TRƯỜNG ĐẠI HỌC BÁCH KHOA HÀ NỘI

VIỆN CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG

──────── \* ───────

ĐỒ ÁN

**TỐT NGHIỆP ĐẠI HỌC**

NGÀNH CÔNG NGHỆ THÔNG TIN

**TÊN ĐỀ TÀI**

**GIÁM SÁT HỆ THỐNG MẠNG BẰNG PHẦN MỀM MÃ NGUỒN MỞ NAGIOS**

Sinh viên thực hiện : **Đỗ Ngọc Ngoạn**

Lớp Truyền Thông và Mạng Máy Tính – K52

Giáo viên hướng dẫn: **Th.S Bành Thị Quỳnh Mai**

HÀ NỘI 01-2013

# PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP

1. Thông tin về sinh viên

Họ và tên sinh viên : Đỗ Ngọc Ngoạn.

Điện thoại liên lạc : 0904060288 Email : ngocngoan060288@gmail.com

Lớp : Truyền thông và Mạng máy tính - K52 Hệ đào tạo : Đại học

Đồ án tốt nghiệp được thực hiện tại:

Viện Công nghệ Thông tin và Truyền thông – Đại học Bách Khoa Hà Nội

Thời gian làm ĐATN: Từ ngày 20 /08 /2012 đến 02 /01 /2013

2. Mục đích nội dung của ĐATN

* Nắm được các kiến thức về giám sát mạng, giao thức giám sát mạng.
* Hiểu về cách thức thực hiện giám sát mạng của một số phần mềm nguồn mở.
* Là cơ sở để triển khai hệ thống giám sát mạng trong thực tiễn.

3. Các nhiệm vụ cụ thể của ĐATN

* Xây dựng nền tảng về lý thuyết trong lĩnh vực quản lý, giám sát hệ thống mạng, cũng như kiến thức về các phần mềm giám sát mạng.
* Xây dựng hệ thống giúp giám sát các thiết bị và dịch vụ cơ bản của mạng, thông báo kịp thời các sự cố, cảnh báo cho quản trị viên.

4. Lời cam đoan của sinh viên:

Tôi - *Đỗ Ngọc Ngoạn* - cam kết ĐATN là công trình nghiên cứu của bản thân tôi dưới sự hướng dẫn của *ThS. Bành Thị Quỳnh Mai*.

Các kết quả nêu trong ĐATN là trung thực, không phải là sao chép toàn văn của bất kỳ công trình nào khác.

|  |  |
| --- | --- |
|  | *Hà Nội, ngày 02 tháng 01 năm 2013*  Tác giả ĐATN  *Đỗ Ngọc Ngoạn* |

5. Xác nhận của giáo viên hướng dẫn về mức độ hoàn thành của ĐATN và cho phép bảo vệ:

|  |  |
| --- | --- |
|  | *Hà Nội, ngày 02 tháng 01 năm 2013*  Giáo viên hướng dẫn  *ThS. Bành Thị Quỳnh Mai* |

# TÓM TẮT NỘI DUNG ĐỒ ÁN TỐT NGHIỆP

Đồ án gồm 4 chương với các nội dung sau :

**Chương 1:** **Tổng quan về giám sát và giao thức giám sát mạng.**

Chương đầu tiên này em sẽ trình bày lý thuyết nền tảng về giám sát hạ tầng mạng, và giao thức SNMP phục vụ cho việc giám sát mạng.

**Chương 2: Các thành phần trong hệ thống giám sát mạng nguồn mở Nagios.**

Chương này em sẽ trình bày về hệ thống giám sát mạng mã nguồn mở Nagios và cơ chế hoạt đông và các thành phần cần thiết trong hệ thống, được sử dụng trong quá trình xây dựng hệ thống.

**Chương 3: Triển khai giám sát các tài nguyên, các dịch vụ mạng đã được cài đặt đối với hệ thống và đưa ra cảnh báo đối với người quản trị viên**

Chương 3 em sẽ tập trung vào việc cài đặt hệ thống kiểm thử, đánh giá hoạt động của hệ thống.

**Chương 4: thực nghiệm các kịch bản giám sát với các kết quả đạt được**

Cuối cùng em trình bày tóm tắt các nội dung kết quả đã làm được, đồng thời nêu các thiếu sót, hạn chế và hướng phát triển cho đồ án trong tương lai.

# MỤC LỤC

[PHIẾU GIAO NHIỆM VỤ ĐỒ ÁN TỐT NGHIỆP 2](#_Toc344790741)

[TÓM TẮT NỘI DUNG ĐỒ ÁN TỐT NGHIỆP 4](#_Toc344790742)

[MỤC LỤC 5](#_Toc344790743)

[LỜI NÓI ĐẦU 8](#_Toc344790744)

[Chương 1 : Tổng quan về giám sát mạng 10](#_Toc344790745)

[1.1. Giới thiệu về giám sát mạng 10](#_Toc344790746)

[1.1.1. Khái niệm 10](#_Toc344790747)

[1.1.2. Mục đích của giám sát mạng 10](#_Toc344790748)

[1.1.3. Các tài nguyên giám sát 11](#_Toc344790749)

[1.2. Giao thức giám sát mạng SNMP 11](#_Toc344790750)

[1.2.1. Tổng quát về SNMP. 11](#_Toc344790751)

[1.2.2. Ưu điểm của SNMP. 12](#_Toc344790752)

[1.2.3. Kiến trúc của SNMP. 12](#_Toc344790753)

[1.2.4. Cơ sở dữ liệu quản trị (MIB) của SNMP. 13](#_Toc344790754)

[1.2.5. Các phương thức của SNMPv1. 15](#_Toc344790755)

[1.3. Định hướng giải quyết vấn đề 16](#_Toc344790756)

[Chương 2 : Công cụ giám sát hệ thống mạng mã nguồn mở Nagios 18](#_Toc344790757)

[2.1. Giới thiệu về Nagios 18](#_Toc344790758)

[2.2. Cơ chế và kiến trúc hoạt động của Nagios 19](#_Toc344790759)

[2.2.1. Cơ chế hoạt động 19](#_Toc344790760)

[2.2.2. Kiến trúc của Nagios 20](#_Toc344790761)

[2.3. Các mô hình giám sát phổ biến 20](#_Toc344790762)

[2.4. Hệ thống các file cấu hình của Nagios 21](#_Toc344790763)

[2.5. Một số các plugin và Add-on của Nagios 22](#_Toc344790764)

[2.6. Triển khai Nagios, các Plugin và các Add-on 24](#_Toc344790765)

[2.6.1. Triển khai trên Nagios Server 24](#_Toc344790766)

[2.6.2. Triển khai trên host Linux 28](#_Toc344790767)

[2.6.3. Triển khai trên host Windows 30](#_Toc344790768)

[2.7. Cấu hình Nagios 31](#_Toc344790769)

[2.7.1. Các file cấu hình mẫu 31](#_Toc344790770)

[2.7.2. Các cảnh báo trong Nagios 31](#_Toc344790771)

[2.7.3. Bộ xử lý sự kiện 32](#_Toc344790772)

[2.7.4. Flapping 33](#_Toc344790773)

[2.7.5. Quan hệ cha/con giữa các host 34](#_Toc344790774)

[Chương 3 : Xây dựng và hệ thống giám sát mạng với Nagios 36](#_Toc344790775)

[3.1. Mô hình giám sát mạng 36](#_Toc344790776)

[3.2. Giám sát tài nguyên mạng 38](#_Toc344790777)

[3.2.1. Cấu hình hệ thống giám sát 38](#_Toc344790778)

[3.2.2. Giám sát tài nguyên của các máy tham gia trong mạng 39](#_Toc344790779)

[3.2.3. Giám sát các tài nguyên trên máy chủ Windows 41](#_Toc344790780)

[3.3. Giám sát các dịch vụ MySQL, LDAP, Jboss, HSM Simulator 42](#_Toc344790781)

[3.3.1. Giám sát dịch vụ MySQL Cluster 43](#_Toc344790782)

[3.3.2. Giám sát dịch vụ LDAP 44](#_Toc344790783)

[3.3.3. Giám sát dịch vụ Jboss và HSM Simiulator 45](#_Toc344790784)

[3.4. Triển khai cảnh báo qua SMS 46](#_Toc344790785)

[3.4.1. Chuẩn bị cài đặt 47](#_Toc344790786)

[3.4.2. Các bước thực hiện 48](#_Toc344790787)

[3.5. Tích hợp công cụ giám sát bằng đồ thị, Nagiosgraph 53](#_Toc344790788)

[Chương 4 : Thực nghiệm giám sát hệ thống mạng 56](#_Toc344790789)

[4.1. Các dịch vụ giám sát 56](#_Toc344790790)

[4.1.1. Cơ chế giám sát 56](#_Toc344790791)

[4.1.2. Nội dung cấu hình cho nhiều máy 57](#_Toc344790792)

[4.2. Thử nghiệm cơ chế cảnh báo qua SMS 58](#_Toc344790793)

[4.2.1. Lọc cảnh báo 58](#_Toc344790794)

[4.2.2. Nội dung cảnh báo 59](#_Toc344790795)

[4.3. Xử lý sự cố tự động 60](#_Toc344790796)

[4.3.1. Xử lý các sự kiện (Event Handler) 60](#_Toc344790797)

[4.3.2. Shell Script khởi động lại dịch vụ 60](#_Toc344790798)

[4.4. Thử nghiệm mô hình Failover Monitoring 62](#_Toc344790799)

[4.5. Đánh giá hệ thống và kết quả đã đạt được 64](#_Toc344790800)

[4.5.1. Các kết quả thu được 64](#_Toc344790801)

[4.5.2. Hướng phát triển 65](#_Toc344790802)

[DANH MỤC CÁC HÌNH VẼ 66](#_Toc344790803)

[DANH MỤC CÁC BẢNG 67](#_Toc344790804)

[DANH MỤC CÁC TỪ VIẾT TẮT 68](#_Toc344790805)

[PHỤ LỤC 69](#_Toc344790806)

[Phụ lục 1 : Định nghĩa các cấu hình cơ bản 69](#_Toc344790807)

[Phụ lục 2 : Các script thực hiện trong đồ án 71](#_Toc344790808)

[TÀI LIỆU THAM KHẢO 73](#_Toc344790809)

# LỜI NÓI ĐẦU

Vào những năm 80 của thế kỉ 20, sự ra đời của mạng máy tính đã tạo nên một bước phát triển nhảy vọt trong lĩnh vực thông tin liên lạc và khả năng chia sẻ tài nguyên. Sự ra đời của mạng máy tính đã hình thành nên một cách thức, một khái niệm, một cái nhìn mới về thông tin liên lạc, chia sẻ tài nguyên, thương mại, dịch vụ … và làm thay đổi gần như tất cả mọi mặt của đời sống xã hội.

Ngày nay, công nghệ thông tin đang được các tổ chức áp dụng rộng rãi, nhằm tối ưu hóa quá trình thông tin liên lạc, chia sẻ trong toàn tổ chức. Với một hệ thống mạng, tất cả các thiết bị tin học trong một tổ chức được liên kết với nhau. Trước khi có mạng, trong các tổ chức, mọi nơi đều phải có kho lưu trữ dữ liệu riêng, do đó các thông tin trong nội bộ không được cập nhật kịp thời; một ứng dụng ở nơi này không thể chia sẻ cho nơi khác. Nhưng với một hệ thống mạng :

* Các thiết bị có thể chia sẻ tài nguyên : giữa các ứng dụng, giữa các kho dữ liệu hay chia sẻ sức mạnh của bộ xử lý…
* Tăng độ tin cậy và an toàn thông tin : vì thông tin được cập nhật theo thời gian thực, khi một máy trạm bị sự cố, các máy còn lại vẫn có khả năng cung cấp dịch vụ chính xác.
* Tiết kiệm : do có thể tận dụng khả năng của hệ thống, chuyên môn hóa các máy tính, chia sẻ tài nguyên .v.v.
* Mạng máy tính còn là một phương tiện thông tin mạnh mẽ trong tổ chức.

Với tốc độ phát triển nhanh chóng cùng với sự tích hợp ngày càng nhiều các dịch vụ tiện ích, mạng máy tính ngày càng trở nên phức tạp và khó kiểm soát. Đặc biệt đối với một hệ thống chứng thực số thì việc giám sát các hoạt động là rất quan trọng bới nếu như có một sự cố xảy ra mà người quản trị không biết hoặc hệ thống không tự khắc phục thì sẽ ảnh hướng nghiêm trọng đến các doanh nghiệp. Thông tin các khách hàng không đúng ảnh hưởng rất lớn đến các giao dịch thương mại, như là các hệ thống thương mại điện tử, hệ thống thanh toán qua mạng, … Vì vậy nhu cầu quản lý, giám sát hoạt động của mạng máy tính cũng như các dịch vụ của nó trở thành yêu cầu tất yếu. Rất nhiều phần mềm giám sát mạng đã được phát triển nhằm đáp ứng nhu cầu giám sát tự động các hệ thống mạng. Với một chi phi thấp và khả năng cấu hình linh hoạt, các phần mềm giám sát mạng mã nguồn mở là một lựa chọn tốt.

Trước những yêu cầu đó, đồ án “**Giám sát hệ thống mạng dựa trên phần mềm mã nguồn mở**” được chọn không chỉ nhằm mục đích nghiên cứu mà còn nhằm phát triển, ứng dụng giám sát hệ thống mạng trong thực tiễn.

Đồ án được xây dựng hướng đến các mục tiêu sau :

* Xây dựng nền tảng về lý thuyết trong lĩnh vực quản lý, giám sát hệ thống mạng. Cũng như kiến thức về các phần mềm giám sát mạng.
* Xây dựng hệ thống giúp giám sát các thiết bị và dịch vụ mạng chứng thực số, như LDAP, MySQL Cluster, Jboss, HSM Simulator.
* Tích hợp các chức năng phối hợp với việc giám sát mạng như vẽ đồ thị, thông báo kịp thời các sự cố, cảnh báo cho quản trị viên qua SMS hoặc Email, khắc phục các sự cố tự động nếu như hệ thống lỗi.

Với những mục tiêu trên, Đồ án được chia thành 2 nội dung cơ bản :

* Cơ sở lý thuyết về giám sát mạng và các phần mềm giám sát mạng nguồn mở.
* Thực nghiệm giám sát hệ thống mạng mô phỏng với phần mềm nguồn mở là Nagios.

Các nội dung được bố cục trong các chương

# Chương 2 : Công cụ giám sát hệ thống mạng mã nguồn mở Nagios

Giám sát mạng là một chức năng đa dạng với nhiều nhiệm vụ mà người quản trị hệ thống mạng máy tính cần phải thực hiện. Và để đơn giản hóa các tác vụ và nâng cao hiệu năng của hệ thống mạng, trong luận án, này em xin đề cập đến phần mềm Nagios, một công cụ mã nguồn mở khá đầy đủ cho việc giám sát mạng toàn diện. Chương này cung cấp các kiến thức cơ bản nhưng rất quan trọng đối với người quản trị mạng về công cụ này.

## Giới thiệu về Nagios

Nagios là một ứng dụng quản trị mạng nguồn mở, giám sát các máy trạm và các dịch vụ, cảnh báo khi có sự cố hoặc khi sự cố được khắc phục. Nagios được phát triển để chạy trên nền tảng Linux, có tính linh hoạt cao trong việc cấu hình các dịch vụ kiểm tra, các cấu hình có thể được đặt trong một file duy nhất hoặc đặt trong các file riêng lẻ cho từng thiết bị khác nhau, hay những mục đích khác nhau.

Nagiso cung cấp các cơ chế :

* Giám sát các dịch vụ mạng (như HTTP, DNS, SNMP, POP3, IMAP, …)
* Giám sát các tài nguyên mạng (như tình trạng ổ đĩa, CPU, …) bao gồm cả hệ điều hành Microsoft Windows với plugin là NSClient++.
* Giám sát bất kỳ thông nào khác (chẳng hạn như nhiệt độ, thông báo, …) có khả năng gửi một tập dữ liệu thông qua mạng tới một plugin xác định.
* Giám sát thông qua việc chạy các kịch bản từ xa như Nagios Remote Plugin Executor (NRPE).
* Giám sát từ xa thông qua SSH hay SSL.
* Cho phép người quản trị có thể phát triển thêm các plugin tích hợp để giám sát những dịch vụ theo tiêu chí của riêng họ thông qua các ngôn ngữ kịch bản (Shell, Python, Perl, Ruby, PHP, …).
* Cho phép định nghĩa các host trong mạng theo thứ tự phân cấp cha-con nhằm phân biệt và phát hiện các host trong tình trạng “DOWN” hay “UNREACHABLE”.
* Thông báo tới tài khoản người dùng khi dịch vụ hoặc host có vấn đề xảy ra để có hướng giải quyết cụ thể.
* Nagios còn có khả năng định nghĩa các xử lý sự kiện nhằm khắc phục sự cố tự động khi dịch vụ hay host gặp sự cố.
* Tự động tạo ra các file log có thể theo dõi theo ngày, tháng.
* Hỗ trợ việc thực thi các host giám sát dự phòng.
* Môi trường web-base cho phép người dùng xem được tình trạng mạng, các cảnh báo, lịch sử các sự cố, …
* Lưu trữ dữ liệu thông qua các tập tin văn bản thay vì sử dụng cơ sở dữ liệu nào khác.

## Cơ chế và kiến trúc hoạt động của Nagios

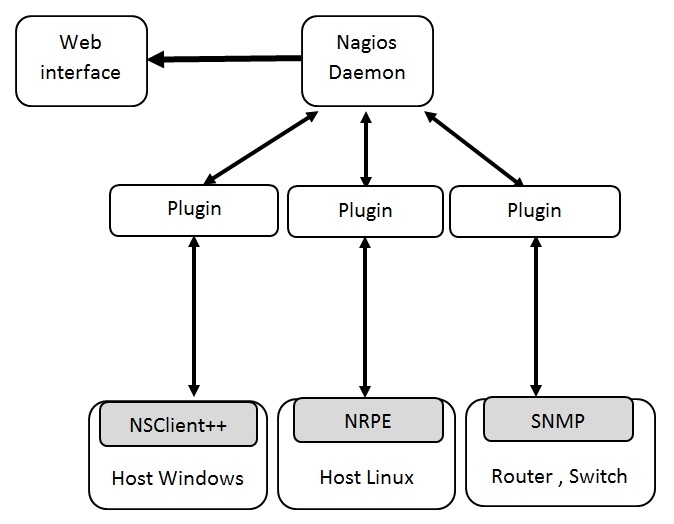
### Cơ chế hoạt động

Toàn bộ hệ thống Nagios có thể chia thành 2 thành phần cơ bản đó là:

* *Nagios Daemon:* làm nhiệm vụ chính là lập lịch kiểm tra định kỳ, nhận kết quả từ các Plugin gửi về và tiến hành phân tích kết quả, hiển thị trên giao diện Web hoặc thông báo tới người quản trị qua Email hoặc tin nhắn SMS.
* *Plugins:* làm nhiệm vụ chính là nhận điều khiển yêu cầu kiểm tra từ Daemon, thực thi các lệnh được yêu cầu, thu thập kết quả kiểm tra và gửi trả lại kết quả cho Daemon.

Nagios còn làm việc với các Add-on, đó là các công cụ mở rộng phục vụ giám sát mạng cho một khía cạnh nào đó. Chẳng hạn như giám sát NRPE sử dụng cho giám sát các host Linux trong khi NSClient++ giám sát các host Windows hay Nagiosgraph thực hiện thu thập số liệu và đưa ra các biểu đồ, …

Ngoài ra, Nagios còn cung cấp một giao diện Web để hiển thị thông tin giám sát. Đây là một tùy chọn trong cài đặt Nagios.



Hình 6 : Sơ đồ tương tác giữa các thành phần trong Nagios

Việc Nagios sử dụng các Plugin riêng biệt kiểm tra cho từng dịch vụ đảm bảo khả năng kiểm tra chính xác trạng thái dịch vụ, cũng như đơn giản hóa việc thực thi lệnh kiểm tra, xử lý kết quả. Các Plugin làm nhiệm vụ kết nối tới các thiết bị, có thể trực tiếp thực thi các lệnh kiểm tra (với các dịch vụ mạng public như HTTP, FTP, POP3 .v.v.), hoặc kết nối với các Daemon đặc biệt được cài đặt trên các thiết bị để chạy các lệnh kiểm tra tài nguyên cục bộ của thiết bị (như Disk, CPU, Memory .v.v. ).

Với các tài nguyên cục bộ của thiết bị từ xa, Nagios không có cơ chế trực tiếp để kiểm tra tình trạng. Do đó, nó cần kết nối với các Plugin được cài đặt cục bộ trên thiết bị từ xa đó để thực thi lệnh kiểm tra. Kết nối đó có thể qua SSH, hay các Daemon đặc biệt chạy trên các nền hệ điều hành khác nhau. Với các máy trạm cài Windows, để kiểm tra được các tài nguyên cục bộ, NSClient++ cần được cài đặt. Trên các máy trạm Linux, có thể sử dụng NRPE để thực thi các plugin từ xa. Các thiết bị định tuyến và các thiết bị khác có hỗ trợ SNMP có thể được kiểm tra thông qua các thông điệp SNMP

### Kiến trúc của Nagios

Hệ thống Nagios bao gồm máy chủ Nagios (Nagios Daemon), hệ thống các Plugin, các file cấu hình, cơ sở dữ liệu lưu trữ trạng thái và log file.

* *Nagios Daemon*: có chức năng lập lịch kiểm tra và xử lý kết quả.
* *Cơ sở dữ liệu lưu trữ trạng thái* (State Retention Database): lưu trữ thông tin trạng thái của các thiết bị và dịch vụ khi các dịch vụ kiểm tra được thực thi. Nó cũng được sử dụng để lưu trữ thông tin trạng thái khi máy chủ khởi động lại.
* *Plugin*: thực hiện những lệnh kiểm tra.
* *File cấu hình* (Configuration file): là những file văn bản rõ chứa những cài đặt, những định nghĩa máy trạm và dịch vụ sẽ được kiểm tra.
* *File nhật ký* (Log file): nhật ký lưu trữ kết quả của việc kiểm tra.

## Các mô hình giám sát phổ biến

Trên thực tế, vì một lý do nào đó mà máy chủ giám sát không hoạt động được, người ta sử dụng thêm ít nhất một máy chủ bổ xung. Máy chủ này có nhiệm vụ đảm bảo cho hệ thống giám sát luôn hoạt động ổn định và có khả năng chịu tải cũng như khắc phục sự cố khi máy chủ giám sát chính gặp sự cố. Tùy thuộc vào từng mô hình hệ thống mạng cụ thể mà sử dụng máy chủ giám sát phụ với các mục đích khác nhau. Thông thường có ba mục đích khác nhau đối với máy chủ giám sát bổ xung. Vì thế, mô hình giám sát mạng được chia thành ba loại theo ba mục đích như vậy sau.

Ba mô hình giám sát mạng gồm:

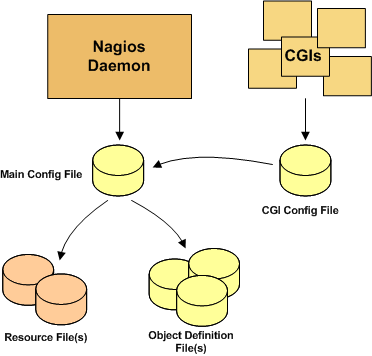
* Mô hình Distributed Monitoring, máy Nagios-Distributed thực hiện việc giám sát máy trạm và gửi thông tin trạng thái cho máy chủ giám sát chính. Mô hình này thường được sử dụng cho các hệ thống phân tán lớn với sự tham gia của nhiều máy khác nhau.
* Mô hình Redundant Monitoring, máy Nagios-Redundant thực hiện việc giám sát máy trạm. Tính năng gửi thông báo tạm thời ngừng hoạt động. Trong trường hợp máy chủ giám sát gặp lỗi, tính năng này sẽ được cho phép trở lại.
* Trong mô hình Failover Monitoring, máy Nagios-Failover thực hiện việc giám sát các thiết bị và dịch vụ như máy chủ giám sát, xong các tính năng thực thi giám sát và thông báo tạm thời được ngừng hoạt động, cho tới khi máy chủ giám sát gặp sự cố.

Trong phần sau của bài luận án này em lựa chọn mô hình giám sát mạng Failover Monitoring cho hệ thống của mình.

## Hệ thống các file cấu hình của Nagios

Nagios hệ thống file cấu hình rất đơn giản, các file tập trung chủ yếu trong thư */usr/local/nagios/etc*, trong đó có một số file đặc biệt sau đây:

* *Main Config File* (file cấu hình chính): thường được đặt tên là *nagios.cfg* và được đặt tại thư mục */usr/local/nagios/etc*, chứa những chỉ thị trỏ tới tất cả các file cấu hình đối tượng và các file liên quan. Các file cấu hình có thể đặt tại các thư mục khác nhau, khi đó trong file cấu hình chính yêu cầu các chỉ thị *cfg\_file* và *cfg\_dir* trỏ tới các file và thư mục chứa các file cấu hình đó. Đây cũng là tính linh hoạt trong cách cấu hình của Nagios.
* *Resource file* (file tài nguyên): được sử dụng để lưu những Macro được định danh bởi người dùng, và những thông tin cấu hình nhạy cảm (như tên người dùng, mật khẩu).
* *Object Definition file* (file định nghĩa đối tượng): được sử dụng để định nghĩa các máy trạm, các dịch vụ, các nhóm máy trạm, định nghĩa các lệnh .v.v. Các file này định nghĩa tất cả các thiết bị được giám sát và cách thức giám sát các thiết bị đó.
* *CGI Config file* (file cấu hình CGI): chứa những chỉ thị tác động tới hoạt động của CGIs. Nó cũng được tham chiếu tới file cấu hình chính để biết cách Nagios được cấu hình và nơi chứa những đối tượng.



Hình 7 : Hệ thống các file cấu hình của Nagios

## Một số các plugin và Add-on của Nagios

Không giống như một số công cụ giám sát khác, Nagios không có các cơ chế bên trong phục vụ cho việc kiểm tra tình trạnh của các máy và các dịch vụ. Thay vào đó, Nagios dựa vào các chương trình mở rộng (gọi là plugin và Add-on) để phục vụ cho các công việc giám sát với những vấn đề riêng biệt.

Thông thường, các plugin đi kèm với Nagios, một số được cài đặt thêm vào, trong khi một số khác phải cài đặt thư viện phát triển để biên dịch mới có được. Các Add-on được biên dịch thành các các file có thể thực thi hoặc các kịch bản (Perl, Shell, …) mà có thể thực hiện từ dòng lệnh để thực hiện kiểm tra tình trạng của các máy hoặc các dịch vụ. Nagios sẽ sử dụng kết quả từ các plugin và Add-on để quyết định trạng thái của các host và các dịch vụ trên mạng. Nagios sẽ thực thi bất cứ khi nào có sự cần thiết kiểm tra trạng thái của một dịch vụ hoặc một host. Plugin làm một số thứ để kiểm tra và sau đó đơn giản là trả lại kết quả cho Nagios. Nagios sẽ xử lý các kết quả nhận được từ các plugin và đưa ra bất kì hành động cần thiết nào. Các plugin hoạt động như một tầng trừu tượng giữa việc giám sát logic hiện thời trong Nagios daemon và các dịch vụ hay các host đang được giám sát. Trong khi đó, các Add-on có thể chạy cả trên Nagios hoặc được cài đặt như một agent gửi thông số về Nagios thông qua các plugin.



Hình 8 : Hoạt động của Nagios và các plugin

Ưu điểm của kiểu kiến trúc plugin là chúng ta có thể theo dõi bất cứ thứ gì có thể nghĩ ra. Nếu có thể tự động hoá quá trình kiểm tra một cái gì đó, ta có thể giám sát nó với Nagios. Hiện đã có rất nhiều các plugin đã được tạo ra để theo dõi tài nguyên cơ bản như tải bộ xử lý, sử dụng đĩa, tốc độ ping, ... Nếu chúng ta muốn kiểm tra cái gì khác, ta co thể xem tài liệu hướng dẫn bổ sung vể chương trình muốn giám sát.

Nhược điểm của kiểu kiến trúc plugin là trên thực tế rằng Nagios không hiểu chi tiết cụ thể về những gì đang được theo dõi, nó chỉ theo dõi những thay đổi trong trạng thái của những tài nguyên. Chỉ có plugin biết chính xác những gì chúng đang theo dõi và làm thế nào để thực hiện việc kiểm tra trên thực tế.

Một số Add-on phổ biến của Nagios gồm có:

* DNX: Là một module mở rộng của Nagios mà tải một phần đáng kể các hoạt động bình thường được thực hiện bởi Nagios tới các mạng phân tán của các máy từ xa. DNX đảm bảo công việc được phân phối công bằng và đồng đều giữa các máy khách đã đăng kí DNX.
* NRPE: cho phép thực thi từ xa các plugin trên các máy Linux/Unix. Điều này cho phép chúng ta giám sát các số liệu máy tính từ xa (việc sử dụng ổ đĩa, tải CPU, …). NRPE cũng cho phép kết nối tới agent của Windows như NSClient++, vì thế ta có thể kiểm tra các số liệu trên máy tính Windows ở xa.
* NSCA: cho phép chúng ta tích hợp các thông báo thụ động và kiểm tra từ máy tính từ xa và các ứng dụng với Nagios. Hữu ích cho các cảnh báo bảo mật, cũng như triển khai Nagios dự phòng và phân phối các thiết lập.
* NSClient++ : là một agent giám sát cho hệ thống của Microsoft Windows làm việc với Nagios.
* Nagiosgraph: là công cụ vẽ đồ thị phân tích đầu ra và hiệu suất dữ liệu từ plugin của Nagios và lưu trữ dữ liệu trong các tập tin RRD. Nagiosgraph hiển thị dữ liệu trong xu hướng của Nagios, như bật các host và dịch vụ, hoặc trong các báo cáo riêng biệt.
* Nconf: là một phần cấu hình cho Nagios dựa trên PHP. Nó khác với các công cụ khác là cung cấp các tính năng doanh nghiệp như các khuôn mẫu, phụ thuộc và khả năng cấu hình một quy mô lớn, mô hình máy chủ Nagios phân phối.
* Nagvis: là một phần trực quan cho Nagios. Có thể được sử dụng để hình dung dữ liệu cho Nagios, ví dụ : hiển thị các quy trình công nghệ thông tin như một hệ thống mail hay một cơ sở hạ tầng mạng.
* NagEventLog: là một agent cho Windows có thể gửi tin nhắn EVENTLOG từ các máy tính Windows kết nối trực tiếp đến NSCA, cho phép chúng ta nhận được thông báo trong Nagios khi các mẫu đăng nhập được phát hiện.
* PNP4Nagios: là một phần cho Nagios phân tích hiệu suất dữ liệu cung cấp bởi các plugin và các lưu trữ một cách tự động trong cơ sở dữ liệu của RRD.
* NDOUtils: cho phép chúng ta xuất dữ liệu hiện tại và lịch sử từ một hoặc nhiều thể hiện của Nagios tới một cơ sở dữ liệu MySQL. Một số chương trình bổ xung sử dụng NDOUtils như một trong những nguồn dữ liệu của chúng. NDOUtils bao gồm một daemon độc lập, một sự kiện Nagios trung gian, và một số tiện ích bổ trợ.

## Triển khai Nagios, các Plugin và các Add-on

### Triển khai trên Nagios Server

Trước khi triển khai hệ thống chúng ta cần chuẩn bị một số bước sau:

* Các giai đoạn cài đặt cần có quyền *root* (dấu *#* đằng trước câu lệnh)
* Các gói cần cài đặt sẵn phục vụ cho hệ thống là *Apache, Perl, PHP, GCC Compiler, GD development library, xinetd*. Nếu chua cài đặt thì ta cài bằng câu lệnh sau:

yum install httpd php perl gcc glibc glibc-common

yum install gd gd-devel xinetd

Tải các gói mã nguồn mới nhất của Nagios 3.3.3, Nagios Plugin 1.4.16 và NRPE 2.14, NSCA 2.9.1 tại các link sau:

<http://nchc.dl.sourceforge.net/project/nagios/nagios-3.x/nagios-3.4.3/nagios-3.4.3.tar.gz>

<http://ncu.dl.sourceforge.net/project/nagiosplug/nagiosplug/1.4.16/nagios-plugins-1.4.16.tar.gz>

<http://nchc.dl.sourceforge.net/project/nagios/nrpe-2.x/nrpe-2.14/nrpe-2.14.tar.gz>

<http://ncu.dl.sourceforge.net/project/nagios/nsca-2.x/nsca-2.9.1/nsca-2.9.1.tar.gz>

1. *Cài đặt Nagios*

Tạo mới user *nagios* và group *nagcmd*

# useradd -m nagios

# passwd nagios

# groupadd nagcmd

Sau đó thêm 2 user *nagios*, *apache* vào group *nagcmd*.

# usermod -a -G nagcmd nagios

# usermod -a -G nagcmd apache

Biên dịch và cài đặt Nagios bằng cách giải nén mã nguồn và chuyển vào thư mục mới được tạo ra

# tar xzf nagios-3.4.3.tar.gz

# cd nagios

Chạy configure script với tham số truyền vào là group *nagcmd*

# ./configure --with-command-group=nagcmd

Biên dịch source code của Nagios

# make all

Cài đặt các binary, init script, sample config file và thiết lập permission trên thư mục chứa external command

# make install

# make install-init

# make install-config

# make install-commandmode

Các file cấu hình mẫu nằm trong thư mục */usr/local/nagios/etc*, chúng ta cần mở file */usr/local/nagios/etc/objects/contacts.cfg* và sửa lại địa chỉ email liên kết với username *nagiosadmin* để nhận các thông báo.

Cài đặt web config file của Nagios vào thư mục */etc/httpd/conf.d* của Apache

# make install-webconf

Tạo tài khoản nagiosadmin và đặt password để truy cập vào web interface

# htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin

Khởi động lại Apache

# service httpd restart

1. *Cài đặt Nagios Plugin*

Trước khi cài đặt nagios Plugin, ta cần phải chắc chắn rằng gói OpenSSL Deverlopment được cài đặt vì gói này hỗ trợ cho biên dịch SSL. Nếu hệ thống chưa cài đặt thì ta cần cài đặt thông qua lệnh sau :

# yum install openssl-devel

Cài đặt các Plugins:

# tar xzf nagios-plugins-1.4.16.tar.gz

# cd nagios-plugins-1.4.16

# ./configure --with-nagios-user=nagios --with-nagios-group=nagios

# make

# make install

Phân quyền các thư mục Plugin với quyền của người dùng là *nagios*

# chown nagios.nagios /usr/local/nagios

# chown -R nagios.nagios /usr/local/nagios/libexec

1. *Cài đặt Add-on NRPE*

Máy chủ Nagios Server trên thực tế đóng vai trò là nhận các thông tin từ các máy chủ được giám sát thông qua Plugin NRPE. Do đó máy chủ Nagios giống như một máy Client cho của NRPE, vì thế ta cần cài đặt NRPE Plugin để nhận thông tin từ máy chủ được giám sát.

# tar xzf nrpe-2.14.tar.gz

# cd nrpe-2.14

# ./configure --enable-command-args

# make all

# make install-plugin

Kiểm tra kết nối giữa NRPE Plugin với NRPE được cài trên máy server từ xa bằng lệnh sau :

# /usr/local/nagios/libexec/check\_nrpe -H <IP máy chủ từ xa>

Nếu lệnh này trả về kết quả là phiên bản của NRPE thì có nghĩa là việc cài đặt thành công (lưu ý là NRPE cũng phải được cài đặt trên máy chủ từ xa dưới dạng NRPE Daemon)

1. *Cài đặt Add-on NSCA*

Nagios sử dụng công cụ NSCA để gửi các kết quả kiểm tra từ ứng dụng của bạn về server Nagios. Công cụ này giúp cho thông tin gửi trên mạng được an toàn hơn vì nó được mã hóa và xác thực. NSCA cũng được sử dụng tương tự như NRPE nhưng nó còn được sử dụng cho các dịch vụ giám sát đã đưa ra trong phần 2.3.

Việc cài đặt NSCA thực hiện qua các lệnh sau:

# tar xzf nsca-2.9.1.tar.gz

# cd nsca-2.9.1

# ./configure

# make all

# cp src/nsca /usr/local/nagios/bin

# src/send\_nsca /usr/local/nagios/bin

# cp sample-config/nsca.cfg /usr/local/nagios/etc

# chown nagios.nagios /usr/local/nagios/etc/nsca.cfg

# chown nagios.nagcmd /usr/local/nagios/etc/nsca.cfg

# chmod g+r /usr/local/nagios/etc/nsca.cfg

# chown nagios.nagios /usr/local/nagios/bin/nsca

# chown nagios.nagios /usr/local/nagios/bin/send\_nsca

Để NSCA chạy dưới dạng một daemin, tạo một file mới */etc/xinetd.d/nsca* với nội dung

# default: on

# description: NSCA

{

flags = REUSE

socket\_type = stream

wait = no

user = nagios

group = nagios

server = /usr/local/nagios/bin/nsca

server\_args = -c /usr/local/nagios/etc/nsca.cfg --inetd

log\_on\_failure += USERID

disable = no

only\_from = 127.0.0.1 0.0.0.0

}

Thêm dòng mới vào file */etc/services*

nrpe 5666/tcp # NRPE Daemon

Khởi động lại xinetd để NSCA có hiệu lực

# chkconfig xinetd on

# service xinetd restart

1. *Khởi chạy Nagios*

Thêm vào danh sách các system service và tự động khởi động cùng hệ thống

# chkconfig --levels 235 httpd on

# chkconfig --levels 235 nagios on

# service nagios start

# service httpd start

Xác nhận các file cấu hình mẫu đang làm việc đúng

# /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

# /usr/local/nagios/bin/nsca -v /usr/local/nagios/etc/nsca.cfg

Nếu không có lỗi gì thì kích hoạt Nagios

# service nagios start

1. *Chỉnh lại thiết lập của SELinux*

Enforcing mode của SELinux mặc định được bật. Điều này có thể dẫn tới thông báo “Internal Server Error” khi thử truy cập Nagios CGI. Kiểm tra nếu SELinux đang ở Enforcing mode

# getenforce

Đặt SELinux trong Permissive mode

# setenforce 0

Để thiết lập này có hiệu lực mãi mãi thì chỉnh sửa lại trong file */etc/selinux/config*, sau đó reboot lại máy.

# sua dong duoi day trong file /etc/selinux/config.

# Gia tri mac dinh la

# SELINUX=enforcing

SELINUX=disabled

Thay vì disable SELinux hoặc enable Permissive mode, có thể sử dụng lệnh sau để chạy các CGI dưới Enforcing/Targeted mode

# chcon -R -t httpd\_sys\_content\_t /usr/local/nagios/sbin/

# chcon -R -t httpd\_sys\_content\_t /usr/local/nagios/share/

Đăng nhập vào bằng giao diện Web :

http://localhost/nagios/

Hệ thống sẽ yêu cầu nhập tên người dùng (nagiosadmin) và mật khẩu đã lập ở trên.

### Triển khai trên host Linux

Với các host Linux, chúng ta cần cài đặt NRPE daemon, ở đây tôi lấy ví dụ cho hai bản phân phối của Linux là hệ điều hành CentOS. Với hệ điều hành CentOS, chúng ta cần thực hiện các bước sau đây, yêu cầu hệ thống đã cài đặt sẵn các gói *make, gcc, perl, openssl, openssl-devel, xinetd*. Nếu hệ thống chưa cài đặt thì có thể thực hiện lệnh sau:

yum install make gcc perl openssl openssl-devel xinetd

Thêm người dùng mới

# useradd -m nagios

# passwd nagios

Tiếp đến là cài đặt Nagios Plugin

# tar xzf nagios-plugins-1.4.16.tar.gz

# cd nagios-plugins-1.4.16

# ./configure --with-nagios-user=nagios --with-nagios-group=nagios

# make

# make install

Sửa lại permission cho các thư mục chứa Nagios plugin

# chown nagios.nagios /usr/local/nagios

# chown -R nagios.nagios /usr/local/nagios/libexec

Cuối cùng là cài đặt NRPE daemon theo các lệnh lần lượt sau đây

# tar xzf nrpe-2.14.tar.gz

# cd nrpe-2.14

# ./configure --enable-command-args

Kết quả hiển thị trên như dưới đây cho biết các thông tin cài đặt đúng

General Options:

-------------------------

NRPE port: 5666

NRPE user: nagios

NRPE group: nagios

Nagios user: nagios

Nagios group: nagios

Tiếp theo là biên dịch và cài đặt cho NRPE khởi động cùng hệ thống

# make all

# make install-plugin

# make install-daemon

# make install-daemon-config

# make install-xinetd

Tiếp theo là cấu hình NRPE cho phép máy chủ Nagios có thể giám sát được, chúng ta sử đổi nội dung của file */etc/xinetd.d/nrpe* và sử đổi nội dung tại dòng

only\_from = 127.0.0.1

và thay đổi địa chỉ 127.0.0.1 thành địa chỉ IP của máy chủ Nagios

only\_from = <nagios\_ip\_address>

Chỉnh sửa file */etc/services* và thêm vào NRPE daemon

nrpe 5666/tcp # NRPE Daemon

Khởi động lại xinetd và thiết lập khởi động cùng hệ thống

# chkconfig xinetd on

# service xinetd restart

Kiểm tra NRPE daemon có đang chạy và lắng nghe cồng 5666 hay không bằng lệnh

# netstat -at | grep nrpe

Kết quả sau chứng tỏ NRPE đang được chạy

tcp 0 0 \*:nrpe \*.\* LISTEN

Kiểm tra NRPE daemon chạy đúng

# /usr/local/nagios/libexec/check\_nrpe -H localhost

Kết quả là *NRPE v2.14* cho biết NRPE được cài đặt đúng.

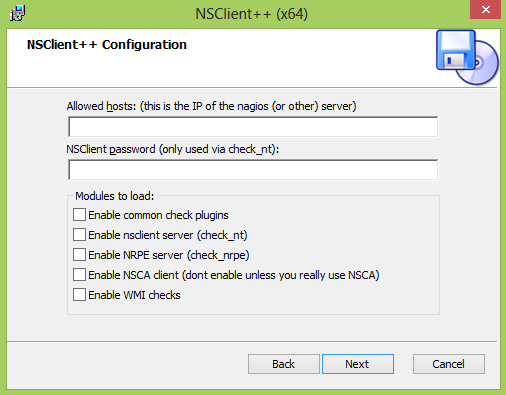
### Triển khai trên host Windows

NSClinet là một add-on cho phép Nagios giám sát các tài nguyên trên hệ thống các host Windows. Thông tin về NSClient có trong trang web <http://www.nsclient.org>. Để tải về bản cài đặt mới nhất, ta truy cập theo địa chỉ sau :

<http://sourceforge.net/projects/nscplus/files/nscplus/>

Ngoài ra ta có thể tải bản chạy thẳng được cung cấp dưới dạng file nén.

Trong quá trình cài đặt vào host Windows ta cần chú ý đến bước xác định địa chỉ và cho phép các plugin được phép truy cập đến Service NSClient++. Mặc định các tùy chọn này được để trống như hình dưới đây.



Hình 9 : Các tùy chọn trong quá trình cài đặt NSClient++

Sau quá trình cài đặt, chúng ta vẫn có thể cấu hình NSClient++ bằng cách sửa đổi nội dung của file *NSC.ini* (đối với phiên bản 0.3.x) hoặc file *nsclient.ini* (đối với phiên bản 0.4.x) trong thư mục cài đặt của NSClient++. Phần cấu hình sẽ được đề cập trong phần sau của báo cáo này. Tuy nhiên để Nagios có thể giám sát được các thông số cơ bản, tôi sẽ lựa chọn hết các tùy chọn trong hình trên và phần địa chỉ IP được phép giám sát host windows cài NSClient thuôc dải mạng 192.168.111.0/24 và mật khẩu truy cập là 1234567

## Cấu hình Nagios

### Các file cấu hình mẫu

Nagios là một chương trình có tính linh hoạt cao trong việc cấu hình. Các cấu hình cho các máy trạm, các dịch vụ có thể được đặt trong cùng một file hoặc được đặt trong các file riêng lẻ. Các file cấu hình này phải được đặt trong chỉ thị *cfg\_file* trong file cấu hình chính *nagios.cfg.* Trong thư mục của Nagios đã có sẵn các file cấu hình mẫu, trước khi thành thạo với Nagios, việc cấu hình sẽ đơn giản hơn rất nhiều nếu sử dụng các cấu hình mẫu.

Trong thư mục của Nagios có các file cấu hình mẫu cơ bản :

* + *Commands.cfg* : dùng để định nghĩa các lệnh kiểm tra dịch vụ, máy trạm.
  + *Contacts.cfg* : dùng để định nghĩa các người dùng, người quản trị.
  + *Localhost.cfg* : dùng để định nghĩa các dịch vụ kiểm tra cho máy cục bộ đang chạy Nagios.
  + *Printer.cfg :* dùng để định nghĩa máy in và các dịch vụ kiểm tra cho máy in.
  + *Switch.cfg* : dùng để định nghĩa switch và các dịch vụ kiểm tra cho switch.
  + *Templates.cfg* : dùng để định nghĩa các máy trạm (host) mẫu, các dịch vụ mẫu.
  + *Windows.cfg* : dùng để định nghĩa mẫu máy trạm chạy hệ điều hành windows.

Nagios định nghĩa các đối tượng là các phần tử gồm các chỉ thị khác nhau trong việc giám sát và cảnh báo. Các file cấu hình trên định nghĩa các đối tượng như :

* Các dịch vụ (Services) và nhóm dịch vụ (Services Groups)
* Các máy giám sát (Hosts) và nhóm máy giám sát (Host Groups)
* Các tài khoản (Contacts) và nhóm tài khoản (Contact Groups)
* Các lệnh (Commands)
* Chu kỳ thời gian (Time Periods)
* Mở rộng cảnh báo (Notification Escalations)
* Phụ thuộc cảnh báo và thực thi (Notification and Execution Dependencies)

### Các cảnh báo trong Nagios

Một trong những tính năng nổi bật của phần mềm Nagios là khả năng gửi cảnh báo linh hoạt tới người quản trị.

Thông báo dịch vụ và máy trạm được gửi đi khi xảy ra các trường hợp sau :

* Khi một trạng thái HARD của dịch vụ hay máy trạm thay đổi (HARD là trạng thái mà dịch vụ hoặc thiết bị đã được kiểm tra lại đủ số lần và đang rơi vào trạng thái lỗi hoặc phục hồi từ trạng thái lỗi).
* Khi một dịch vụ hoặc máy trạm đang trong tình trạng “non-OK” và đã hết thời gian trong bước lặp gửi thông báo lại.

Các thông báo lỗi sẽ được gửi tới những thành viên được định nghĩa trong định nghĩa contact, và contact\_group. Nagios sẽ loại bỏ đi các thanh viên trùng lặp trong các nhóm trước khi nó xử lý gửi thông báo đi.

Trước khi thông báo được gửi nó, nó phải qua một số bộ lọc nhất định.

* Bộ lọc mức chương trình (Program-Wide Filter). Đây là bộ lọc đầu tiên mà thông báo phải đi qua. Bộ lọc này định nghĩa thông báo có được cho phép gửi đi hay không thông qua chỉ thị enable\_notifications trong file cấu hình chính. Nhưng lựa chọn này có thể được thay đổi qua giao diện Web trong suốt thời gian chạy.
* Bộ lọc mức máy trạm và dịch vụ (Service and Host Filters). Bộ lọc đầu tiên kiểm tra nếu dịch vụ và máy trạm đang trong thời gian được lập lịch ngừng hoạt động, thông báo sẽ không được gửi đi. Bộ lọc thứ 2 kiểm tra nếu trạng thái của dịch vụ và máy trạm là thay đổi thường xuyên (trạng thái Flapping), thông báo cũng sẽ không được gửi. Bộ lọc thứ 3 chỉ ra các trạng thái cụ thể của dịch vụ và máy trạm được chỉ định trong định nghĩa mới được gửi đi. Bộ lọc thứ 4 ở mức dịch vụ và máy trạm đó là bộ lọc thời gian gửi thông báo – notification period, thông báo chỉ được gửi đi trong khoảng thời gian quy định tại đây. Bộ lọc cuối cùng là khi thời gian bước lặp gửi thông báo đã hết, và trạng thái của dịch vụ vẫn không thay đổi, thông báo sẽ được gửi đi.
* Bộ lọc mức contact (Contact Filters). Bộ lọc xác định thành viên nào được gửi thông báo và thời gian được phép gửi thông báo (notification\_period).

Nagios hỗ trợ nhiều phương thức cảnh báo như email, tin nhắn SMS, cảnh báo bằng âm thanh, máy nhắn tin .v.v.

### Bộ xử lý sự kiện

Bộ xử lý sự kiện có thể là một lệnh hay một script, được thực hiện khi trạng thái của dịch vụ hoặc của máy trạm thay đổi, giúp phục hồi lỗi trước khi thông báo được gửi đi. Nó có thể được dùng để :

* Khởi động lại dịch vụ bị lỗi.
* Nhật ký thông tin sự kiện vào cơ sở dữ liệu.
* Khởi động lại thiết bị.

Có 2 loại bộ xử lý sự kiện là :

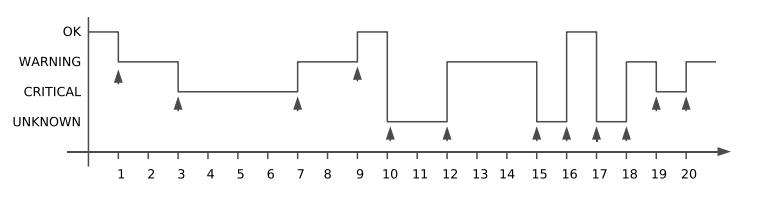
* Bộ xử lý sự kiện chung cho tất cả các thiết bị và dịch vụ.
* Bộ xử lý sự kiện riêng cho từng máy trạm và dịch vụ.

Bộ xử lý sự kiện được thực thi khi dịch vụ hoặc máy trạm ở trạng thái mềm (SOFT – trạng thái mà thiết bị hoặc dịch vụ chưa được kiểm tra lại hết số lần quy định), hoặc bắt đầu rơi vào trạng thái cứng (HARD), hoặc bắt đầu phục hồi từ một trạng thái mềm hay cứng. Với trạng thái phục hồi hoặc rơi vào lỗi cứng, nó được thực thi ngay sau khi thông báo đã được gửi đi.

### Flapping

Nếu trạng thái của host/ dịch vụ không ổn định, thay đổi liên tục. Người quản trị sẽ nhận được rất nhiều thông báo trong một khoảng thời gian ngắn. Nó không chỉ gây khó chịu mà còn làm rối loạn việc xác định vấn đề lỗi. Nagios có thể phát hiện vấn đề này và đặt trạng thái đó là flapping.

Để phát hiện tình trạng flap của một dịch vụ, Nagios lưu lại 21 kết quả kiểm tra dịch vụ gần nhất, tức là tối đa lưu lại 20 lần thay đổi trạng thái của dịch vụ. Hình dưới đây mô tả sự thay đổi trạng thái của một dịch vụ:



Hình 10 : Phát hiện Flaping trong Nagios

Từ hình trên ta có thể thấy là trong 20 lần kiểm tra, dịch vụ thay đổi trạng thái 12 lần. Nagios dựa vào số liệu này để thông báo dịch vụ đang rơi vào tình trạng *flapping* hoặc thoát khởi tình trạng *flapping*. Khi flapping xảy ra, Nagios sẽ ghi sự kiện này vào tệp log, đặt thông tin flap vào phần comment của dịch vụ và dừng hành động thông báo trạng thái dịch vụ.

Phát hiện flap được cấu hình ở 2 vị trí; tệp cấu hình chính *nagios.cfg* (cài đặt cấu hình nói chung) và trong định nghĩa của từng dịch vụ cụ thể.

Ví dụ như trong tệp cấu hình chính:

enable\_flap\_detection=1 // cho phép phát hiện flap.

low\_service\_flap\_threshold=5.0 //ngưỡng dưới flap

high\_service\_flap\_threshold=20.0 //ngưỡng trên flap

Đoạn cấu hình trên có nghĩa là nếu có từ 5 lần trở lên dịch vụ được ghi nhận là thay đổi trong 20 lần kiểm tra thì dịch vụ đó được đặt vào tình trạng flapping.

Ta có thể thiết đặt tùy chọn phát hiện flapping và đặt ngưỡng flap cho từng dịch vụ trong phần định nghĩa đối tượng dịch vụ.

define service{

host\_name linux01

flap\_detection\_enabled 1

low\_flap\_threshold 6.0

high\_flap\_threshold 20.0

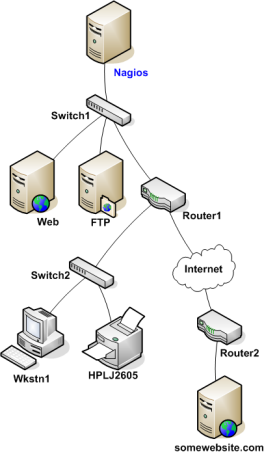
...

}

Tương tự phát hiện flapping đối với host.

### Quan hệ cha/con giữa các host

Nagios là phần mềm chưa có khả năng tự phát hiện ra các node và kiến trúc của mạng. Công việc này do người dùng tự định nghĩa và quyết định theo quy tắc nhất định. Nagios được coi là trung tâm giám sát. Các thiết bị(A) có đường kết nối vật lý trực tiếp đến server Nagios được có mối quan hệ là con của Nagios. Các thiết bị kết nối trực tiếp đến A được coi là con của A.. Cứ như vậy kiến trúc mạng được định nghĩa và mở rộng qua mối quan hệ cha/con này, với Nagios là trung tâm.



Hình 11 : Mối quan hệ host cha/con.

Ví dụ mạng có kiến trúc như trên. Khi đó ta có Switch1 được coi là con của Nagios. Web, FTP, Router1 là con của Switch1, Switch2 được coi là con của Router1 … Tất cả mối quan hệ này đều phải do người dùng định nghĩa qua tùy chọn parents trong mỗi định nghĩa đối tượng. ví dụ:

define host{

host\_name

…

}

define host{

host\_name Switch1

…

parents Nagios

}

define host{

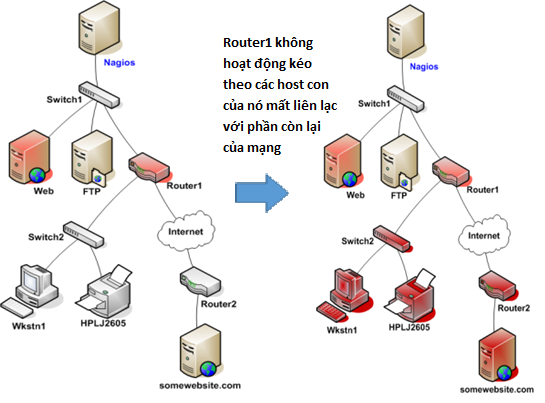
host\_name Web

…

parents Switch1

}

Như ví dụ hình bên dưới, ta tắt host web và router1. Một hành động kiểm tra được thực hiện và trả về kết quả cho Nagios. Trường hợp này Nagios kết luận host web và router1 ở trạng thái DOWN bởi vì host cha Switch1 hoạt động bình thường. Trong khi đó các host nằm sau router1 được kết luận là UNREACHABLE (Không xác định). Vì Nagios không thể liên lạc được với chúng vì router1 bị tắt kéo theo mất đường kết nối đến các host này.



Hình 12 : Phân biệt DOWN-UNREACHABLE.

Việc phân biệt trạng thái DOWN-UNREACHABLE của host giúp các nhà quản trị dễ dàng hơn trong việc xác định được nguyên nhân và vị trí của lỗi sảy ra trên mạng khi nhận được thông báo sự cố.

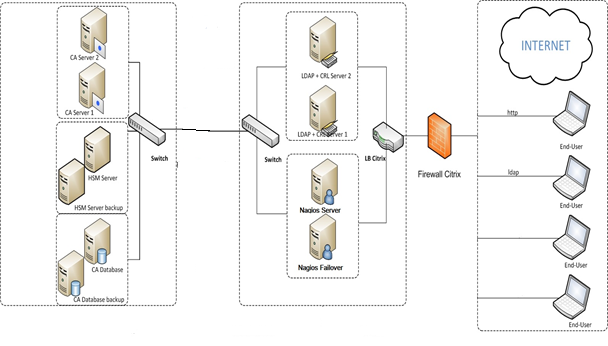
# Chương 3 : Xây dựng và hệ thống giám sát mạng với Nagios

## Mô hình giám sát mạng

Việc tích hợp này nhằm đem lại khả năng quản lý tập trung, thống nhất, đánh giá được hoạt động tổng thể của cơ sở hạ tầng CNTT, giảm thiểu được chi phí hoạt động và thời gian xử lý sự cố, thời gian hỗ trợ của các nhóm kỹ thuật của doanh nghiệp.

Giải pháp hệ thống giám sát dựa trên Nagios cung cấp cho doanh nghiệp một console tập trung duy nhất để giám sát toàn bộ họat động của các thành phần trong cơ sở hạ tầng của doanh nghiệp thay vì sử dụng các công cụ quản lý, giám sát riêng rẽ cho từng hệ thống như trước đây. Đồng thời, giải pháp cũng sẽ cung cấp cho người quản lý doanh nghiệp các báo cáo về toàn bộ họat động của hệ thống hạ tầng CNTT, giúp họ có thể đánh giá được họat động của hệ thống CNTT của doanh nghiệp.

Sau đây là những hệ thống quản lý thành phần bên trong của hệ thống giám sát ta cần phải xây dựng:

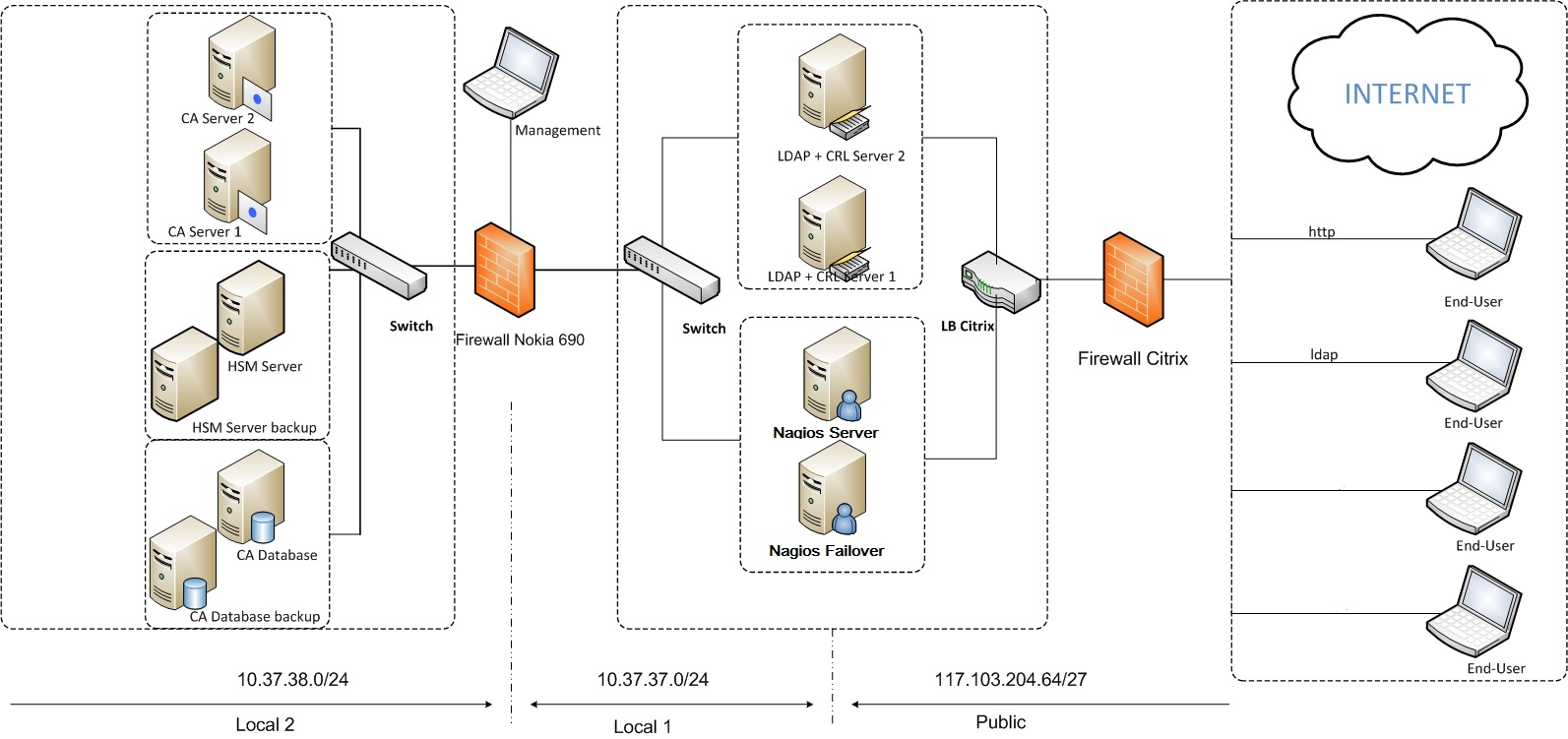


Hình 13 : Mô hình mạng giám sát chung

Hệ thống mạng LAN mô phỏng với các máy tính gồm:

* Hai máy chủ cài đặt phần mềm Nagios là Nagios-Server đóng vai trò là máy chủ giám sát trong mô hình thử nghiệm.
* Các máy khác đóng vai trò là các máy chủ dịch vụ và máy trạm được giám sát với một số dịch vụ được cài đặt trên đó.

Những thành phần trong hệ thống chủ yếu nằm trong cùng một mạng LAN với nhau, điều này dẫn đến những nguy cơ về an ninh. Vì thế để đảm bảo vấn đề an toàn an ninh thì cần phải xây dựng mô hình mạng như hình 14. Mô hình này gọi là mô hình nâng cao của mô hình mạng trong hình 13. Ở mô hình nâng cao, Nagios giám sát các máy chủ thông qua tường lửa, đồng thời giam sát luôn tường lửa. Tuy nhiên cần phải khai báo thêm các thông số như quan hệ cha/con trong mô hình nâng cao để Nagios có thểm hoạt động đúng với những yêu cầu của người quản trị.



Hình 14 : Mô hình mạng giám sát nâng cao

Trong mô hình giám sát nâng cao, ta đặt thêm một máy giám sát Nagios Server được đặt là máy Slave (trong mô hình nâng cao có tên là Nagios Failover) có nhiệm vụ giám sát như máy Master (Nagios Server trong hình 14) nhưng tắt các chức năng giám sát đi cho đến khi máy Master bị lỗi.

Dưới đây là một số mô tả tổng quan về từng thiết bị cần giám sát trong mô hình trên

|  |  |  |
| --- | --- | --- |
| Máy chủ | Nền tảng | Mô tả |
| Nagios Server | CentOS 6 | * Máy chủ giám sát Nagios . * Là máy chủ giám sát (Master). * Các dịch vụ được giám sát là : các tài nguyên trên máy (Disk, CPU Load, User, Memory, Swap) và các máy chủ cài đặt các dịch vụ CA, HSM, Cơ sở dữ liệu CA, LDAP. * Thực hiện việc gửi tin nhắn cảnh báo qua SMS hay Email |
| Nagios Failover | CentOS 6 | * Cài đặt Nagios, đóng vai trò là máy giám sát phụ (Slave). * Thực hiện chức năng chịu lỗi cho hệ thống giám sát. * Trong điều kiện máy chủ Nagios Server (Master) chạy bình thường, máy Slave ngừng kiểm tra dịch vụ cũng như gửi thông báo. Nó được kích hoạt trở lại khi máy Master gặp lỗi. |
| CA Server | CentOS 6 | * Cài đặt dịch vụ chứng thực số Jboss, ẸBCA * Có nhiệm vụ tạo và quản lý chứng thực số |
| HSM Server | Windows Server 2008 | * Cài đặt dịch vụ HSM Simulator * Nhiệm vụ chủ yếu của Server này là ký lên các chứng thực số. |
| CA Database | CentOS 6 | * Cài đặt dịch vụ cơ sở dữ liệu MySQL Cluster * Lưu cở sở dữ liệu chứng thực số |
| LDAP Server | CentOS 6 | * Cài đặt dịch vụ quản lý tập trung LDAP * Lưu thông tin chứng thực số CA và public cho người dùng từ xa kết nối đến để tra cứu thông tin |

Bảng 1 : Các thành phần của hệ thống

Máy chủ Nagios giám sát các các máy chủ trong mô hình có chung các dịch vụ và tài nguyên hệ thống nên cần được nhóm lại để tiện cho việc cấu hình. Chẳng hạn như hai máy chủ chạy dịch vụ LDAP sẽ được nhóm lại thành một nhóm, việc làm này nhằm mục đích đơn giản hóa các cấu hình giám sát, dễ sửa đổi sau này mỗi khi có cập nhật.

Các nhóm máy được đưa vào trong một file cấu hình tên là “*hostgroups.cfg”*, file cấu hình *“service\_groups.cfg”* cũng có chức năng như “*hostgroups.cfg”* và cả hai file này cùng nằm trong thư mục */usr/local/nagios/server/*. Để thư mục này có hiệu lực, ta thêm dòng sau vào file cấu hình chính *nagios.cfg* một dòng với nội dung:

cfg\_dir=/usr/local/nagios/etc/servers

Phần tiếp theo của chương này đề cập đền việc cài đặt và cấu hình cho các kịch bản sau:

* Giám sát tài nguyên hệ thống mạng trên các server.
* Giám sát các dịch vụ cơ sở dữ liệu MySQL, LDAP, Jboss, HSM Simulator.
* Thực hiện cài đặt và cấu hình cho cảnh báo qua SMS, sử dụng D-Com 3G.
* Khắc phục sự cố đối mạng với các server và dịch vụ.

## Giám sát tài nguyên mạng

### Cấu hình hệ thống giám sát

Để hệ thống giám sát các thiết bị, điều đầu tiên người quản trị mạng cần phải làm là thiết lập các định nghĩa sau:

* Định nghĩa host: gồm các thông tin về thiết bị, mỗi host có một định nghĩa riêng cho mọi dịch vụ kết nối tới nó. Có thể định nghĩa mỗi host trong nhiều file hoặc chung một file. Thông thường mỗi định nghĩa host có bốn trường cơ bản như ví dụ sau:

define host{

use linux-server

host\_name localhost

alias localhost

address 127.0.0.1

}

* Định nghĩa dịch vụ: bao gồm các thông tin cơ bản về dịch vụ. Các định nghĩa dịch vụ phải được kết hợp trên một host cụ thể. Khi giám sát một dịch vụ mới, phải định nghĩa cho dịch vụ đó, kèm theo một định nghĩa máy trạm cụ thể mà dịch vụ chạy trên đó (nếu máy trạm đã tồn tại, không phải định nghĩa lại máy trạm đó). Tất cả định nghĩa dịch vụ cho các host có thể được đặt cùng trong một file hoặc trong nhiều file riêng rẽ. Ví dụ:

define service{

use local-service

host\_name localhost

service\_description PING

check\_command check\_ping!100.0,20%!500.0,60%

}

* Định nghĩa lệnh kiểm tra: định nghĩa thông thường được đặt trong file *command.cfg* nằm trong thư mục */usr/local/nagios/etc/objects.* Tuy nhiên, người quản trị có thể định nghĩa lệnh trong một file khác nhưng cần phải thêm một tùy chọn sử dụng file này trong file cấu hình chính *nagios.cfg*. Các lệnh được định nghĩa một lần cho tất cả các dịch vụ và máy trạm. Chẳng hạn:

define command{

command\_name check\_local\_swap

command\_line $USER1$/check\_swap -w $ARG1$ -c $ARG2$

}

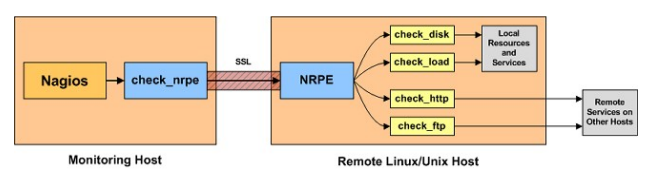
Trong đó *$USER1$* và *$HOSTADDRESS$* là các macro được định nghĩa sẵn trong Nagios *($USER1$* tương ứng đường dẫn */usr/local/nagios/libexec*; và *$HOSTADDRESS$* tương ứng với địa chỉ được miêu tả tại trường *host\_name* trong định nghĩa máy trạm), *$ARG1$* và *$ARG2$* là các tham số dòng lệnh (người quản trị, có thể truyền tối đa 32 tham số bắt đầu từ *$ARG1$*).

### Giám sát tài nguyên của các máy tham gia trong mạng

Các máy tham gia trong mô hình mạng đã trình bày trong phần 3.1 ở trên đều cài đặt hệ điều hành CentOS 6.3 nên các máy này đều được cài đặt Add-on NRPE phục vụ cho mục đích giám sát.

NRPE hoạt động theo nguyên tắc:

* Trên máy chủ Nagios Server sử dụng plugin “check\_nrpe” (là một file nhị phân thực thi trong thư mục */usr/local/nagios/etc/objects*) để truyền lệnh tới chương trình NRPE Daemon trên máy chủ được giám sát.
* Tại máy chủ cài đặt NRPE Daemon (đóng vai trò là Agent của Nagios) sẽ thực hiện các kiểm tra cục bộ hoặc trên các máy khác với các lệnh đã định nghĩa trong file *nrpe.cfg* nằm tại thư mục */usr/local/nagios/etc*.
* NRPE Daemon sau đó gửi kết quả tới Nagios Server để hiển thị trên trình duyệt WEB thông báo cho người quản trị biết.



Hình 15 : Nguyên tắc hoạt động của NRPE

Để giám sát các máy Linux từ xa ta sử dung NRPE. Nên tạo một file mới trong thư mục */usr/local/nagios/etc/server/*. File này sẽ chứa các thông tin cần theo dõi về các dịch vụ chẳng hạn như DNS, HTTP,... hay các thông số kiểm tra khác như CPU, dung lượng ổ đĩa, …

Ví dụ ta cần giám sát số lượng user đăng nhập trong máy chủ cần giám sát thì các bước thực hiện như sau:

* Trên máy chủ Nagios định nghĩa lệnh “check\_nrpe” trong file *command.cfg* như sau:

define command {

command\_name check\_nrpe

command\_line $USER1$/check\_nrpe -H $HOSTADDRESS$ -c $ARG1$ -a $ARG2$

}

* Trong file *nrpe.cfg* của máy chủ cần giám sát ta định nghĩa lệnh:

command[check\_users]=/usr/local/nagios/libexec/check\_users -w 5 -c 10

Nếu cần truyền tham số trên NRPE Daemon thì chỉnh sửa dòng:

dont\_blame\_nrpe=0 # không cho phép truyền tham số

thành:

dont\_blame\_nrpe=1 # Được phép truyền tham số

và định nghĩa lệnh như sau :

command[check\_users]=/usr/local/nagios/libexec/check\_users -w $ARG1$ -c $ARG2$

* Sau đó định nghĩa một dịch vụ trong file cấu hình của máy giám sát, chẳng hạn như cần giám sát trên máy Ubuntu Server

# Nếu hông cho phép truyền tham số

define service {

use local-service

host\_name Ubuntu-Server

service\_description SDA1 Space

check\_command check\_nrpe!check\_users

}

# Nếu cho phép truyền tham số

define service {

use local-service

host\_name Ubuntu-Server

service\_description SDA1 Space

check\_command check\_nrpe!check\_users!5!10

}

Lưu ý:

* Tham số cần truyền bắt buộc là địa chỉ IP của máy chủ cần giám sát được xác định bởi macro *$HOSTADDRESS$* của Nagios. Tuy nhiên macro này không cần thiết phải truyền cho dịch vụ khai báo trên do đã được định nghĩa trong phần host.
* Các tham số dòng lệnh cách nhau bằng dấu chấm than ‘!’.

Việc giám sát các tài nguyên mạng trong mô hình đã đưa ra trong phần 3.1 được chia thành hai file cấu hình, một cho các host chạy hệ điều hành CentOS và một cho các host chạy hệ điều hành Windows Server 2008. Trong thư mục */usr/local/nagios/etc/server/*, tôi tạo ra hai file tương ứng là *“centos.cfg”* và *“windows.cfg”* với mục đích trên.

### Giám sát các tài nguyên trên máy chủ Windows

Các máy chủ windows có thể được giám sát thông qua plugin *check\_nt*, được định nghĩa trong file *windows.cfg*. Ví dụ như kiểm tra windows đã được bật trong bao lâu, ta định nghĩa:

define service{

use generic-service

host\_name NgocNgoan

service\_description Uptime

check\_command check\_nt!UPTIME

}

Lưu ý rằng NSClient++ đã được cấu hình (trong quá trình cài đặt) cho phép *check\_nt* có thể làm việc được. Ngoài ra, check\_nrpe cũng có thể được sử dụng để kiểm tra tài nguyên của các máy chủ Windows. Tuy nhiên NSClient++ không có cấu hình sẵn cho phép NRPE có thể làm việc. Để cấu hình cho phép NRPE làm việc, ta thêm nội dung sau vào file cấu hình *nsclient.ini* (phiên bản 0.4.x)

allow arguments=1

performance data=1

port=5666

Đoạn cấu hình trên cho phép NRPE được truyền với các tham số, cho phép gửi dữ liệu trở lại Nagios và mở cổng 5666 để kết nối. Sau đó khởi động lại NSClient trên Command Prompt (yêu cầu phải có quyền *administrator*) với các lệnh:

net stop nscp

net start nscp

## Triển khai cảnh báo qua SMS

Phần này nghiên cứu và áp dụng tính năng cảnh báo SMS có trong Nagios sử dụng phần mềm Gammu và thiết bị USB 3G để nhanh chóng gửi thông tin, cảnh báo về trạng thái hoạt động của các host/service/network device đang chạy trong hệ thống.

Mô hình cảnh báo được mô tả như hình dưới đây



Hình 18 : Mô hình cảnh báo qua SMS

Trong đó:

* Máy Monitor sẽ gồm :
  + Phần mềm Nagios giúp giám sát các bộ phận, thông số quan trọng của hạ tầng CNTT như : tài nguyên hệ thống (CPU, RAM, disk, …), giao thức mạng (HTTP, SSH, SMTP, …), các dịch vụ (DNS, DHCP, …), các máy chủ (Windows, UNIX, …), các thiết bị mạng (router, switch, firewall, …).
  + Phần mềm Gammu giúp truy cập tới các thiết bị điện thoại (trong đó có USB 3G), điều khiển việc gửi nhận SMS cùng nhiều chức năng khác về quản lý cuộc gọi và danh bạ.
  + Thiết bị D-Com 3G đóng vai trò làm GSM Modem, liên lạc với nhà mạng di động để thực hiện việc gửi tin nhắn SMS.
* Khi có các sự kiện ngưng trễ hoặc khôi phục hoạt động từ các thành phần được giám sát, Nagios sẽ tự động tạo ra các thông báo. Nội dung của thông điệp cảnh báo này sẽ được truyền cho Gammu và từ đó đẩy tới thiết bị USB 3G rồi gửi tới số điện thoại của người quản trị.

### Chuẩn bị cài đặt

Trước khi cài đặt phần mềm Gammu, hệ thống của tôi cần chuẩn bị các yêu cầu sau:

* Một USB 3G, ở đây thử nghiệm với thiết bị D-Com của mạng Viettel.
* Source package của Gammu, phần mềm giúp điều khiển các mobile phone, USB modem để thực hiện gửi và nhận SMS.
  + Các phụ thuộc mà Gammu bắt buộc phải có là: CMake, pkg-config
  + Các gói phụ thuộc tùy chọn giúp mở rộng tính năng cho Gammu là: *Bluez-libs, libusb-1.0, libCURL, libiconv, Gettext, MySQL, PostgreSQL, unixODBC, libdbi, Python SQLite + libdbi-drivers + SQLite*.
  + Ngoài ra, cần thêm các gói make, gcc để build và compile chương trình từ source.
  + Tải về source của các package cần thiết tại các liên kết sau:

<http://sourceforge.net/projects/gammu/files/gammu/1.32.0/gammu-1.32.0.tar.bz2>

<http://www.cmake.org/files/v2.8/cmake-2.8.8.tar.gz>

<http://pkgconfig.freedesktop.org/releases/pkg-config-0.27.tar.gz>

Hoặc cài đặt nhanh thông qua lệnh yum:

# yum install cmake pkgconfig

* Có thể cần cài thêm gói usb\_modeswitch để chuyển từ chế độ storage sang modem nếu máy chủ CentOS chạy Nagios không nhận ra D-Com 3G như là một USB modem.
* Các gói phụ thuộc cần thiết cho usb\_modeswitch là: *libusb-devel, tcl*
* Cần tới quyền root trong quá trình cài đặt và cấu hình hệ thống.
* Tải về source tại:

<http://www.draisberghof.de/usb_modeswitch/usb-modeswitch-data-20121109.tar.bz2>

<http://www.draisberghof.de/usb_modeswitch/usb-modeswitch-1.2.5.tar.bz2>

### Các bước thực hiện

#### Kết nối USB 3G tới máy chủ Nagios

Kiểm tra xem CentOS đã nhận ra USB 3G là một GSM modem hay chưa:

# dmesg | grep GSM

* Nếu thấy output như sau thì đã ổn:

USB Serial support registered for GSM modem (1-port)

option 1-1:1.0: GSM modem (1-port) converter detected

usb 1-1: GSM modem (1-port) converter now attached to ttyUSB0

option 1-1:1.1: GSM modem (1-port) converter detected

usb 1-1: GSM modem (1-port) converter now attached to ttyUSB1

option 1-1:1.2: GSM modem (1-port) converter detected

usb 1-1: GSM modem (1-port) converter now attached to ttyUSB2

option 1-1:1.3: GSM modem (1-port) converter detected

usb 1-1: GSM modem (1-port) converter now attached to ttyUSB3

option: v0.7.2:USB Driver for GSM modems

(để ý là device file cho thiết bị là */dev/ttyUSB3*)

* Nếu output trống rỗng thì cần compile và install gói *usb\_modeswitch*

# tar jxf usb-modeswitch-data-20121109.tar.bz2

# cd usb-modeswitch-data-20121109

# make install

# tar jxf usb-modeswitch-1.2.5.tar.bz2

# cd usb-modeswitch-1.2.5

# make install

#### Biên dịch và cài đặt Gammu

Giải nén source package của Gammu và chuyển vào folder vừa được giải nén:

# tar jxvf gammu-1.32.0.tar.bz2

# cd gammu-1.32.0

Chạy các lệnh sau để configure, compile và install Gammu:

# ./configure

# make

# make install

Kiểm tra bằng lệnh:

# gammu

Nếu nhận được thông báo lỗi liên quan tới library thì chạy 2 dòng lệnh sau:

# ln -s /usr/local/lib/libGammu.so.7 /usr/lib/libGammu.so.7

# ln -s /usr/local/lib/libgsmsd.so.7 /usr/lib/libgsmsd.so.7

Nếu nền tảng là CentOS 64-bit thì thay thư mục “*lib*” thành “*lib64*” trong hai lệnh trên.

#### Cấu hình Gammu để gửi SMS

Tạo config file cho Gammu:

# vi /etc/gammurc

Nội dung của file là:

[gammu]

port = /dev/ttyUSB3

connection = at19200

Ngoài ra, có thể sử dụng lệnh sau để cấu hình dễ dàng cấu hình các thông số trên:

$ gammu-config

Kiểm tra USB 3G được nhận dạng:

# gammu --identify

Device : /dev/ttyUSB3

Manufacturer : ZTE CORPORATION

Model : unknown (MF190S)

Firmware : BD\_MF190SV1.0.0B01

IMEI : 864482000915806

Ngoài ra còn có 2 lệnh khác để theo dõi hoạt động của thiết bị là:

# gammu --monitor

# gammu --networkinfo

Nagios chạy với quyền của user nagios, vậy nên nếu muốn Nagios gửi thành công SMS, user nagios phải có quyền truy cập tới device file, config file và binary file của Gammu. Chạy các lệnh sau để gán các quyền thích hợp cho user nagios:

# cp /etc/gammurc /home/nagios/.gammurc

# chown nagios.nagios /home/nagios/.gammurc

# chmod 4755 /usr/local/bin/gammu

# usermod -a -G dialout nagios

# usermod -a -G dialout apache

Chuyển qua user nagios và thử gửi một SMS mẫu:

# su - nagios

# echo “test SMS nagios” | gammu –sendsms TEXT +8490xxxxxxx

If you want break, press Ctrl+C…

Sending SMS 1/1….waiting for network answer..OK, message reference=181

Nếu số điện thoại trong câu lệnh ở trên nhận được thông điệp “test SMS nagios” thì việc cấu hình để CentOS gửi SMS đã thành công.

#### Cấu hình Nagios để gửi cảnh báo qua SMS

Thêm 2 command mới sau vào config file */usr/local/nagios/etc/commands.cfg*

# ‘notify-host-by-sms’ command definition

define command{

command\_name notify-host-by-sms

command\_line /usr/bin/printf “%b” “\*\*\* Nagios \*\*\*\n Notification Type: $NOTIFICATIONTYPE$\n Host: $HOSTNAME$\n State: $HOSTSTATE$\n Address: $HOSTADDRESS$\n Info: $HOSTOUTPUT$\n Date/Time: $LONGDATETIME$” | /usr/local/bin/gammu --sendsms TEXT $CONTACTPAGER$

}

# ‘notify-service-by-sms’ command definition

define command{

command\_name notify-service-by-sms

command\_line /usr/bin/printf “%b” “\*\*\* Nagios \*\*\*\n Notification Type: $NOTIFICATIONTYPE$\n Service: $SERVICEDESC$\n Host: $HOSTALIAS$\n Address: $HOSTADDRESS$\n State: $SERVICESTATE$\n Date/Time: $LONGDATETIME$\n Additional Info: $SERVICEOUTPUT$” | /usr/local/bin/gammu --sendsms TEXT $CONTACTPAGER$

Thêm 1 contact mới sau vào config file */usr/local/nagios/etc/contacts.cfg*

define contact{

contact\_name admin

use generic-contact

alias Ngoc Ngoan

service\_notification\_options w,u,c,r

host\_notification\_options d,u,r

service\_notification\_commands notify-service-by-sms,notify-service-by-email

host\_notification\_commands notify-host-by-sms,notify-host-by-email

email admin@gmail.com

pager +84983xxxxxx

}

Kiểm tra cấu hình và khởi động lại Nagios

# /usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

# /etc/init.d/nagios restart

# /etc/init.d/nagios reload

Thử ngưng hoạt động một *host/service/netwok device* nào đó thì sau một khoảng thời gian quy định trước, Nagios sẽ kiểm tra trạng thái và gửi đi SMS thông báo tình trạng của chúng tới số điện thoại của người quản trị.

#### Gửi cảnh báo theo nhóm

Sau các bước đã thực hiện tại phần trên, tôi sẽ trình bày các bước để khởi tạo và định nghĩa các tài khoản (contact) cho những cá nhân và nhóm sẽ nhận được cảnh báo khi một máy tính, thiết bị hay dịch vụ nào đó trong hệ thống xảy ra sự cố.

Tùy chỉnh lại mẫu *generic-contact* mà Nagios cung cấp sẵn sau khi cài đặt để các contact được tạo ở bước sau sẽ nhận được thông báo qua SMS. Mở tập tin */usr/local/nagios/etc/objects/templates.cfg* và sửa mục *generic-contact* như sau:

define contact{

name generic-contact

service\_notification\_period 24×7

host\_notification\_period 24×7

service\_notification\_options w,u,c,r

host\_notification\_options d,u,r

service\_notification\_commands notify-service-by-sms, notify-service-by-email

host\_notification\_commands notify-service-by-sms, notify-host-by-email

register 0

}

Tiếp đến, thêm mới các contact cho những người muốn nhận thông báo qua email và SMS từ Nagios. Mở tập tin */usr/local/nagios/etc/objects/contacts.cfg* và bổ sung nội dung mẫu sau:

define contact{

contact\_name ngocngoan

use generic-contact

alias Ngoc Ngoan (Sysadmin)

email ngocngoan@gmail.com

pager +84983xxxxxx //thay bằng sđt thực

}

define contact{

contact\_name user

use generic-contact

alias DB User (Database)

email user@gmail.com

pager +84123xxxxxx //thay bằng sđt thực

}

Giờ ta sẽ gom nhóm các *contact* để gửi các thông báo thích hợp. Ví dụ, các thông báo liên quan tới thiết bị mạng chỉ được gửi tới nhóm network, thông báo liên quan tới máy chủ sẽ được gửi tới nhóm system, thông báo liên quan tới CSDL thì gửi tới nhóm database, v.v… Cũng trong tập tin *contacts.cfg* trên, tạo thêm các *contactgroup* theo mẫu sau:

define contactgroup{

contactgroup\_name db-admins

alias Database Administrators

members user,ngocngoan

}

define contactgroup{

contactgroup\_name ca-admins

alias Linux System Administrator

members ngocngoan

}

Việc khai báo các contact ở trên không có nghĩa là họ sẽ nhận được thông báo mà ta cần liên kết các contactgroup tới một dịch vụ hoặc máy tính nào đó cần giám sát. Ví dụ, trong thư mục */usr/local/nagios/etc/objects/* tạo ra 2 tập tin *ca-server.cfg* và *db-server.cfg* rồi định nghĩa như sau:

Trong file trong *ca-server.cfg*

define host{

use linux-server

host\_name ca-server

alias Corporate Email Server

address 192.168.1.16

contact\_groups ca-admins //nhóm Unix sẽ nhận notify

}

Trong file *db-server.cfg*

define service{

use generic-service

host\_name mysql-db

service\_description MySQL Database Status

contact\_groups db-admins //nhóm DB sẽ nhận notify

check\_command check\_nrpe!check\_mysql\_db

}

Cuối cùng, chạy các lệnh sau để kiểm tra cấu hình và khởi động lại Nagios

#/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

# /etc/init.d/nagios restart

# /etc/init.d/nagios reload

Kiểm tra hoạt động gửi theo nhóm bằng cách thử ngưng hoạt động của một máy tính, thiết bị hay dịch vụ nào đó thì sau một khoảng thời gian quy định trước, Nagios sẽ kiểm tra trạng thái và gửi đi SMS và email thông báo tình trạng của chúng tới số điện thoại và hộp thư của người quản trị.

## Tích hợp công cụ giám sát bằng đồ thị, Nagiosgraph

Nagiosgraph là một cụ giám sát bằng đồ thị được thiết kế để khai thác sức mạnh của công cụ lưu trữ dữ liệu và đồ thị RRDTool. Nagiosgraph là một add-on của Nagios, cung cấp một bộ thu thập thông tin nhanh, các mẫu đồ thị tiên tiến, đa phương pháp thu nhập dữ liệu. Nagios giám sát một hoặc nhiều dịch vụ trên mỗi host, trong khi đó Nagiosgraph trích xuất thông tin từ đầu ra của Nagios, xử lý và sau đó đưa vào trong một hoặc nhiều file cơ sở dữ liệu round-robin (RRD). Các kịch bản CGI hiển thị dữ liệu từ file RRD trên trình duyệt web. Đầu ra CGI có thể được nhúng trực tiếp vào Nagios để đồ thị hiển thị như các bản ghi khác.

Nagiosgraph là một giao diện đơn giản giữa Nagios và các file RRD, nó hoạt động trong hai chế độ. Hoặc là thu thập dữ liệu từ các dịch vụ giám sát hoặc là hiển thị đồ thị từ dữ liệu thu thập được. Tất cả các dữ liệu thu thập được lưu trong các file RRD sử dụng công cụ RRDtool. Nagiosgraph sẽ xử lý các dữ liệu để hiển thị các thông tin trên trình duyệt, quá trình xử lý sẽ thực hiện theo một trong hai dạng : xử lý theo nhóm hoặc xử lý trực tiếp. Nếu xử lý theo nhóm, dữ liệu sẽ được lưu trữ trong một file, sau đó Nagios sẽ thường xuyên cập nhật thông tin trong một khoảng thời gian định trước. Nếu xử lý trực tiếp, nagios sẽ lưu dữ liệu của mỗi dịch vụ trong một file tương ứng, sau đó cập nhật trực tiếp từ các file đó. Quá trình xử lý trực tiếp sẽ tốn nhiều tài nguyên CPU hơn xử lý theo nhóm vì dữ liệu liên tục được cập nhật.

Để cài đặt Nagiosgraph thì hệ thống phải chắc chắn đã cài đặt sẵn các gói hỗ trợ.

# yum install perl-rrdtool perl-GD perl-Time-HiRes

Sau đó tải mã nguồn tại

<http://sourceforge.net/projects/nagiosgraph/files/nagiosgraph/>

Giải nén mã nguồn và cài đặt

# tar xzf nagiosgraph-1.4.4.tar.gz

# cd nagiosgraph-1.4.4

# ./install.pl --prefix=/usr/local/nagiosgraph

Sau khi cài đặt xong, Nagiosgraph sẽ không hiển thị luôn. Để hiển thị được biểu đồ, ta phải thực hiện thêm các bước cấu hình. Trong đồ án này em sử dụng Nagioshraph xử lý trực tiếp dữ liệu, trong file cấu hình chính nagios.cfg thêm hai dòng sau:

process\_performance\_data=1

service\_perfdata\_command=process-service-perfdata

Định nghĩa thêm lệnh mới để xử lý các thông tin hiển thị

define command{

command\_name process-service-perfdata

command\_line /usr/local/nagiosgraph/bin/insert.pl "$LASTSERVICECHECK$||$HOSTNAME$||$SERVICEDESC$||$SERVICEOUTPUT$||$SERVICEPERFDATA$"

}

Kiểm tra cấu hình và khởi động lại Nagios

#/usr/local/nagios/bin/nagios -v /usr/local/nagios/etc/nagios.cfg

# /etc/init.d/nagios restart

Tới đây, Nagios vẫn chưa hiển thị được đồ thị trên nền web, do đó ta phải thêm đoạn cấu hình sau trong file cấu hình web, chẳng hạn như *httpd.cfg* trong thư mục */etc/httpd/conf.d*

ScriptAlias /nagiosgraph/cgi /usr/local/nagiosgraph/cgi

<Directory "/usr/local/nagiosgraph/cgi">

Options ExecCGI

AllowOverride None

Order allow,deny

Allow from all

</Directory>

Alias /nagiosgraph "/usr/local/nagiosgraph/share"

<Directory "/usr/local/nagiosgraph/share">

Options None

AllowOverride None

Order allow,deny

Allow from all

</Directory>

Khởi động lại dịch vụ web

# /etc/init.d/httpd restart

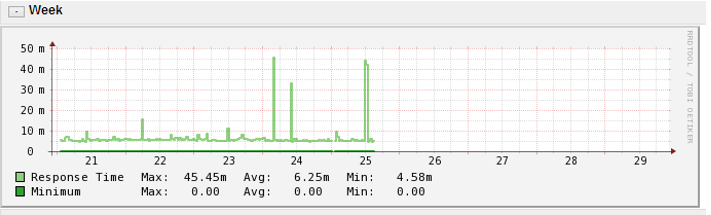
Kiểm tra Nagiosgraph hoạt động bằng cách chạy file *showconfig.cgi*

http://server/nagiosgraph/cgi/showconfig.cgi

Kiểm tra chế độ vẽ đồ thị bằng cách chạy file *show.cgi*

http://server/nagiosgraph/cgi/show.cgi

quá trình này sẽ hiển thị trên trang web danh sách các host và dịch vụ giám sát.



Hình 19 : Ví dụ giám sát dịch vụ LDAP bằng biểu đồ theo tuần