

# **Nagios®**

## EN SERVIDOR Debian



## Con Clientes

### Windows

### Linux Debian/Ubuntu

### CentOS

### Mac OS X

# Índice

• Introducción a Nagios	3.
• Introducción a SNMP	4.
• Hardware utilizado	5.
• Máquinas Virtuales	5.
• Instalación de Nagios3	7.
• Archivos principales Nagios3	14.
• Directorios relevantes Nagios3	14.
• Opciones y menús de Nagios3	14.
• Solucionando la primera alerta	19.
• Añadiendo equipos a monitorear	21.
◦ Equipos Windows	22.
◦ Debian/Ubuntu	29.
◦ Routers/Switches	31.
◦ Mac OS X	34.
◦ CentOS	45.
• Introducir equipos en un grupo	47.
• Asignar iconos para el "statusmap"	47.
• Otras opciones del menú de Nagios3	48.
• Configuración de contactos	54.
• Horarios de notificaciones	55.
• Chequeos máximos antes de notificar	55.
• Alertas a través de e-mail	56.
• Alertas a través de SMS	58.
• Touchmon, Nagios en tu iDevice	59.
• Problemas durante el desarrollo	62.
• Experiencia con Nagios	62.
• Conclusión después del PI	62.
• Agradecimientos	62.

# Nagios como sistema de monitorización

Nagios es un sistema de monitorización SNMP open source. Monitorea los hosts y servicios que se especifiquen, alertando cuando algo sale mal y nuevamente cuando esta bien.

Originalmente tuvo el nombre de [Netsaint](#), fue creado y es mantenido actualmente por Ethan Galstad, junto con un grupo de desarrolladores de software que mantienen también varios plugins.

Nagios fue originalmente diseñado para ser ejecutado en Linux, pero también se ejecuta bien en variantes de Unix.

Nagios está licenciada bajo la [GNU General Public License](#) Version 2 publicada por la Free Software Foundation.

- Monitoreo de servicios de red (SMTP, POP3, HTTP, NTTP, ICMP, SNMP).
- Monitoreo de los recursos de un host (carga del procesador, uso de los discos, logs del sistema) en varios sistemas operativos, incluso Microsoft Windows con el plugin [NRPE\\_NT](#).
- Monitoreo remoto, a través de túneles SSL cifrados o SSH.
- Diseño simple de plugins, que permiten a los usuarios desarrollar sus propios chequeos de servicios dependiendo de sus necesidades, usando sus herramientas preferidas (Bash, C++, Perl, Ruby, Python, PHP, C#, etc.).
- Chequeo de servicios paralizados.
- Posibilidad de definir la jerarquía de la red, permitiendo distinguir entre host caídos y host inaccesibles.
- Notificaciones a los contactos cuando ocurren problemas en servicios o hosts, así como cuando son resueltos (via email, pager, SMS, o cualquier método definido por el usuario junto con su correspondiente plugin).
- Posibilidad de definir manejadores de eventos que ejecuten al ocurrir un evento de un servicio o host para resoluciones de problemas proactivas.
- Rotación automática del archive de log.
- Soporte para implementar host de monitores redundantes.
- Interfaz web opcional, para observar el estado de la red actual, notificaciones, historial de problemas, archivos de logs, etc.

Para más información dejo un enlace a la documentación oficial en formato PDF [AQUÍ](#)

# SNMP

El SNMP es un protocolo de la capa de aplicación de la suite de protocolos TCP/IP, que facilita el intercambio de información administrativa entre dispositivos de red a fin de que los administradores puedan supervisar el desempeño de la red, buscar y resolver sus problemas, y planear su crecimiento.

El funcionamiento de SNMP es sencillo (como lo sugiere su nombre), aunque su implementación es un poco más compleja ya que utiliza la capa de transporte de TCP/IP mediante el envío de datagramas UDP, lo cual lo hace poco fiable (en UDP no se garantiza la recepción de los paquetes enviados, como en TCP).

El esquema es sencillo, sin embargo su complejidad se incrementa a la hora de definir las variables (y su formato). Las variables ofrecidas para consulta por los agentes SNMP se definen a través de una MIB (Management Information Base, Base de Información de Gestión). La MIB (hay sólo una aunque existen múltiples extensiones a ésta) es una forma de determinar la información que ofrece un dispositivo SNMP y la forma en que se representa.

Una red administrada a través SNMP consiste de tres componentes claves:

- Dispositivos administrados.
- Agentes.
- Sistemas administradores de red (NMS's o gestores)

Un dispositivo administrado es un nodo de red que contiene un agente SNMP y reside en una red administrada. Estos recogen y almacenan información de administración, la cual es puesta a disposición de los NMS's usando SNMP. Los dispositivos administrados, a veces llamados elementos de red, pueden ser routers, servidores de acceso, switches, bridges, hubs, computadores o impresoras.

Un agente es un módulo de software de administración de red que reside en un dispositivo administrado. Un agente posee un conocimiento local de información de administración, la cual es traducida a un formato compatible con SNMP. Ofrece unas determinadas variables al exterior, para consulta o cambio. Así mismo, un agente puede enviar alertas a otros agentes para reportar ciertas condiciones y cambios de estado a un proceso de administración. Cada agente SNMP ofrece información dentro de una MIB, tanto de la general (definida en los distintos RFCs) como de aquellas extensiones que desee proveer cada uno de los fabricantes. Así, los fabricantes de routers han extendido las MIBs estándar incluyendo información específica de sus equipos.

Un NMS ejecuta aplicaciones que supervisan y controlan a los dispositivos administrados por medio de la recepción de las alertas (traps o notificaciones) enviadas por los dispositivos administrados. Estos gestores proporcionan el volumen de recursos de procesamiento y memoria requeridos para la administración de la red. En cualquier red administrada debe existir al menos un gestor.

Con SNMP se puede monitorear el estado de un enlace punto a punto para detectar cuando está congestionado y tomar así medidas oportunas, se puede hacer que una impresora alerte al administrador cuando se ha quedado sin papel, o que un servidor envíe una alerta cuando la carga se incrementa significativamente. SNMP también permite modificar remotamente la configuración de dispositivos, de forma que se pueden cambiar las direcciones IP de un sistema a través de su agente SNMP, u obligar a la ejecución de comandos (si el agente ofrece las funcionalidades necesarias).

# Hardware

Para este proyecto usaré 5 máquinas virtuales y el equipo anfitrión.

Como el tema que nos ocupa no es la virtualización usaré VirtualBox sobre Windows 7 en vez de otras opciones como Xen o KVM sobre Linux.

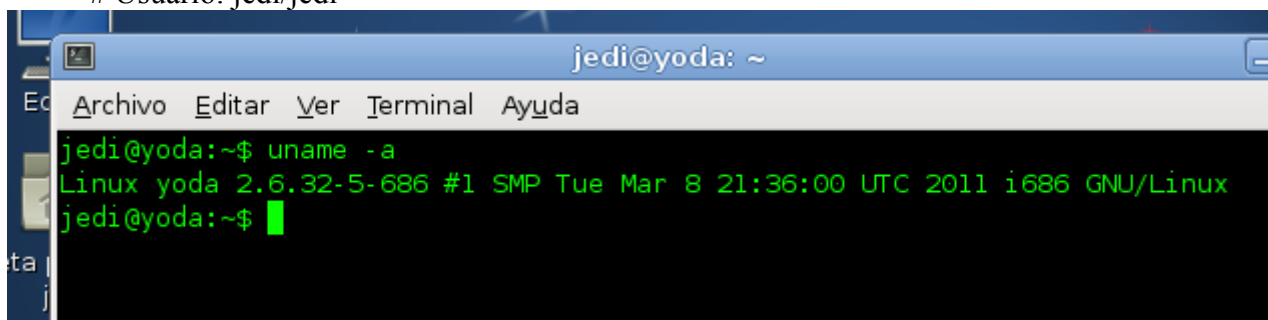
El equipo anfitrión es un HP Pavilion DV6 2165es:

- Intel i5 M430.
- 4GB ram DDR3.
- SO Windows 7 Home Premium 64 bits.

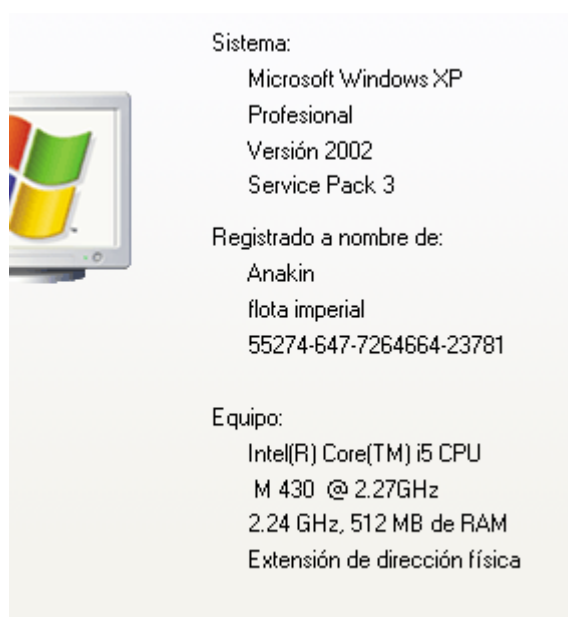
## Máquinas Virtuales

Tendremos 5 máquinas virtuales, una linux Debian que haré de servidor, una CentOS, una Mac OS X, una Ubuntu y una Windows como supuestos clientes para hacer las pruebas.

- Yoda: Servidor Debian 6.0.1 32 bits 1024MB ram HDD 8GB.  
# Root: yoda/yoda.  
# Usuario: jedi/jedi



- Anakin: Equipo de trabajo Windows XP SP3 32bits 512ram HDD 7,5GB.  
# Contraseña de administrador: anakin.



- Obin-Wan: Equipo de trabajo con Ubuntu 11.04 32bits 512MB ram HDD 8GB.

```
root@obiwan: /home/obi-wan
Archivo Editar Ver Buscar Terminal Ayuda
root@obiwan:/home/obi-wan# uname -a
Linux obiwan 2.6.38-8-generic #42-Ubuntu SMP Mon Apr 11 03:31:24 UTC 2011 x86_64
x86_64 x86_64 GNU/Linux
root@obiwan:/home/obi-wan#
```

- DarkMouth: Equipo de trabajo con Mac OS X Leopard 1GB ram HDD 20GB.



- C3PO: Equipo de trabajo con CentOS 1,5GB ram 15GB HDD.



# Instalación Nagios

Configuramos los repositorios de la siguiente forma (#nano /etc/apt/sources.list):

##OFICIALES

deb http://ftp.us.debian.org/debian/ squeeze main contrib non-free

deb-src http://ftp.us.debian.org/debian/ squeeze main contrib non-free

##SEGURIDAD

deb http://security.debian.org/ squeeze/updates main contrib non-free

deb-src http://security.debian.org/ squeeze/updates main contrib non-free

##MULTIMEDIA

deb http://www.debian-multimedia.org/ squeeze main non-free

deb-src http://www.debian-multimedia.org/ squeeze main non-free

Primero instalaremos el servidor en la máquina "yoda". Antes de hacer nada, comprobamos que tenemos conexión a internet y hacemos desde el terminal:

- #aptitude update
- #aptitude upgrade

Ahora instalaremos el servidor web "Apache2":

- #aptitude search apache2

```
root@yoda:/home/jedi# aptitude search apache2
p  apache2                - Apache HTTP Server metapackage
p  apache2-dbg            - Apache debugging symbols
v  apache2-dev            -
p  apache2-doc            - Apache HTTP Server documentation
v  apache2-mpm            -
p  apache2-mpm-event      - Apache HTTP Server - event driven model
p  apache2-mpm-itk        - multiuser MPM for Apache 2.2
p  apache2-mpm-prefork    - Apache HTTP Server - traditional non-threaded
p  apache2-mpm-worker     - Apache HTTP Server - high speed threaded r
p  apache2-prefork-dev    - Apache development headers - non-threaded
p  apache2-suexec         - Standard suexec program for Apache 2 mod_s
p  apache2-suexec-custom  - Configurable suexec program for Apache 2 r
p  apache2-threaded-dev   - Apache development headers - threaded MPM
p  apache2-utils          - utility programs for webserver
p  apache2.2-bin          - Apache HTTP Server common binary files
p  apache2.2-common       - Apache HTTP Server common files
p  libapache2-mod-authnz-external - authenticate Apache against external auth
p  libapache2-mod-php5    - server-side, HTML-embedded scripting langu
p  libapache2-mod-php5filter - server-side, HTML-embedded scripting langu
p  libapache2-svn         - Subversion server modules for Apache
p  mahara-apache2         - Electronic portfolio, weblog, and resume b
p  rt3.8-apache2         - Apache 2 specific files for request-tracke
root@yoda:/home/jedi# aptitude install apache2
```

· #aptitude install apache2

```
root@yoda:/home/jedi# aptitude install apache2
Se instalarán los siguiente paquetes NUEVOS:
  apache2 apache2-mpm-worker{a} apache2-utils{a} apache2.2-bin{a}
  apache2.2-common{a} libapr1{a} libaprutil1{a} libaprutil1-dbd-sqlite3{a}
  libaprutil1-ldap{a}
0 paquetes actualizados, 9 nuevos instalados, 0 para eliminar y 0 sin actualizar
.
Necesito descargar 2053 kB de ficheros. Después de desempaquetar se usarán 6844
kB.
¿Quiere continuar? [Y/n/?] Y
```

Ahora instalamos las siguientes librerías:

```
root@yoda:/home/jedi# aptitude install libapache2-mod-php5
```

Encontraremos un conflicto y nos dara como solución borrar el paquete "apache2-mpm-worker", a lo que aceptamos.

Instalamos el siguiente paquete:

```
root@yoda:/home/jedi# aptitude search build-essential
p   build-essential          - Lista informativa de los paquetes build-es
root@yoda:/home/jedi#
```

Comprobamos que se ha instalado correctamente:

```
root@yoda:/home/jedi# dpkg -l build-essential
Deseado=Desconocido/Instalar/Eliminar/Purgar/Retener
| Estado=No/Instalado/Config-files/Desempaquetado/Medio-conf/Medio-inst/espera-disparo/p
endiente-disparo
|/ Err?=(ninguno)/Requiere-reinst (Estado,Err: mayúsc.=malo)
||/ Nombre                Versión                Descripción
+++-----+-----+-----+
ii build-essential        11.5                   Informational list of build-essential packages
root@yoda:/home/jedi#
```

Procedemos a la instalación real del propio Nagios:

· Hacemos una búsqueda para ver los posibles paquetes.

```
root@yoda:/home/jedi# aptitude search nagios3
p   nagios3                  - A host/service/network monitoring and managem
p   nagios3-cgi              - cgi files for nagios3
p   nagios3-common           - support files for nagios3
p   nagios3-core             - A host/service/network monitoring and managem
p   nagios3-dbg              - debugging symbols and debug stuff for nagios3
p   nagios3-doc              - documentation for nagios3
p   ndoutils-nagios3-mysql   - This provides the NDOUtils for Nagios with MyS
root@yoda:/home/jedi#
```



- Instalamos el paquete "nagios3".

```
root@yoda:/home/jedi# aptitude install nagios3
Se instalarán los siguiente paquetes NUEVOS:
 fancontrol{a} fping{a} libgd2-noxpm{a} libmysqlclient16{a} libnet-snmp-perl{a}
 libperl5.10{a} libpq5{a} libradiusclient-ng2{a} libsensors4{a} libsnmp-base{a}
 libsnmp15{a} lm-sensors{a} mysql-common{a} nagios-images{a} nagios-plugins{a}
 nagios-plugins-basic{a} nagios-plugins-standard{a} nagios3 nagios3-cgi{a}
 nagios3-common{a} nagios3-core{a} qstat{a} samba-common{a} samba-common-bin{a}
 smbclient{a} snmp{a}
0 paquetes actualizados, 26 nuevos instalados, 0 para eliminar y 0 sin actualizar.
Necesito descargar 32,7 MB de ficheros. Después de desempaquetar se usarán 90,4 MB.
¿Quiere continuar? [Y/n/?] █
```

- Durante la instalación nos preguntará la contraseña de administrador para Nagios, introducimos una.

Configuración de paquetes

I

**Configuración de nagios3-cgi**

Por favor, introduzca la contraseña para el nuevo usuario «nagiosadmin».

Estos son el usuario y contraseña que usará para ingresar a su instalación de nagios, después de que termine la configuración. Si no define una contraseña tendrá que configurar nagios de forma manual.

Clave de administración web de Nagios

\*\*\*\*\*

<Aceptar>

- Dejamos el nombre del grupo de trabajo por defecto.

Configuración de paquetes

**Samba Server**

Indique el grupo de trabajo de este equipo. Este parámetro controla el grupo de trabajo en el que aparecerá el equipo si se usa como un servidor, el grupo de trabajo a usar cuando explore la red con los distintos interfaces, y el nombre de dominio usado con el parámetro «security=domain».

Nombre del dominio o del grupo de trabajo:

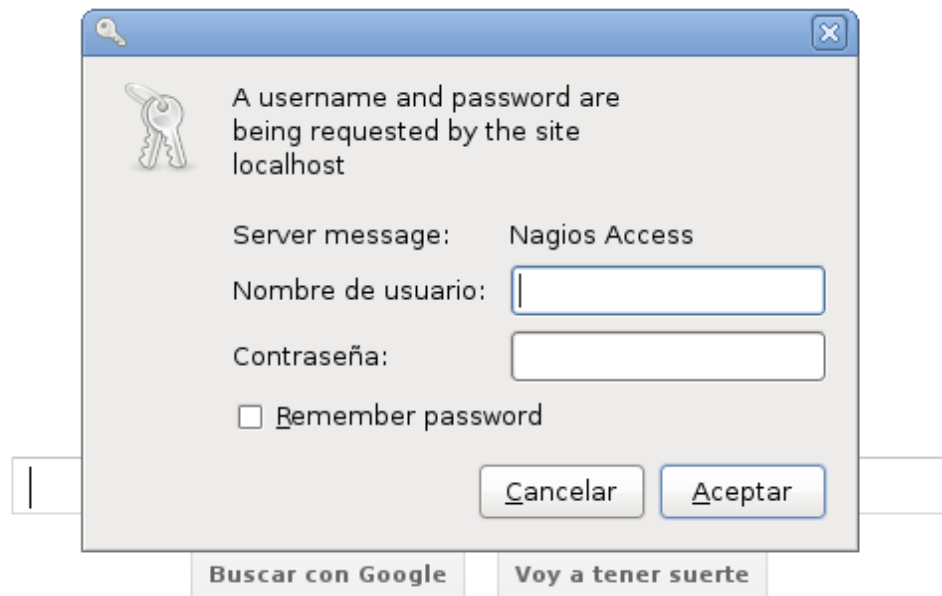
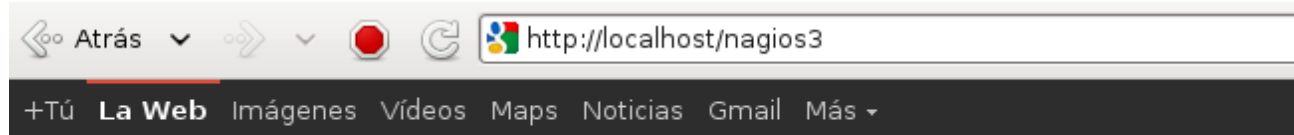
WORKGROUP

<Aceptar>

· Comprobamos que se ha instalado correctamente.

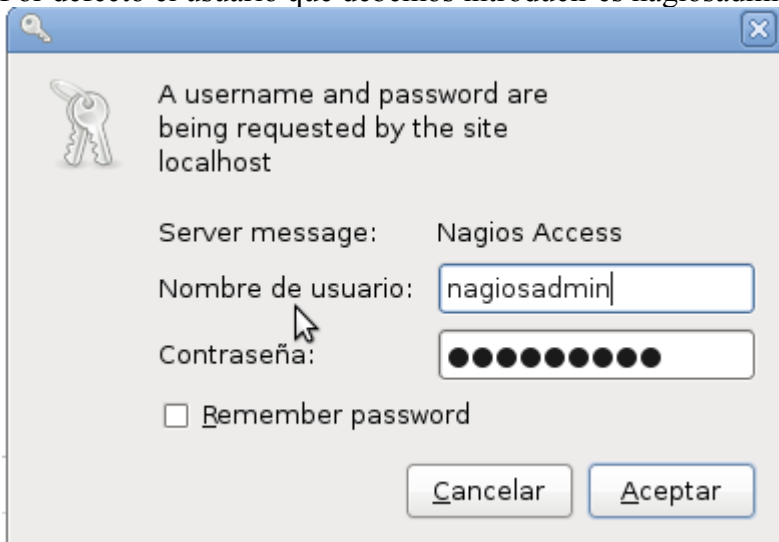
```
root@yoda:/home/jedi# dpkg -l nagios3
Deseario=Desconocido/Instalar/Eliminar/Purgar/Retener
| Estado=No/Instalado/Config-files/Desempaquetado/Medio-conf/Medio-inst/espera-disparo/p
endiente-disparo
|/ Err?=(ninguno)/Requiere-reinst (Estado,Err: mayúsc.=malo)
||/ Nombre Versión Descripción
+++-----
ii nagios3 3.2.1-2 A host/service/network monitoring and management
root@yoda:/home/jedi#
```

En un navegador web accedemos a <http://localhost/nagios3>

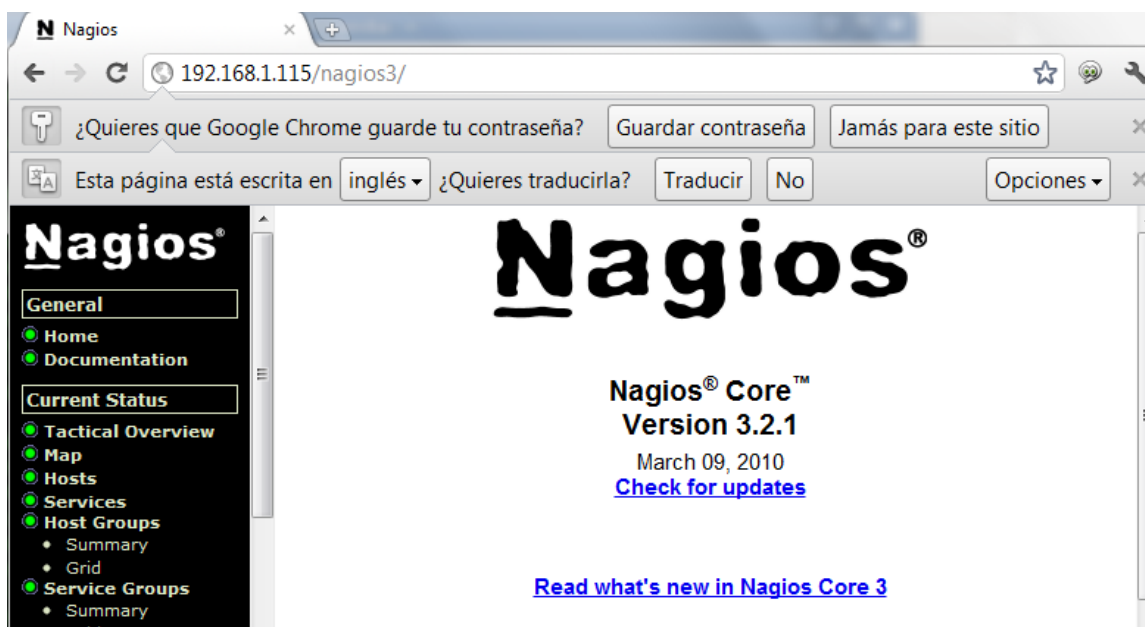
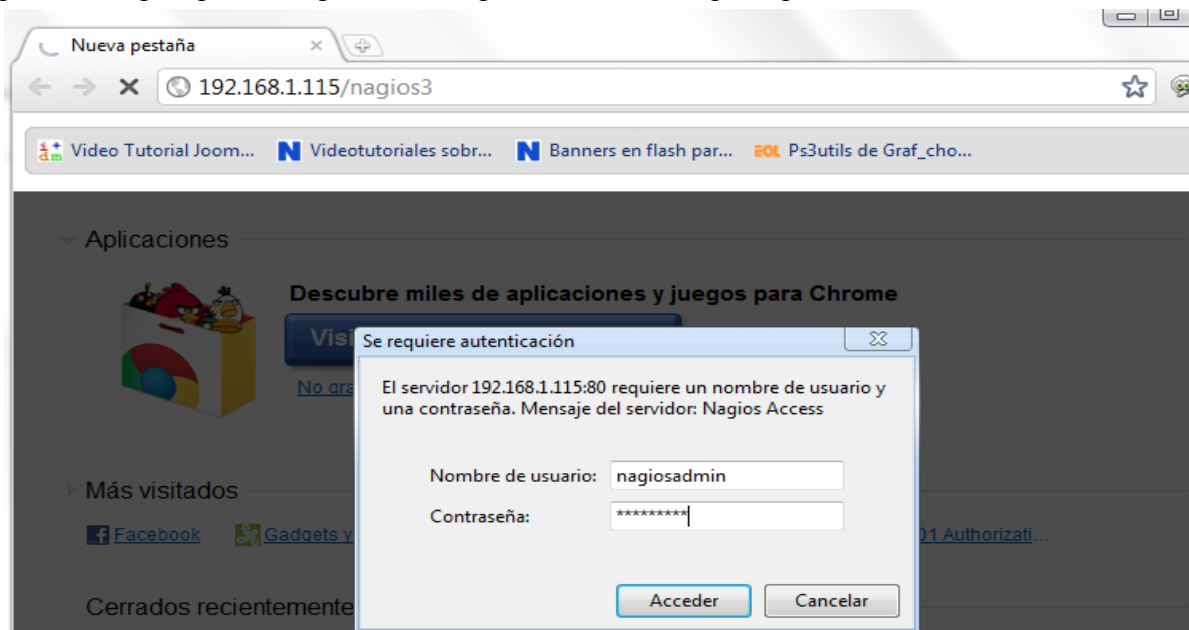


Introducimos la contraseña que previamente establecimos para logearnos como administradores de Nagios.

Por defecto el usuario que debemos introducir es nagiosadmin.



En un entorno real normalmente no trabajaríamos directamente sobre el servidor a partir de este punto siempre que fuese posible, así que desde otra máquina probamos el acceso.



Como comprobamos, podemos acceder perfectamente y trabajar de manera más cómoda.

Vamos a realizar otra comprobación.

```
root@yoda:/home/jedi# nagios3 -v /etc/nagios3/nagios.cfg

Nagios Core 3.2.1
Copyright (c) 2009-2010 Nagios Core Development Team and Community Contributors
Copyright (c) 1999-2009 Ethan Galstad
Last Modified: 03-09-2010
License: GPL

Website: http://www.nagios.org
Reading configuration data...
  Read main config file okay...
Processing object config file '/etc/nagios3/commands.cfg'...
Processing object config directory '/etc/nagios-plugins/config'...
Processing object config file '/etc/nagios-plugins/config/users.cfg'...
Processing object config file '/etc/nagios-plugins/config/telnet.cfg'...
Processing object config file '/etc/nagios-plugins/config/fping.cfg'...
Processing object config file '/etc/nagios-plugins/config/real.cfg'...
Processing object config file '/etc/nagios-plugins/config/ssh.cfg'...
Processing object config file '/etc/nagios-plugins/config/ftp.cfg'...
Processing object config file '/etc/nagios-plugins/config/pgsql.cfg'...
Processing object config file '/etc/nagios-plugins/config/flexlm.cfg'...
Processing object config file '/etc/nagios-plugins/config/apt.cfg'...
Processing object config file '/etc/nagios-plugins/config/netware.cfg'...
Processing object config file '/etc/nagios-plugins/config/ifstatus.cfg'...
Processing object config file '/etc/nagios-plugins/config/disk-smb.cfg'...
Processing object config file '/etc/nagios-plugins/config/http.cfg'...
Processing object config file '/etc/nagios-plugins/config/tcp_udp.cfg'...
Processing object config file '/etc/nagios-plugins/config/ldap.cfg'...
Processing object config file '/etc/nagios-plugins/config/ldap.cfg'...
Processing object config file '/etc/nagios-plugins/config/ntp.cfg'...
Processing object config file '/etc/nagios-plugins/config/ping.cfg'...
Processing object config file '/etc/nagios-plugins/config/nt.cfg'...
Processing object config file '/etc/nagios-plugins/config/mailq.cfg'...
Processing object config file '/etc/nagios-plugins/config/load.cfg'...
Processing object config file '/etc/nagios-plugins/config/hppjd.cfg'...
Processing object config file '/etc/nagios-plugins/config/mysql.cfg'...
Processing object config file '/etc/nagios-plugins/config/dns.cfg'...
Processing object config file '/etc/nagios-plugins/config/rpc-nfs.cfg'...
Processing object config file '/etc/nagios-plugins/config/mrtg.cfg'...
Processing object config file '/etc/nagios-plugins/config/radius.cfg'...
Processing object config file '/etc/nagios-plugins/config/snmp.cfg'...
Processing object config file '/etc/nagios-plugins/config/dhcp.cfg'...
Processing object config file '/etc/nagios-plugins/config/dummy.cfg'...
Processing object config file '/etc/nagios-plugins/config/procs.cfg'...
Processing object config file '/etc/nagios-plugins/config/breeze.cfg'...
Processing object config file '/etc/nagios-plugins/config/mail.cfg'...
Processing object config file '/etc/nagios-plugins/config/news.cfg'...
Processing object config file '/etc/nagios-plugins/config/disk.cfg'...
Processing object config file '/etc/nagios-plugins/config/games.cfg'...
Processing object config directory '/etc/nagios3/conf.d'...
Processing object config file '/etc/nagios3/conf.d/extinfo_nagios2.cfg'...
Processing object config file '/etc/nagios3/conf.d/localhost_nagios2.cfg'...
Processing object config file '/etc/nagios3/conf.d/timeperiods_nagios2.cfg'...
Processing object config file '/etc/nagios3/conf.d/services_nagios2.cfg'...
Processing object config file '/etc/nagios3/conf.d/generic-host_nagios2.cfg'...
Processing object config file '/etc/nagios3/conf.d/hostgroups_nagios2.cfg'...
```

```
Processing object config file '/etc/nagios3/conf.d/hostgroups_nagios2.cfg'...
Processing object config file '/etc/nagios3/conf.d/generic-service_nagios2.cfg'.
..
Processing object config file '/etc/nagios3/conf.d/contacts_nagios2.cfg'...
  Read object config files okay...

Running pre-flight check on configuration data...

Checking services...
  Checked 6 services.
Checking hosts...
  Checked 1 hosts.
Checking host groups...
  Checked 4 host groups.
Checking service groups...
  Checked 0 service groups.
Checking contacts...
  Checked 1 contacts.
Checking contact groups...
  Checked 1 contact groups.
Checking service escalations...
  Checked 0 service escalations.
Checking service dependencies...
  Checked 0 service dependencies.
Checking host escalations...
  Checked 0 host escalations.
Checking host dependencies...
  Checked 0 host dependencies.
Checking commands...
  Checked 153 commands.
Checking time periods...
  Checked 4 time periods.
Checking for circular paths between hosts...
Checking for circular host and service dependencies...
Checking global event handlers...
Checking obsessive compulsive processor commands...
Checking misc settings...

Total Warnings: 0
Total Errors:   0

Things look okay - No serious problems were detected during the pre-flight check
root@yoda:/home/jedi#
```

Como podemos comprobar no tenemos ninguna alerta ni ningún error.

## **Archivos principales de configuración**

Siguiendo el estandar, los archivos de configuración de Nagios están en /etc/nagios3/ , aquí podemos encontrar archivos como:

- apache2.conf: Configuración del servidor web apache2.
- cgi.cfg: Configuración de rutas y parámetros.
- commands.cfg: Comandos predefinidos y los personalizados por el usuario.
- nagios.cfg: Configuración principal de Nagio.

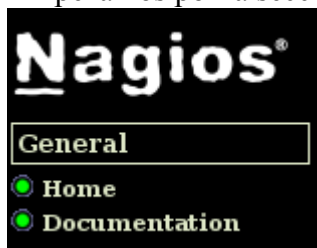
## **Directorios relevantes para Nagios**

- /etc/apache2/conf.d/ : Enlace simbólico al archivo de configuración del servidor Web.
- /etc/nagios3/ : Contiene archivos de configuración generales.
- /etc/nagios3/conf.d/ : Contiene archivos especificos para los dispositivos y servicios monitorizados.
- /etc/lib/nagios/plugin: Contiene los archivos binarios encargados de monitorizar los protocolos (http,ssh,etc...).
- /usr/share/doc/nagios\* : Contiene documentación y ejemplos sobre el uso de los plugins.
- /usr/share/nagios/htdocs/ :Contiene los archivos .html utilizados por el servidor web.
- /usr/share/nagios/htdocs/images/logos/ :Contiene imágenes para identificar los dispositivos en un mapa.
- /var/log/nagios3/ :Contiene el registro de las alertas detectadas por Nagios.

## **Opciones y Menus de Nagios**

Vamos a ver que nos ofrece cada apartado del menu que tiene la interfaz Web de Nagios.

- Empezamos por la sección "General", en concreto el apartado "Home".



Aquí aparece la versión actual que tenemos de Nagios. De cuando es la versión y el tipo de licencia que tiene.

# Nagios®

**Nagios® Core™**  
**Version 3.2.1**

March 09, 2010

[Check for updates](#)

[Read what's new in Nagios Core 3](#)

Copyright © 2009 Nagios Core Development Team and Community Contributors.  
Copyright © 1999-2009 Ethan Galstad.  
See the THANKS file for more information on contributors.

Nagios Core is licensed under the GNU General Public License and is provided AS IS with NO WARRANTY OF ANY KIND, INCLUDING THE WARRANTY OF DESIGN, MERCHANTABILITY, AND FITNESS FOR A PARTICULAR PURPOSE. Nagios, Nagios Core and the Nagios logo are trademarks, servicemarks, registered trademarks or registered servicemarks owned by Nagios Enterprises, LLC. Usage of the Nagios marks are governed by our [trademark policy](#).

**Nagios®**  
**Enterprises**

MONITORED BY  
**Nagios®**  
NAGIOS CORE

SOURCEFORGE.NET

Como vemos, esta versión tiene más de año y medio, y eso es por que es la versión de los repositorios Debian, aunque desde la Web de Nagios podríamos bajar la última versión. Pero como los miembros de Debian no aseguran su estabilidad nos quedamos con esta

- En el apartador "Current Status" nos encontramos con:
  - "Tactical Overview" : Aquí veremos de forma general el estado de la red monitorizada.

Network Outages				Network Health	
0 Outages				Host Health:	<div></div>
				Service Health:	<div></div>
Hosts					
0 Down		0 Unreachable		1 Up	
				0 Pending	
Services					
1 Critical		0 Warning		0 Unknown	
				5 Ok	
				0 Pending	
<b>1 Unhandled Problems</b>					
Monitoring Features					
Flap Detection		Notifications		Event Handlers	
Enabled		Enabled		Enabled	
All Services Enabled		All Services Enabled		All Services Enabled	
No Services Flapping		All Hosts Enabled		All Hosts Enabled	
All Hosts Enabled					
No Hosts Flapping					
Active Checks		Passive Checks			
Enabled		Enabled			
All Services Enabled		All Services Enabled			
All Hosts Enabled		All Hosts Enabled			

Como vemos tenemos una alerta crítica, la cual solucionaremos más tarde.

- En "Map" nos encontraremos con un esquema gráfico de los que es la estructura monitorizada. De la cual podríamos hacer filtro para ver lo que nos interesara.

**Network Map For All Hosts**  
 Last Updated: Sat Oct 29 11:55:16 CEST 2011  
 Updated every 90 seconds  
 Nagios® Core™ 3.2.1 - [www.nagios.org](http://www.nagios.org)  
 Logged in as nagiosadmin  
[View Status Detail For All Hosts](#)  
[View Status Overview For All Hosts](#)

Layout Method: Collapsed tree  
 Drawing Layers: All Servers  
 Debian GNU/Linux Servers  
 HTTP servers  
 SSH servers  
 Suppress popups: ☐  
 Scaling factor: 0.0  
 Layer mode: Include  
 Exclude  
 Update

- En "Host" vemos los equipos que tenemos monitorizados. De momento solo tenemos 1 (el localhost) cuando añadamos más equipos volveremos a esta sección.

**Current Network Status**  
 Last Updated: Sat Oct 29 11:57:44 CEST 2011  
 Updated every 90 seconds  
 Nagios® Core™ 3.2.1 - [www.nagios.org](http://www.nagios.org)  
 Logged in as nagiosadmin  
[View Service Status Detail For All Host Groups](#)  
[View Status Overview For All Host Groups](#)  
[View Status Summary For All Host Groups](#)  
[View Status Grid For All Host Groups](#)

**Host Status Totals**  

Up	Down	Unreachable	Pending
1	0	0	0

All Problems	All Types
0	1

**Service Status Totals**  

Ok	Warning	Unknown	Critical	Pending
5	0	0	1	0

All Problems	All Types
1	6

### Host Status Details For All Host Groups

Host ↑↓	Status ↑↓	Last Check ↑↓	Duration ↑↓	Status Information
<a href="#">localhost</a>	UP	2011-10-29 11:54:16	13d 9h 59m 27s	PING OK - Packet loss = 0%, RTA = 0.06 ms

1 Matching Host Entries Displayed



- En "Services" vemos el estado de los servicios que queremos monitorizar, por ahora solo vemos los que vienen predefinidos.

**Current Network Status**  
 Last Updated: Sat Oct 29 12:03:40 CEST 2011  
 Updated every 90 seconds  
 Nagios® Core™ 3.2.1 - [www.nagios.org](http://www.nagios.org)  
 Logged in as nagiosadmin

[View History For all hosts](#)  
[View Notifications For All Hosts](#)  
[View Host Status Detail For All Hosts](#)

**Host Status Totals**

Up	Down	Unreachable	Pending
1	0	0	0

All Problems	All Types
0	1

**Service Status Totals**

Ok	Warning	Unknown	Critical	Pending
5	0	0	1	0

All Problems	All Types
1	6

### Service Status Details For All Hosts

Host ↑↓	Service ↑↓	Status ↑↓	Last Check ↑↓	Duration ↑↓	Attempt ↑↓	Status Information
localhost	Current Load	OK	2011-10-29 11:59:16	13d 10h 5m 23s	1/4	OK - load average: 0.00, 0.00, 0.00
	Current Users	OK	2011-10-29 12:00:06	13d 10h 4m 33s	1/4	USERS OK - 1 users currently logged in
	Disk Space	OK	2011-10-29 12:00:56	13d 10h 3m 43s	1/4	DISK OK
	HTTP	OK	2011-10-29 12:01:46	13d 10h 2m 53s	1/4	HTTP OK: HTTP/1.1 200 OK - 453 bytes in 0.001 second response time
	SSH	CRITICAL	2011-10-29 12:02:36	13d 10h 2m 3s	4/4	Connection refused
	Total Processes	OK	2011-10-29 12:03:26	13d 10h 1m 13s	1/4	PROCS OK: 105 processes

6 Matching Service Entries Displayed

- En "Host groups" nos encontraremos con el estado de los dispositivos clasificados por grupos tales como servidores web, servidores ssh, etc...

**Current Network Status**  
 Last Updated: Sat Oct 29 12:10:40 CEST 2011  
 Updated every 90 seconds  
 Nagios® Core™ 3.2.1 - [www.nagios.org](http://www.nagios.org)  
 Logged in as nagiosadmin

[View Service Status Detail For All Host Groups](#)  
[View Host Status Detail For All Host Groups](#)  
[View Status Summary For All Host Groups](#)  
[View Status Grid For All Host Groups](#)

**Host Status Totals**

Up	Down	Unreachable	Pending
1	0	0	0

All Problems	All Types
0	1



**Service Status Totals**

Ok	Warning	Unknown	Critical	Pending
5	0	0	1	0



All Problems	All Types
1	6

### Service Overview For All Host Groups



[All Servers \(all\)](#)

Host	Status	Services	Actions
localhost	UP	1 OK 1 CRITICAL	 



[Debian GNU/Linux Servers \(debian-servers\)](#)

Host	Status	Services	Actions
localhost	UP	1 OK 1 CRITICAL	 

[HTTP servers \(http-servers\)](#)





Host	Status	Services	Actions
localhost	UP	1 OK 1 CRITICAL	 

[SSH servers \(ssh-servers\)](#)





Host	Status	Services	Actions
localhost	UP	1 OK 1 CRITICAL	 

## Status Grid For All Host Groups





[All Servers \(all\)](#)

Host	Services	Actions
<a href="#">localhost</a> 	<a href="#">Current Load</a> <a href="#">Current Users</a> <a href="#">Disk Space</a> <a href="#">HTTP</a> <a href="#">SSH</a> <a href="#">Total Processes</a>	  





[Debian GNU/Linux Servers \(debian-servers\)](#)

Host	Services	Actions
<a href="#">localhost</a> 	<a href="#">Current Load</a> <a href="#">Current Users</a> <a href="#">Disk Space</a> <a href="#">HTTP</a> <a href="#">SSH</a> <a href="#">Total Processes</a>	  

[HTTP servers \(http-servers\)](#)

Host	Services	Actions
<a href="#">localhost</a> 	<a href="#">Current Load</a> <a href="#">Current Users</a> <a href="#">Disk Space</a> <a href="#">HTTP</a> <a href="#">SSH</a> <a href="#">Total Processes</a>	  

[SSH servers \(ssh-servers\)](#)

Host	Services	Actions
<a href="#">localhost</a> 	<a href="#">Current Load</a> <a href="#">Current Users</a> <a href="#">Disk Space</a> <a href="#">HTTP</a> <a href="#">SSH</a> <a href="#">Total Processes</a>	  

- En "Service Groups" nos muestra el estado de los equipos agrupados en servicios. En principio no viene definido ningún grupo. Más adelante veremos como definirlos.

**Current Network Status**  
 Last Updated: Sun Oct 30 13:19:42 CET 2011  
 Updated every 90 seconds  
 Nagios® Core™ 3.2.1 - [www.nagios.org](http://www.nagios.org)  
 Logged in as nagiosadmin

[View Service Status Detail For All Service Groups](#)  
[View Status Summary For All Service Groups](#)  
[View Service Status Grid For All Service Groups](#)

**Host Status Totals**

Up	Down	Unreachable	Pending
1	0	0	0

All Problems	All Types
0	1

**Service Status Totals**

Ok	Warning	Unknown	Critical	Pending
5	0	0	1	0

All Problems	All Types
1	6

## Service Overview For All Service Groups

There are no service groups defined.

- En "Problems" veremos las distintas alertas agrupadas por "Hosts", "Services" o "Network".

**Current Network Status**  
 Last Updated: Sun Oct 30 13:21:31 CET 2011  
 Updated every 90 seconds  
 Nagios® Core™ 3.2.1 - [www.nagios.org](http://www.nagios.org)  
 Logged in as nagiosadmin

[View History For all hosts](#)  
[View Notifications For All Hosts](#)  
[View Host Status Detail For All Hosts](#)

**Host Status Totals**

Up	Down	Unreachable	Pending
1	0	0	0

All Problems	All Types
0	1








**Service Status Totals**

Ok	Warning	Unknown	Critical	Pending
5	0	0	1	0

All Problems	All Types
1	6

## Service Status Details For All Hosts

**Display Filters:**  
 Host Status Types: All  
 Host Properties: Any  
 Service Status Types: All Problems  
 Service Properties: Any

Host 	Service 	Status 	Last Check 	Duration 	Attempt 	Status Information
<a href="#">localhost</a> 	<a href="#">SSH</a>	CRITICAL	2011-10-30 13:20:56	14d 12h 19m 54s	4/4	Connection refused

1 Matching Service Entries Displayed

Por ahora veremos hasta aquí en cuanto al menú de Nagios, ya que es en lo que nos vamos a centrar.

## Solucionando la primera alerta

Lo primero que vamos a hacer es solucionar la alerta que nos muestra del localhost (donde está instalado Nagios), y centrarnos en las alertas que vamos a ir probocando en las máquinas clientes.

Supondremos que estamos en una rutina diaria de monitorización.

- Accedemos a "Tactical Overview" para ver si tenemos algún problema en cualquier dispositivo.



Services				
1 Critical	0 Warning	0 Unknown	5 Ok	0 Pending
1 Unhandled Problems				

- Como vemos tenemos un servicio en estado crítico. Pinchamos sobre "1 Unhandled Problems".

Display Filters:	
Host	Pending   Up
Status	
Types:	
Host	Any
Properties:	
Service	Critical
Status	
Types:	
Service	Not In Scheduled Downtime & Has Not Been
Properties:	Acknowledged & Active Checks Enabled

Service Status Details For All Hosts

Host ↑↓	Service ↑↓	Status ↑↓	Last Check ↑↓	Duration ↑↓	Attempt ↑↓	Status Information
localhost	SSH	CRITICAL	2011-10-30 13:25:56	14d 12h 28m 51s	4/4	Connection refused

Como podemos observar, nos muestra el equipo (en este caso es el propio servidor), el servicio afectado y el estado, y en "Status information" nos muestra el problema que tiene el servicio, que como podemos ver, es conexión rechazada.

- En este caso lo primero es comprobar que tenemos el paquete "ssh" instalado para poder recibir conexiones.

```
root@yoda:/home/jedi# dpkg -l ssh
Deseado=Desconocido/Instalar/Eliminar/Purgar/Retener
| Estado=No/Instalado/Config-files/Desempaquetado/Medio-conf/Medio-inst/espera-d
isparo/pendiente-disparo
|/ Err?=(ninguno)/Requiere-reinst (Estado,Err: mayúsc.=malo)
||/ Nombre          Versión          Descripción
+++-----
un  ssh              <ninguna>          (no hay ninguna descripción disponible)
root@yoda:/home/jedi#
```

- No está instalado, así que vamos a buscarlo e instalarlo (hacer previamente un update de los paquetes).

```
root@yoda:/home/jedi# dpkg -l ssh
Deseado=Desconocido/Instalar/Eliminar/Purgar/Retener
| Estado=No/Instalado/Config-files/Desempaquetado/Medio-conf/Medio-inst/espera-d
isparo/pendiente-disparo
|/ Err?=(ninguno)/Requiere-reinst (Estado,Err: mayúsc.=malo)
||/ Nombre          Versión          Descripción
+++-----
ii  ssh              1:5.5p1-6+sque secure shell client and server (metapackage)
root@yoda:/home/jedi#
```

Ya lo tenemos instalado, volvamos a Nagios para ver el estado de la alerta.

- Accedemos otra vez al apartado donde nos mostraba la alerta.

**Display Filters:**  
 Host Pending | Up  
 Status  
 Types:  
 Host Any  
 Properties:  
 Service Critical  
 Status  
 Types:  
 Service Not In Scheduled Downtime & Has Not Been  
 Properties: Acknowledged & Active Checks Enabled

#### Service Status Details For All Hosts

Host ↑↓	Service ↑↓	Status ↑↓	Last Check ↑↓	Duration ↑↓	Attempt ↑↓	Status Information
localhost	SSH	CRITICAL	2011-10-30 13:25:56	14d 12h 28m 51s	4/4	Connection refused

- Como vemos aun aparece la alerta, pero en unos instante Nagios vuelve hacer el chequeo y ya desaparece.

Host ↑↓	Service ↑↓	Status ↑↓	Last Check ↑↓	Duration ↑↓	Attempt ↑↓	Status Information
localhost	Current Load	OK	2011-10-30 13:42:36	14d 12h 45m 53s	1/4	OK - load average: 0.00, 0.02, 0.03
	Current Users	OK	2011-10-30 13:43:26	14d 12h 45m 3s	1/4	USERS OK - 2 users currently logged in
	Disk Space	OK	2011-10-30 13:39:16	14d 12h 44m 13s	1/4	DISK OK
	HTTP	OK	2011-10-30 13:40:06	14d 12h 43m 23s	1/4	HTTP OK: HTTP/1.1 200 OK - 453 bytes in 0.001 second response time
	SSH	OK	2011-10-30 13:40:56	0d 0h 3m 14s	1/4	SSH OK - OpenSSH_5.5p1 Debian-6+squeeze1 (protocol 2.0)
	Total Processes	OK	2011-10-30 13:41:46	14d 12h 41m 43s	1/4	PROCS OK: 111 processes

6 Matching Service Entries Displayed

- Vamos a habilitar una opción para que no tengamos que esperar. Editamos con nano el archivo /etc/nagios3/nagios.cfg. Buscamos la línea "check\_external\_commands".

```
# EXTERNAL COMMAND OPTION
# This option allows you to specify whether or not Nagios should check
# for external commands (in the command file defined below). By default
# Nagios will *not* check for external commands, just to be on the
# cautious side. If you want to be able to use the CGI command interface
# you will have to enable this.
# Values: 0 = disable commands, 1 = enable commands

check_external_commands=1
```

Tenemos que cambiar el valor 0 por 1.

Una vez solucionado la alerta en el servidor de Nagios vamos a empezar a añadir equipos.

## Añadiendo elementos a monitorear

Lo primero que necesitamos en las máquinas a monitorear es un cliente SNMP que nos facilite la información sobre el sistema.

En sistemas como Debian o Ubuntu nos basta con:

```
#aptitude install snmp
```

Comprobamos que se a instalado correctamente:

```
root@yoda:/home/jedi# dpkg -l snmp
Deseado=Desconocido/Instalar/Eliminar/Purgar/Retener
| Estado=No/Instalado/Config-files/Desempaquetado/Medio-conf/Medio-inst/espera-d
isparo/pendiente-di-paro
|/ Err?=(ninguno)/Requiere-reinst (Estado,Err: mayúsc.=malo)
||/ Nombre Versión Descripción
+++-+-----+-----+-----+
ii snmp 5.4.3~dfsg-2 SNMP (Simple Network Management Protocol) ap
root@yoda:/home/jedi#
```

En equipos Windows necesitamos instalar "NSClient ++" que es el cliente de Nagios para Windows.

Para equipos Mac OS X necesitamos "net-snmp", en este sistema necesitamos compilar el "net-snmp" por lo que tendremos que instalar "xCode".

Vamos a ir añadiendo equipos que serán los clientes a monitorear.

- Editamos el archivo /etc/nagios3/nagios.cfg y buscamos la linea "cfg\_file"

```
# Commands definitions
cfg_file=/etc/nagios3/commands.cfg

#Dispositivos monitorizados
cfg_file=/etc/nagios3/dispositivos.cfg
```

Tenemos que añadir las dos líneas últimas que os muestro, siendo la primera de estas un comentario y la otra la ruta al archivo de configuración.

- Creamos el archivo "dispositivos.cfg".

```
root@yoda:/etc/nagios3# touch dispositivos.cfg
root@yoda:/etc/nagios3#
```

- Por defecto, cuando instalamos Nagios, el archivo /var/lib/nagios3/rw/nagios.cmd el cual nos permite realizar chequeos desde el navegador, viene con propietario nagios:nagios, debemos darle permisos al usuario de apache.

```
root@yoda:/home/jedi# dpkg-statoverride --update --add nagios www-data 2710 /var
/lib/nagios3/rw
root@yoda:/home/jedi# dpkg-statoverride --update --add nagios nagios 751 /var/li
b/nagios3
root@yoda:/home/jedi#
```

- Reiniciamos el servicio de Nagios para aplicar los cambios.

```
root@yoda:/etc/nagios3# /etc/init.d/nagios3 restart
Restarting nagios3 monitoring daemon: nagios3
.
root@yoda:/etc/nagios3#
```

- Editamos el archivo `/etc/nagios3/dispositivos.cfg` y vamos a introducir los equipos. Esta va a ser la estructura principal que voy a tener para equipos que sean linux en el archivo (esto es a gusto personal).

```

GNU nano 2.2.4                                Fichero: dispositivos.cfg

##### Dispositivos a monitorear #####
#####Administrador Alberto A. Mariscal Casado#####
#####Proyecto Integrado#####

#### Máquinas Linux ####

```

Como vemos solo he creado la sección para máquinas Linux ya que las máquinas windows irán en otro archivo con su plantilla correspondiente.

## Monitorizando equipos Windows

- Como vamos a configurar Nagios para monitorizar una máquina Windows necesitamos hacer algunos ajustes:

- Editamos el fichero `/etc/nagios3/nagios.cfg`.
- Descomentamos la línea siguiente:

```

# Definitions for monitoring a Windows machine
cfg_file=/etc/nagios3/objects/windows.cfg

```

- Creamos una nueva entrada en el archivo de la captura anterior para la máquina Windows.

```

GNU nano 2.2.4                                Fichero: windows.cfg

# Aqui definiremos las máquinas Windows

define host{
use      windows-servers ; Nombre de la plantilla que vamos a usar
host_name      Anakin    ; Nombre que le daremos al HOST
alias      DarkVader     ; Nombre asociado al host
address 192.168.0.128    ; Direccion IP de la maquina windows
}

```

- Creamos otro archivo en el mismo directorio, llamado "templates.cfg".

```

GNU nano 2.2.4                                Fichero: templates.cfg                                Modificado

#### Dispositivos Windows ####
define host{
name      windows-servers ; Nombre de la plantilla
use      generic-host     ; Un atributo general
check_period      24x7      ; Periodos de chequeos
check_interval    5         ; Tiempo entre chequeo y chequeo
retry_interval    1         ; tiempo para reintentar un chequeo
max_check_attempts 10       ; optativo pero recomendable
check_command     check-host-alive ; Comandos generales de chequeo si el sistema esta "Arriba"
notification_period      24x7 ;
notification_interval    30  ;
notification_options     d,r ;
contact_groups           admins ; Grupo de usuarios que reciben las notificaciones
#hostgroups              windows-servers ; Como no hay un grupo definido aun, me daria error.
}

```

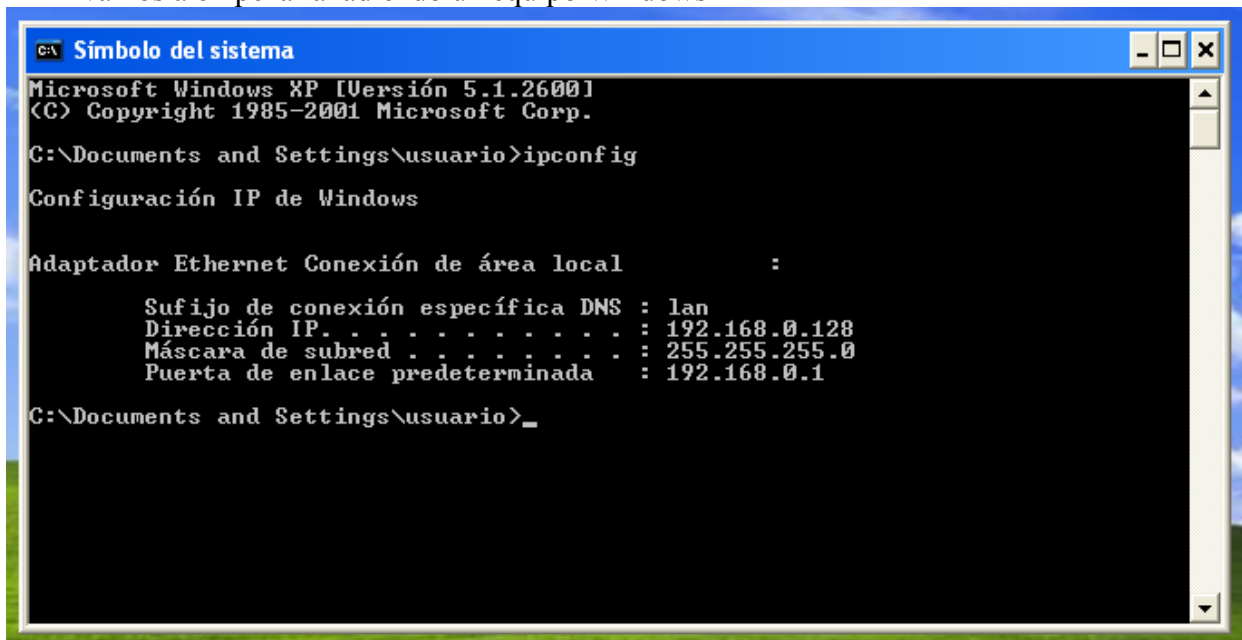
- Definimos que servicios se van a monitorear de la máquina Windows.

```
# Obtener versión del cliente
define service{
  use generic-service
  host_name Anakin
  service_description Version NSClient++
  check_command check_nt!CLIENTVERSION
}

#Tiempo de actividad del equipo
define service{
  use generic-service
  host_name Anakin
  service_description Tiempo Activo
  check_command check_nt!UPTIME
}
```

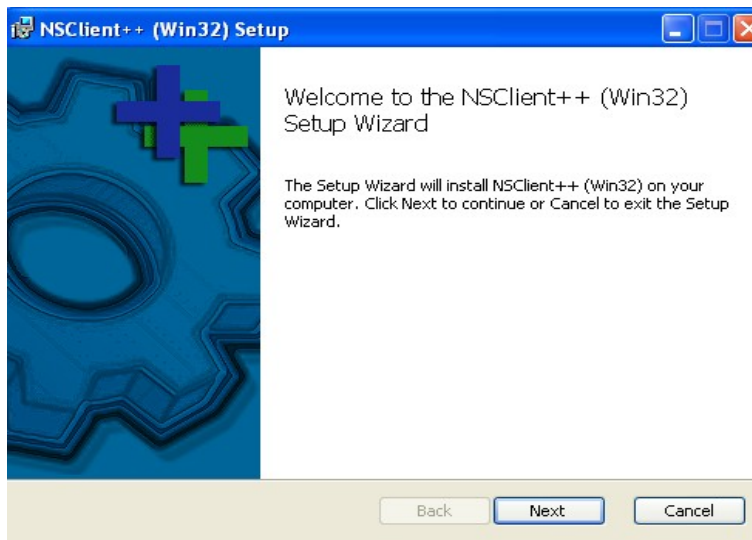
Como vemos usamos el comando "check\_nt" el cual es especial para las máquinas Windows.

- Vamos a empezar añadiendo un equipo Windows XP

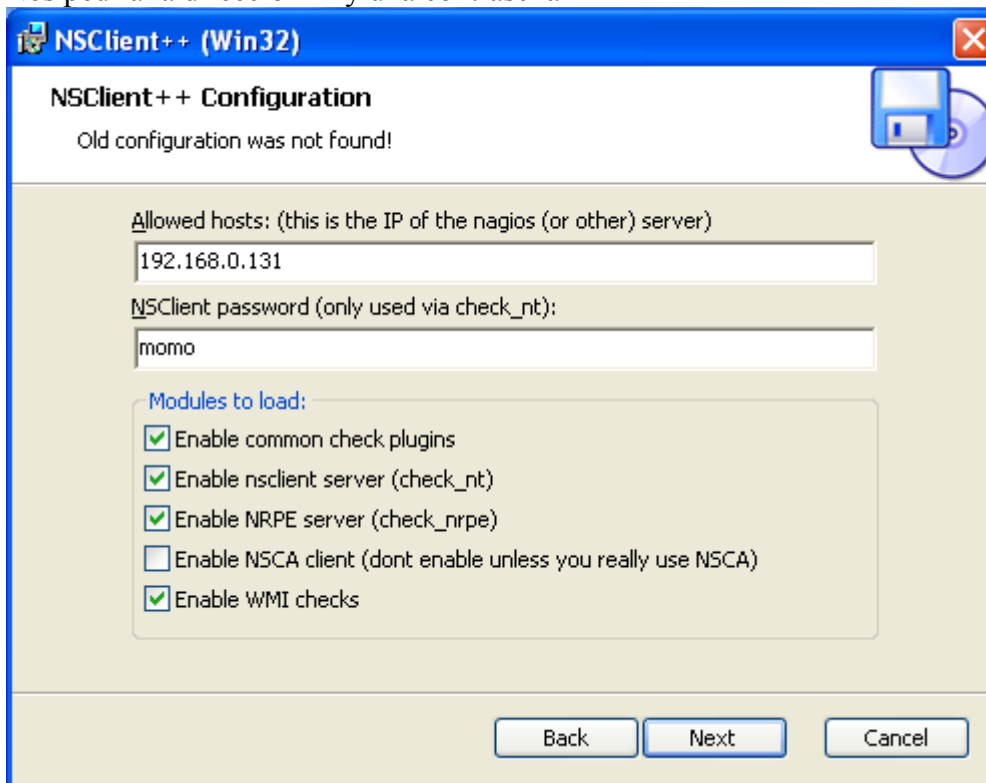


Estos son los parámetros básicos que se necesitan para empezar a monitorizar. Como es una máquina Windows necesitamos hacer varias cosas:

- Instalar un agente en la máquina Windows cliente: NSClient++.

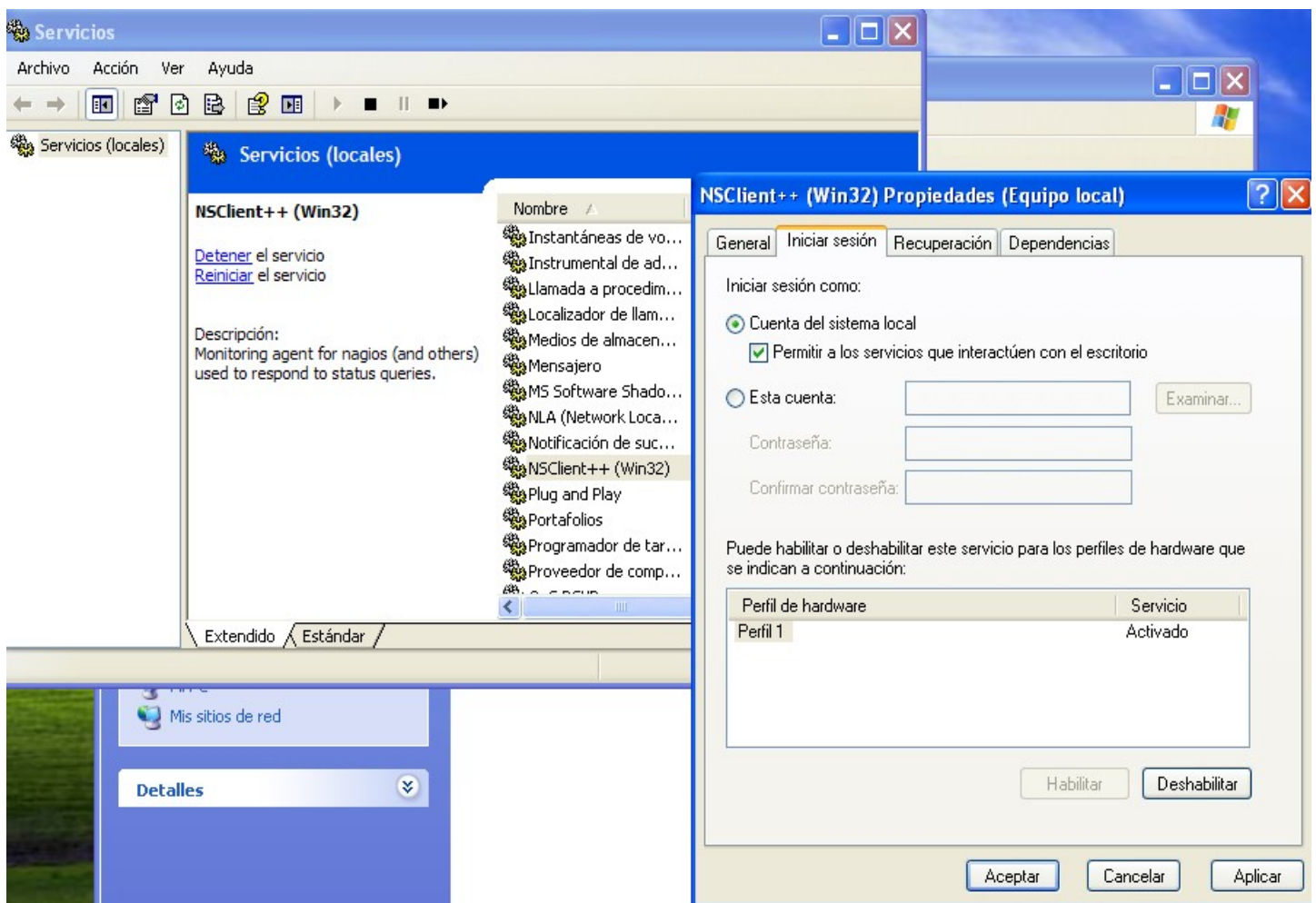


- Nos pedirá la dirección IP y una contraseña

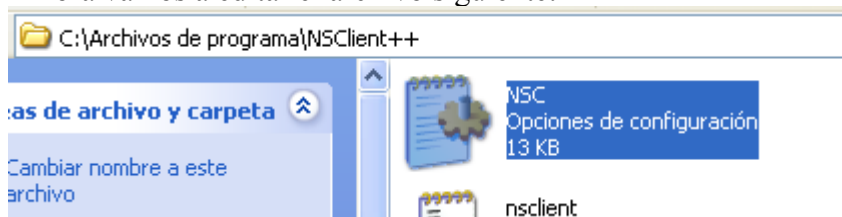




- Después iremos al panel de control, herramientas administrativas, servicios y buscaremos el de NSClient++ y habilitaremos el "Permitir a los servicios que interactúen con el escritorio".



- Ahora vamos a editar el archivo siguiente:



- Descomentamos las siguientes líneas tal como aparece en las imágenes.
  - En esta primera permitiremos que se cargen ciertos módulos.

```
FileLogger.dll
CheckSystem.dll
CheckDisk.dll
NSClientListener.dll
NRPEListener.dll
SysTray.dll
CheckEventLog.dll
CheckHelpers.dll
; CheckWMI.dll
CheckNSCP.dll
;
; script to check external scripts
; CheckExternalScripts.dll
;
```

- Aquí le indicamos la IP del servidor Nagios:

```

;# ALLOWED HOST ADDRESSES
; This is a comma-delimited list of IP address
; If leave this blank anyone can access the dea
; The syntax is host or ip/mask so 192.168.0.0/
allowed_hosts=192.168.0.131/32

```

- Le decimos por que puerto debe de escuchar.

```

[NSClient]
;# ALLOWED HOST ADDRESSES
; This is a comma-delimited list of IP address of hosts that are allowed to
; If you leave this blank the global version will be used instead.
;allowed_hosts=
;
;# NSCLIENT PORT NUMBER
; This is the port the NSClientListener.dll will listen to.
port=12489

```

- Comprobamos desde una consola de linux que podemos obtener informacion:

```

root@yoda:/etc/nagios-plugins/config# /usr/lib/nagios/plugins/check_nt -H 192.16
8.1.49 -p 12489 -v USEDDISKSPACE -d SHOWALL -l c -s 48956668h
c:\ - total: 7,49 Gb - used: 2,74 Gb (37%) - free 4,75 Gb (63%) | 'c:\ Used Spac
e'=2,74Gb;0,00;0,00;0,00;7,49
root@yoda:/etc/nagios-plugins/config#

```

Donde:

- -H 192.168.1.49 es la máquina Windows.
- -p 12489 es el puerto configurado en el cliente Windows.
- -s 48956668h es la password del cliente Windows.
- Debemos hacer unos pequeños cambios en la definición del comando check\_nt y lo dejaremos como mostramos a continuación:

```

GNU nano 2.2.4          Fichero: /etc/nagios-plugins/config/nt.cfg          Modificado
# If you are confused about this command definition, cause you was
# reading other suggestions, please have a look into
# /usr/share/doc/nagios-plugins/README.Debian

# 'check_nt' command definition
define command {
    command_name     check_nt
    command_line      /usr/lib/nagios/plugins/check_nt -H '$HOSTADDRESS$' -p 12489 -s 48956668h -v '$ARG1$' '$ARG2$'
}

```

- Reiniciamos el servicio de Nagios en el servidor y comprobamos que la ha detectado.

**Host Status: Anakin**

<b>Name:</b>	Anakin
<b>Alias:</b>	DarkVader
<b>Address:</b>	192.168.0.128
<b>State:</b>	Up
<b>Status Information:</b>	PING OK - Packet loss = 0%, RTA = 0.51 ms
<b>State Duration:</b>	0d 0h 0m 44s
<b>Last Status Check:</b>	2011-10-30 18:08:51
<b>Last State Change:</b>	2011-10-30 18:08:57
<b>Parent Host(s):</b>	None (This is a root host)
<b>Immediate Child Hosts:</b>	0

**Services:**

**Layout Method:** Circular (Marked Up)

**Drawing Layers:** All Servers, Debian GNU/Linux Servers, HTTP servers, SSH servers

**Layer mode:** ☐ Include, ☒ Exclude

**Suppress popups:** ☐

**Scaling factor:** 0.0

**Update**

- Vamos añadir servicios a monitorear en el equipo Windows. Editamos de nuevo el archivo "/etc/nagios3/objects/windows.cfg".
  - Servicio de "Ping":

```
### Servicios para las máquinas Windows ###
# Alerta de Ping
define service{
  use generic-service      ; Es un atributo general
  host_name Anakin         ; Equipos que tendran dicha alerta de servicio
  service_description PING ; Breve descripcion de la alerta
  check_command check_ping!200.0,20%!600.0,60% ; Comando de la alerta
  normal_check_interval 5
  retry_check_interval 1
}
```

- Versión del cliente instalado en la máquina Windows.

```
# Obtener versión del cliente
define service{
  use generic-service
  host_name Anakin
  service_description Version NSClient++
  check_command check_nt!CLIENTVERSION
}
```

- Tiempo que lleva la máquina operativa desde el último reinicio/encendido.

```
#Tiempo de actividad del equipo
define service{
  use generic-service
  host_name Anakin
  service_description Tiempo Activo
  check_command check_nt!UPTIME
}
```

- Uso de la memoria RAM.

```
# Uso de la memoria fisica (RAM)
define service{
  use generic-service
  host_name Anakin
  service_description Uso Memoria RAM
  check_command check_nt!MEMUSE!-w 80 -c 90
}
```


- Reiniciamos el servicio de Nagios3 y comprobamos que nos aparecen los servicios en la máquina Windows.

<a href="#">Anakin</a>	<a href="#">Espacio en C</a>	UNKNOWN	2011-11-12 18:30:27	0d 1h 12m 15s	4/4	wrong -l argument
	<a href="#">Explorer.exe</a>	UNKNOWN	2011-11-12 18:31:36	0d 1h 12m 6s	4/4	No service/process specified
	<a href="#">PING</a>	OK	2011-11-12 18:26:57	0d 0h 4m 44s	1/4	PING OK - Packet loss = 0%, RTA = 0.51 ms
	<a href="#">Tiempo Activo</a>	OK	2011-11-12 18:31:17	0d 0h 0m 24s	1/4	System Uptime - 0 day(s) 0 hour(s) 52 minute(s)
	<a href="#">Uso CPU</a>	OK	2011-11-12 18:30:05	0d 0h 1m 36s	1/4	CPU Load 0% ( 5 min average)
	<a href="#">Uso Memoria RAM</a>	OK	2011-11-12 18:31:29	0d 0h 0m 16s	1/4	Memory usage: total:1246,75 Mb - used: 118,99 Mb (10%) - free: 1127,75 Mb (90%)
	<a href="#">Version NSClient++</a>	OK	2011-11-12 18:31:36	0d 0h 0m 5s	1/4	NSClient++ 0.3.9.327 2011-08-16

Vemos que al lado de cada servicio aparece un "bocadillo" del estilo de los "comics", eso indican que tienen un comentario, el cual a podido ser puesto por un administrador o por el propio servicio de "Nagios".

### Service Comments

 [Add a new comment](#)  [Delete all comments](#)

Entry Time	Author	Comment	Comment ID	Persistent	Type	Expires	Actions
2011-11-12 18:26:01	(Nagios Process)	Notifications for this service are being suppressed because it was detected as having been flapping between different states (24.3% change >= 20.0% threshold). When the service state stabilizes and the flapping stops, notifications will be re-enabled.	32	No	Flap Detection	N/A	

## Monitorizando equipos GNU/Linux Debian/Ubuntu

En esta ocasión vamos a monitorizar un equipo Ubuntu.

- Necesitamos saber la IP de dicha máquina (o el nombre), aunque lo ideal es tener un dns dinámico en la red.

```
root@obiwan:/home/obi-wan# ifconfig
eth0      Link encap:Ethernet  direcciónHW 08:00:27:81:f0:ea
          Direc. inet:192.168.1.16 Difus.:192.168.1.255 Másc:255.255.255.0
          Dirección inet6: fe80::a00:27ff:fe81:f0ea/64 Alcance:Enlace
          ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST MTU:1500 Métrica:1
          Paquetes RX:374 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:63 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:1000
          Bytes RX:35388 (35.3 KB)  TX bytes:14382 (14.3 KB)

lo        Link encap:Bucle local
          Direc. inet:127.0.0.1  Másc:255.0.0.0
          Dirección inet6: ::1/128 Alcance:Anfitrión
          ACTIVO BUCLE FUNCIONANDO MTU:16436 Métrica:1
          Paquetes RX:124 errores:0 perdidos:0 overruns:0 frame:0
          Paquetes TX:124 errores:0 perdidos:0 overruns:0 carrier:0
          colisiones:0 long.colaTX:0
          Bytes RX:9600 (9.6 KB)  TX bytes:9600 (9.6 KB)

root@obiwan:/home/obi-wan#
```

- Añadimos la máquina al servidor. Editamos el archivo /etc/nagios3/dispositivos.cfg

```
##### Dispositivos a monitorear #####
#####Administrador Alberto A. Mariscal Casado#####
#####Proyecto Integrado#####
#### Máquinas Linux ####
define host{
use generic-host
host_name Obi-Wan
alias obiwan
address 192.168.1.16 ;cuidado con el cambio de IP en la LAN.
check_command check-host-alive
max_check_attempts 20
notification_interval 60
notification_period 24x7
notification_options d,u,r
}
```

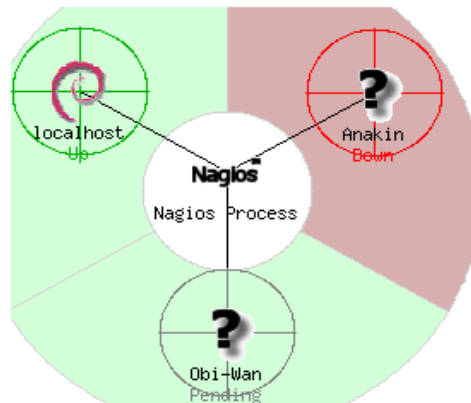
- Creamos un grupo y añadimos esta máquina. Se introducen los datos en el mismo archivo que hemos introducido la máquina.

```
### Grupos ###
define hostgroup{
hostgroup_name equipos_linux
alias pclinux
members Obi-Wan
}
```

- Reiniciamos el servicio. `#/etc/init.d/nagios3 restart`.

```
root@yoda:/etc/nagios3# /etc/init.d/nagios3 restart
Restarting nagios3 monitoring daemon: nagios3
.
root@yoda:/etc/nagios3#
```

- Accedemos al administrador web de nagios y comprobamos que se ha añadido el nuevo equipo.



- Al principio nos aparecerá como host caído, vamos a solucionarlo.

Host ↑↓	Status ↑↓	Last Check ↑↓	Duration ↑↓	Status Information
Anakin	DOWN	2011-11-06 20:23:12	0d 0h 24m 47s	PING CRITICAL - Packet loss = 100%
Obi-Wan	DOWN	2011-11-06 20:24:44	0d 0h 0m 46s	CRITICAL - Host Unreachable (192.168.1.16)
localhost	UP	2011-11-06 20:23:34	21d 19h 27m 21s	PING OK - Packet loss = 0%, RTA = 0.04 ms

- Volvemos a editar el archivo anterior (`/etc/nagios3/dispositivos.cfg`). Añadimos el servicio PING.

```
###Servicios a monitorizar ###
#Servicio a PING
define service{
  use generic-service
  host_name Obi-Wan
  service_description PING
  check_command check_ping!200.0,20%!600.0,60%
  normal_check_interval 5
  retry_check_interval 1
}
```

- Vamos a ver otro servicio importante a tener en cuenta en una red LAN y más aun si tenemos salida a Internet. El servicio SSH

```
# Monitorizar Acceso SSH
define service{
  use generic-service
  host_name Obi-Wan
  service_description Estado SSH
  check_command check_ssh
}
```



- Comprobamos que el estado es "OK", siempre y cuando la máquina tenga instalado el paquete "ssh". También vemos que tiene un tipo de servicio "Estado HTTP" y está "OK" por que he instalado "apache2" en la máquina cliente.

<a href="#">Obi-Wan</a>	<a href="#">Disco Duro</a>	CRITICAL	2011-11-06 23:06:58	0d 0h 34m 43s	4/4	DISK CRITICAL - free space: / 5365 MB (73% inode=85%): /lib/init/rw 505 MB (100% inode=99%): /dev 500 MB (99% inode=99%): /dev/shm 505 MB (100% inode=99%):
	<a href="#">Estado HTTP</a>	OK	2011-11-06 23:07:52	0d 0h 0m 53s	1/4	HTTP OK: HTTP/1.1 200 OK - 453 bytes in 0,005 second response time
	<a href="#">Estado IF</a>	CRITICAL	2011-11-06 23:05:58	0d 0h 12m 44s	4/4	CRITICAL: No response from remote host '192.168.1.20' for 13.6.1.2.1.2.2.1.8 with snmp version 1
	<a href="#">Estado SSH</a>	OK	2011-11-06 23:04:46	0d 0h 23m 55s	1/4	SSH OK - OpenSSH_5.8p1 Debian-1ubuntu3 (protocol 2.0)
	<a href="#">PING</a>	OK	2011-11-06 23:04:43	0d 1h 47m 29s	1/4	PING OK - Packet loss = 0%, RTA = 1.78 ms

## Monitorizando Routers

Igual que hemos hecho con los equipos de trabajo y servidores vamos a configurar Nagios para monitorizar un router, el cual nos da acceso a Internet.

Vamos a configurar el router.

- Editamos el archivo /etc/nagios3/nagios.cfg

```
# Definitions for monitoring a router/switch
cfg_file=/etc/nagios3/objects/switch.cfg
```

Descomentando la línea donde indica el archivo de configuración para los switches.

- Editamos el archivo /etc/nagios3/objects/templates.cfg para crear la plantilla correspondiente.

```
### Routers ###
define host{
name generic-switch ; Nombre de la plantilla
use generic-host ;
check_period 24x7 ; Tiempo de monitoreo de los Switchs
check_interval 5 ; Son chequeados cada 5 minutos.
retry_interval 1 ;
max_check_attempts 10 ;
check_command check-host-alive ; Comando por defecto para ver si un router esta arriba (levantado)
notification_period 24x7 ; Las notificaciones se envian a cualquier hora
notification_interval 30 ; las notificaciones se reenvian cada 30 minutos
notification_options d,r ;
contact_groups admins ; por defecto las notificaciones se envian a los usuarios Admins
register 0 ; Esto caracteriza que es una plantilla
}
```

- Creamos el archivo /etc/nagios/objects/switch.cfg

```
### Routers/Switches ###
define host{
use generic-switch ; el tipo de plantilla general para switch/routers
host_name cisco-epc3825 ; Cisco ONO 50MB
alias cisco50MB ; Un alias descriptivo
address 192.168.1.1 ; IP del router, normalmente la puerta de enlace
hostgroups switches ; Esto indica en que grupo lo clasificamos
}
```

- En el mismo archivo, debajo de la definición de los host, creamos un apartado para definir los servicios. En esta ocasión creamos el servicio "PING", el cual usaremos para comprobar que el dispositivo esta "Arriba".

```
### Servicios a monitorear ###
define service{
  use generic-service ; plantilla general de servicios.
  host_name cisco-epc3825 ; quien usara este servicio.
  service_description PING ; Descripcion de la función del servicio.
  check_command check_ping!200.0,20%!600.0,60% ; Comando del servicio
  normal_check_interval 2 ; El chequeo del servicio se realiza cada 2 minutos en condiciones normales.
  retry_check_interval 1 ; Vuelve a chequear el servicio si no presenta condiciones normales.
}
```

- Reiniciamos el servicio de nagios (/etc/init.d/nagios3 restart) y comprobamos que ya nos aparece en el sistema web de monitorización.

```
root@yoda:/etc/nagios3/objects# /etc/init.d/nagios3 restart
Restarting nagios3 monitoring daemon: nagios3
.
root@yoda:/etc/nagios3/objects#
```

- Comprobamos que el dispositivo aparece en Nagios y esta levantado.

Host ↑↓	Status ↑↓	Last Check ↑↓	Duration ↑↓	Status Information
Anakin	DOWN	2011-11-08 21:58:35	2d 2h 0m 9s	PING CRITICAL - Packet loss = 100%
Obi-Wan	UP	2011-11-08 21:59:50	2d 0h 42m 22s	PING OK - Packet loss = 0%, RTA = 0.38 ms
cisco-epc3825	UP	2011-11-08 22:00:44	0d 0h 1m 20s+	PING OK - Packet loss = 0%, RTA = 136.77 ms
localhost	UP	2011-11-08 21:13:03	23d 21h 2m 43s	PING OK - Packet loss = 0%, RTA = 0.05 ms

Llegados a este punto en el que tenemos clientes y el dispositivo de red configurados vamos a organizarlos en el mapa.



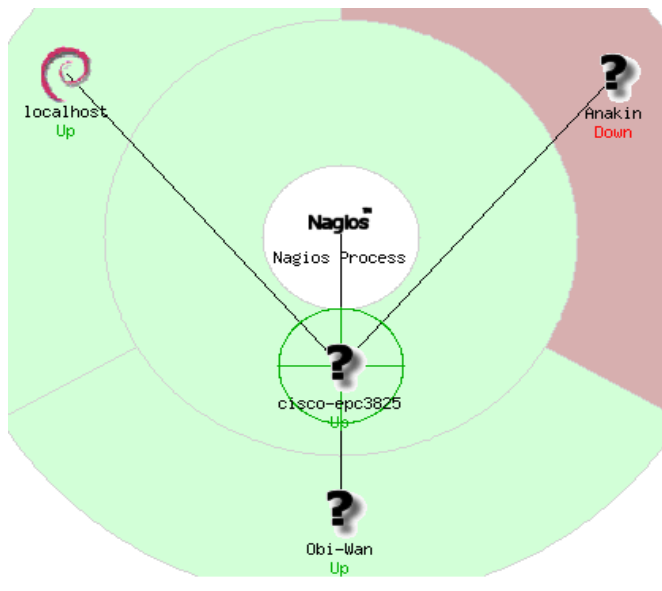


Como vemos los equipos cuelgan todos del "Nagios Process" pero para representarlo como en realidad estan deberían colgar de "cisco-epc3825" que es la puerta de enlace y el punto de interconexión de los equipos.

- Como ejemplo vamos a editar la configuración para el equipo "Obi-Wan". Editamos el archivo "/etc/nagios3/dispositivos.cfg" y añadimos la etiqueta "parents" seguido del dispositivo.

```
#### Máquinas Linux ####
define host{
use generic-host
host_name Obi-Wan
alias obiwan
address 192.168.1.20 ;cuidado con el cambio de IP en la LAN.
check_command check-host-alive
max_check_attempts 20
notification_interval 60
notification_period 24x7
notification_options d,u,r
parents cisco-epc3825
}
```

- Reiniciamos el servicio de Nagios y comprobamos que se ha actualizado.



Podemos monitorizar muchos servicios de los routers que soportan SNMP, en mi caso no es soportado, pero algunos de ellos serían:

- UpTime del dispositivo:

```
define service{
use generic-service ;
host_name cisco-epc3825
service_description Uptime
check_command check_snmp!-C public -o sysUpTime.0
}
```

- Ancho de banda de un puerto específico:  

```
define service{  
  use generic-service ; Inherit values from a template  
  host_name linksys-srw224p  
  service_description Port 1 Bandwidth Usage  
  check_command check_local_mrtgtraf!/var/lib/mrtg/192.168.1.253_1.log!AVG!  
  1000000,2000000!5000000,5000000!10  
}
```

En el ejemplo anterior, la opción `"/var/lib/mrtg/192.168.1.1_1.log"` que es pasada al comando `check_local_mrtgtraf` le dice al plugin cual archivo de registros de MRTG se va a leer. La opción `"AVG"` le dice que deberá utilizar estadísticas de promedio de ancho de banda. Las opciones `"1000000,2000000"` son los rangos de precaución (en bytes) para las tasas de tráfico entrante. Las opciones `"5000000,5000000"` son rangos críticos (en bytes) para tráfico saliente. La opción `"10"` causa que el plugin regrese un estado CRÍTICO (CRITICAL) si el archivo de registros no se ha actualizado en 10 minutos (se debe actualizar cada 5 minutos).

## **Monitorizando Mac OS X**

De entre las principales plataformas de servidores/estaciones de trabajo que se usan en la actualidad, nos queda ver como podríamos monitorizar un equipo de Apple con Mac OSX.

Durante la instalación del OS X tendremos que elegir una unidad de disco en la cual instalar el sistema operativo. Por defecto no nos aparecerá ninguno.

En general Mac OS X es muy parecido a cualquier distribución linux, en casos como iOS para dispositivos móviles de Apple muchos paquetes de instalación son .deb como por ejemplo los que instalamos en Debian, de hecho se instalan igual si accedemos por ssh al terminar:

- `#dpkg -l nombredelpaquete.deb`

Como es lógico este S.O. tiene peculiaridades respecto a los sistemas operativos libre linux, suponiendo que somos novatos en la administración y temas más avanzados que un simple usuario en este SO debemos acudir a la web oficial y ver la documentación.

La manera que vemos de poder hacer chequeos al Mac OSX es a través de "SNMP".

La dirección a la documentación oficial de este caso es: "[Apple SNMP](#)"

Vamos a necesitar un compilador de C para poder compilar el net-snmp. Accedemos a la web de descargas de herramientas de desarrollo de Apple [aquí](#).

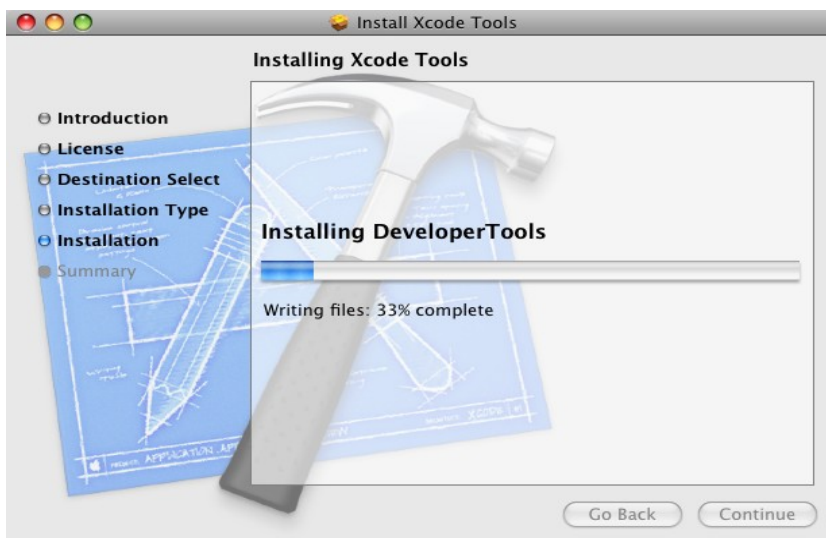
- Buscamos "xCode" y en este caso voy a descargar la versión 3.0 ya que la versión del OS X tampoco es la más actualizada (10.5.6). Descargamos el archivo .dmg y el PDF que es la documentación que lo acompaña.

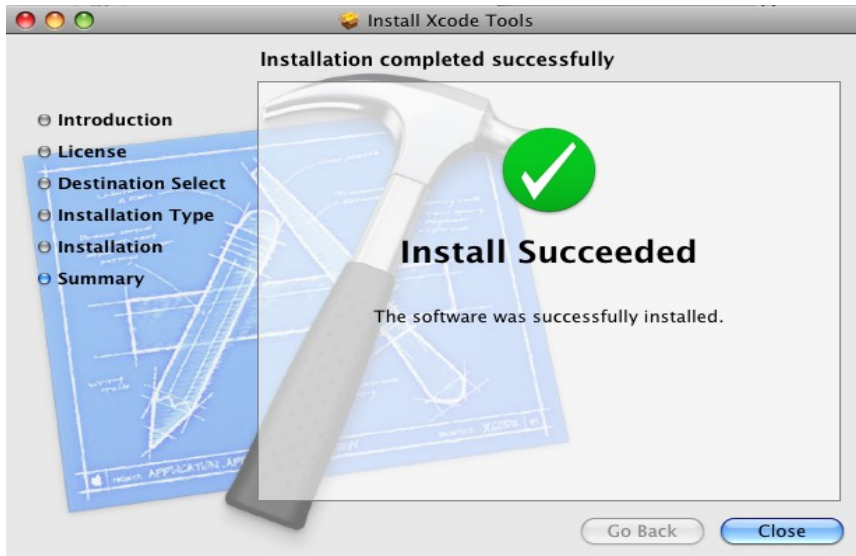


- Una vez descargado se nos abrirá el .dmg (parecido a un .iso).



- Damos doble click sobre "XcodeTools.mpkg" y seguimos el instalador, dejando las opciones por defecto.

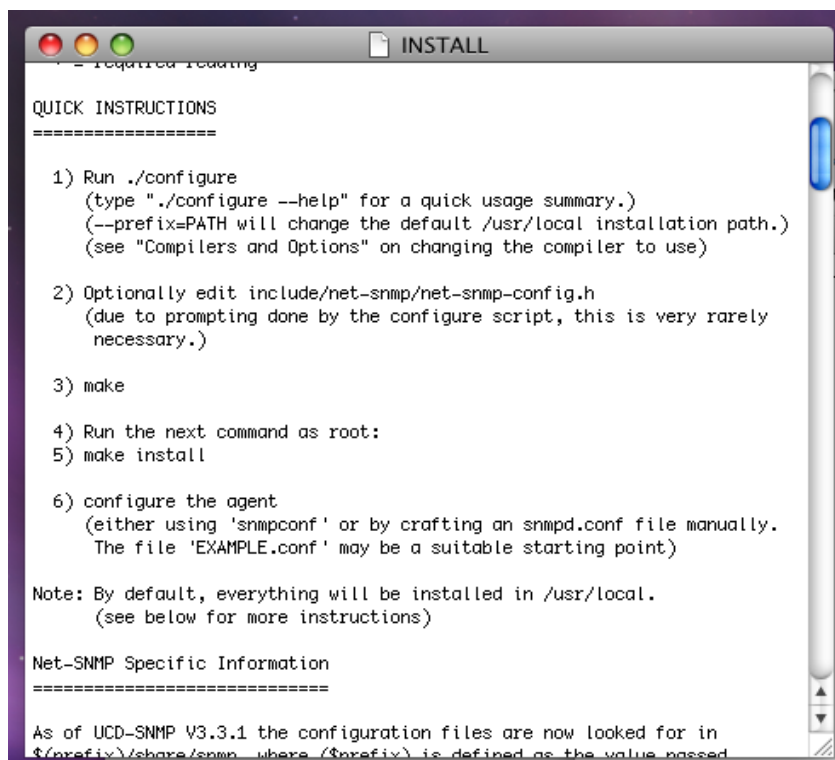




Para que nagios realice sus chequeos vamos a instalar un cliente "SNMP" en este caso "net-snmp", que lo descargamos de la siguiente dirección: [NET-SNMP](http://net-snmp.sourceforge.net/).

Una vez descargado abrimos la carpeta y abrimos el archivo llamado "INSTALL", el cual nos muestra como instalar lo que acabamos de descargar.

- Seguimos los pasos que nos indica el archivo de texto.



As of UCD-SNMP V3.3.1 the configuration files are now looked for in  
`$(prefix)/share/snmp`, where `(prefix)` is defined as the value passed

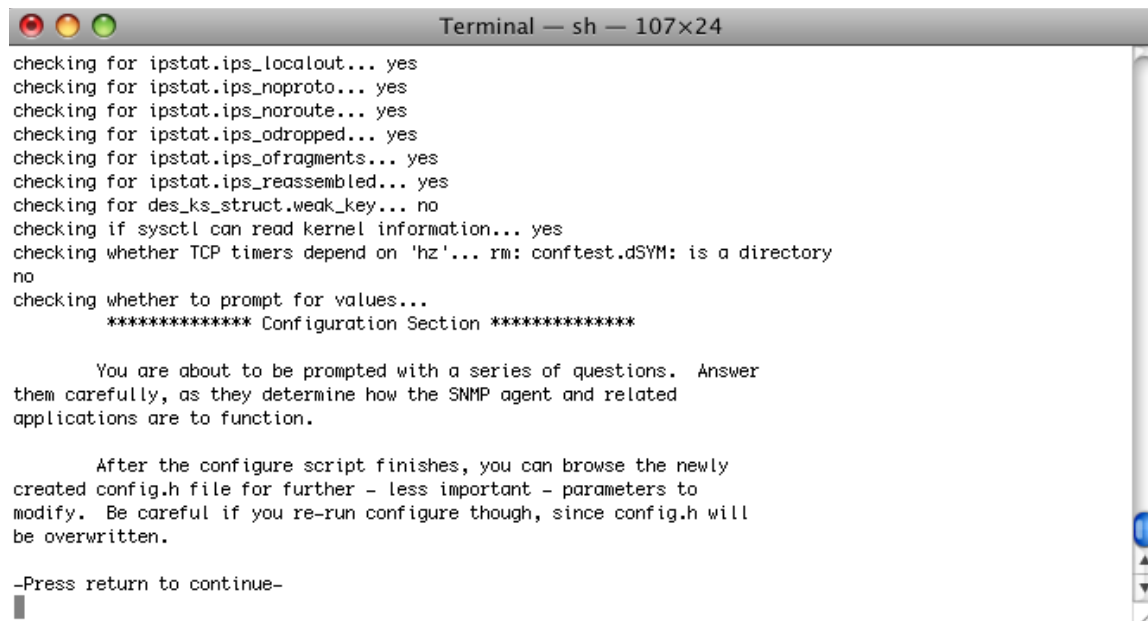
- Ejecutamos en consola el comando `./configure`



```

alberto-mariscals-hp-pavilion-dv6-notebook-pc:net-snmp-5.3.4 alberto$ ./configure
checking what to build and install... agent apps man local mibs
using default temporary file pattern /tmp/snmpdXXXXXX
using default "enterprise.net-snmp"
using default enterprise sysOID "NET-SNMP-MIB::netSnmpAgentOIDs..."
using default notifications "NET-SNMP-MIB::netSnmpNotifications"
using OS default send buffer size for server sockets
using OS default rcv buffer size for server sockets
using OS default send buffer size for client sockets
using OS default rcv buffer size for client sockets
checking if I need to feed myself to ksh... no
checking for gcc... gcc
checking for C compiler default output file name... a.out
checking whether the C compiler works... yes
checking whether we are cross compiling... no
checking for suffix of executables...
checking for suffix of object files... o
checking whether we are using the GNU C compiler... yes
checking whether gcc accepts -g... yes
checking for gcc option to accept ANSI C... none needed
checking how to run the C preprocessor... gcc -E
checking for egrep... grep -E
checking for AIX... no
checking build system type... i386-apple-darwin9.6.0

```



```

checking for ipstat.ips_localout... yes
checking for ipstat.ips_noproto... yes
checking for ipstat.ips_noroute... yes
checking for ipstat.ips_odropped... yes
checking for ipstat.ips_ofragments... yes
checking for ipstat.ips_reassembled... yes
checking for des_ks_struct.weak_key... no
checking if sysctl can read kernel information... yes
checking whether TCP timers depend on 'hz'... rm: conftest.dSYM: is a directory
no
checking whether to prompt for values...
***** Configuration Section *****

You are about to be prompted with a series of questions. Answer
them carefully, as they determine how the SNMP agent and related
applications are to function.

After the configure script finishes, you can browse the newly
created config.h file for further - less important - parameters to
modify. Be careful if you re-run configure though, since config.h will
be overwritten.

-Press return to continue-

```

- Nos preguntara que versión de "SNMP" queremos usar. Las diferencias las podemos ver [AQUÍ](#) . Por defecto vamos a usar la "v3".

\*\*\* Default SNMP Version:

Starting with Net-SNMP 5.0, you can choose the default version of the SNMP protocol to use when no version is given explicitly on the command line, or via an 'snmp.conf' file. In the past this was set to SNMPv1, but you can use this to switch to SNMPv3 if desired. SNMPv3 will provide a more secure management environment (and thus you're encouraged to switch to SNMPv3), but may break existing scripts that rely on the old behaviour. (Though such scripts will probably need to be changed to use the '-c' community flag anyway, as the SNMPv1 command line usage has changed as well.).

At this prompt you can select "1", "2" (for SNMPv2c), or "3" as the default version for the command tools (snmpget, ...) to use. This can always be overridden at runtime using the -v flag to the tools, or by using the "defVersion" token in your snmp.conf file.

Providing the --with-default-snmp-version="x" parameter to ./configure will avoid this prompt.

Default version of SNMP to use (3):  
setting Default version of SNMP to use to... 3  
checking System Contact Information...

- Nos preguntará quien será el usuario al que contactara el servicio. Dejamos el usuario por defecto.

\*\*\* System Contact Information:

Describes who should be contacted about the host the agent is running on. This information is available in the MIB-II tree. This can also be over-ridden using the "syscontact" syntax in the agent's configuration files.

Providing the --with-sys-contact="contact" parameter to ./configure will avoid this prompt.

System Contact Information (alberto@): █

- Configuraremos el directorio donde se guardará el log del agente.

\*\*\* Logfile location:

Enter the default location for the snmpd agent to dump information & errors to. If not defined (enter the keyword "none" at the prompt below) the agent will use stdout and stderr instead. (Note: This value can be over-ridden using command line options.)

Providing the --with-logfile="path" parameter to ./configure will avoid this prompt.

Location to write logfile (/var/log/snmpd.log): █

- Seleccionamos el directorio donde guardaremos los archivos de configuración.

\*\*\* snmpd persistent storage location:

Enter a directory for the SNMP library to store persistent data in the form of a configuration file. This default location is different than the old default location (which was for ucd-snmp). If you stay with the new path, I'll ask you in a second if you wish to copy your files over to the new location (once only). If you pick some other path than the default, you'll have to copy them yourself. There is nothing wrong with picking the old path (/var/ucd-snmp) if you'd rather.

Providing the --with-persistent-directory="path" parameter to ./configure will avoid this prompt.

Location to write persistent information (/var/net-snmp): █

```
Location to write persistent information (/var/net-snmp):
setting Location to write persistent information to... /var/net-snmp
configure: creating ./config.status
config.status: creating Makefile
config.status: creating snmplib/Makefile
config.status: creating apps/Makefile
config.status: creating apps/snmpnetstat/Makefile
config.status: creating agent/Makefile
config.status: creating agent/helpers/Makefile
config.status: creating agent/mibgroup/Makefile
config.status: creating local/Makefile
config.status: creating testing/Makefile
config.status: creating man/Makefile
config.status: creating mibs/Makefile
config.status: creating net-snmp-config
config.status: creating include/net-snmp/net-snmp-config.h
config.status: executing default commands
```

- Al final nos muestra un resumen.

```
-----
Net-SNMP configuration summary:
-----

SNMP Versions Supported: 1 2c 3
Net-SNMP Version: 5.3.4
Building for: darwin9
Network transport support: Callback Unix TCP UDP
SNMPv3 Security Modules: usm
Agent MIB code: mibII ucd_snmp snmpv3mibs notification notification-log
-mib target agent_mibs agentx disman/event-mib disman/schedule utilities
SNMP Perl modules: disabled
Embedded perl support: disabled
Authentication support: MD5 SHA1
Encryption support: DES AES
WARNING: New version of the Event MIB which may be subtly different from the original implementation - configure with 'disman/old-event-mib' for the previous version
-----

alberto-mariscals-hp-pavilion-dv6-notebook-pc:net-snmp-5.3.4 alberto$ █
```

- Ahora ejecutamos el comando "make".

```
alberto-mariscals-hp-pavilion-dv6-notebook-pc:net-snmp-5.3.4 alberto$ make
gcc -E -Iinclude -I./include -I./agent/mibgroup -I. -I. -DDONT_INC_STRUCTS -DBINDIR=/usr/local/bin -x c ./sedscrip
in | egrep '^[a-zA-Z]' | sed 's/REMOVE//g;s# */#/#g;s# */#/#g;s# */#/#g;s# */#/#g;' > sedscrip
echo 's#DATADIR#/usr/local/share#g' >> sedscrip
echo 's#LIBDIR#/usr/local/lib#g' >> sedscrip
echo 's#BINDIR#/usr/local/bin#g' >> sedscrip
echo 's#PERSISTENT_DIRECTORY#/var/net-snmp#g' >> sedscrip
echo 's#SYSCONFDIR#/usr/local/etc#g' >> sedscrip
.....
making all in /Users/alberto/Downloads/net-snmp-5.3.4/mibs
chmod a+x net-snmp-config
touch net-snmp-config-x
alberto-mariscals-hp-pavilion-dv6-notebook-pc:net-snmp-5.3.4 alberto$ █
```

- Activamos el usuario root.

```
alberto-mariscals-hp-pavilion-dv6-notebook-pc:net-snmp-5.3.4 alberto$ dsenableroot -u alberto
user password:
root password:
verify root password:

dsenableroot:: ***Successfully enabled root user.
alberto-mariscals-hp-pavilion-dv6-notebook-pc:net-snmp-5.3.4 alberto$ su
Password:
sh-3.2# █
```



- Ahora ejecutamos el comando "make install".

```
sh-3.2# make install
creating directory /usr/local/include/net-snmp
mkdir /usr/local
mkdir /usr/local/include
mkdir /usr/local/include/net-snmp
/usr/bin/install -c -m 644 ./include/net-snmp/version.h /usr/local/include/net-snmp/version.h
installing version.h in /usr/local/include/net-snmp

.....
install: installed SNMPv2-TM.txt in /usr/local/share/snmp/mibs
/usr/bin/install -c -m 644 ./DISMAN-EVENT-MIB.txt /usr/local/share/snmp/mibs/DISMAN-EVENT-MIB.txt
install: installed DISMAN-EVENT-MIB.txt in /usr/local/share/snmp/mibs
/usr/bin/install -c -m 644 ./DISMAN-SCHEDULE-MIB.txt /usr/local/share/snmp/mibs/DISMAN-SCHEDULE-MIB.txt
install: installed DISMAN-SCHEDULE-MIB.txt in /usr/local/share/snmp/mibs
sh-3.2#
```

Vamos a agregar un cliente MAC con OSX. Necesitamos tener una plantilla, que en este caso vamos a utilizar la misma de Windows copiandola para cambiar nombres y tener una organización.

- Creamos el siguiente apartado.

```
#### Dispositivos MAC OS X ####
define host{
name                mac-osx ; Nombre de la plantilla
use                 generic-host ; Un atributo general
check_period        24x7      ; Periodos de chequeos
check_interval      5         ; Tiempo entre chequeo y chequeo
retry_interval      1         ; tiempo para reintentar un chequeo
max_check_attempts  10        ; optativo pero recomendable
check_command       check-host-alive ; Comandos generales de chequeo $
notification_period 24x7      ;
notification_interval 30      ;
notification_options d,r      ;
contact_groups      admins    ; Grupo de usuarios que reciben las not$
host_groups         OSX ; Como no hay un grupo definido aun, me daría error.
}
```

- Creamos el archivo "/etc/nagios3/objects/macintosh.cfg. Creamos una entrada para el host y también un grupo al que pertenecerá.

```
GNU nano 2.2.4      Archivo: macintosh.cfg

# Aquí definiremos las máquinas con MAC OS X

### Máquina de Dark-Vader ###
define host{
use mac-osx ; Nombre de la plantilla que vamos a usar
host_name LordVader ; Nombre que le daremos al HOST
alias LordSith ; Nombre asociado al host
address 192.168.1.17 ; Dirección IP de la máquina windows
parents cisco-epc3825
}

#### Grupos de máquinas Mac OS X ####
define hostgroup{
hostgroup_name OSX ; Es el nombre que tendrá el grupo
alias DispositivosOSX ; Un alias para el grupo
members LordVader ; Miembros que pertenecerán al grupo.
}
```



- Reiniciamos el servicio de Nagios ("/etc/init.d/nagios3 restart") y comprobamos que detecta el host levantado.

#### Host Status Details For All Host Groups

Host ↑↓	Status ↑↓	Last Check ↑↓	Duration ↑↓	Status Information
<a href="#">Anakin</a>	DOWN	2011-11-16 22:46:23	2d 2h 28m 15s	CRITICAL - Host Unreachable (192.168.1.56)
<a href="#">DarkMouth</a>	UP	2011-11-16 22:51:24	0d 0h 0m 7s	PING OK - Packet loss = 0%, RTA = 5.34 ms
<a href="#">Obi-Wan</a>	DOWN	2011-11-16 22:48:23	4d 7h 46m 11s	CRITICAL - Host Unreachable (192.168.1.20)
<a href="#">cisco-ipc3825</a>	UP	2011-11-16 22:50:53	8d 0h 49m 30s	PING OK - Packet loss = 0%, RTA = 307.71 ms
<a href="#">localhost</a>	UP	2011-11-16 22:47:03	31d 21h 53m 22s	PING OK - Packet loss = 0%, RTA = 0.03 ms

5 Matching Host Entries Displayed

Vamos a configurar Nagios para que monitoree algunos servicios de la máquina.

Como hemos instalado el "net-snmp" la configuración de los servicios a monitorear es igual que el de las máquinas linux, pero en este caso en el archivo de los equipos Mac.

```

GNU nano 2.2.4          Fichero: macintosh.cfg

alias DispositivosOSX          ; Un alias para el grupo
members DarkMouth              ; Miembros que pertenecieran al grupo.
}

I
#### Servicios a monitorear para dispositivos MAC OS X ####
#Servicio a PING
define service{
use generic-service
host_name DarkMouth
service_description PING
check_command check_ping!200.0,20%!600.0,60%
normal_check_interval 5
retry_check_interval 1
}

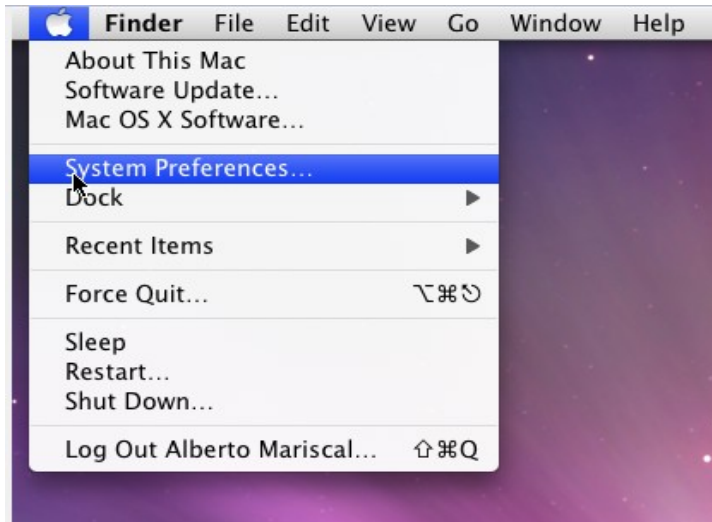
```

Reiniciamos el servicio de Nagios y comprobamos que ya aparecen en el administrador Web.

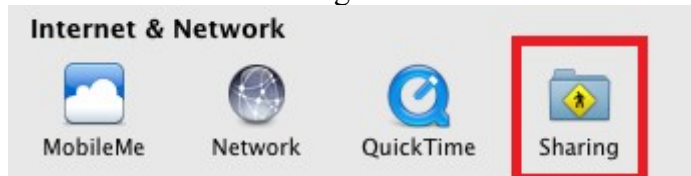
<a href="#">DarkMouth</a>	<a href="#">Disco Duro</a>	CRITICAL	2011-11-17 18:42:15	0d 0h 24m 43s	4/4	DISK CRITICAL - free space: / 53 56 MB (73% inode=85%): /lib/init/rw 505 MB (100% inode=99%): /dev 500 MB (99% inode=99%): /dev/shm 505 MB (100% inode=99%):
	<a href="#">Estado HTTP</a>	CRITICAL	2011-11-17 18:43:16	0d 0h 23m 42s	4/4	Conexión rehusada
	<a href="#">Estado SSH</a>	CRITICAL	2011-11-17 18:40:18	0d 0h 21m 40s	4/4	No existe ninguna ruta hasta el 'host'
	<a href="#">PING</a>	OK	2011-11-17 18:42:40	0d 0h 21m 18s	1/4	PING OK - Packet loss = 0%, RTA = 0.60 ms

Vamos a instalar un servidor Web y a habilitar el acceso ssh para arreglar esas alarmas.

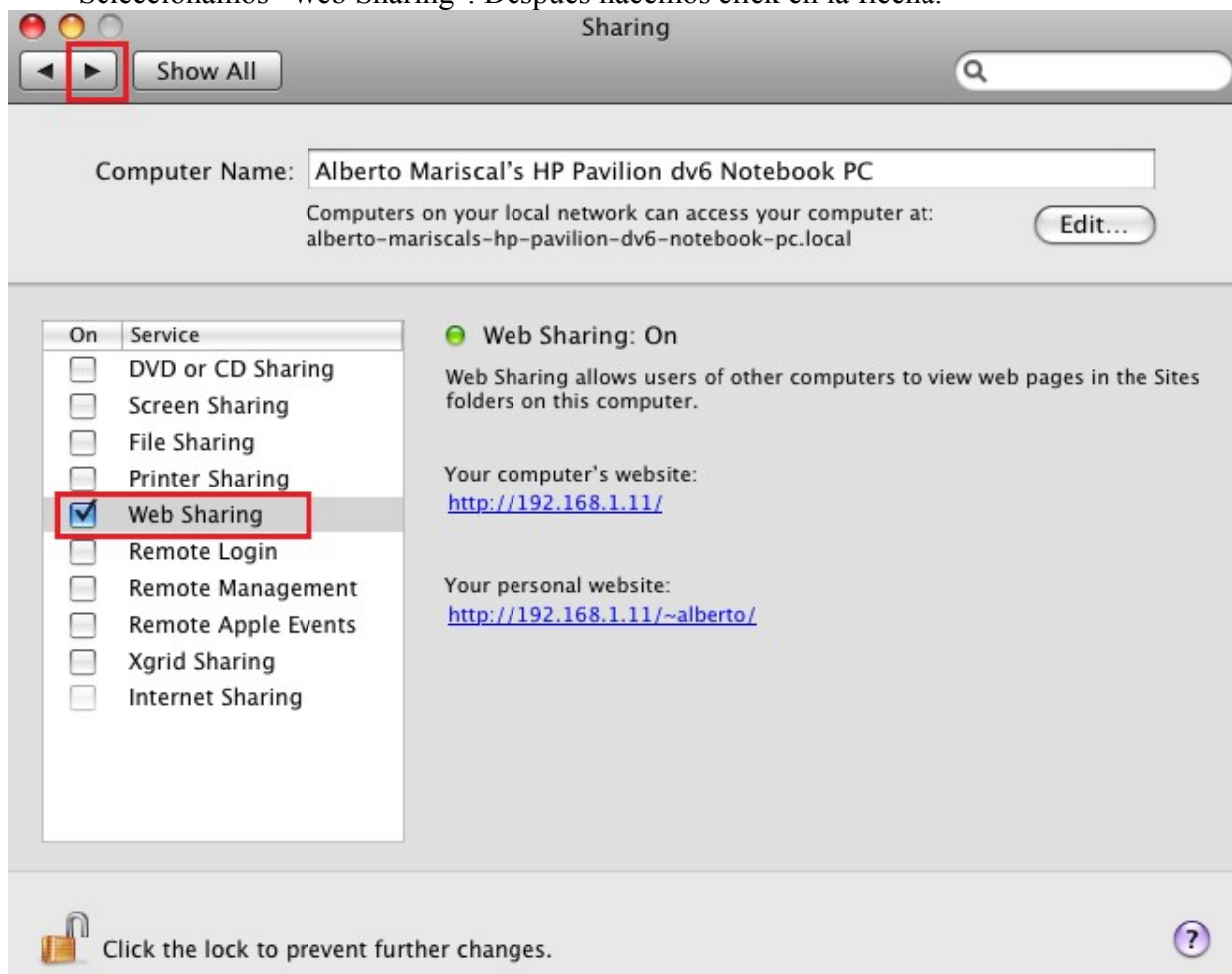
- En la barra superior de la pantalla seleccionamos el logo de "Apple" => "System Preferences..."



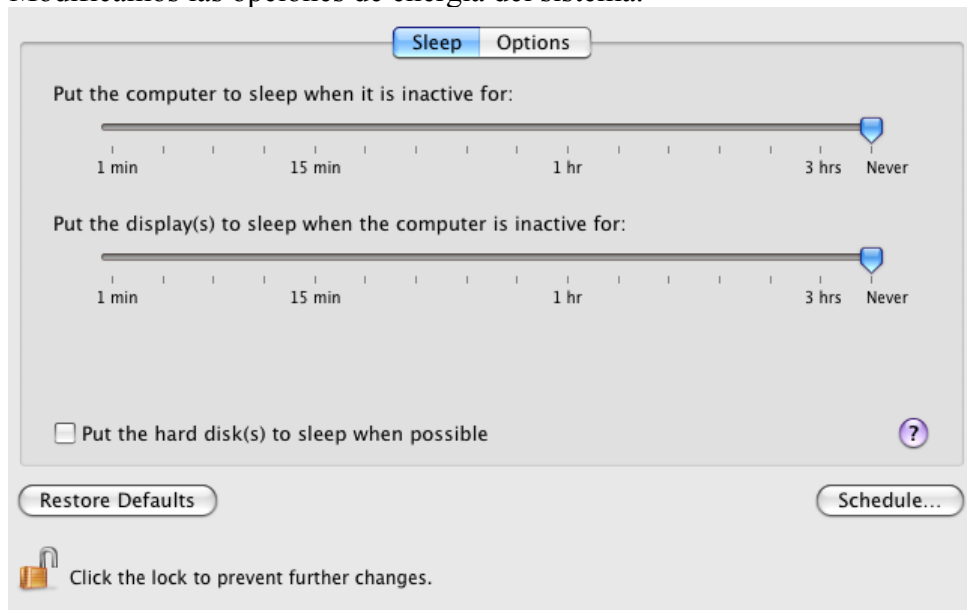
- Hacemos click en "Sharing".



- Seleccionamos "Web Sharing". Despues hacemos click en la flecha.



- Modificamos las opciones de energía del sistema.



Al finalizar podemos cerrar la ventana.

- Abrimos un navegador Web y accedemos a la dirección de la máquina.  
**¡Funcionó! ¡El Servidor de Red Apache ha sido instalado en ese sitio!**

Entonces los dueños de esta máquina han instalado el Servidor de Red Apache con éxito. Ahora deben añadir contenido a este directorio y reemplazar esta página.

Esta página y no es lo que esperaba, por favor **contacte al administrador de este sitio**. (Trate de enviar correo electrónico a <Webmaster@domain>.) Aunque es seguro que no tiene ninguna conexión con el Apache Group, por eso favor de no enviar correo sobre este sitio o su contenido a los autores de Apache. Si lo ha

no sido incluida en esta distribución.

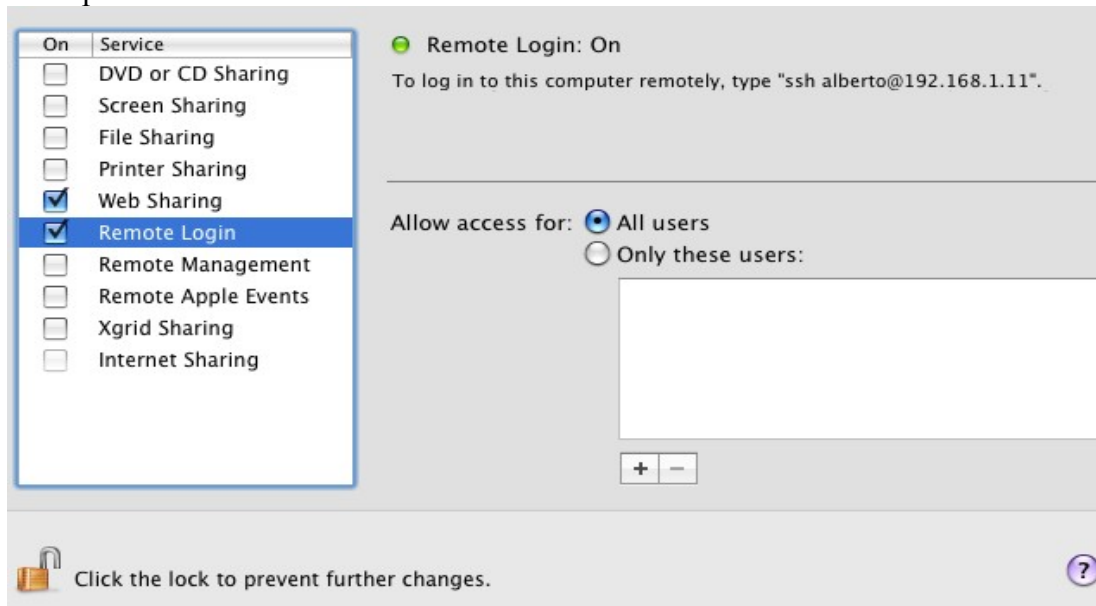
Se recomienda a usar la siguiente imagen para indicar que su sitio es servido por Apache. ¡Gracias por usar Apache!



- Comprobamos en Nagios.

<a href="#">DarkMouth</a>	<a href="#">Disco Duro</a>	CRITICAL	2011-11-17 19:42:15	0d 1h 23m 57s	4/4	DISK CRITICAL - free space: / 5355 MB (73% inode=85%); /lib/init/rw 505 MB (100% inode=99%); /dev 500 MB (99% inode=99%); /dev/shm 505 MB (100% inode=99%);
	<a href="#">Estado HTTP</a>	OK	2011-11-17 19:43:08	0d 0h 0m 4s	1/4	HTTP OK: HTTP/1.1 200 OK - 1859 bytes in 0.006 second response time
	<a href="#">Estado SSH</a>	CRITICAL	2011-11-17 19:40:18	0d 1h 20m 54s	4/4	Conexión rechazada
	<a href="#">PING</a>	OK	2011-11-17 19:40:45	0d 0h 2m 27s	1/4	PING OK - Packet loss = 0%, RTA = 2.35 ms

- Accedemos a "System Preferences..." y seleccionamos "Remote Login", podríamos elegir entre permitir acceso a usuarios específicos o todos los usuarios.



- Comprobamos en Nagios que ya tenemos en "OK" el servicio "ssh".

<a href="#">DarkMouth</a>	<a href="#">Disco Duro</a>	CRITICAL	2011-11-17 20:17:15	0d 2h 1m 23s	4/4	DISK CRITICAL - free space: / 53 55 MB (73% inode=85%): /lib/init/rw 505 MB (100% inode=99%): /dev 500 MB (99% inode=99%): /dev/shm 505 MB (100% inode=99%):
	<a href="#">Estado HTTP</a>	OK	2011-11-17 20:18:08	0d 0h 37m 30s	1/4	HTTP OK: HTTP/1.1 200 OK - 1859 bytes in 0,003 second response time
	<a href="#">Estado SSH</a>	OK	2011-11-17 20:20:18	0d 0h 10m 20s	1/4	SSH OK - OpenSSH_5.1 (protocol 2.0)
	<a href="#">PING</a>	OK	2011-11-17 20:15:45	0d 0h 39m 53s	1/4	PING OK - Packet loss = 0%, RTA = 0.52 ms

## Monitorear equipos con CentOS

Lo primero es comprobar que tiene el cliente snmp instalado.

```
#yum search snmp
```

```
===== Matched: snmp =====
net-snmp.i686 : A collection of SNMP protocol tools and libraries
net-snmp-devel.i686 : The development environment for the NET-SNMP project
net-snmp-libs.i686 : The NET-SNMP runtime libraries
net-snmp-perl.i686 : The perl NET-SNMP module and the mib2c tool
net-snmp-python.i686 : The Python 'netsnmp' module for the NET-SNMP
net-snmp-utils.i686 : Network management utilities using SNMP, from the NET-SNMP
                        : project
perl-SNMP_Session.noarch : SNMP support for Perl 5
cluster-snmp.i686 : Red Hat Enterprise Linux Cluster Suite - SNMP agent
php-snmp.i686 : A module for PHP applications that query SNMP-managed devices
openhpi-subagent.i686 : NetSNMP subagent for OpenHPI
arpwatch.i686 : Network monitoring tools for tracking IP addresses on a network
pywbem.noarch : Python WBEM Client and Provider Interface
[root@localhost usuario]#
```

Vemos el resultado de la búsqueda, e instalamos el "net-snmp-utils".

```
#yum install net-snmp-utils
```

```
=====
Package                Arch      Version              Repository      Size
=====
Installing:
net-snmp-utils          i686      1:5.5-27.el6_0.1     updates        165 k
Transaction Summary
=====
Install      1 Package(s)
Upgrade      0 Package(s)

Total download size: 165 k
Installed size: 288 k
Is this ok [y/N]: y
Downloading Packages:
net-snmp-utils-5.5-27.el6_0.1.i686.rpm      | 165 kB      00:03
Running rpm_check_debug
Running Transaction Test
Transaction Test Succeeded
Running Transaction
█
```

```
Installing      : 1:net-snmp-utils-5.5-27.el6_0.1.i686
```

1/1

```
Installed:
```

```
net-snmp-utils.i686 1:5.5-27.el6_0.1
```

```
Complete!
```

```
[root@localhost usuario]#
```

Comprobamos que se ha instalado correctamente:

```
[root@localhost usuario]# rpm -qa net-snmp-utils
net-snmp-utils-5.5-27.el6_0.1.i686
[root@localhost usuario]#
```

Al introducir el comando nos listara todos los paquetes que contengan esa cadena en su nombre.

Vamos a añadir la máquina "C3PO" con CentOS en los equipos que monitoriza Nagios.

Como en esta ocasion no tenemos varias/muchas máquinas CentOS podemos incluirla en el archivo que tenemos ya creado para máquinas Linux en general.

- Editamos el archivo `"/etc/nagios3/dispositivos.cfg"`.

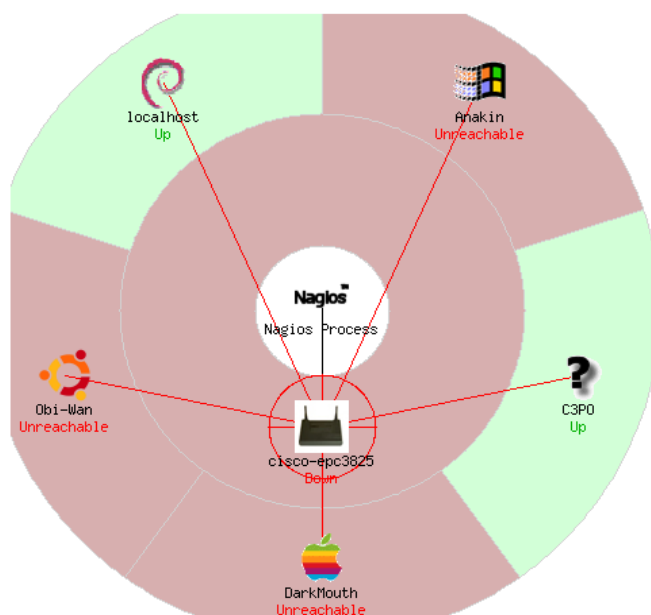
```
## Equipos con CentOS
define host{
  use generic-host
  host_name C3PO
  alias c3po
  address 192.168.1.27
  check_command check-host-alive
  max_check_attempts 20
  notification_interval 60
  notification_period 24x7
  notification_options d,u,r
  parents cisco-epc3825
}
```

- Añadimos un servicio a monitorizar (en el mismo archivo que se añade la máquina).

```
###Servicios a monitorizar###
#Servicio a PING
define service{
  use generic-service
  host_name Obi-Wan,C3PO
  service_description PING
  check_command check_ping!200.0,20%!600.0,60%
  normal_check_interval 5
  retry_check_interval 1
}
```

Como vemos, hemos añadido la nueva máquina a un servicio que ya esta usando otra, así nos ahorramos añadir dos veces el servicio.

- Reiniciamos el servicio de Nagios y comprobamos que se ha añadido la nueva máquina.



<a href="#">C3PO</a>	<a href="#">Disco Duro</a>	CRITICAL	2011-11-28 16:40:43	0d 0h 2m 38s	2/4	DISK CRITICAL - free space: / 5342 MB (73% inode=85%): /lib/init/rw 505 MB (100% inode=99%): /dev 500 MB (99% inode=99%): /dev/shm 505 MB (100% inode=99%):
	<a href="#">Estado SSH</a>	OK	2011-11-28 16:40:11	0d 0h 2m 10s	1/4	SSH OK - OpenSSH_5.3 (protocol 2.0)
	<a href="#">PING</a>	OK	2011-11-28 16:36:49	0d 0h 7m 56s	1/4	PING OK - Packet loss = 0%, RTA = 0.40 ms

## Introducir un equipo en grupos

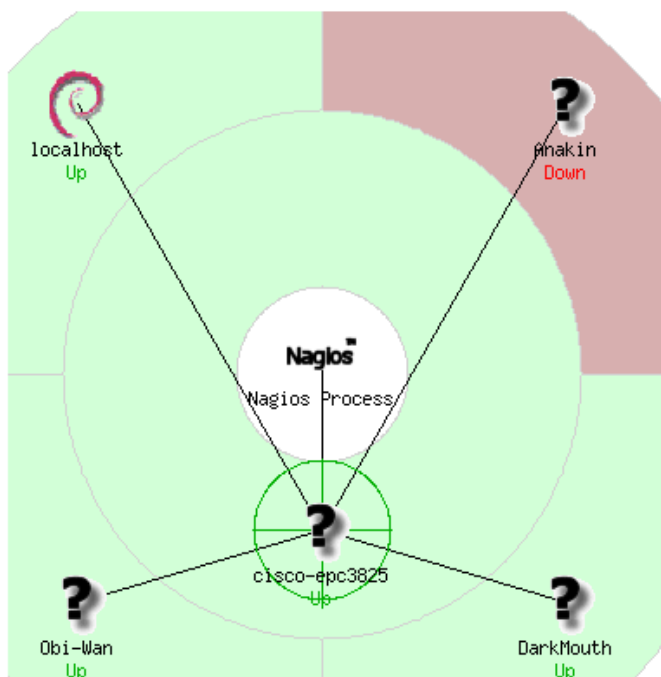
Como ejemplo vamos a usar el equipo Anakin en Nagios, pero puede hacerse igual con cualquier máquina. Necesitaremos meterlo en un grupo y especificarle que procesos queremos que se monitoreen de dicho equipo.

- Vamos a ver como introducirlo en un grupo y configurarle una alerta.

```
#### Grupos de máquinas Windows ####
define hostgroup{
  hostgroup_name PC-Windows      ; Es el nombre que tendrá el grupo
  alias EquiposWindows           ; Un alias para el grupo
  members Anakin                 ; Miembros que pertenecieran al grupo.
}
```

## Asignar iconos para el "Status Map"

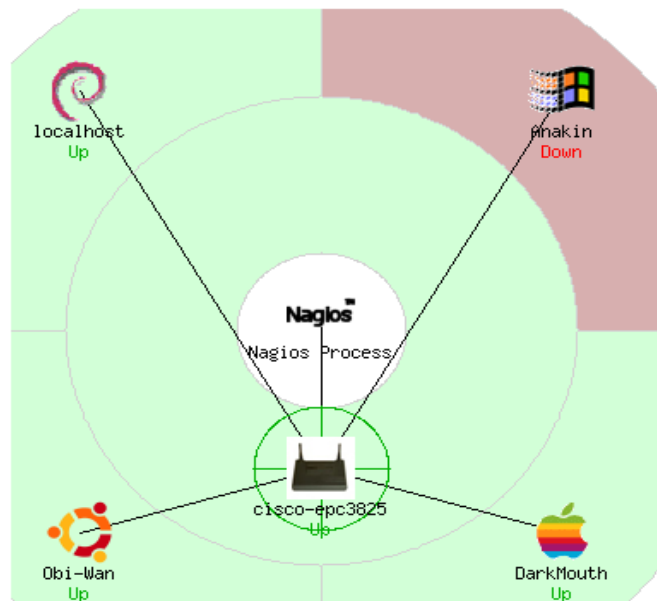
Si recordamos, Nagios posee una opción con la que vemos una especie de mapa de como es nuestra red monitorizada por Nagios. Por defecto no vienen iconos asignado a los dispositivos que monitorizamos, por lo que debemos configurarlo.



- Las imagenes están en "/usr/share/nagios/htdocs/images/logos/", encontraremos los logos divididos en carpetas. Lo primero será previsualizar los iconos y elegir el que queremos para cada dispositivo.
- Accedemos al archivo de configuración de cada dispositivos y añadimos la siguiente etiqueta:

```
##Informacion adicional de los dispositivos
define hostextinfo{
  host_name Obi-Wan
  statusmap_image base/ubuntu.png
}
```

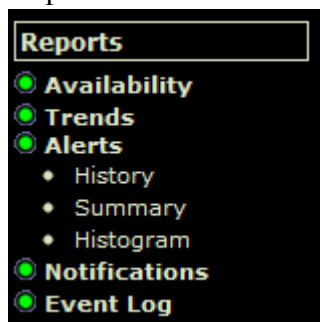
- Haremos lo mismo con todos los dispositivos que queramos luego reiniciamos el servicio de Nagios y comprobamos que se ha producido el cambio:



## Otras opciones del menú de Nagios

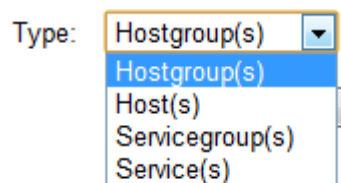
Una vez tenemos nuestra estructura de red montada y lista para monitorizar, vamos a ver el resto de opciones del menú de la interfaz web de Nagios.

- Reportes:



- Availability o Disponibilidad: Nos muestra un informe de la disponibilidad de según lo que necesitamos.

### **Step 1: Select Report Type**





- Vamos a elegir para el ejemplo "Hostgroup(s)".  
Elegimos el grupo que deseamos tener el informe, en este caso el de OSX.

### Step 2: Select Hostgroup

Hostgroup(s):

\*\* ALL HOSTGROUPS \*\*

\*\* ALL HOSTGROUPS \*\*

OSX

PC-Windows

all

debian-servers

equipos\_linux

http-servers

ssh-servers

switches

Configuramos las opciones del reporte:

### Step 3: Select Report Options

Report Period:

If Custom Report Period...

Start Date (Inclusive):

End Date (Inclusive):

Report time Period:

Assume Initial States:

Assume State Retention:

Assume States During Program Downtime:

Include Soft States:

First Assumed Host State:

First Assumed Service State:

Backtracked Archives (To Scan For Initial States):

Creamos el "Reporte de Disponibilidad".

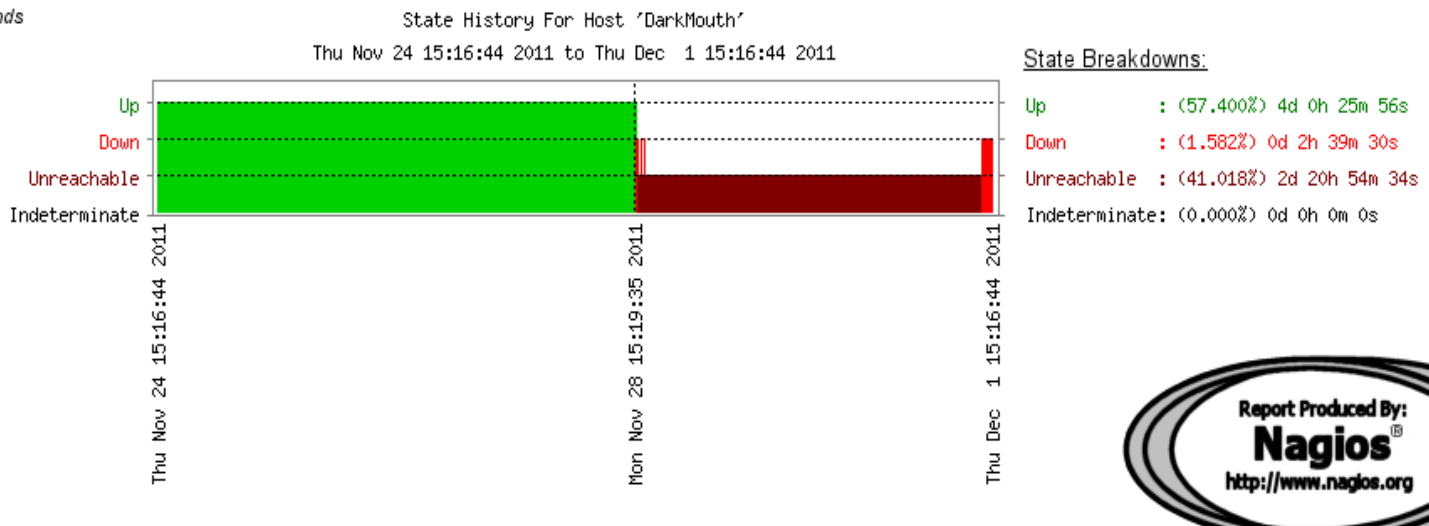
### Hostgroup 'OSX' Host State Breakdowns:

Host	% Time Up	% Time Down	% Time Unreachable	% Time Undetermined
DarkMouth	57.622% (57.622%)	1.360% (1.360%)	41.018% (41.018%)	0.000%
Average	57.622% (57.622%)	1.360% (1.360%)	41.018% (41.018%)	0.000%

57% de Disponibilidad, 1.36% ha estado caído y un 41% era inaccesible.

- Trends o Tendencias: nos lo muestra graficamente.

Trends



- Alerts o Alertas:
  - Hostiry: nos muestra un listado de alertas por orden cronológico.

	December 01, 2011 14:00	
	[2011-12-01 14:46:03] Nagios 3.2.1 starting... (PID=3563)	
	December 01, 2011 13:00	
	[2011-12-01 13:53:40] Caught SIGTERM, shutting down...	
	[2011-12-01 13:37:10] HOST ALERT: C3PO;DOWN;HARD;20;CRITICAL - Host Unreachable (192.168.1.27)	
	[2011-12-01 13:36:00] HOST ALERT: C3PO;DOWN;SOFT;19;CRITICAL - Host Unreachable (192.168.1.27)	
	[2011-12-01 13:34:50] HOST ALERT: C3PO;DOWN;SOFT;18;CRITICAL - Host Unreachable (192.168.1.27)	
	[2011-12-01 13:33:40] HOST ALERT: C3PO;DOWN;SOFT;17;CRITICAL - Host Unreachable (192.168.1.27)	

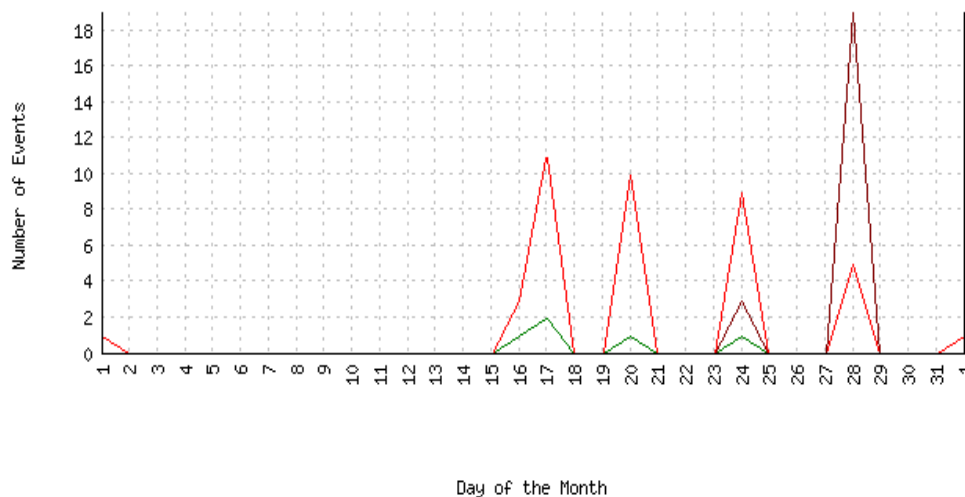
- Summary o Resumen: lo vemos de forma resumida según la configuración que le indiquemos.

Displaying most recent 25 of 196 total matching alerts

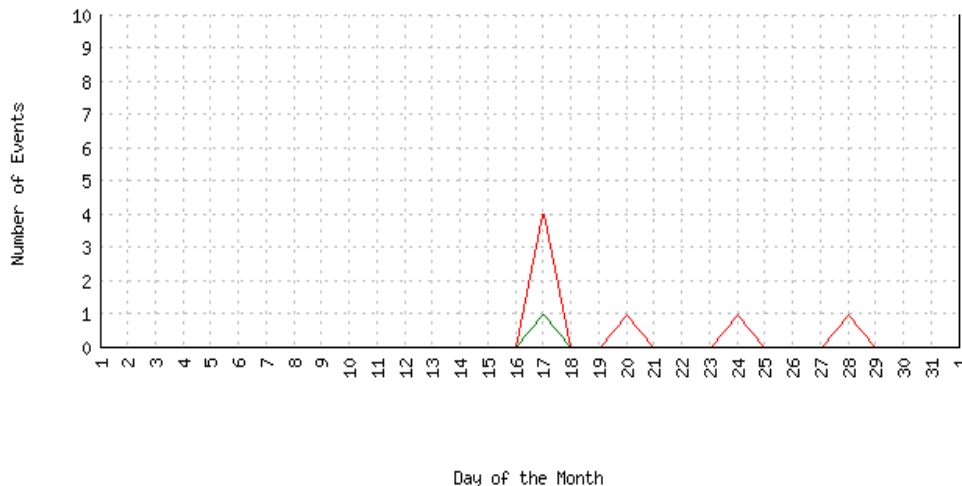
Time	Alert Type	Host	Service	State	State Type	Information
2011-12-01 13:37:10	Host Alert	C3PO	N/A	DOWN	HARD	CRITICAL - Host Unreachable (192.168.1.27)
2011-12-01 13:36:00	Host Alert	C3PO	N/A	DOWN	SOFT	CRITICAL - Host Unreachable (192.168.1.27)
2011-12-01 13:34:50	Host Alert	C3PO	N/A	DOWN	SOFT	CRITICAL - Host Unreachable (192.168.1.27)
2011-12-01 13:33:40	Host Alert	C3PO	N/A	DOWN	SOFT	CRITICAL - Host Unreachable (192.168.1.27)
2011-12-01 13:32:30	Host Alert	C3PO	N/A	DOWN	SOFT	CRITICAL - Host Unreachable (192.168.1.27)
2011-12-01 13:31:20	Host Alert	C3PO	N/A	DOWN	SOFT	CRITICAL - Host Unreachable (192.168.1.27)
2011-12-01 13:30:10	Host Alert	C3PO	N/A	DOWN	SOFT	CRITICAL - Host Unreachable (192.168.1.27)
2011-12-01 13:29:00	Host Alert	C3PO	N/A	DOWN	SOFT	CRITICAL - Host Unreachable (192.168.1.27)
2011-12-01 13:27:50	Host Alert	C3PO	N/A	DOWN	SOFT	CRITICAL - Host Unreachable (192.168.1.27)
2011-12-01 13:26:40	Host Alert	C3PO	N/A	DOWN	SOFT	CRITICAL - Host Unreachable (192.168.1.27)
2011-12-01 13:25:30	Host Alert	C3PO	N/A	DOWN	SOFT	CRITICAL - Host Unreachable (192.168.1.27)
2011-12-01 13:24:20	Host Alert	C3PO	N/A	DOWN	SOFT	CRITICAL - Host Unreachable (192.168.1.27)

- Histogram o Histograma: Podemos ver una grafica historica por "Hosts" o "Servicios".

Histogram

Event History For Host 'DarkMouth'  
Thu Nov 24 15:29:27 2011 to Thu Dec 1 15:29:27 2011

Histogram



Event History For Service 'Estado HTTP' On Host 'DarkMouth'  
Thu Nov 24 15:30:10 2011 to Thu Dec 1 15:30:10 2011

- Notifications o Notificaciones:  
Nos muestra una lista de las notificaciones que se mandan por email u otros medios de todos los dispositivos ordenados cronologicamente.

Host	Service	Type	Time	Contact	Notification Command	Information
<a href="#">Anakin</a>	N/A	HOST DOWN	2011-12-01 15:16:23	<a href="#">root</a>	<a href="#">notify-host-by-email</a>	CRITICAL - Host Unreachable (192.168.1.56)
<a href="#">DarkMouth</a>	N/A	HOST DOWN	2011-12-01 15:16:23	<a href="#">root</a>	<a href="#">notify-host-by-email</a>	CRITICAL - Host Unreachable (192.168.1.141)
<a href="#">Obi-Wan</a>	N/A	HOST DOWN	2011-12-01 14:47:53	<a href="#">root</a>	<a href="#">notify-host-by-email</a>	CRITICAL - Host Unreachable (192.168.1.20)
<a href="#">C3PO</a>	N/A	HOST DOWN	2011-12-01 14:47:03	<a href="#">root</a>	<a href="#">notify-host-by-email</a>	CRITICAL - Host Unreachable (192.168.1.27)
<a href="#">Anakin</a>	N/A	HOST DOWN	2011-12-01 14:46:13	<a href="#">root</a>	<a href="#">notify-host-by-email</a>	CRITICAL - Host Unreachable (192.168.1.56)
<a href="#">DarkMouth</a>	N/A	HOST DOWN	2011-12-01 14:46:13	<a href="#">root</a>	<a href="#">notify-host-by-email</a>	CRITICAL - Host Unreachable (192.168.1.141)
<a href="#">C3PO</a>	N/A	HOST DOWN	2011-12-01 13:37:10	<a href="#">root</a>	<a href="#">notify-host-by-email</a>	CRITICAL - Host Unreachable (192.168.1.27)
<a href="#">cisco-epc3825</a>	<a href="#">PING</a>	OK	2011-11-28 17:18:30	<a href="#">root</a>	<a href="#">notify-service-by-email</a>	PING OK - Packet loss = 0%, RTA = 4.93 ms
<a href="#">cisco-epc3825</a>	<a href="#">PING</a>	CRITICAL	2011-11-28 16:54:30	<a href="#">root</a>	<a href="#">notify-service-by-email</a>	CRITICAL - Host Unreachable (192.168.1.1)

- Event log: Nos muestra el archivo de log que tienen "todos" los servicios que podemos instalar en linux, pero desde la interfaz Web, en el cual se visualiza mas comodamente, pero que al fin y al cabo lo que vemos son las entradas del archivo de log.

December 01, 2011 15:00

 [2011-12-01 15:16:23] HOST NOTIFICATION: root;Anakin;DOWN;notify-host-by-email;CRITICAL - Host Unreachable (192.168.1.56)  
 [2011-12-01 15:16:23] HOST NOTIFICATION: root;DarkMouth;DOWN;notify-host-by-email;CRITICAL - Host Unreachable (192.168.1.141)

December 01, 2011 14:00

 [2011-12-01 14:47:53] HOST NOTIFICATION: root;Obi-Wan;DOWN;notify-host-by-email;CRITICAL - Host Unreachable (192.168.1.20)  
 [2011-12-01 14:47:03] HOST NOTIFICATION: root;C3PO;DOWN;notify-host-by-email;CRITICAL - Host Unreachable (192.168.1.27)  
 [2011-12-01 14:46:13] HOST NOTIFICATION: root;Anakin;DOWN;notify-host-by-email;CRITICAL - Host Unreachable (192.168.1.56)  
 [2011-12-01 14:46:13] HOST NOTIFICATION: root;DarkMouth;DOWN;notify-host-by-email;CRITICAL - Host Unreachable (192.168.1.141)  
 [2011-12-01 14:46:03] Finished daemonizing... (New PID=3565)  
 [2011-12-01 14:46:03] LOG VERSION: 2.0  
 [2011-12-01 14:46:03] Local time is Thu Dec 01 14:46:03 CET 2011  
 [2011-12-01 14:46:03] Nagios 3.2.1 starting... (PID=3563)













December 01, 2011 13:00

- System: Nos muestra información sobre el estado del servicio de localhost, y la configuración de los equipos introducidos a monitorear.

**Process Information**

Program Version:	3.2.1
Program Start Time:	2011-12-01 14:46:03
Total Running Time:	0d 0h 54m 5s
Last External Command Check:	2011-12-01 15:40:03
Last Log File Rotation:	N/A
Nagios PID	3565
Notifications Enabled?	YES
Service Checks Being Executed?	YES
Passive Service Checks Being Accepted?	YES
Host Checks Being Executed?	YES
Passive Host Checks Being Accepted?	YES
Event Handlers Enabled?	Yes
Obsessing Over Services?	No
Obsessing Over Hosts?	No
Flap Detection Enabled?	Yes
Performance Data Being Processed?	No

**Process Commands**

	<a href="#">Shutdown the Nagios process</a>
	<a href="#">Restart the Nagios process</a>
	<a href="#">Disable notifications</a>
	<a href="#">Stop executing service checks</a>
	<a href="#">Stop accepting passive service checks</a>
	<a href="#">Stop executing host checks</a>
	<a href="#">Stop accepting passive host checks</a>
	<a href="#">Disable event handlers</a>
	<a href="#">Start obsessing over services</a>
	<a href="#">Start obsessing over hosts</a>
	<a href="#">Disable flap detection</a>
	<a href="#">Enable performance data</a>

**Services**

Service															
Host	Description	Max. Check Attempts	Normal Check Interval	Retry Check Interval	Check Command	Check Period	Parallelize	Volatile	Obsess Over	Enable Active Checks	Enable Passive Checks	Check Freshness	Freshness Threshold	Default Contacts/Groups	E
<a href="#">Anakin</a>	Espacio en C	4	0h 5m 0s	0h 1m 0s	<a href="#">check_nt\USEDDISKSPACE-1 c- w 80 -c 90</a>	24x7	Yes	No	Yes	Yes	Yes	No	Auto-determined value	<a href="#">admins</a>	Y
<a href="#">Anakin</a>	Explorer.exe	4	0h 5m 0s	0h 1m 0s	<a href="#">check_nt\PROCSTATE-d SHOWALL -l explorer.exe</a>	24x7	Yes	No	Yes	Yes	Yes	No	Auto-determined value	<a href="#">admins</a>	Y
<a href="#">Anakin</a>	PING	4	0h 5m 0s	0h 1m 0s	<a href="#">check_ping!200.0.20%!600.0.60%</a>	24x7	Yes	No	Yes	Yes	Yes	No	Auto-determined value	<a href="#">admins</a>	Y
<a href="#">Anakin</a>	Tiempo Activo	4	0h 5m 0s	0h 1m 0s	<a href="#">check_nt\UPTIME</a>	24x7	Yes	No	Yes	Yes	Yes	No	Auto-determined value	<a href="#">admins</a>	Y

## Configuración de contactos

Lo primero que debemos hacer es añadir los contactos al archivo de contactos.

```
### Contactos ###
define contact{
name      contactos
service_notification_period    laboral
host_notification_period      laboral
service_notification_options w,u,c,r,f,s
host_notification_options d,u,r,f,s
service_notification_commands notify-service-by-email
host_notification_commands notificar-host
register 0
}
```

Creamos el archivo "/etc/nagios3/objects/contacts.cfg":

En el mismo archivo definimos el grupo de administradores:

```
### Grupos de contactos ###
define contactgroup{
contactgroup_name admins
alias Administradores
members Alberto
}
```

Solo nos queda definir el contacto, en el mismo archivo añadimos las siguientes líneas:

```
#### Contactos ####
define contact{
contact_name Alberto
use contactos ; es la plantilla definida arriba
alias alberto
email alberto.mariscal.c@gmail.com
}
```

Accedemos a la interfaz web de Nagios y comprobamos que se ha añadido correctamente:

### Contacts

Contact Name	Alias	Email Address	Pager Address/Number	Service Notification Options	Host Notification Options	Service Notification Period	Host Notification Period	Service Notification Commands	Host Notification Commands	Retention Options
Alberto	alberto	<a href="mailto:alberto.mariscal.c@gmail.com">alberto.mariscal.c@gmail.com</a>		Unknown, Warning, Critical, Recovery, Flapping, Downtime	Down, Unreachable, Recovery, Flapping, Downtime	<a href="#">24x7</a>	<a href="#">24x7</a>	<a href="#">notify-service-by-email</a>	<a href="#">notificar-host</a>	Status Information Non-Status Information
root	Root	<a href="mailto:root@localhost">root@localhost</a>		Unknown, Warning, Critical, Recovery	Down, Recovery	<a href="#">24x7</a>	<a href="#">24x7</a>	<a href="#">notify-service-by-email</a>	<a href="#">notify-host-by-email</a>	Status Information Non-Status Information

## Configuración horarios de notificaciones

A la hora de monitorear un sistema, veremos que hay servicios cuya prioridad no es tan alta o que tiene posibilidad de recuperar sin ayuda, para estos casos nagios tiene diferentes periodos de configuración ya creados, pero nosotros podemos crear otros a nuestra medida.

Editamos el archivo "/etc/nagios3/conf.d/times\_periods\_nagios2.cfg":

En este primero nos enviarían las notificaciones 24 horas al día 7 días a la semana:

```
GNU nano 2.2.4 Fichero: timeperiods_nagios2.cfg
#####
# timeperiods.cfg
#####

# This defines a timeperiod where all times are valid for checks,
# notifications, etc. The classic "24x7" support nightmare. :-)

define timeperiod{
    timeperiod_name 24x7
    alias           24 Hours A Day, 7 Days A Week
    sunday          00:00-24:00
    monday          00:00-24:00
    tuesday         00:00-24:00
    wednesday       00:00-24:00
    thursday        00:00-24:00
    friday          00:00-24:00
    saturday        00:00-24:00
}
```

En este otro solo en un supuesto horario de oficina:

```
# The complement of workhours
define timeperiod{
    timeperiod_name nonworkhours
    alias           Non-Work Hours
    sunday          00:00-24:00
    monday          00:00-09:00,17:00-24:00
    tuesday         00:00-09:00,17:00-24:00
    wednesday       00:00-09:00,17:00-24:00
    thursday        00:00-09:00,17:00-24:00
    friday          00:00-09:00,17:00-24:00
    saturday        00:00-24:00
}
```

Para indicar en un servicio el timeperiod:

```
notification_period 24x7
```

## Configurar número de chequeos máximos antes de notificar

En principio Nagios está configurado para que se notifiquen las alarmas después de 4 chequeos fallidos o con alarmas.

Cuando estemos configurando cualquier tipo de servicio que vaya a ser monitoreado lo especificaremos así:

```
max_check_attempts 4
```

Donde el número son los chequeos máximos permitidos antes de mandar la notificación.

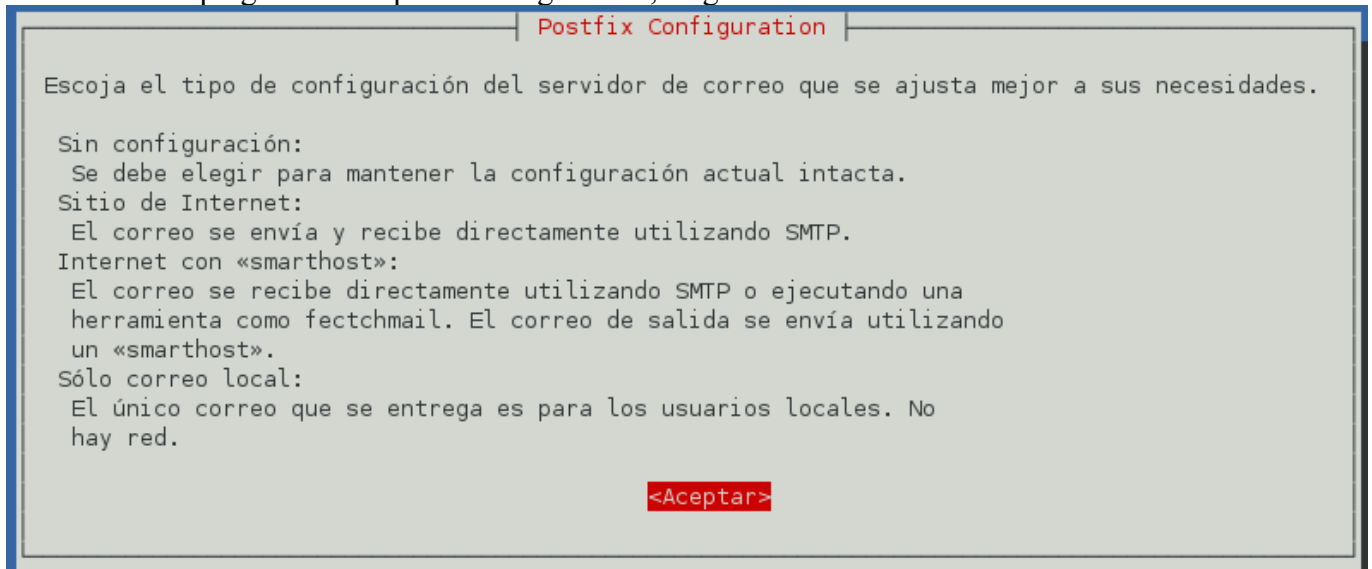


## Configuración de alertas a través de E-mail

Para poder enviar e-mail desde nuestro servidor necesitamos instalar un servidor de correos electronicos que se encargue de ello.

Vamos a instalar Postfix:

- #aptitude install postfix
- Nos preguntará el tipo de configuración, elegimos "Sitio de Internet".



- Comprobamos que el paquete se ha instalado correctamente:

```
root@yoda:/etc/nagios3# dpkg -l postfix
Deseado=Desconocido/Instalar/Eliminar/Purgar/Retener
| Estado=No/Instalado/Config-files/Desempaquetado/Medio-conf/Medio-inst/espera-disparo/pendiente-disparo
|/ Err?=(ninguno)/Requiere-reinst (Estado,Err: mayúsc.=malo)
||/ Nombre Versión Descripción
++- - - - -
ii postfix 2.7.1-1+squeezel High-performance mail transport agent
root@yoda:/etc/nagios3#
```

- Reiniciamos el servicio de postfix.

Una vez instalado y configurado nuestro servidor de email lo primero editamos el archivo "/etc/nagios3/commands.cfg" y añadimos el siguiente comando:

```
###Comando notificaciones grupo Administrator
define command{
command_name notificar-host
commandline /tmp/email.pl $CONTACTEMAIL$ $NOTIFICATIONTYPE$ $HOSTNAME$ $HOSTSTATE$ $HOSTADDRESS$
}
```



```

GNU nano 2.2.4          Fichero: email.pl

### Script envio de email
# Alberto A. Mariscal Casado
#!/usr/bin/perl

##Indicamos las variables con sus valores
$destinatario=$ARGV[0];
$tipo=$ARGV[1];
$host=$ARGV[2];
$estado=$ARGV[3];
$ip=$ARGV[4];
##Indicamos los datos del email.
$smtp= Net::SMTP-> new("smtp.ejemplo.es");
$smtp->mail("nagios@ejemplo.es");
$smtp->to("$destinatario");
$smtp->data();
$smtp->datasend("To: $destinatario\n");
$smtp->datasend("Subject: NAGIOS - $tipo: $host con estado $estado\n");
$smtp->datasend("Equipo: $host ($ip)\n");
$smtp->datasend("Estado actual: $estado\n");
$smtp->datasend();
$smtp->quit;

```

Como vemos hay un "/tmp/email.pl" que es un script en perl que vamos a realizar para el envio de email.

Reiniciamos el servicio de Nagios y comprobamos que no nos devuelve ningun erro.

Miramos tambien el log y vemos que el servicio se detiene y se activa de nuevo sin problemas.

```

[1322766799] Successfully shutdown... (PID=14010)
[1322766799] Nagios 3.2.1 starting... (PID=14215)
[1322766799] Local time is Thu Dec 01 20:13:19 CET 2011
[1322766799] LOG VERSION: 2.0
[1322766799] Finished daemonizing... (New PID=14216)
[1322767297] Caught SIGTERM, shutting down...
[1322767297] Successfully shutdown... (PID=14216)
[1322767297] Nagios 3.2.1 starting... (PID=14532)
[1322767297] Local time is Thu Dec 01 20:21:37 CET 2011
[1322767297] LOG VERSION: 2.0
[1322767297] Finished daemonizing... (New PID=14533)

```

- Comprobamos que las alarmas llegan al correo de los usuarios locales:

```

U 1 root@yoda          Sun Oct 16 01:26 31/1112 apt-listchanges: noticias de yoda
U 2 nagios@yoda        Sun Oct 16 02:04 30/682  ** PROBLEM Service Alert: localhost/SSH is CR
U 3 root@yoda          Sun Oct 30 13:16 18/591  Anacron job 'cron.daily' on yoda
U 4 nagios@yoda        Sun Oct 30 13:41 30/708  ** RECOVERY Service Alert: localhost/SSH is O
U 5 nagios@yoda        Sun Oct 30 14:46 25/646  ** PROBLEM Host Alert: anakin is DOWN **
U 6 nagios@yoda        Sun Oct 30 15:49 25/646  ** PROBLEM Host Alert: anakin is DOWN **
U 7 nagios@yoda        Sun Oct 30 17:56 25/653  ** PROBLEM Host Alert: anakin is DOWN **
U 8 nagios@yoda        Sun Oct 30 19:19 30/715  ** PROBLEM Service Alert: DarkVader/Version N
U 9 nagios@yoda        Sun Oct 30 19:33 30/705  ** PROBLEM Service Alert: DarkVader/Tiempo Ac
U 10 nagios@yoda       Sun Oct 30 19:45 30/694  ** PROBLEM Service Alert: DarkVader/Uso CPU i
U 11 nagios@yoda       Sun Oct 30 20:37 30/711  ** PROBLEM Service Alert: DarkVader/Explorer.
U 12 nagios@yoda       Sun Oct 30 20:37 30/704  ** PROBLEM Service Alert: DarkVader/Espacio e
U 13 root@yoda         Sun Nov 06 19:55 18/591  Anacron job 'cron.daily' on yoda
U 14 nagios@yoda       Sun Nov 06 20:00 25/652  ** PROBLEM Host Alert: Anakin is DOWN **
U 15 nagios@yoda       Sun Nov 06 20:31 25/652  ** PROBLEM Host Alert: Anakin is DOWN **
U 16 nagios@yoda       Sun Nov 06 20:52 30/695  ** PROBLEM Service Alert: obiwan/PING is CRIT
U 17 nagios@yoda       Sun Nov 06 21:02 25/652  ** PROBLEM Host Alert: Anakin is DOWN **
U 18 nagios@yoda       Sun Nov 06 21:15 25/661  ** PROBLEM Host Alert: Obi-Wan is DOWN **
U 19 nagios@yoda       Sun Nov 06 21:18 25/659  ** RECOVERY Host Alert: Obi-Wan is UP **
U 20 nagios@yoda       Sun Nov 06 21:21 30/692  ** RECOVERY Service Alert: obiwan/PING is OK

```

## Configuración de alertas a través de SMS

Para esta parte me dado de alta en Descom, una web que permite el envío de SMS y el cual podemos aprovechar para usarlo con Nagios, y en su propia web tenemos como hacerlo.

- Nos registramos y nos regalaran 10 créditos (sms).
- `#mkdir -p /usr/local/bin`
- `#wget -O /usr/local/bin/dcsms.pl http://www.descomsms.com/developer/dcsms.pl`
- `chmod+x /usr/local/bin/dcsms.pl`
- Hacemos una prueba de envío:

```
root@yoda:/usr/local/bin# dcsms.pl -i "187833" -u "iverson88" -p "48956668h" -m "665952210" -t "pr
ueba sms" -s "DescomSMS"
Ok: Mensaje Enviado (Saldo disponible: 9)
root@yoda:/usr/local/bin#
```



- Configuramos Nagios para que haga uso de este servicio:
  - Definimos los comandos en el archivo `"/etc/nagios3/commands.cfg"`

```
##Notificaciones por SMS
define command{
command_name notify-host-by-dcsms
command_line /usr/local/bin/dcsms.pl -i 187833 -u iverson88 -p 48956668h -m "$CONTACTPAGER$" -t "$NOTIFICATIONTYPE$"
}
define command{
command_name notify-service-by-dcsms
command_line /usr/bin/perl /usr/local/bin/dcsms.pl -i 187833 -u iverson88 -p 48956668h -m "$NOTIFICATIONTYPE$: $HOSTNAME$"
}
```

- Reiniciamos el servicio y comprobamos que se envían:

**December 01, 2011**  
**23:00**

```
[2011-12-01 23:37:43] EXTERNAL COMMAND: ENABLE_HOST_SVC_NOTIFICATIONS;DarkMouth
[2011-12-01 23:37:39] EXTERNAL COMMAND: ENABLE_HOST_SVC_CHECKS;DarkMouth
[2011-12-01 23:37:38] HOST NOTIFICATION: Alberto;DarkMouth;DOWN;notify-host-by-email;CRITICAL - Host Unreachable (192.168.1.141)
[2011-12-01 23:37:38] HOST NOTIFICATION: Alberto;DarkMouth;DOWN;notify-host-by-dsmsg;CRITICAL - Host Unreachable (192.168.1.141)
[2011-12-01 23:37:34] EXTERNAL COMMAND: SCHEDULE_FORCED_HOST_CHECK;DarkMouth;1322779052
```

Por problemas de créditos (es de pago) no tengo capturas de mensaje recibido en el móvil.

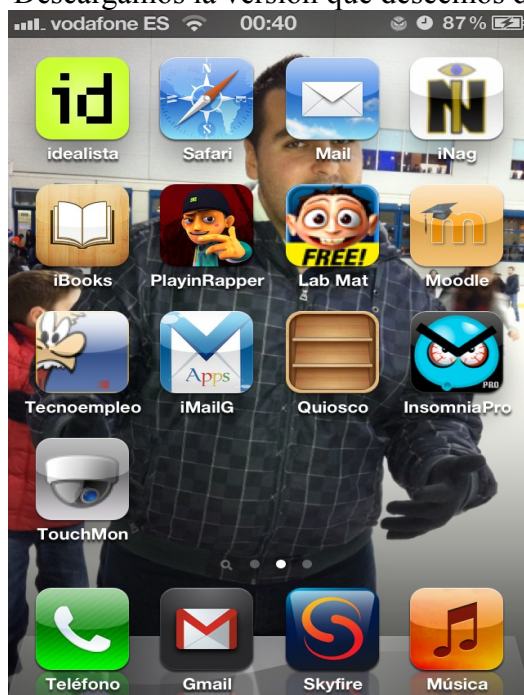
## **Touchmon, acceso a Nagios desde tu iDevice**

Con esta herramienta, podemos acceder a toda la información que nos reporta Nagios, todo desde nuestro iDevice (para Android disponemos de otras herramientas).

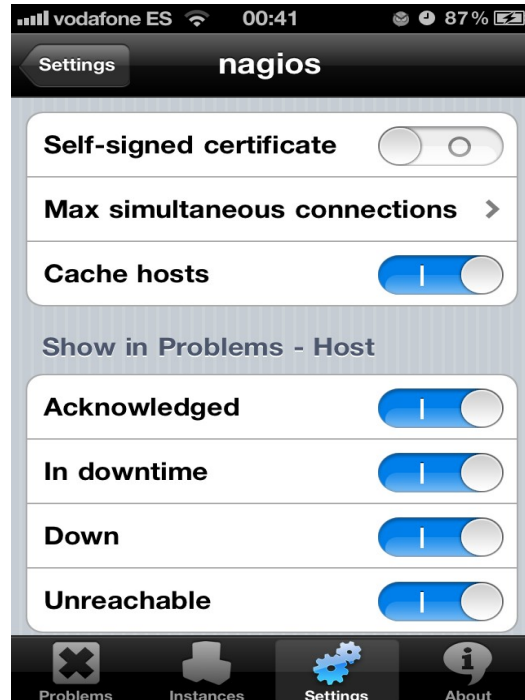
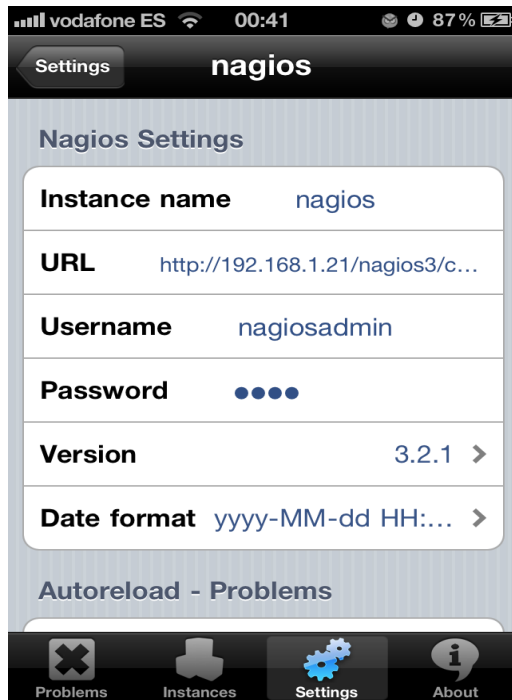
Esta herramienta tiene versiones gratuitas y de pago, podemos probar la gratis desde la "appstore" o probar la de pago desde "Installous"...

En mi caso he probado la versión completa, en un Iphone 4G sobre iOS 5.0.1 con jailbreak tethered.

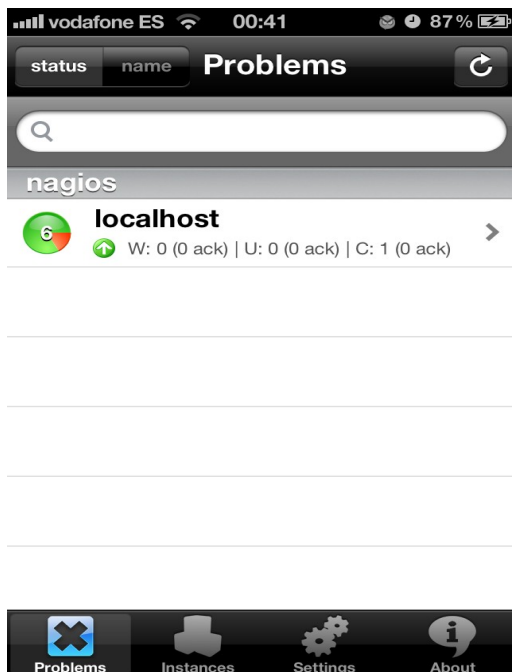
- Descargamos la versión que deseemos de "touchmon".

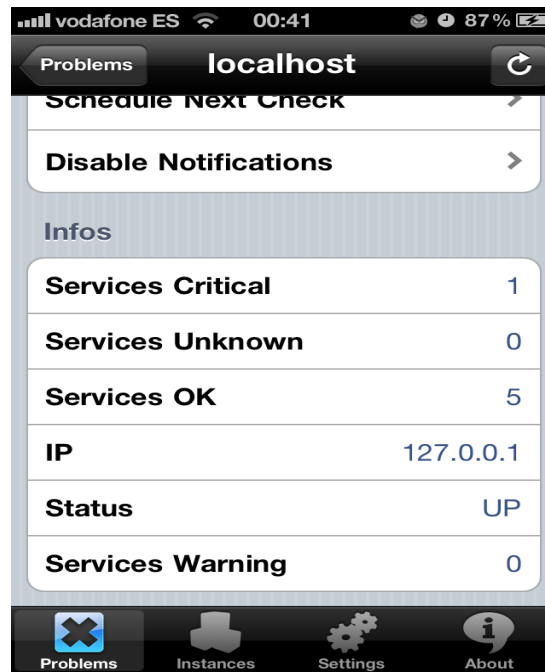
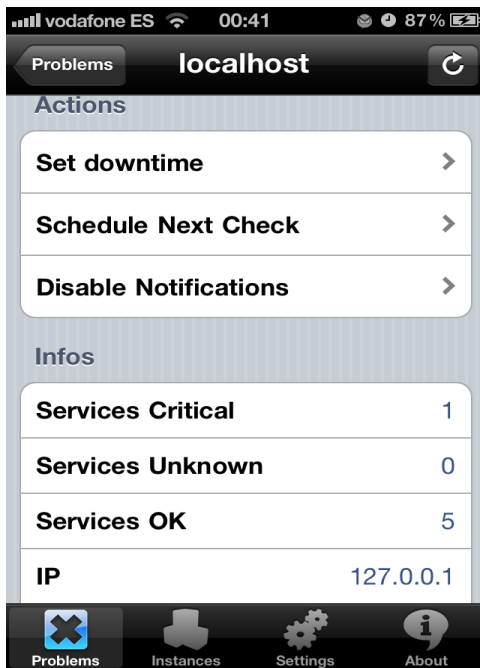


- Lo primero es ir a "Settings". Muy importante poner en la URL la dirección desde que accedemos desde cualquier ordenador con acceso a Nagios3 pero añadiendo un dato quedando "<http://192.168.1.21/nagios3/cgi-bin>" en mi caso, terminamos introduciendo el login normal para acceder a Nagios y la versión de nuestro servidor.

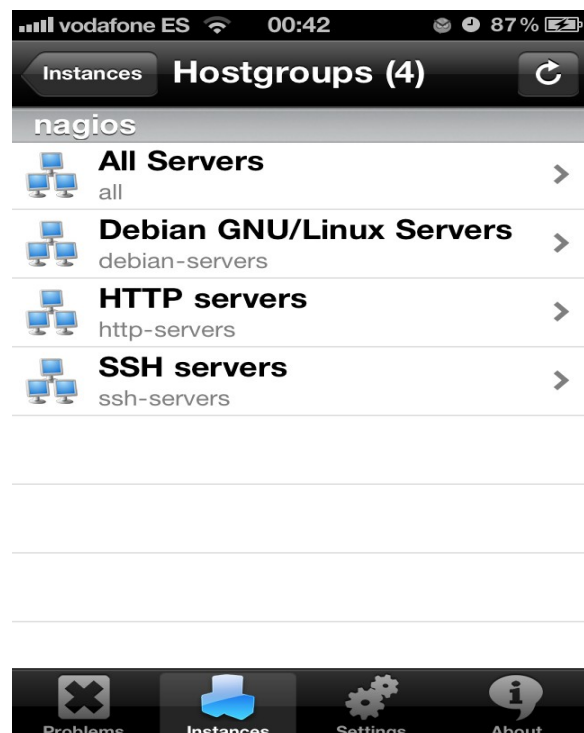


- En "Problems" podemos ver los dispositivos que presentan alguna alarma:





- En "Instances" podemos ver los dispositivos monitoreados, individualmente o en grupos:



## **Problemas encontrados en el desarrollo del PI**

- Cliente de Nagios para Windows, aleatoriamente el proceso se colgaba. Se solucionó instalando una versión diferente del cliente.
- Lector de CD en VM de Mac OS X, hay que poner el disco duro y el lector en el mismo IDE, para que lo detecte, este detalle no lo he encontrado por google.
- Poca información del Nagios empaquetado por los desarrolladores de Debian, lo que dificulta el encontrar soluciones a los problemas que van saliendo.

## **Experiencia con Nagios**

Después de este proyecto en el cual he intentado conocer en profundidad como administrar Nagios, puedo decir que es un sistema algo complicado de usar aunque no lo parezca.

Cuando sigues un "How to" parece sencillo pero cuando empezamos a incluir máquinas y servicios a monitorizar nos damos cuenta que no es en realidad como ponen en esos "How to" de sencillo, ya que salen bastantes errores y problemas, y la dificultad de averiguar si es en el servidor o en el cliente el cual nos muestra su estado.

No podemos pedir que rinda y sea sencillo como otro con los que he trabajado pero que son privativos y/o de pago como GFI Remote Management, el cual con instalar un simple cliente nos aparece directamente en el servidor y podemos acceder al instante al cliente por que lleva integrado el software de control remoto Team Viewer.

## **Conclusión después del PI**

Aunque he trabajado monitorizando servidores y equipos de escritorio y actualmente estoy monitorizando estaciones de telefonía 2G y 3G en Vodafone, gracias a este proyecto he aprendido como funciona realmente, como el servidor pide/recibe datos a los clientes. También cabe destacar lo que puede enseñar un proyecto como este referente al protocolo SNMP.

Y bueno aunque no es el centro del proyecto, también he aprendido un poco sobre otros sistemas operativos como son Mac OS X o la distribución Linux CentOS.

## **Agradecimientos**

Gracias a el compañero "ballstud" (Eduardo) de [www.elotrolado.net](http://www.elotrolado.net) por su ayuda con aclaraciones sobre el Unix de Mac OS X y sus compiladores de C.

Gracias al compañero Francisco C. del departamento de redes de Vodafone España por las explicaciones del protocolo SNMP.

Gracias a Jose Manuel Ferrete por su apoyo durante el periodo de realización de este proyecto cuando convivíamos en Madrid.

Gracias a los profesores Alberto Molina, Raul Ruiz, Juan Tagua, Jesús Moreno, Jose Domingo y Manuel Tienda del CFGS ASI del IES Gonzalo Nazareno por lo que nos han enseñado durante estos años.