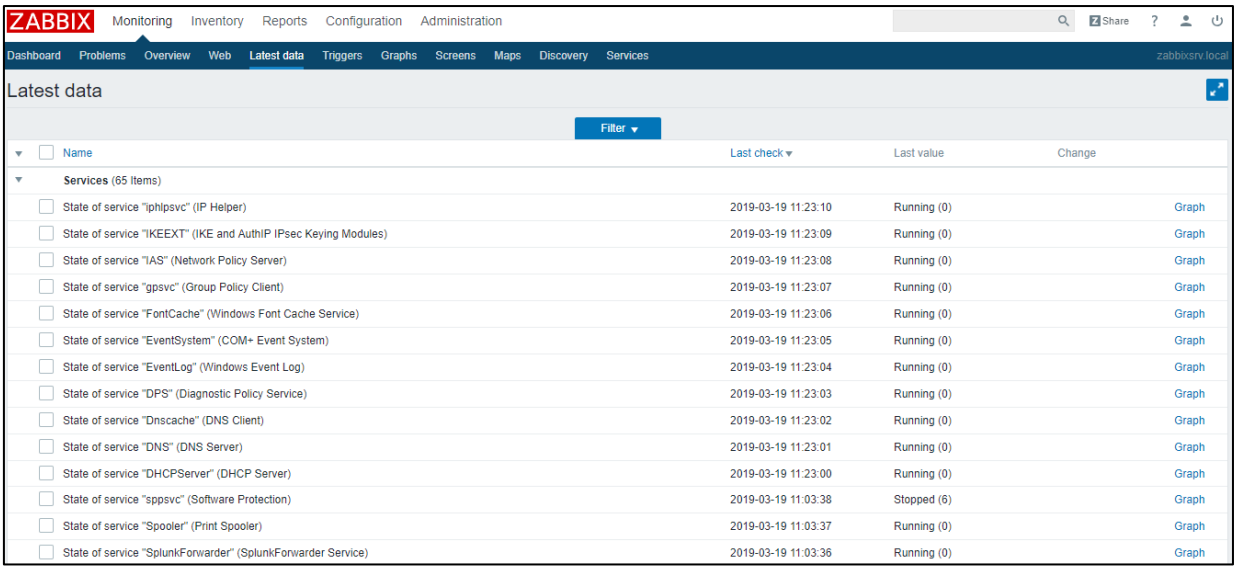
Cụ thể là hình ảnh báo lỗi của zabbix agent: Get value from agent failed: cannot connect to [[192.168.1.138]:10050]: [111] Connection refused – lỗi kết nối không thành tới host AD02 với địa chỉ 192.168.1.138 qua port 10050.




*Hình 4.4.1.1: Trạng thái của hosts*

## b. Giám sát trạng thái hoạt động của các dịch vụ

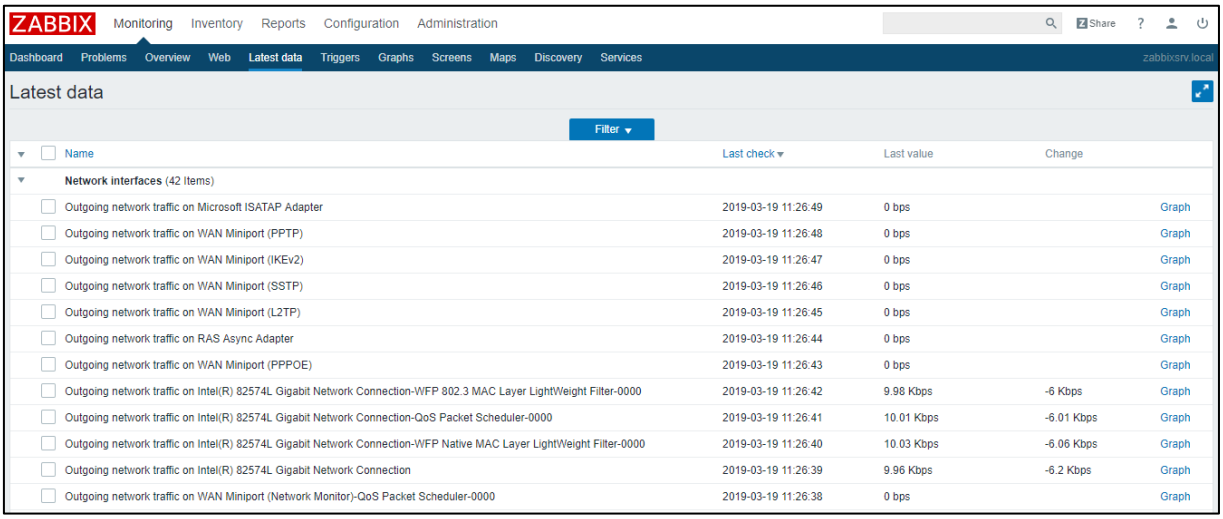
Trong zabbix (Web) việc việc hiện thị trạng thái dịch vụ được thể hiện trong mục Latest data của Monitoring




*Hình 4.4.1.2: Hình ảnh trạng thái các dịch vụ*

Trong hình ảnh trên chúng ta quan sát thấy tình trạng của các thiết bị đang running như dịch vụ DNS, DHCP, hay đang stop như Software Protection.

### c. Trạng thái của interfaces

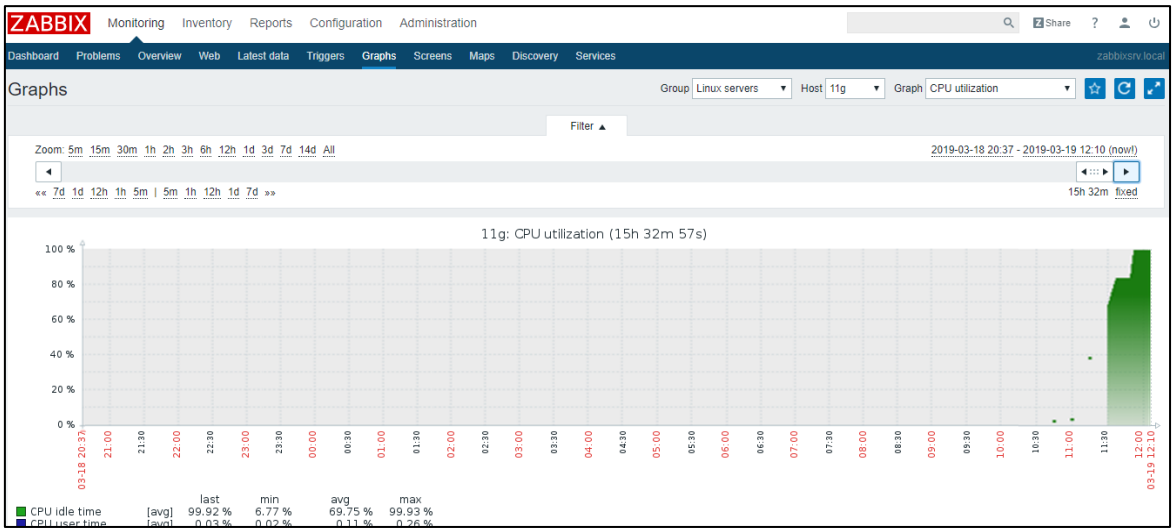



*Hình 4.4.1.3: Trạng thái của interface trên host AD-SOTA*

Chúng ta có thể quan sát được các interface, cổng mạng của Intel(R) hay các host như isatap …

## 4.4.2. Giám sát tài nguyên của host

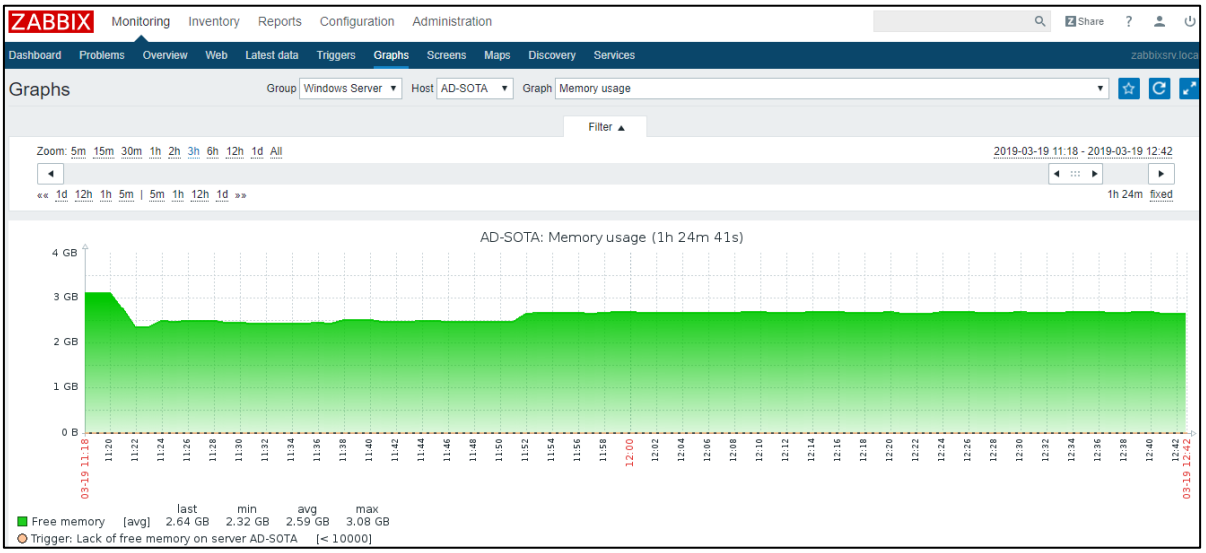
### a. Tài nguyên CPU




*Hình 4.4.2.1: Biểu đồ sử dụng CPU của host 11g*

Dựa vào biểu đồ trên chúng ta quan sát được lúc 11h30 CPU sử dụng 70% dung lượng, đến 12h00 CPU sử dụng là 100%, từ đó chúng ta có thể thiết lập các chế độ cảnh báo về việc sử dụng tài nguyên CPU.

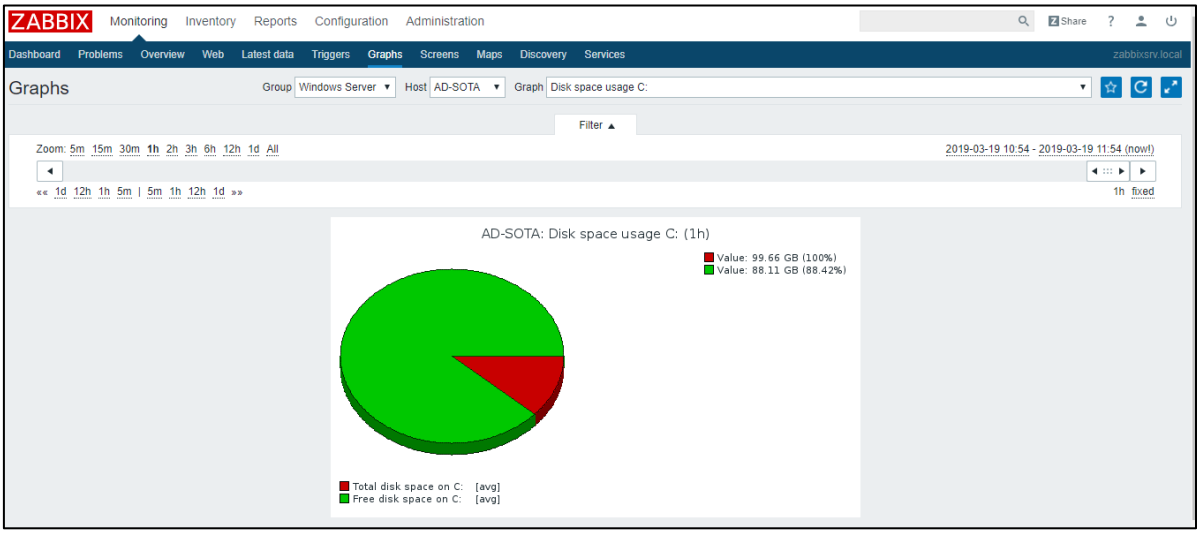
## b. Tài nguyên RAM




*Hình 4.4.2.2: Biểu đồ sử dụng RAM của AD-SOTA*

Dựa vào biểu đồ trên chúng ta quan sát được lúc 12h42 dung lượng Ram đang dùng đến là 2.64 GB/ 4GB (hay 4GB tương ứng tổng số dung lượng RAM).

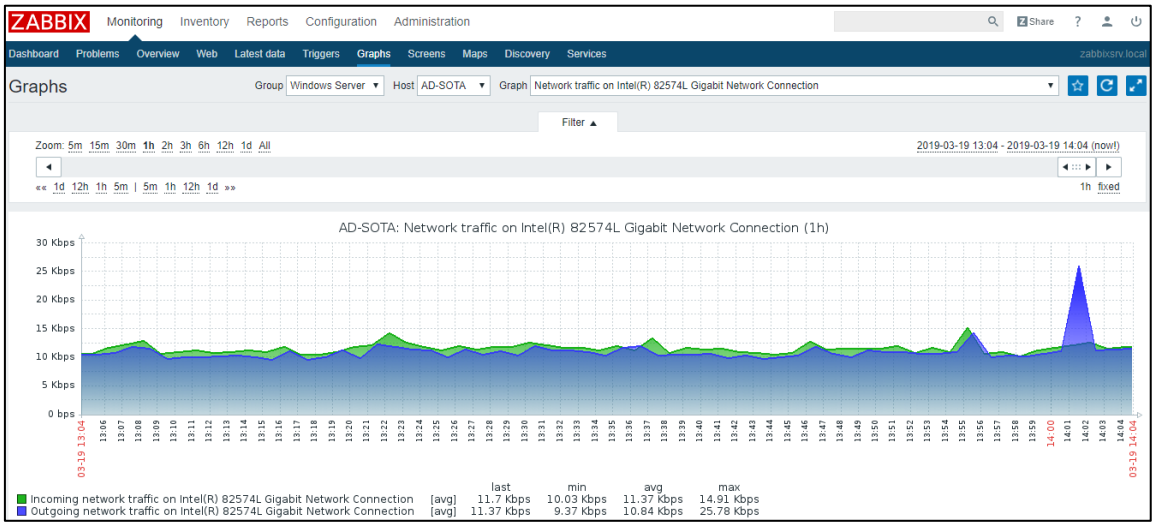
## c. Tài nguyên Disk




*Hình 4.4.2.3: Biểu đồ sử dụng ổ cứng (ổ C) của host AD-SOTA*

Trên hình 4.4.2.3 chúng ta biết được tổng dung lượng ổ C là 100GB (màu đỏ) 100% vì tại AD-SOTA đang 1 ổ C duy nhất. Màu xanh lá cây là lưu lượng còn trống 88.42%.

### 4.4.3. Giám sát lưu lượng mạng trên các host

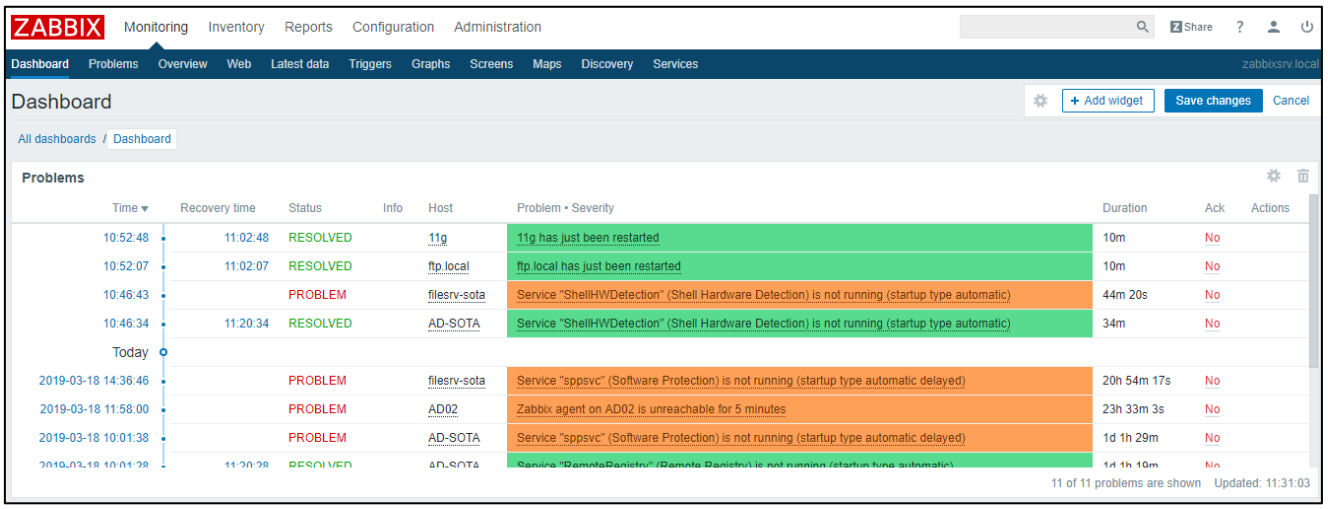



*Hình 4.4.3.1: Biểu đồ lưu lượng vào ra trên 1 interface*

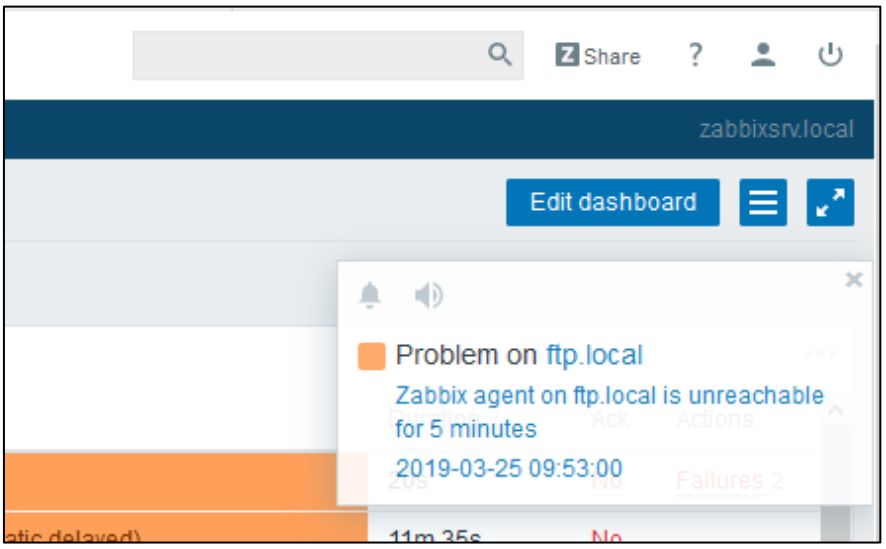
Lưu lượng vào (incomming) tương ứng với màu xanh lá cây đạt đỉnh điểm lúc 13h55 phút với tốc độ là 14.91 Kbps và màu xanh lam tương ứng với lưu lượng ra (outgoing) là 25.78 Kbps.

### 4.4.4. Cảnh báo sự cố

#### a. Cảnh báo trạng thái của host hay dịch vụ trên host bị down hay thay đổi

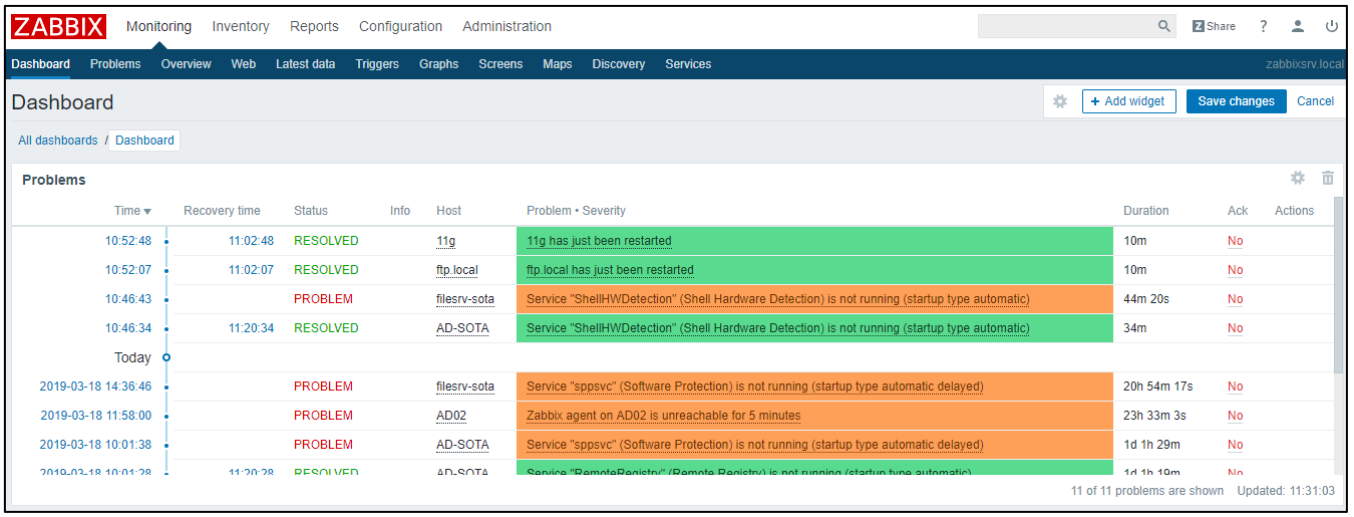



*Hình 4.4.4.1: Màn hình hiển thị cảnh báo khi host AD02 không thể kết nối*




*Hình 4.4.4.2: Hiện thị chuông cảnh báo*

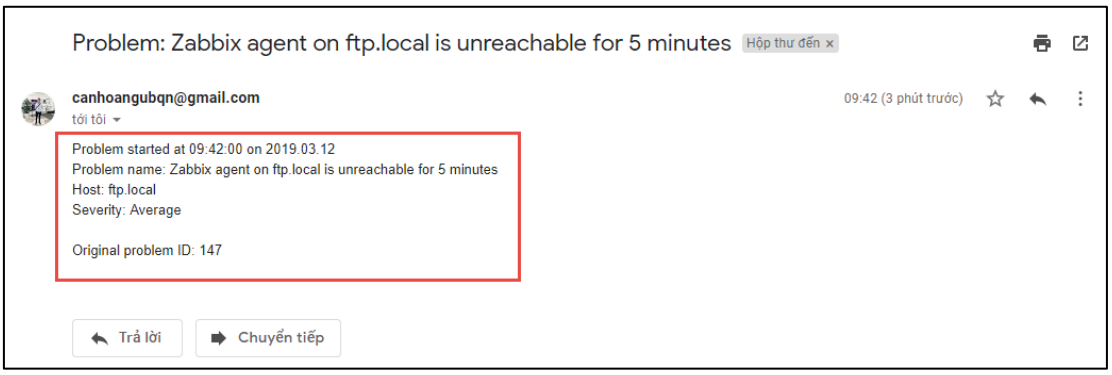
Chúng ta có thể thấy rõ là host AD02 đang không thể truy cập, và thời gian cụ thể host này mất kết nối.




*Hình 4.4.4.3: Cảnh báo khi dịch vụ sppsvc trên host filesrv-sota không khởi động*

Tượng tự như cảnh báo trạng thái của host thì cảnh báo services cũng hiện thị đầy đủ tên services, lỗi đang xảy ra và thời gian xảy ra 1 cách cụ thể trực quan.

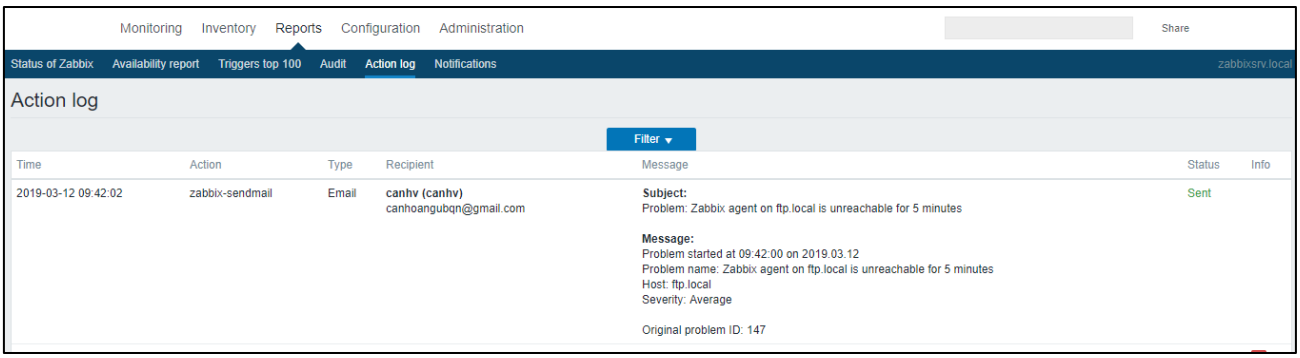
## b. Cảnh báo qua mail




*Hình 4.4.4.4: Cảnh báo qua mail khi server shutdows quá thời gian truy cập*

Trong hình 4.4.4.5, chúng ta có thể thấy được thời gian truy cập server là 5 phút. Các thông số chúng ta quan tâm phần message gửi về bao gồm:

- Problem: là biểu hiện của cảnh báo
- Severity: mức độ cảnh báo gồm 5 cấp độ: Information, Warning, Average, High, Disaster, trong đó Disaster là cấp độ cao nhất
- Item values: là những giá trị mà của cảnh báo
- Time là thời điểm nhận được thông báo
- Status: là tình trạng thông báo có 3 biểu hiện:
    + Falied: lỗi không gửi được thông báo
    + Sent: đã được gửi đi
    + Proccess: là đang được xử lý
- Recipient: ở đây hệ thống được thiết lập gửi thông báo tới Admin và địa chỉ gmail là canhoanguqbn@gmail.com.




*Hình 4.4.4.5: Chi tiết cảnh báo*

# KẾT LUẬN

Trong đồ án này em đã nghiên cứu, tìm hiểu xây dựng hệ thống giám sát mạng dựa trên phần mềm nguồn mở Zabbix.

Đồ án thực hiện thành công mô hình giám sát mạng sử dụng phần mềm Zabbix đáp ứng được cơ bản yên cầu quản trị mạng của 1 hệ thống bao gồm:

+ Quản lý được dữ liệu quan trọng, và các thông tin cơ bản của các thiết bị trong hệ thống kịp thời chính xác.
+ Xây dựng thành công cơ chế cảnh báo online 1 và hiện thị cảnh báo trên màn hình tiện ích quan trọng cho việc quản trị.

Em đã áp dụng các kiến thức của các môn học như quản trị mạng, mạng máy tính … vào mô hình và thực tế. Đã học hỏi được thêm nhiều kinh nghiệm về cách thức tổ chức, xây dựng hệ thống giám sát cũng như quy hoạch hệ thống .Tuy nhiên, do thời gian và khả năng có hạn, nên em chưa đi sâu tìm hiểu được thêm những vấn đề cần thiết của hệ thống. Em đã cố gắng nhưng mô hình mới chỉ dừng ở mức độ theo dõi, giám sát máy chủ như giám sát tài nguyên máy, dung lượng traffic, tình trạng của host.

Trong thời gian tới em sẽ phát triển và nghiên cứu sâu hơn về hệ thống giám sát mạng Zabbix và các công cụ hỗ trợ giám sát mạng, giám sát sâu hơn nhưng vấn đề cần thiết của hệ thống. Phát triển các chức năng trên Zabbix như: giám sát hạ tầng mạng bao gồm các thiết bị router, switch, firewall,… Cảnh báo qua SMS.

# TÀI LIỆU THAM KHẢO

[1]. https://vi.wikipedia.org/

[2]. http://www.vncert.gov.vn/baiviet.php?id=4

[3]. https://github.com/Ducnm37/All/blob/master/Monitor/SNMP.md

[4]. https://www.zabbix.com/documentation/3.4/ trang chủ tài liệu của zabbix hỗ trợ người dùng.

[5].http://luanvan123.info/threads/nghien-cuu-va-trien-khai-he-thong-giam-sat-ha-tang-cntt-tren-phan-mem-ma-nguon-mo-zabbix.95114/

[7]. Essential SNMP / Douglas R. Mauro and Kevin J. Schmidt

[8]. SMNP toàn tập – Nguyễn Thanh Diệp