Práctica Nagios

Pablo Jiménez Mateo y Mihaita Alexandru Lupoiu

Diciembre 2014

Índice

1. Introducción a Nagios

Nagios es un sistema de monitorización de redes ampliamente utilizado, de código abierto, que vigila los equipos (hardware) y servicios (software) que se especifiquen, alertando cuando el comportamiento de los mismos no sea el deseado. Entre sus características principales figuran la monitorización de servicios de red (SMTP, POP3, HTTP, SNMP...), la monitorización de los recursos de sistemas hardware (carga del procesador, uso de los discos, memoria, estado de los puertos...), independencia de sistemas operativos, posibilidad de monitorización remota mediante túneles SSL cifrados o SSH, y la posibilidad de programar plugins específicos para nuevos sistemas[2].

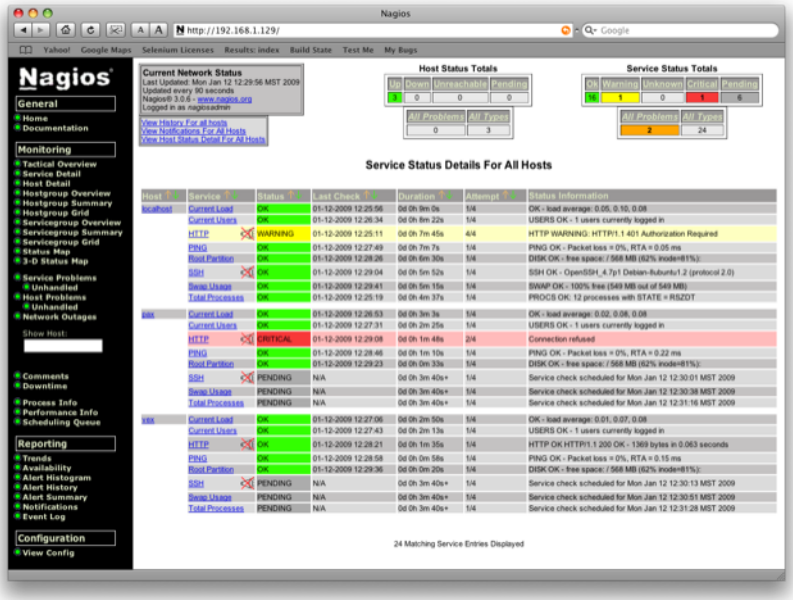


Figura 1: Interfaz de Nagios

Se trata de un software que proporciona una gran versatilidad para consultar prácticamente cualquier parámetro de interés de un sistema, y genera alertas, que pueden ser recibidas por los responsables correspondientes mediante (entre otros medios) correo electrónico y mensajes SMS, cuando estos parámetros exceden de los márgenes definidos por el administrador de red.

Nagios[1] fue originalmente diseñado para ser ejecutado en GNU/Linux, pero también se ejecuta bien en variantes de Unix. Está licenciado bajo la GNU General Public License Version 2 publicada por la Free Software Fundation.

1.1. Características

- Monitorización de Servicios de Red
- Monitorización de los recursos de equipos hardware (carga del procesador, uso de los discos, logs del sistema) en varios sistemas operativos, incluso Microsoft Windows con los plugins $NRPE_NT$ o $NSClient++$.
- Monitorización remota, a través de túneles SSL cifrados o SSH.
- Monitorización de Host y sus recursos como CPU, Memoria, Discos, etc
- Diseño simple de plugins, que permiten a los usuarios desarrollar sus propias comprobaciones de servicios dependiendo de sus necesidades, usando sus herramientas preferidas (Bash, C++, Perl, Ruby, Python, PHP, C#...).
- Comprobación de servicios paralizados.
- Posibilidad de definir la jerarquía de la red, permitiendo distinguir entre host caídos y host inaccesibles.
- Capacidad de Definir Host/Servicios padres o hijos, lo que permite detectar el origen del problema en caso de no ser de la propia máquina
- Notificaciones a los contactos cuando ocurren problemas en servicios o hosts, así como cuando son resueltos (a través del correo electrónico, buscapersonas, Jabber, SMS, o cualquier método definido por el usuario junto con su correspondiente complemento).
- Posibilidad de definir manejadores de eventos que ejecuten al ocurrir un evento de un servicio o host para resoluciones de problemas proactivas.
- Log de eventos
- Interface Web para la visualización de estados de servicio, históricos, Archivo de Log, etc
- Integración con herramientas que la comunidad ha desarrollado
- Multiplataforma, aunque fue desarrollado originalmente para correr sobre Linux
- Visualización del estado de la red en tiempo real a través de interfaz web, con la posibilidad de generar informes y gráficas de comportamiento de los sistemas monitorizados, y visualización del listado de notificaciones enviadas, historial de problemas, archivos de registros....

1.2. Beneficios de utilizar Nagios

- Supervisión contínua de la plataforma de TI
- Esto te permite mejorar los tiempos de disponibilidad de los servicios
- Alertar al equipo de TI ante alertas preventivas (Warning) o críticas (Critical)
- Reaccionar de manera preventiva y no reactiva ante los eventos que nagios detecte.
- Aumenta la productividad de las TIC
- Generar reportes de los eventos.
- Planificar mantenimiento de tu hardware o servicios.
- Planificar el cambio o renovación de la Infraestructura de TIC

1.3. Alternativas a Nagios

Algunas de las alternativas existentes a Nagios son:

- Pandora FMS[3] es un software de código abierto que sirve para monitorizar y medir todo tipo de elementos. Monitoriza sistemas, aplicaciones o dispositivos. Permite saber el estado de cada elemento de un sistema a lo largo del tiempo. Pandora FMS está orientado a grandes entornos, y permite gestionar con y sin agentes, varios miles de sistemas, por lo que se puede emplear en grandes clusters, centros de datos y redes de todo tipo.

Figura 2: Interfaz Pandora FMS

- Zenoss (Zenoss Core)[4] es una aplicación de informática de código abierto, plataforma para la gestión de red y servidores basada en el servidor de aplicaciones Zope. Zenoss Core provee una interfaz web que permite a los administradores de sistemas monitorizar disponibilidad, inventario/configuración, desempeño y eventos.

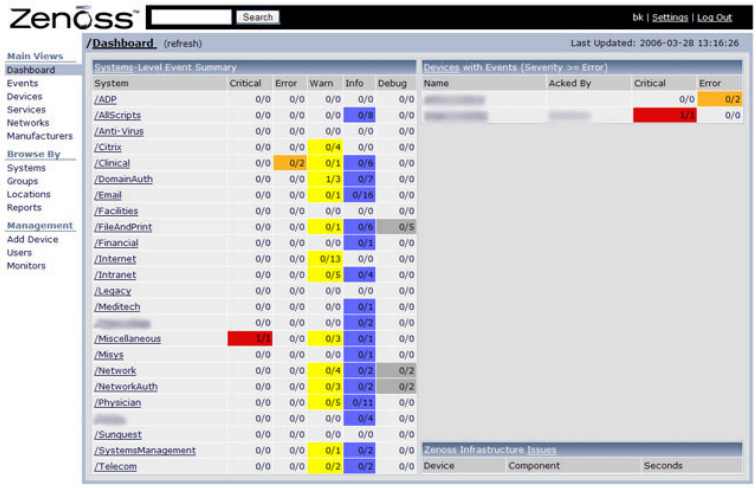


Figura 3: Interfaz Zenoss

2. Funcionamiento de Nagios

El funcionamiento básico de Nagios consiste en una arquitectura cliente-servidor mediante ejecución de polling periódico de comprobaciones de recursos (con agente) y servicios (sin agente) sobre sistemas cliente. Cuando se detecta un error la plataforma es capaz de enviar una notificación (sobre diferentes modos de comunicación) a los contactos administrativos, informando del estado del servicio que ha provocado el error, incluyendo informes de estado, de logs e históricos web).

Una gran parte de la monitorización se puede hacer remotamente. Todo servicio publicado en la red, se puede monitorizar sin necesidad de instalar nada en el host monitorizar. Sin embargo hay recursos que no se pueden monitorizar remotamente, como por ejemplo:

- Espacio libre en disco
- Memoria en uso
- Carga de CPU

- Servicios activos
- Cantidad de usuarios logueados

Para ello es necesario instalar agentes. Estos agentes, mediante plugins, comprueban el estado de los recursos y envían los datos al servidor Nagios. El tipo de conexión es pull, es decir, el servidor Nagios contacta cada cierto intervalo de tiempo a los agentes, les indica qué comandos ejecutar y obtiene los resultados.

Existen varios protocolos utilizados para la comunicación entre agentes y Nagios:

- NRPE Nagios Remote plugin Executor. Es el protocolo más utilizado y recomendado por Nagios. Se accede mediante el plugin check_nrpe.
- NSCA Nagios Service Check Acceptor.
- NSCP: protocolo nativo de NSClient++
- NRDP: reemplazo NSCA.
- Syslog: protocolo estándar para transmisión de logs en Unix.
- SNMP: Simple Network Management Protocol. Protocolo estándar para administración y monitorización de dispositivos de red.

Nagios tiene un núcleo de la aplicación que forma la lógica de control de negocio. De la aplicación contiene el software necesario para realizar la monitorización de los servicios y máquinas de la red para la que está preparado. Hace uso de diversos componentes que vienen con la aplicación, y puede hacer uso de otros componentes realizados por terceras personas.

Aunque permite la captura de paquetes SNMP Trap para notificar sucesos, no es un sistema de monitorización y gestión basado en SNMP sino que realiza su labor basándose en una gran cantidad de pequeños módulos software que realizan comprobaciones de parte de la red.

Muestra los resultados de la monitorización y del uso de los diversos componentes en una interfaz web a través de un conjunto de CGI's y de un conjunto de páginas HTML que vienen incorporadas de serie. Y que permiten al administrador una completa visión de qué ocurre, dónde y en algunos casos, por qué. Por último, si se compila para ello, Nagios guardará los históricos en una base de datos para que al detener y reanudar el servicio de monitorización, todos los datos sigan como iban, sin cambios.

Así que en las siguientes páginas, antes de profundizar en el uso de Nagios, vamos a explicar cómo se realiza correctamente la instalación de este sistema, que no es nada trivial; al menos no es algo rápido.

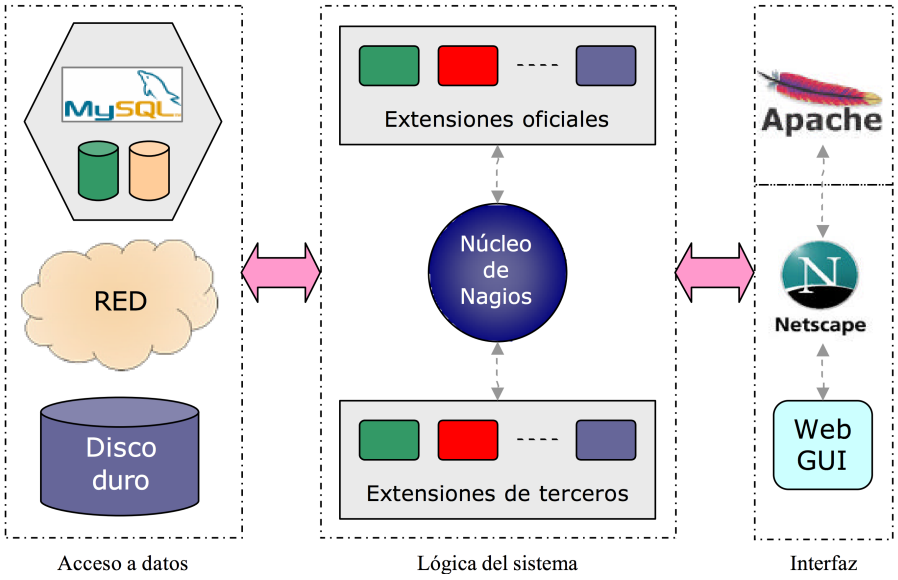


Figura 4: Estructura Nagios

3. Instalación de Nagios

En este apartado se explica cómo instalar Nagios de dos formas, desde repositorio y también compilando su código fuente. Ambos métodos tienen como distribución objetivo CentOs/RedHat.

3.1. Instalación desde repositorio

Suponiendo que se tiene acceso root a la máquina, el comando a ejecutar como superusuario es el siguiente

```
yum install nagios*
```

esto se encargará de instalar Nagios, sus dependencias y los plugins. Los plugins nos permiten un mayor control de los sistemas a monitorizar así como una visualización más completa de la información.

3.2. Instalación desde código fuente

En primer lugar debemos crear el usuario y grupo al que pertenecerá la ejecución de Nagios. Esto se realiza con los siguientes comandos en modo superusuario

```
groupadd nagios
groupadd nagcmd
useradd nagios -g nagios
useradd nagios -g nagcmd
```

El usuario Nagios será el que tendrá en ejecución el daemon correspondiente. Además hemos creado dos grupos, **nagios** y **nagcmd**

- **nagios**: El grupo principal de Nagios.
- **nagcmd**: Este es el grupo que permite la ejecución de comandos externos a través de la interfaz web.

Ahora necesitamos instalar las dependencias, incluyendo el servidor web. De nuevo como superusuario ejecutamos lo siguiente

```
yum -y install httpd php glibc glibc-common gd gd-devel net-snmp
yum -y groupinstall "Development Tools"
```

httpd y **php** serán los encargados del servidor web. **glibc**, **glibc-common**, **gd** y **gd-devel** son librerías para el manejo de gráficos y **net-snmp** es el que incluye las librerías para manejar el protocolo SNMP, que nos permitirá monitorizar los ordenadores en red.

Development Tools instala todos los paquetes necesarios para poder compilar los códigos fuente. Incluye, como pequeño ejemplo, gcc y make.

Ahora ya estamos listos para descargar el código fuente. En el momento de escribir esta memoria la última versión disponible es la 4.0.8 y es la que se usará a lo largo de la memoria, pero los pasos deberían ser iguales o muy similares en las versiones posteriores.

En primer lugar, descargamos y descomprimimos las fuentes de Nagios Core. Nagios Core es el engine que tiene toda la lógica de la aplicación, sobre él se añadirán plugins y demás para personalizarlo.

```
wget http://netcologne.dl.sourceforge.net/project/nagios/
            nagios-4.x/nagios-4.0.8/nagios-4.0.8.tar.gz
tar -xzf nagios-4.0.8.tar.gz
```

Accedemos a la carpeta y realizamos un **./configure**

```
cd nagios-4.0.8
./configure
```

./configure comprueba que las dependencias del programa estén satisfechas (por lo menos las esenciales). En caso de no cumplirse algún requisito indispensable el programa saldrá con un mensaje de error indicando qué se debe instalar. Si el programa considera que se cumplen los requisitos mínimos generará un fichero **Makefile** con los comandos necesarios para compilar e instalar el programa.

Una vez el comando ha acabado ejecutamos

```
make all
make install
make install-init
make install-commandmode
make install-config
make install-webconf
```

- **all**: Compila el programa dejando los ejecutables preparados para ejecutarse desde local o para instalarse.
- **install**: Copia los ejecutables base a sus respectivas carpetas para que se puedan encontrar desde el $PATH.
- **install-init**: Este comando instala el fichero en /etc/rc.d/init.d para poder iniciarlo y pararlo de manera cómoda.
- **install-commandmode**: Instala y configura los permisos en el directorio para almacenar los ficheros externos de comandos.
- **install-config**: Este comando instala plantillas de configuración en /usr/local/nagios/etc para que sea más fácil su edición.
- **install-webconf**: Instala el fichero de configuración de Apache.

Ahora vamos a compilar e instalar los plugins desde código fuente, a fecha de escribir este artículo la última versión de los plugins es la 2.0.3

```
wget http://nagios-plugins.org/download/nagios-plugins-2.0.3.tar.gz
tar -xzf nagios-plugins-2.0.3.tar.gz
cd nagios-plugins-2.0.3
```

Una vez descargados procedemos a su configuración, compilación e instalación.

```
./configure
make
make install
```

Con esto tendríamos instalado Nagios y sus plugins desde código fuente, ahora sólo queda configurarlos. Esto se explica más adelante.

4. Configuración

4.1. Configuración de Nagios

En esta sección se explica cómo configurar Nagios paso a paso.

En primer lugar debemos crear los usuarios. En esta memoria crearemos un usuario administrador y otro que sólo pueda consultar la interfaz web.

```
htpasswd -c /usr/local/nagios/etc/htpasswd.users administrador
htpasswd /usr/local/nagios/etc/htpasswd.users usuario
```

En ambos casos se pedirá que se introduzca una contraseña, en nuestro ejemplo serán **labora2000** y **practicas** respectivamente. El parámetro -c usado en el primer comando crea el fichero si no existe, de existir lo sobreescribe.

El siguiente paso es cambiar el e-mail de contacto. Este e-mail será el que recibirá todas las notificaciones de Nagios. Para ello editamos el fichero /usr/local/nagios/etc/objects/contacts.cfg y cambiamos la siguiente línea para que contenga nuestro e-mail

```
email  usuario@gmail.com; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS ******
```

Ahora debemos decirle a Apache que carge el módulo PHP o la web no funcionará correctamente. Para ello modificamos el fichero /etc/httpd/conf/httpd.conf y añadimos la línea

```
LoadModule php5_module modules/libphp5.so
```

con esto nos aseguramos de que funcione PHP en nuestro servidor web.

Por último configuramos los usuarios de Nagios. Para ello hay que editar el fichero /usr/local/nagios/etc/cgi.cfg

Hay que editar las líneas siguientes para incluir los usuarios que hemos creado

```
authorized_for_system_information=administrador,usuario
authorized_for_configuration_information=administrador
authorized_for_system_commands=administrador
authorized_for_all_services=administrador,usuario
authorized_for_all_hosts=administrador,usuario
authorized_for_all_service_commands=administrador
authorized_for_all_host_commands=administrador
```

Las líneas hacen lo siguiente:

- **authorized_for_system_information**: Estos usuarios podrán ver los procesos e información de los sistemas en la interfaz web.
- **authorized_for_configuration_information**: Estos usuarios podrán ver los comandos y la configuración de los sistemas a través de la interfaz web.
- **authorized_for_system_commands**: Estos usuarios podrán apagar y reiniciar Nagios.
- **authorized_for_all_services**: Estos usuarios podrán ver la información de todos los servicios monitorizados.
- **authorized_for_all_hosts**: Estos ususarios podrán ver la información de todos los sistemas monitorizados.
- **authorized_for_all_service_commands**: Estos usuarios podrán ejecutar comandos sobre todos los servicios.
- **authorized_for_all_host_commands**: Estos usuarios podrán ejecutar comandos sobre todos los sistemas.

De esta manera sólo le damos poderes totales al usuario **administrador** y poderes de consulta al usuario **usuario**.

Con esto hemos acabado la configuración de Nagios, pero no es muy útil si no añadimos algunos hosts externos para monitorizar.

4.2. Configuración de los hosts externos

Para comenzar la configuración, crearemos el fichero /usr/local/nagios/etc/objects/hosts.cfg que contendrá todos los parámetros, hosts y servicios a configurar.

La plantilla básica que vamos a usar para configurar cada uno de los hosts es la siguiente

```
define host {

    host_name                PC_108
    alias                    108 lab
    address                  150.128.49.108
    check_command            check-host-alive
    max_check_attempts       3
    check_period             24x7
    notification_intervaL    180
```

```
    notification_period      24x7
    notification_optionS     d,r,u

}
```

- **host_name**: Un nombre corto y único para identificar el ordenador.
- **alias**: Un nombre largo o descripción para identificar el ordenador.
- **address**: La dirección IP
- **check_command**: Comando que se ejecutará para saber si el host está disponible o no.
- **max_check_attempts**: El número de intentos que realizará Nagios para comprobar si un host no está disponible después de que este no le responda.
- **check_period**: Indica el intervalo sobre el cual se puede hacer checks sobre el servidor.
- **notification_interval**: El intervalo que tardará Nagios en decirnos que el host todavía no está disponible.
- **notification_period**: Indica el intervalo sobre el cual se pueden enviar notificaciones a los usuarios.
- **notification_options**: Indica cuándo enviar notificaciones. **d**: Indicar cuando el host no está disponible. **u**: Indicar cuándo el host no es accesible. **r**: Indicar cuándo se ha recuperado el servidor.

Tendremos un bloque como el anterior **para cada uno** de los hosts que pretendemos monitorizar.

El siguiente paso es el de configurar grupos, de manera que se pueda asignar la monitorización de un servicio al grupo entero en lugar de ir host por host. La plantilla es la siguiente

```
define hostgroup {

    hostgroup_name    grupolab1
    alias             Grupo laboratorio
    members           PC_108, PC_107, PC_105

}
```

- **hostgroup_name**: Nombre por el que se identifica el grupo.

- **alias**: Pequeña descripción del grupo.
- **members**: Hosts por nombre que pertenecen al grupo.

Y por último la plantilla usada para definir los servicios que queremos monitorizar es la siguiente

```
define service{

    hostgroup_name          grupolab1
    service_description     SSH
    check_period            24x7
    max_check_attempts      4
    contact_groups          admins
    notification_options    w,u,c,r
    notification_interval   960
    notification_period     24x7
    check_command           check_ssh

}
```

- **hostgroup_name**: El grupo o grupos a los que se le aplicará este servicio.
- **service_description**: Una pequeña descripción del servicio.
- **check_period**: Indica el intervalo sobre el cual se puede hacer checks sobre el servicio.
- **max_check_attempts**: El número de intentos que realizará Nagios para comprobar si un servicio no está disponible después de que este no le responda.
- **contact_groups**: A qué grupo hay que enviarle las notificaciones.
- **notification_options**: Indica bajo qué condiciones se envían notificaciones. **w**: Cuando está en WARNING. **u**: Cuando está en UNKNOWN. **c**: Cuando está en CRITICAL. **r**: Cuando se ha recuperado.
- **notification_interval**: El intervalo que tardará Nagios en decirnos que el servicio todavía no está disponible.
- **notification_period**: Indica el intervalo sobre el cual se pueden enviar notificaciones a los usuarios.
- **check_command**: El comando que se ejecutará en este servicio.

El ejemplo anterior monitoriza el estado del servicio SSH en el grupo grupolab1.

Ahora queda decirle a Nagios que lea el fichero que acabamos de crear, para ello hay que editar el fichero /usr/local/nagios/etc/nagios.cfg y hay que añadir la línea

```
cfg_file=/usr/local/nagios/etc/objects/hosts.cfg
```

Ahora sólo queda reiniciar los servicios y todo estará funcionando.

```
service httpd restart
service nagios restart
```

5. Acceso a Nagios mediante el navegador

Para acceder a la interfaz web de Nagios una vez seguidos los pasos anteriores, no hay más que ir a `http://localhost/nagios/`. Nos pedirá usuario y contraseña, si has estado atento a la memoria sabrás cuáles son.

La página principal es la siguiente

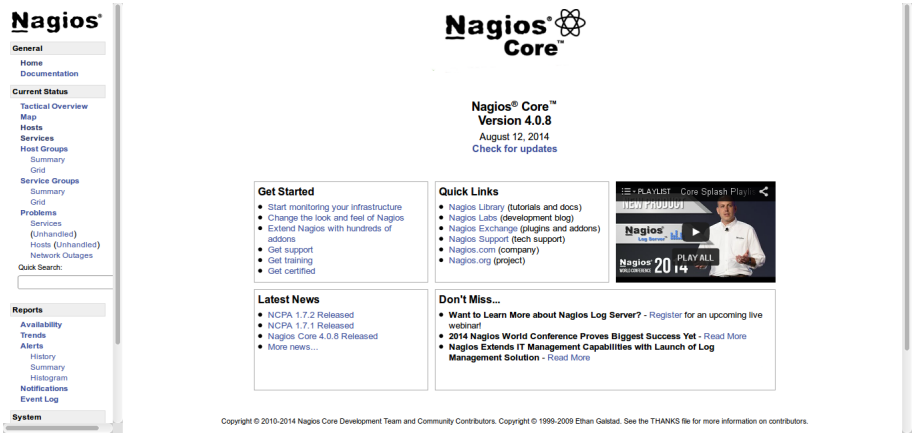


Figura 5: Página principal.

En la columna izquierda se puede pinchar en el enlace Hosts para poder ver el estado de los hosts que monitorizamos.

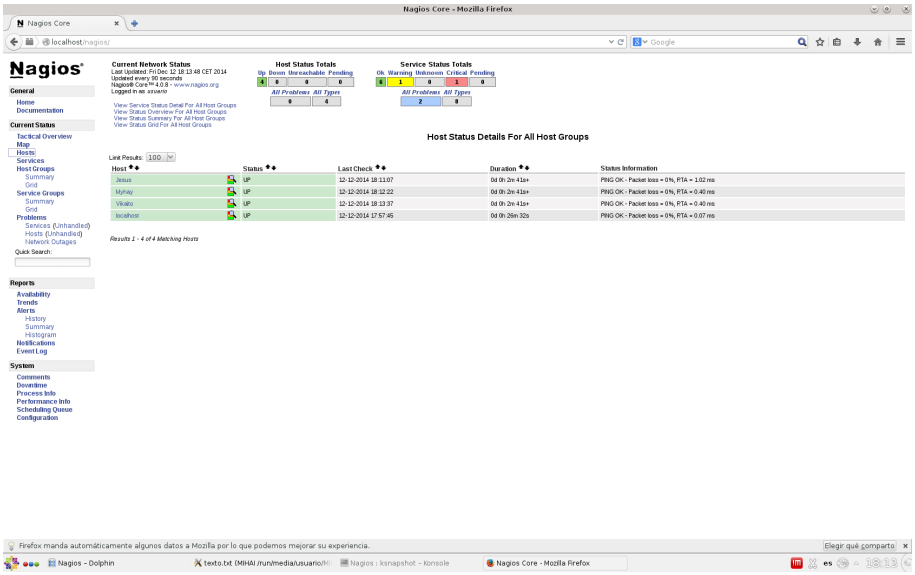
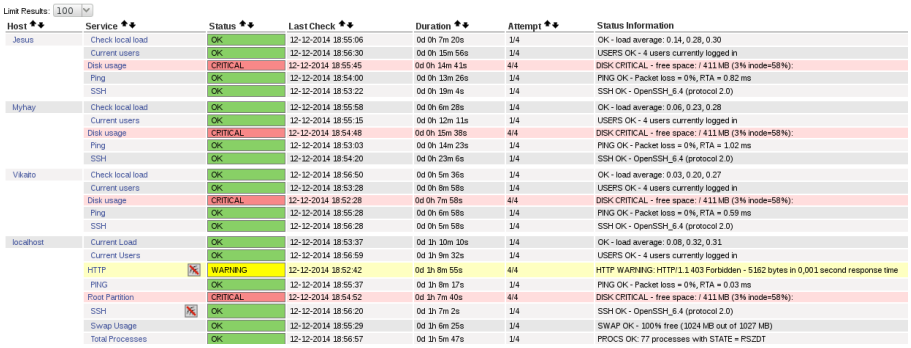


Figura 6: Página de hosts.

Si pinchamos en la opción de Services de la columna izquierda podremos ver los servicios que han sido configurados para cada host y sus estados



Limit Results: 100

Host	Service	Status	Last Check	Duration	Attempt	Status Information
Jesus	Check local load	OK	12-12-2014 18:55:06	0d 0h 7m 20s	1/4	OK - load average: 0.14, 0.28, 0.30
	Current users	OK	12-12-2014 18:56:30	0d 0h 15m 56s	1/4	USERS OK - 4 users currently logged in
	Disk usage	CRITICAL	12-12-2014 18:55:45	0d 0h 14m 41s	4/4	DISK CRITICAL - free space: / 411 MB (3% inode=58%):
	Ping	OK	12-12-2014 18:54:00	0d 0h 13m 26s	1/4	PING OK - Packet loss = 0%, RTA = 0.82 ms
	SSH	OK	12-12-2014 18:53:22	0d 0h 19m 4s	1/4	SSH OK - OpenSSH_6.4 (protocol 2.0)
Myhay	Check local load	OK	12-12-2014 18:55:58	0d 0h 6m 28s	1/4	OK - load average: 0.06, 0.23, 0.28
	Current users	OK	12-12-2014 18:55:15	0d 0h 12m 11s	1/4	USERS OK - 4 users currently logged in
	Disk usage	CRITICAL	12-12-2014 18:54:48	0d 0h 15m 38s	4/4	DISK CRITICAL - free space: / 411 MB (3% inode=58%):
	Ping	OK	12-12-2014 18:53:03	0d 0h 14m 23s	1/4	PING OK - Packet loss = 0%, RTA = 1.02 ms
	SSH	OK	12-12-2014 18:54:20	0d 0h 23m 6s	1/4	SSH OK - OpenSSH_6.4 (protocol 2.0)
Vikaito	Check local load	OK	12-12-2014 18:56:50	0d 0h 5m 36s	1/4	OK - load average: 0.03, 0.20, 0.27
	Current users	OK	12-12-2014 18:53:28	0d 0h 8m 58s	1/4	USERS OK - 4 users currently logged in
	Disk usage	CRITICAL	12-12-2014 18:52:28	0d 0h 7m 58s	4/4	DISK CRITICAL - free space: / 411 MB (3% inode=58%):
	Ping	OK	12-12-2014 18:55:28	0d 0h 6m 58s	1/4	PING OK - Packet loss = 0%, RTA = 0.59 ms
	SSH	OK	12-12-2014 18:56:28	0d 0h 5m 58s	1/4	SSH OK - OpenSSH_6.4 (protocol 2.0)
localhost	Current Load	OK	12-12-2014 18:53:37	0d 1h 10m 10s	1/4	OK - load average: 0.08, 0.32, 0.31
	Current Users	OK	12-12-2014 18:56:59	0d 1h 9m 32s	1/4	USERS OK - 4 users currently logged in
	HTTP	WARNING	12-12-2014 18:52:42	0d 1h 8m 55s	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 5162 bytes in 0,001 second response time
	PING	OK	12-12-2014 18:55:37	0d 1h 8m 17s	1/4	PING OK - Packet loss = 0%, RTA = 0.03 ms
	Root Partition	CRITICAL	12-12-2014 18:54:52	0d 1h 7m 40s	4/4	DISK CRITICAL - free space: / 411 MB (3% inode=58%):
	SSH	OK	12-12-2014 18:56:20	0d 1h 7m 2s	1/4	SSH OK - OpenSSH_6.4 (protocol 2.0)
	Swap Usage	OK	12-12-2014 18:55:29	0d 1h 6m 25s	1/4	SWAP OK - 100% free (1024 MB out of 1027 MB)
	Total Processes	OK	12-12-2014 18:56:57	0d 1h 5m 47s	1/4	PROCS OK: 77 processes with STATE = RSZDT

Figura 7: Página de servicios.

También se puede ver un mapa de la topología de la red si pinchamos en Map

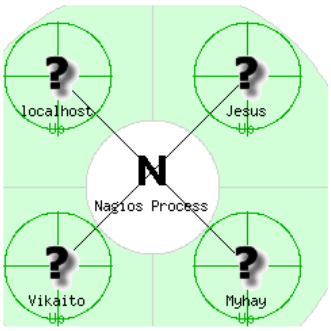


Figura 8: Página de la topología de red.

Se puede jugar con la interfaz para ver cómo funciona con mayor profundidad.

6. Conclusión

Nagios es un sistema muy potente pero que requiere bastante tiempo o conocimientos para su correcta configuración. Es una herramienta para usuarios avanzados con conocimientos de sistemas Linux.

7. Experiencia de instalación y configuración de Nagios

Se deben detallar los pasos que se han seguido para la instalación y configuración de Nagios tanto para monitorizar el sistema local como sistemas remotos.

Referencias

[1] Página web del proyecto de Nagios: `http://www.nagios.org`

[2] Wikipedia Nagios: `http://es.wikipedia.org/wiki/Nagios`

[3] Página web del proyecto de PandoraFMS: `http://pandorafms.com/`

[4] Página web del proyecto de Zenoss: `http://www.zenoss.com/`