

Miguel Casares Martínez

Sistemas de monitoreo de redes e instalación y configuración de Nagios



2^a ASIR-A Administración de sistemas informáticos en red

I.E.S ZAIDÍN VERGELES – GRANADA

Tutor: Miguel Ángel Moreno Garrido

ÍNDICE

Introducción	3
Implantación de un sistema de monitoreo en una empresa	3
Explicación de las herramientas de monitoreo de redes	4
Explicación de Nagios	7
Instalación de Nagios	8
Monitoreo de equipos Windows y Linux	12
Monitoreo de router o switch	18
Creación de grupos de servicios y de hosts	20
Interfaz web de Nagios	22
Edición del tiempo de monitoreo y de reintento de un host o servicio	22
Como activar las notificaciones por correo de Nagios	24
Conclusiones	32
Bibliografía	33

-Introducción:

El objetivo de este proyecto es adentrarnos en el monitoreo de redes y en su implementación en una empresa, ver los programas más importantes, además de llevar a la práctica la instalación y la configuración de un sistema con uno de los programas más importantes como es Nagios.

El monitoreo de redes consiste en utilizar herramientas de software y hardware para hacer un seguimiento constante de los datos de rendimiento, funcionamiento y actividad. Con esto se pueden encontrar problemas causados por sobrecarga o fallos en los servidores. También nos ayuda a prevenir problemas futuros y reducir el tiempo de solución de estos.

PALABRAS CLAVE: Nagios,monitoreo de redes, datos, administración y configuración.

-Implantación de un sistema de monitoreo de redes en una empresa:

En la mayoría de empresas de tamaño mediano(unos 200 empleados), no cuentan con un sistema de monitoreo de redes.

Con este sistema, además de las ventajas que hemos tratado en la introducción, nos ayuda a mantener actualizado el sistema, ya que al tener todo monitorizado, veremos si nos hace falta más ancho de banda de red o que sistemas fallan más y se deberían sustituir. Se conseguirá tener un inventario de todos los equipos, servidores y routers que tenemos, además de la estructura de red de la empresa. También se incrementará la seguridad, ya que en algunos casos tenemos la posibilidad de implementar backups y prevenir el acceso de hackers maliciosos.

Los datos que nos aporta el sistema también nos servirán para controlar el crecimiento de nuestra empresa, por ejemplo los datos de uso.

Hay ciertos pasos que debemos seguir al implantar una herramienta de monitoreo:

-Comparación de herramientas: Cada herramienta es diferente, no hay una que sea la mejor, debemos analizar el tamaño de nuestra empresa y que tipo de uso le vamos a dar para elegir una u otra.

-Análisis de costes: Aunque en un principio, implantar un sistema de monitoreo de redes, nos va a producir un coste económico, a la larga es rentable y se consiguen ahorro de costes y de tiempo, al reducir tiempo buscando los fallos y estar más optimizado el sistema

-Requisitos del personal: Obviamente manejo de la herramienta que vayamos a usar, interpretación correcta de todos los datos a los que dispondremos, solución fácil de los problemas que se nos plantean en el sistema y donde se originan.

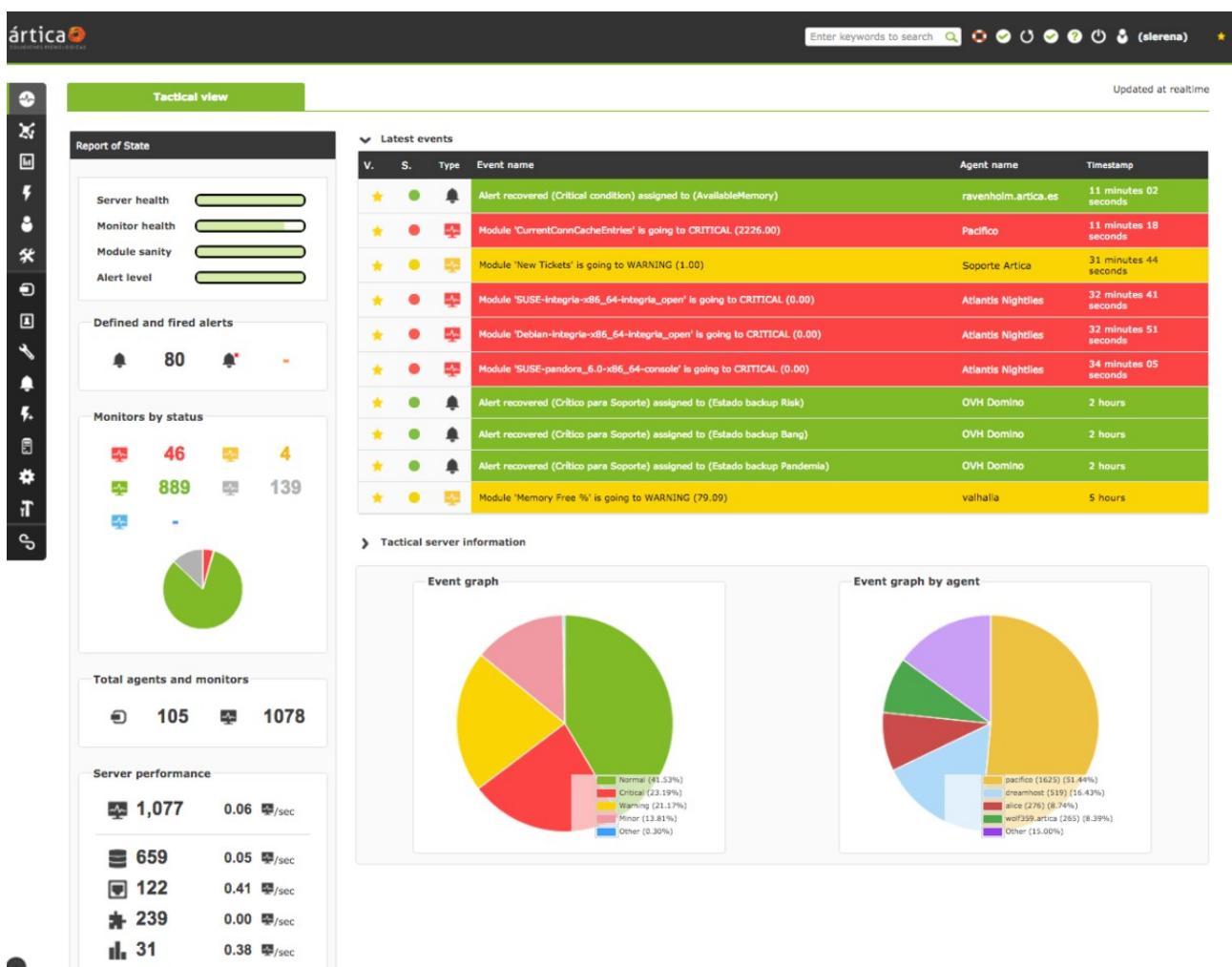
-Análisis de resultados: Tras el paso de un periodo de tiempo con la herramienta implementada, debemos estudiar los beneficios que nos ha originado ,si es viable mantenerla en la empresa, si decidimos usar otra herramienta o nos vale la pena llevar la herramienta a un nivel mejor, como por ejemplo pasar de Nagios Core a Nagios XI .

-Explicación de las herramientas de monitoreo de redes:

-PandoraFMS:

Es un software de monitorización desarrollado por PandoraFMS Enterprise que tiene una versión free software, esta monitoriza hasta 10000 nodos y monitoriza red, servidores y aplicaciones. Algunas de sus características son que utiliza el protocolo ICMP, aplica perfiles de monitoreo de forma automática. Otra característica interesante es que es la de poder incorporar herramientas de código libre para ayudar a gestionar la información, estas herramientas se pueden incorporar mediante servidores, para hacer las tareas de monitoreo. Además de la detección automática de la topología de red.

Como ventajas podemos decir que tiene una administración simple mediante interfaz web, como la mayoría de herramientas, posibilidad de personalización y el uso de la misma herramienta para diferentes entornos.



-Zabbix:

Es un sistema de monitorización de redes de código libre, desarrollado por Zabbix SIA y que fue creado por Alexei Vladishev. Mediante una interfaz basada en web y sencilla de usar que permite la administración remota de la red. Permite un monitoreo extenso, permite a los usuarios configurar alertas para todo lo que ocurra, también ofrece funciones relacionadas con la visualización y capacidad de los datos.

Como ventaja podemos decir que tiene una comunidad activa y es potente a bajo nivel y como desventaja que disminuye su rendimiento a partir de 1000 nodos y no tiene informes en tiempo real.

Algunas otras de sus características son:

- Configuración flexible
- Ejecución de comandos remotos
- Funciona con casi todos los S.O basados en Unix, ya sea AIX, Linux, FreeBSD, Solaris o FreeBSD
- Agentes nativos para sus versiones de Windows y de estos sistemas operativos.
- Gráficos integrados
- Configuración de permisos por usuarios y grupos

The screenshot shows the Zabbix web interface with the following sections:

- Favourite maps:** Local network (Maps), New host CPU load (Graphs), Zabbix server (Screens).
- Last 20 issues:**

HOST	ISSUE	LAST CHANGE	AGE	INFO	ACK	ACTIONS
New host	CPU load too high on 'New host' for two minutes	2016-02-12 08:50:19	22s	No	1	
New host	New host has just been restarted	2016-02-12 08:47:59	2m 42s	No	1	
Zabbix server 1	Zabbix server 1 has just been restarted	2016-02-12 08:46:31	4m 10s	No	1	
Zabbix server 1	Lack of free swap space on Zabbix server 1	2015-08-11 23:29:28	6m 4d 10h	Yes	4	

4 of 4 issues are shown Updated: 08:50:41
- Status of Zabbix:**

PARAMETER	VALUE	DETAILS
Zabbix server is running	Yes	localhost:10051
Number of hosts (enabled/disabled/templates)	54	10 / 1 / 43
Number of items (enabled/disabled/not supported)	356	350 / 0 / 6
Number of triggers (enabled/disabled [problem/ok])	95	94 / 1 [4 / 90]
Number of users (online)	3	2
- System status:**

HOST GROUP	DISASTER	HIGH	AVERAGE	WARNING	INFORMATION	NOT CLASSIFIED
Clouds	0	0	0	0	0	0
Database servers	0	0	0	0	0	0
Discovered hosts	0	0	0	1	1	0
JB applications	0	0	0	0	0	0
Linux servers	0	1	0	0	1	0
Network devices	0	0	0	0	0	0
SNMP hosts	0	0	0	0	0	0
Virtual machines	0	0	0	0	0	0
Web servers	0	0	0	0	0	0
Windows servers	0	0	0	0	0	0
Zabbix servers	0	0	0	1	1	0

Updated: 08:50:40
- Discovery status:**

DISCOVERY RULE	UP	DOWN
Local network2	19	1

Updated: 08:50:39
- Web monitoring:**

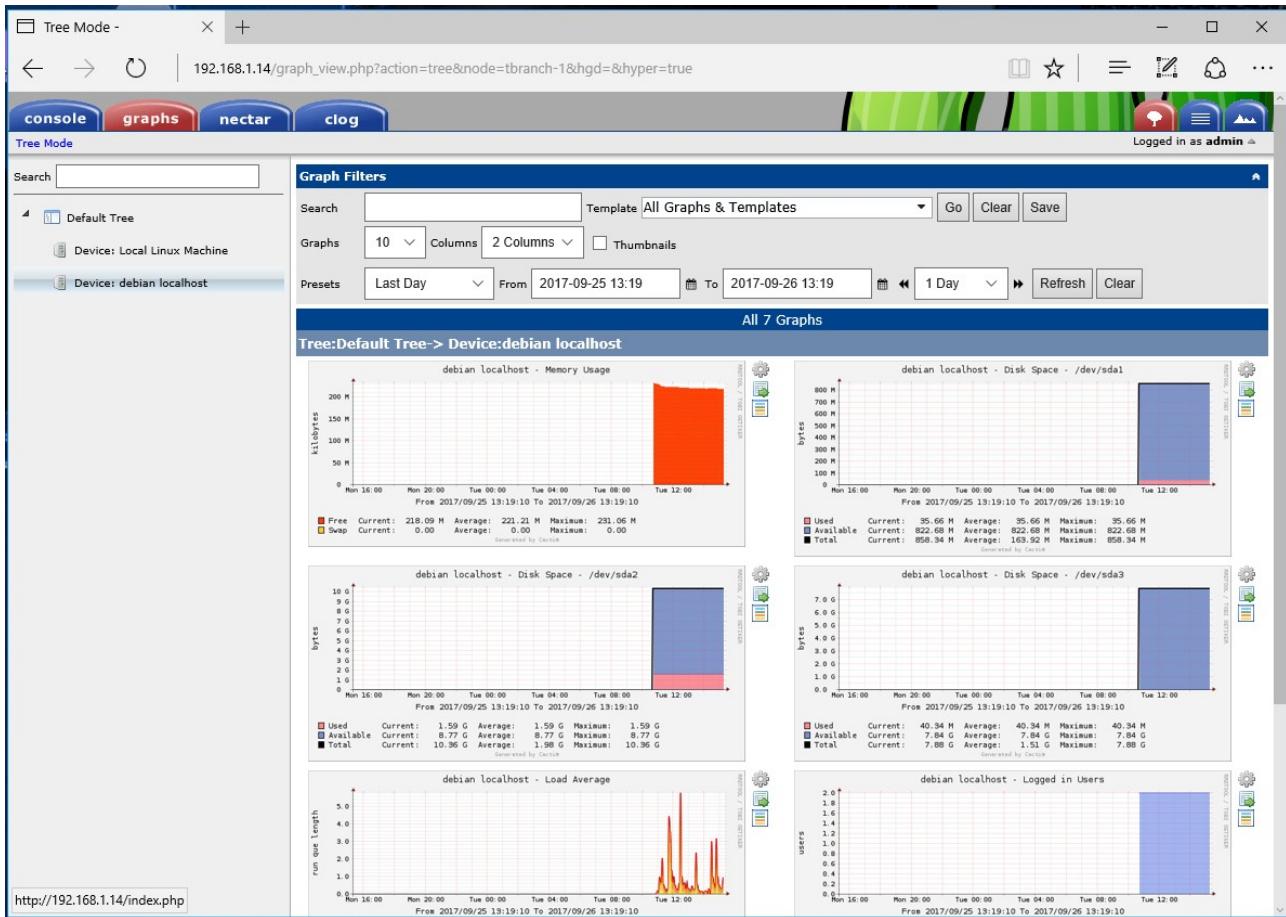
HOST GROUP	OK	FAILED	UNKNOWN
Discovered hosts	1	0	0
Zabbix servers	1	0	0

Updated: 08:50:40
- Host status:**

-Cacti:

Es un sistema open source desarrollado en PHP. Incluye monitoreo en tiempo real. Utiliza el protocolo SNMP, cuenta con una consola de administración.

Una ventaja de este sistema es la fácil configuración de esta consola de administración. Una desventaja de este sistema es la poca claridad de sus gráficas y la imposibilidad de incluir en una misma configuración a varios equipos.



Hay más herramientas de monitoreo menos conocidas como puede ser Whatsup Gold, cuyo punto fuerte es a la hora de balancear la carga, pero no es Open Software; PRTG Network Monitor, el punto fuerte de esta herramienta es su fácil manejo, tiene una versión OpenSoftware pero es muy limitada.

Conviene nombrar también las herramientas de Manage Engine, Observium y Op5 Monitor.

-Explicación de Nagios:

Es un sistema de monitorización de redes muy utilizado de software libre que fue escrito por Ethan Galstad, es muy completo y complejo. Está creado en lenguaje C y contiene también algo de php y trabaja en sistemas GNU/Linux y algunos tipos de Unix.

Puedes monitorizar servicios de red como POP3, SMTP, DNS, HTTP, SSH o HTTPS

Con este sistema se pueden monitorizar servidores, teléfonos IP, switch, router, PCs e impresoras.

Una característica muy importante es que se pueden desarrollar plugins en diferentes lenguajes para que los usuarios puedan realizar diferentes tareas ya sea para monitorizar servicios como HTTP, POP3, IMAP, FTP, SSH, DHCP, ver información como el uso de la CPU, los usuarios activos y los recursos del sistema que están en uso o como archivos de registro. Otro tipo de plugins son los desarrollados por la comunidad nosotros vamos a poner en práctica la opción de desarrollar uno de estos plugins.

Según opiniones de los usuarios al principio, la configuración es un poco compleja pero una vez que te adaptas a ella llegas a dominarla. Tiene varias versiones dependiendo de la monitorización que queramos hacer, más compleja o más sencilla, la básica, que se llama Nagios Core y es la que vamos a instalar, es gratuita.

Otras versiones son :

-Nagios XI que contiene una prueba gratuita de 30 días y según pone en la descripción de su página “le permite acceder a todas las funciones diseñadas para implementaciones a escala empresarial”

-Nagios Log Server está orientado a los datos, ya sea recopilarlos, analizarlos y almacenarlos también permite según su propia descripción “ver, ordenar y configurar registros de forma rápida y sencilla desde cualquier origen de cualquier red determinada”.

-Nagios Fusion está orientado a resolver problemas de múltiples redes y con separación geográfica. Permite la centralización y visualizar servidores Nagios XI y Core

Página web oficial de Nagios:[Nagios - The Industry Standard In IT Infrastructure Monitoring.](http://www.nagios.org)

-Instalación de Nagios Core:

Una vez explicada la herramienta pasamos a su instalación, requiere de aprendizaje y las primeras veces puede ser un poco tedioso.

Requerimientos:

- Un GNU-LINUX y compilador de C
- Apache
- Librerías gráficas GD, JPEG y PNG
- MySQL (Opcional)

Paso a paso:

Primero de todo instalamos apache y los requerimientos que falten. Nos vamos a la pagina web de Nagios y en la opción Nagios Core descargamos la última versión y ponemos este comando para la configuración inicial:

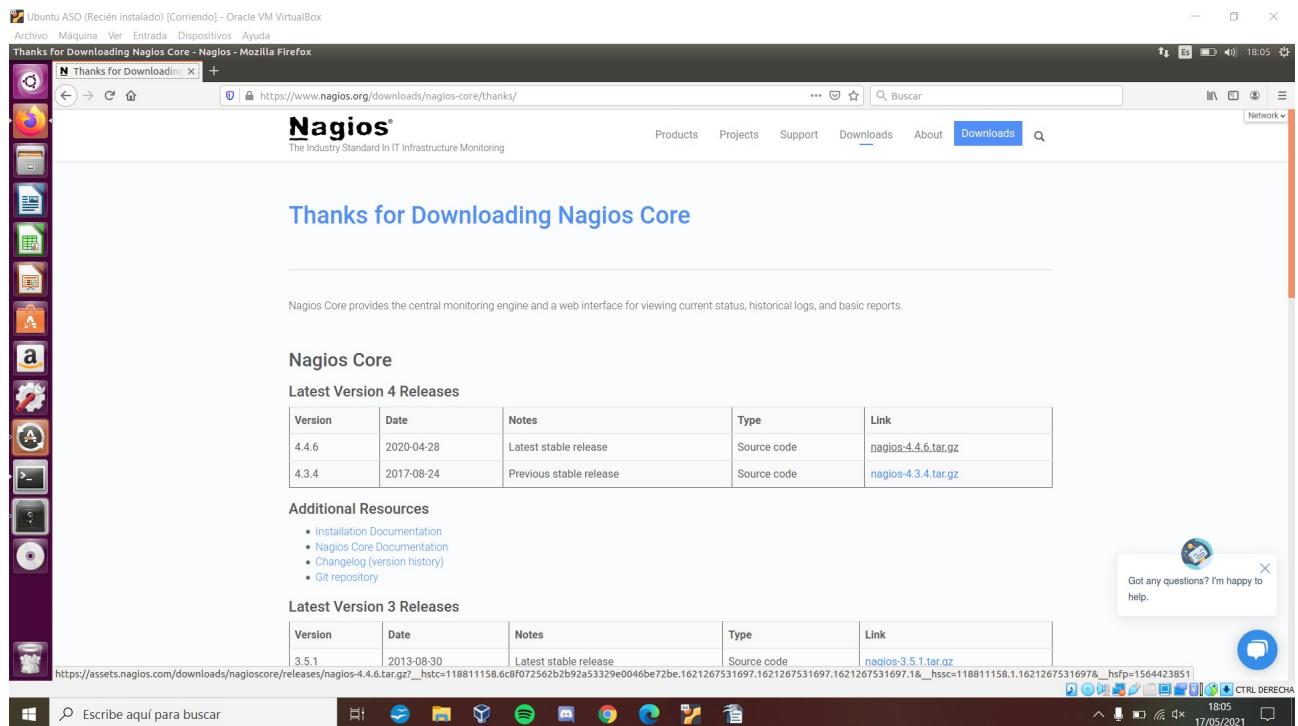
`./configure --prefix=/usr/local/nagios --with-cgiurl=/nagios/cgi-bin --with-htmurl=/nagios/ --with-nagios-user=nagios --with-nagios-group=nagios --with-command-group=nagios`

Conviene crear un usuario y un grupo para Nagios para que se opere de forma separada de los demás. Después tendremos que habilitar el sitio en el apache con a2ensite

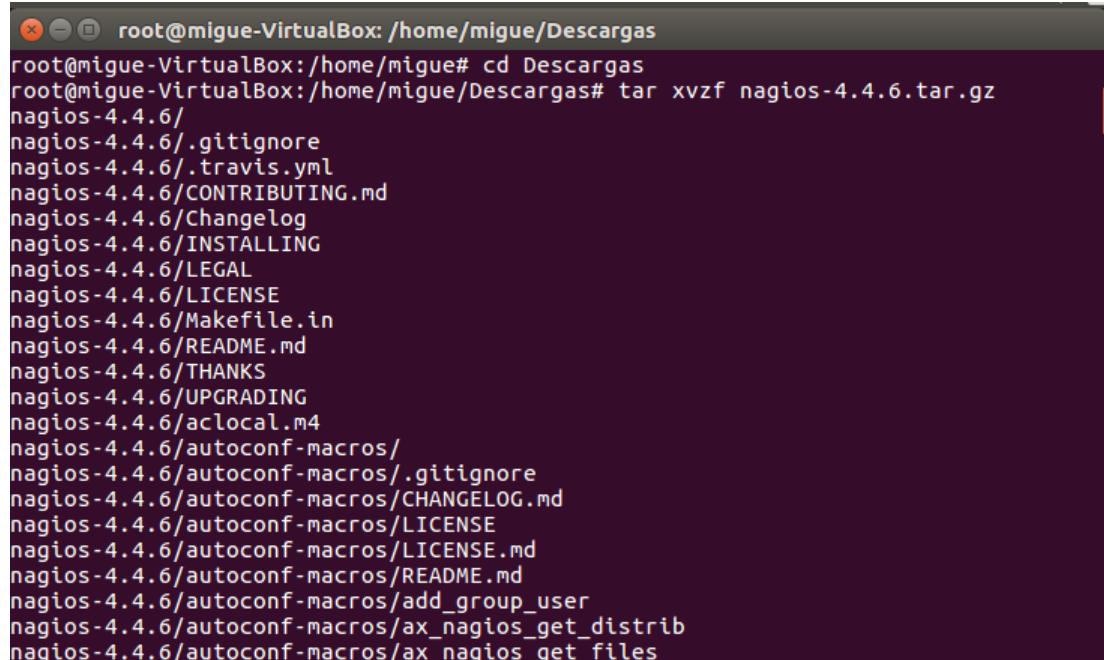
Una vez instalado para iniciararlo: `/etc/init.d/nagios start`

Y para detenelo: `/etc/init.d/nagios stop`

Esta es la página que nos encontramos al instalar Nagios Core:

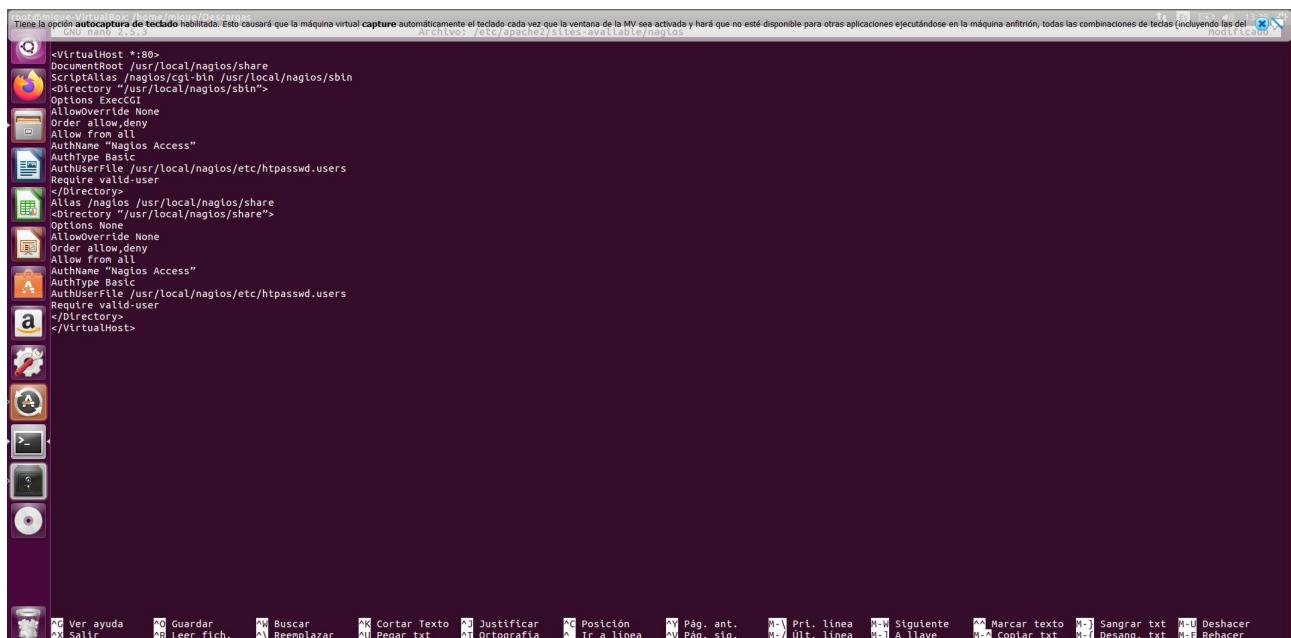


Una vez descargado, seguimos los pasos que vienen en la guía de Nagios Core, básicamente hay que descomprimir e instalar cada paquete. Una vez hecho esto, habilitar en el apache el sitio Nagios con el comando a2ensite.



```
root@migue-VirtualBox: /home/migue/Descargas
root@migue-VirtualBox:/home/migue# cd Descargas
root@migue-VirtualBox:/home/migue/Desktop# tar xvzf nagios-4.4.6.tar.gz
nagios-4.4.6/
nagios-4.4.6/.gitignore
nagios-4.4.6/.travis.yml
nagios-4.4.6/CONTRIBUTING.md
nagios-4.4.6/Changelog
nagios-4.4.6/INSTALLING
nagios-4.4.6/LEGAL
nagios-4.4.6/LICENSE
nagios-4.4.6/Makefile.in
nagios-4.4.6/README.md
nagios-4.4.6/THANKS
nagios-4.4.6/UPGRADING
nagios-4.4.6/aclocal.m4
nagios-4.4.6/autoconf-macros/
nagios-4.4.6/autoconf-macros/.gitignore
nagios-4.4.6/autoconf-macros/CHANGELOG.md
nagios-4.4.6/autoconf-macros/LICENSE
nagios-4.4.6/autoconf-macros/LICENSE.md
nagios-4.4.6/autoconf-macros/README.md
nagios-4.4.6/autoconf-macros/add_group_user
nagios-4.4.6/autoconf-macros/ax_nagios_get_distrib
nagios-4.4.6/autoconf-macros/ax_nagios_get_files
```

Este es el archivo de el sitio de Nagios Core en apache2:

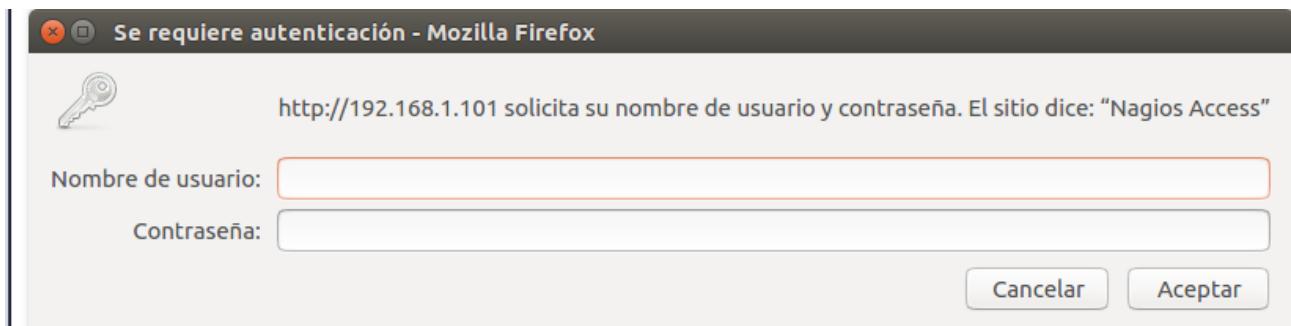


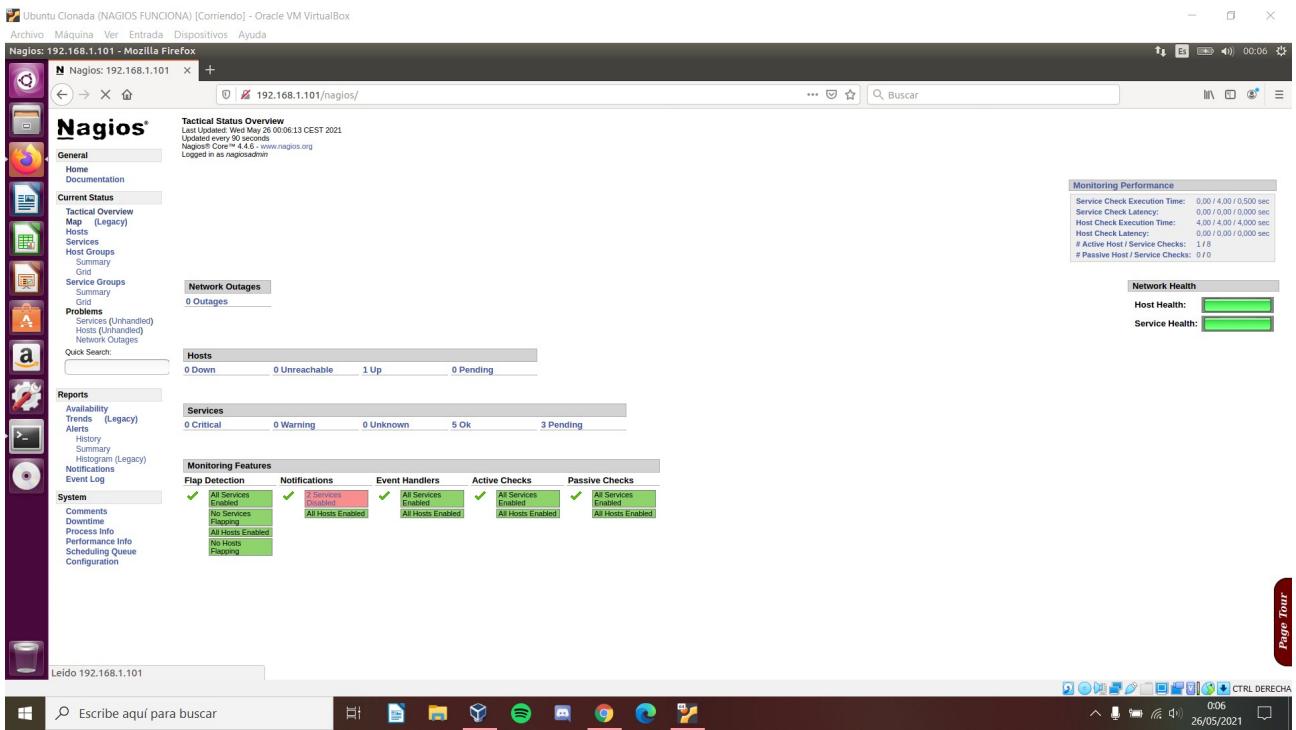
También hay que instalar los plugins que nos ayudarán a que Nagios funcione correctamente y nos añadirán características, en la captura lo descomprimimos y comprobamos que se han instalado correctamente.

```
Ubuntu ASO (Recién instalado) [Corriendo] - Oracle VM VirtualBox
Archivo Máquina Ver Entrada Dispositivos Ayuda
root@miguel-VirtualBox:/home/miguel/Descargas/nagios-plugins-2.0#
root@miguel-VirtualBox:/home/miguel/Descargas/nagios-plugins-2.0# ./install-data-all
test -z "$USR/local/nagios/lbexec" || /bin/mkdir -p "$USR/local/nagios/lbexec"
/usr/bin/install -c -m 755 check_breeze check_disk_smb check_flexlm check_ircd check_log check_oracle check_rpc check_sensors check_wave check_ifstatus check_ifoperstatus check_mailq check_file_age utils.sh util
ls -l $USR/local/nagios/lbexec
make[2]: se sale del directorio '/home/miguel/Descargas/nagios-plugins-2.0/plugins-scripts'
make[1]: se sale del directorio '/home/miguel/Descargas/nagios-plugins-2.0/plugins-scripts'
Making install in plugins-root
make[2]: se entra en el directorio '/home/miguel/Descargas/nagios-plugins-2.0/plugins-root'
make[2]: se entra en el directorio '/home/miguel/Descargas/nagios-plugins-2.0/plugins-root'
/usr/bin/install -c -m 755 check_dhcp /usr/local/nagios/lbexec/check_dhcp
chown root /usr/local/nagios/lbexec/check_dhcp
chmod ug+x,us $USR/local/nagios/lbexec/check_dhcp
/usr/bin/install -c -m 755 check_icmp /usr/local/nagios/lbexec/check_icmp
chown root /usr/local/nagios/lbexec/check_icmp
chmod ug+x,us $USR/local/nagios/lbexec/check_icmp
chown root /usr/local/nagios/lbexec/check_ncp
make[2]: No se hace nada para 'install-data-all'.
make[2]: se sale del directorio '/home/miguel/Descargas/nagios-plugins-2.0/plugins-root'
make[1]: se sale del directorio '/home/miguel/Descargas/nagios-plugins-2.0/plugins-root'
Making install in po
make[2]: se entra en el directorio '/home/miguel/Descargas/nagios-plugins-2.0/po'
/bin/mkdir -p $USR/local/nagios/share
installing fr.gmo as /$USR/local/nagios/share/locale/fr/LC_MESSAGES/nagios-plugins.mo
installing de.gmo as /$USR/local/nagios/share/locale/de/LC_MESSAGES/nagios-plugins.mo
if test "nagios-plugins" = "gettext-tools"; then \
    /usr/bin/install -c -m 755 ./po/file /$USR/local/nagios/share/gettext/po/file; \
    for file in Makefile.in Remove-potcde.sed Makevars.template; do \
        /usr/bin/install -c -m 644 ./file $USR/local/nagios/share/gettext/po/$file; \
done; \
for file in Makevars; do \
    rm -f $USR/local/nagios/share/gettext/po/$file; \
done; \
else \
    :; \
fi
make[1]: se sale del directorio '/home/miguel/Descargas/nagios-plugins-2.0/po'
make[1]: se entra en el directorio '/home/miguel/Descargas/nagios-plugins-2.0'
make[2]: se entra en el directorio '/home/miguel/Descargas/nagios-plugins-2.0'
make[2]: No se hace nada para 'install-exec-all'.
make[2]: No se hace nada para 'install-data-all'.
make[1]: se sale del directorio '/home/miguel/Descargas/nagios-plugins-2.0'
make[1]: se sale del directorio '/home/miguel/Descargas/nagios-plugins-2.0'
root@miguel-VirtualBox:/home/miguel/Descargas/nagios-plugins-2.0# ls $USR/local/nagios/lbexec
check_apt      check_dummy      check_ircd      check_ntp_peer      check_sensors      check_wave
check_breeze   check_file_age  check_load     check_ntp_time     check_sntp      negate
check_by_ssh   check_flexlm   check_log     check_nvtstat     check_svn      urtize
check_cron     check_flexnet  check_mailq   check_ntpmap     check_tftp     utilip.pm
check_cluster  check_http    check_mrtg    check_overcr    check_tcp      utilis.sh
check_dhcp     check_icmp    check_mrtgraf  check_ping     check_time
check_dig      check_tde_smart check_nagios  check_pop      check_udp
check_disk     check_ifoperstatus check_ntp    check_procs    check_ups
check_email    check_ntpstatus check_ntpmap  check_shm      check_uptime
check_fds     check_smb      check_overcr  check_tcp      check_users
check_dns     check_imap     check_ntpmap  check_procs    check_uptime
root@miguel-VirtualBox:/home/miguel/Descargas/nagios-plugins-2.0#
```

Con esto chequeamos que están todos los ficheros y cada uno con sus permisos. En el caso de que tengamos algún error ,con el comando systemctl status nagios podremos ver si funciona el servicio, y con el comando journalctl -xe podremos ver en que fichero e incluso en que línea está el error. Esto también nos servirá al añadir equipos para monitorizar si nos salen errores en los ficheros que veremos ahora.

Una vez finalizada la instalación nos vamos a la IP de nuestro equipo, ponemos la contraseña que hemos establecido anteriormente y tendremos la interfaz web de Nagios:





Con la interfaz web de nagios podemos hacer varias cosas remotamente:

- Resumen general del sistema
- Acceder a la documentación online
- Ver el estado actual de los servicios y hosts
- Visualizar mapas de estado y 3D
- Resumen de los problemas
- Añadir comentarios a las situaciones, hosts y servicios
- Añadiendo equipos windows y linux a nuestro Nagios:**

Para añadir un equipo a nuestro sistema Nagios tendremos que instalar un agente en el equipo que queremos monitorizar, como puede ser NSClient++, que es el más usado, u otros como NRPE.

0.5.2.35

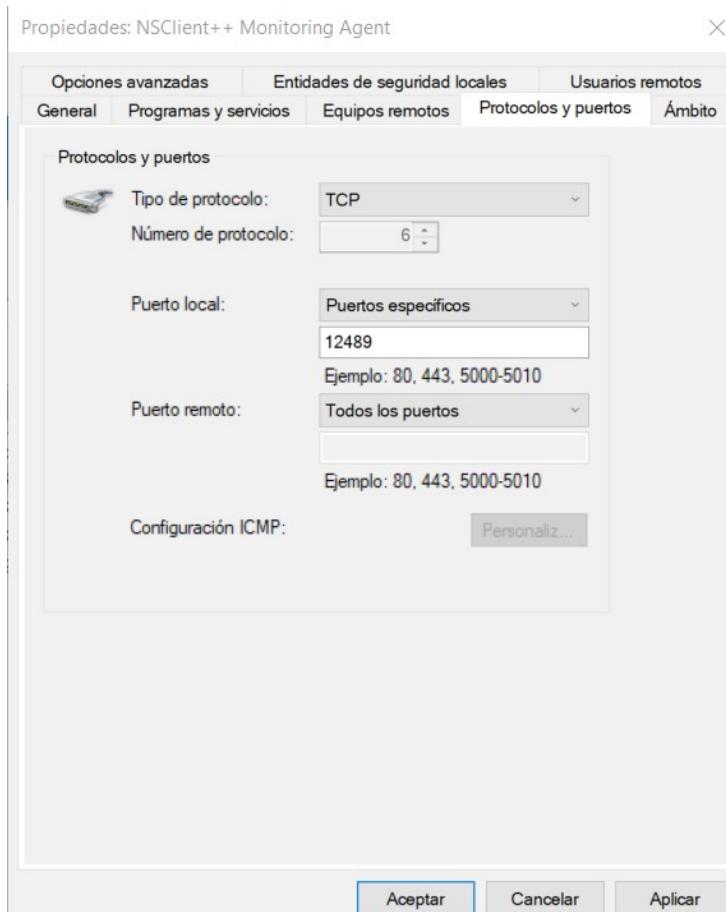
See [github](#) for details

Feel free to [browse other 0.5.2 releases](#)

Windows Windows XP and beyond 0.5.2.35 x64 Win32	Linux Redhat/Centos 0.5.2.35 EL6 x64	Source Code zip 0.5.2.35 zip tgz
---	--	---

La instalación del programa NSClient es sencilla, solo necesitaremos en un momento de la instalación poner la IP del servidor donde está Nagios Core.

También tendremos que editar el archivo de configuración de NSClient para habilitar cada servicio(cambiar disabled por enabled) y abrirle el puerto 12489 en el cortafuegos que es el que usa para comunicarse con el servidor.



```

nsclient: Bloc de notas
Archivo Edición Formato Ver Ayuda

; Undocumented key
allowed hosts = 192.168.1.101

; in flight - TODO
[/settings/NRPE/server]

; Undocumented key
verify mode = none

; Undocumented key
insecure = true

; in flight - TODO
[/modules]

; Undocumented key
CheckExternalScripts = disabled

; Undocumented key
CheckHelpers = disabled

; Undocumented key
CheckEventLog = disabled

; Undocumented key
CheckNSCP = disabled

; Undocumented key
CheckDisk = disabled

; Undocumented key
CheckSystem = disabled

; Undocumented key
NSClientServer = enabled

; Undocumented key
NRPEServer = enabled

```

Una vez hecho esto nos tocará, en nuestro servidor linux, y la carpeta “/usr/local/nagios/etc/objects/”, editar el archivo windows.cfg donde añadiremos el nombre, alias e IP del equipo a monitorizar en el parámetro define-host y cambiaremos simplemente el parámetro host_name para cada servicio.

```
GNU nano 2.5.3                               Archivo: windows.cfg                               Modificado

}

# Create a service for monitoring memory usage
# Change the host_name to match the name of the host you defined above

define service {

    use          generic-service
    host_name    win10
    service_description Memory Usage
    check_command check_nt!MEMUSE!-w 80 -c 90
}

# Create a service for monitoring C:\ disk usage
# Change the host_name to match the name of the host you defined above

^G Ver ayuda ^O Guardar      ^W Buscar      ^K Cortar Texto ^J Justificar ^C Posición
^X Salir      ^R Leer fich.  ^\ Reemplazar   ^U Pegar txt   ^T Ortografía ^_ Ir a línea
```

Una vez editado este archivo, en el archivo nagios.cfg, donde descomentaremos la linea del archivo windows.cfg. También es posible ignorar el archivo que viene por defecto de windows, crear uno nosotros y añadirlo en el archivo nagios.

```
GNU nano 2.5.3                               Archivo: /usr/local/nagios/etc/nagios.cfg

# Definitions for monitoring the local (Linux) host
cfg_file=/usr/local/nagios/etc/objects/localhost.cfg

# Definitions for monitoring a Windows machine
#cfg_file=/usr/local/nagios/etc/objects/windows.cfg

# Definitions for monitoring a router/switch
#cfg_file=/usr/local/nagios/etc/objects/switch.cfg

# Definitions for monitoring a network printer
#cfg_file=/usr/local/nagios/etc/objects/printer.cfg

# You can also tell Nagios to process all config files (with a .cfg
# extension) in a particular directory by using the cfg_dir
# directive as shown below:

cfg_dir=/usr/local/nagios/etc/servers
#cfg_dir=/usr/local/nagios/etc/printers
#cfg_dir=/usr/local/nagios/etc/switches

^G Ver ayuda ^O Guardar      ^W Buscar      ^K Cortar Texto ^J Justificar ^C Posición
^X Salir      ^R Leer fich.  ^\ Reemplazar   ^U Pegar txt   ^T Ortografía ^_ Ir a línea
```

Una vez hecho todo esto, y vigilando cosas como tener la máquina virtual en adaptador puente para que se pueda comunicar, reiniciamos nagios y tendremos conectada nuestra máquina windows y nos aparecerá tal que así en la web en la página servicios. La máquina de arriba es el propio servidor y la de abajo el cliente windows.

El servicio del internet explorer aparecerá crítico cuando no esté iniciado y el servicio W3SVC no está instalado. Además la máquina nos podría aparecer como caída debido al cortafuegos aunque aparezcan como correcto los servicios de la misma.

Host Status Totals		Service Status Totals	
Up	Down	Unreachable	Pending
2	0	0	0
All Problems	All Types	All Problems	All Types
0	2	3	15

Service Status Details For All Hosts

Host	Service	Status	Last Check	Duration	Attempt	Status Information
localhost	Current Load	OK	05-27-2021 23:21:34	0d 1m 53m 30s	1/4	OK - load average: 0.04, 0.26, 0.31
localhost	Current Users	OK	05-27-2021 23:21:34	0d 1m 53m 1s	1/4	USERS OK - 1 users currently logged in
HTTP	HTTP	OK	05-27-2021 23:22:29	0d 1m 52m 24s	1/4	HTTP OK - HTTP/1.1 200 OK - 11595 bytes in 0.000 second response time
PING	PING	OK	05-27-2021 23:23:28	0d 1m 51m 46s	1/4	PING OK - Packet loss = 0%, RTA = 0.08 ms
	Root Partition	OK	05-27-2021 23:24:26	0d 1m 51m 9s	1/4	DISK OK - free space / 2504 MB (33.35% used=559M): connect to address 127.0.0.1 and port 22: Conexión rehusada
SSH	SSH	CRITICAL	05-27-2021 23:21:34	0d 1h 26m 8s	4/4	SWAP OK - 98% free (1994 MB out of 2045 MB)
	Swap Usage	OK	05-27-2021 23:21:34	0d 1h 29m 8s	1/4	PROCS OK: 50 processes with STATE = R/SZD
	Total Processes	OK	05-27-2021 23:21:34	0d 1h 29m 8s	1/4	c: - total: 476.15 Gb - used: 147.46 Gb (31%) - free: 328.69 Gb (69%)
win10	C:\ Drive Space	OK	05-27-2021 23:23:21	0d 0h 1m 18s	1/3	CPU Load 9% (5 min average)
	CPU Load	OK	05-27-2021 23:17:00	0d 0h 7m 39s	1/3	Explorer.exe: not running
	Memory Usage	CRITICAL	05-27-2021 23:17:00	0d 0h 8m 41s	3/3	Memory usage: total=83 MB - used=6717.47 MB (76%) - free=2163.36 MB (24%)
	NSClient++ Version	OK	05-27-2021 23:16:34	0d 0h 28m 5s	1/3	NSClient++ 0.5.2.35 2018-01-28
	Uptime	OK	05-27-2021 23:17:06	0d 0h 7m 33s	1/3	System Uptime - 0 day(s) 7 hours(s) 1 minute(s)
	W3SVC	UNKNOWN	05-27-2021 23:17:51	0d 1h 16m 48s	3/3	Failed to open service W3SVC: 424: El servicio especificado no existe como servicio instalado.

Una vez monitorizado un equipo windows, pasamos a monitorizar uno linux:

Es un poco más complicado que hacerlo en la máquina windows ya que habrá que instalar más paquetes y crear y editar más archivos de configuración.

Primero tendremos que instalar el NRPE tanto en el servidor como en el cliente. Una vez instalado, en el servidor nos iremos a la carpeta /usr/local/nagios/etc/objects y crearemos un archivo que se llame linux.cfg. Este archivo tendrá la misma estructura que el de windows, con una sección para definir el host y otra para definir los servicios. En el apartado check command, se podrá check nrpe ya que es el agente que usaremos y el nombre del comando que queremos (están en el archivo de configuración de nrpe en el cliente los posibles).

```

GNU nano 2.5.3                               Archivo: linux.cfg
### SERVIDORES LINUX ###
define host{
    use           generic-service
    host_name     linux
    alias         linux 1
    check_interval 1
    address      192.168.1.50
}

### SERVICIOS ###
define service{
    use           generic-service
    host_name     linux
    service_description Uptime
    check_interval 1
    check_command  check_nrpe!check_uptime
}

define service{
    use           generic-service
    host_name     linux_
    service_description Carga actual sistema
    check_command  check_nrpe!check_load
}

define service{
    use           generic-service
    host_name     linux_
    service_description Disco Duro
    check_command  check_nrpe!check_hda1
}

define service{
    use           generic-service
    host_name     linux_
    service_description Chequeo ping
    check_command  check_ping!500.0,30%$0.0,70%
}

define service{
    use           generic-service
    host_name     linux_
    service_description Swap
    check_command  check_nrpe!check_swap
}

define service{
    use           generic-service
}

```

Archivo: linux.cfg

Ver ayuda Ver Salir Guardar Leer fich. Buscar Cortar Texto Justificar Posición Ir a linea Pág. ant. Pág. sig. Pri. linea Últ. linea Siguiente A llave Marcar texto Sangrar txt Desangr. txt Deshacer Ctrl Derecha

Ahora nos iremos al archivo de configuración de nagios, nagios.cfg en la carpeta /usr/local/nagios/etc e igual que descomentamos la línea del archivo de windows, creamos otra igual pero con el archivo de equipos linux. Una vez hecho esto, reiniciamos Nagios.

Cuando hayamos acabado, el servidor linux nos quedará así en la interfaz web de nagios.

Host Status Totals		Service Status Totals					
Up	Down	Unreachable	Pending	All Problems	All Types	All Problems	All Types
2	1	0	0	13	0	3	5
				All Problems	All Types	All Problems	All Types
				1	3	8	21

Service Status Details For All Hosts

Host	Service	Status	Last Check	Duration	Attempt	Status Information
linux	Carga actual sistema	CRITICAL	05-29-2021 16:28:04	0d 0h 29m 56s	3/3	CRITICAL - load average: 0.11, 0.54, 0.61
	Chequeo ping	OK	05-29-2021 16:28:09	0d 0h 29m 44s	1/3	PING OK - 0ms latency, 100% packet loss, 0ms
	Disco Duro	CRITICAL	05-29-2021 16:28:04	0d 0h 29m 48s	3/3	DISK CRITICAL - /dev/hd0d is not accessible: No such file or directory
	Swap	UNKNOWN	05-29-2021 16:28:04	0d 0h 0m 48s	3/3	NRPE: Command check_swap not defined
	Uptime	UNKNOWN	05-29-2021 16:28:04	0d 0h 0m 48s	3/3	NRPE: Command check_uptime not defined
	Usuarios activos	OK	05-29-2021 16:28:04	0d 0h 0m 48s	1/3	USERS OK - 1 users currently logged in
localhost	Current Load	OK	05-29-2021 16:23:51	0d 23h 23m 54s	1/4	OK - load average: 0.05, 0.19, 0.27
	Current Users	OK	05-29-2021 16:25:23	0d 23h 23m 16s	1/4	USERS OK - 1 users currently logged in
	HTTP	OK	05-29-2021 16:26:32	0d 23h 22m 39s	1/4	HTTP OK: HTTP/1.2 200 OK - 11595 bytes in 0.000 second response time
	PING	OK	05-29-2021 16:27:42	0d 23h 22m 1s	1/4	PING OK - Packet loss = 0%, RTA = 0.06 ms
	Root Partition	OK	05-29-2021 16:23:51	0d 23h 21m 24s	1/4	DISK OK - free space: /2465 MB (32.82% inode=54%)
	SSH	CRITICAL	05-29-2021 16:23:51	0d 5h 56m 23s	4/4	connect to address 127.0.0.1 and port 22: Conexión rechazada
	Swap Usage	SWAP OK	05-29-2021 16:23:51	0d 5h 59m 23s	1/4	SWAP OK - 92% free (1881 MB out of 2045 MB)
	Total Processes	OK	05-29-2021 16:23:51	0d 5h 59m 23s	1/4	PROCS OK: 58 processes with STATE = R/S/Z/T
win10	C1 Drive Space	OK	05-29-2021 16:20:46	0d 4h 41m 33s	1/3	c - total: 476.15 Gb - used: 151.46 Gb (32%) - free 324.69 Gb (68%)
	CPU Load	OK	05-29-2021 16:21:56	0d 4h 37m 54s	1/3	CPU Load 9% (5 min average)
	Explorer	CRITICAL	05-29-2021 16:23:05	0d 4h 36m 56s	3/3	Explorer.exe: not running
	Memory Usage	CRITICAL	05-29-2021 16:24:14	0d 0h 14m 38s	3/3	Memory usage: total 9262.47 MB - used: 8373.28 MB (90%) - free: 889.19 MB (10%)
	NSClient++ Version	OK	05-29-2021 16:25:24	0d 4h 58m 20s	1/3	NSClient++ 0.5.2.35 2018-01-28
	Uptime	OK	05-29-2021 16:26:33	0d 4h 37m 48s	1/3	System Uptime - 1 day(s) 21 hour(s) 7 minute(s)
	W3SVC	UNKNOWN	05-29-2021 16:27:42	0d 5h 47m 3s	3/3	Failed to open service W3SVC: 424. El servicio especificado no existe como servicio instalado.

Results 1 - 21 of 21 Matching Services

Como podemos ver, hay varios errores que subsanaremos ahora. Hay que editar el archivo de configuración de nrpe en el cliente linux (/usr/local/nagios/etc/nrpe.cfg).

Primero, para solucionar el error del disco, que en sí es que no encuentra la partición, buscamos como se llama nuestra partición de la que queremos observar el espacio (en mi caso /dev/sda1) y la cambiamos por la que viene por defecto que es /dev/hda1.

```
ux_: /usr/local/nagios/etc
root@linux:~/home/migue
migue@linux:~$ sudo su
[sudo] password for migue:
migue@linux:~/home/migue$ df -h
Filesystem      Tamaño Usados  Disp Usos Montado en
udev            979M       0  979M   0% /dev
tmpfs           201M     3.7M  197M  2% /run
/dev/sda1        7.8G    5.7G  1.7G  78% /
tmpfs           100M   252K 100M  1% /dev/shm
tmpfs           5.0M   4.0K  5.0M  1% /run/lock
tmpfs           100M   100M  0% /sys/fs/cgroup
tmpfs           201M   60K  200M  1% /run/user/1000
/dev/sr0          59M     59M  0  100% /media/migue/VBox_GAs_6.1.141
root@linux:~/home/migue$ [REDACTED]
```

```
command[check_users]=/usr/local/nagios/libexec/check_users -w 5 -c 10
command[check_load]=/usr/local/nagios/libexec/check_load -r -w .15,.10 -.05 -.30,.25,.20
command[check_hd1]=/usr/local/nagios/libexec/check_disk -w 20% -c 10% -p /dev/sda1
command[check_zombie_procs]=/usr/local/nagios/libexec/check_procs -w 5 -c 10 -s Z
command[check_total_procs]=/usr/local/nagios/libexec/check_procs -w 150 -c 200

# The following examples allow user-supplied arguments and can
# only be used if the NRPE daemon was compiled with support for
# command arguments *AND* the dont_blame_nrpe directive in this
# config file is set to '1'. This poses a potential security risk, so
# make sure you read the SECURITY file before doing this.

### MISC SYSTEM METRICS ####
command[check_users]=/usr/local/nagios/libexec/check_users $ARG1$
command[check_load]=/usr/local/nagios/libexec/check_load $ARG1$
command[check_disk]=/usr/local/nagios/libexec/check_disk $ARG1$
command[check_swap]=/usr/local/nagios/libexec/check_swap $ARG1$
command[check_cpu_stats]=/usr/local/nagios/libexec/check_cpu_stats.sh $ARG1$
command[check_mem]=/usr/local/nagios/libexec/custom_check_mem -n $ARG1$

## GENERIC SERVICES ##
command[check_int_service]=sudo /usr/local/nagios/libexec/check_int_service $ARG1$
command[check_services]=/usr/local/nagios/libexec/check_services -p $ARG1$
```

Una vez solucionado este error, los otros dos consisten en definir los comandos que no están definidos, aunque están comentados como ejemplo y nos podemos basar en eso. En el caso del comando del swap, tendremos que definir que porcentaje consideraremos warning y crítico.

```
root@linux:/usr/local/nagios/etc          [ ] [ ] [ ] 16:09 30
GNU nano 2.5.3                           Archivo: /usr/local/nagios/etc/nrpe.cfg
Modificado

# COMMAND DEFINITIONS
# Command definitions that this daemon will run. Definitions
# are in the following format:
#
# command[<command_name>]=<command_line>
#
# When the daemon receives a request to return the results of <command_name>
# it will execute the command specified by the <command_line> argument.
#
# Unlike Nagios, the command line cannot contain macros - it must be
# typed exactly as it should be executed.
#
# Note: Any plugins that are used in the command lines must reside
# on the machine that this daemon is running on! The examples below
# assume that you have plugins installed in a /usr/local/nagios/libexec
# directory. Also note that you will have to modify the definitions below
# to match the argument format the plugins expect. Remember, these are
# examples only!
#
# The following examples use hardcoded command arguments...
# This is by far the most secure method of using NRPE
#
# command[check_users]=/usr/local/nagios/libexec/check_users -w 5 -c 10
# command[check_load]=/usr/local/nagios/libexec/check_load -w .10,.05 -c .30,.25,.20
# command[check_hdload]=/usr/local/nagios/libexec/check_disk -w 20% -c 10% -p /dev/sda1
# command[check_zombie_procs]=/usr/local/nagios/libexec/check_procs -w 5 -c 10 -s Z
# command[check_low_procs]=/usr/local/nagios/libexec/check_procs -w 150 -c 200
# command[check_swap]=/usr/local/nagios/libexec/check_swap -w 15% -c 10%
# command[check_uptime]=/usr/local/nagios/libexec/check_uptime

# The following examples allow user-supplied arguments and can
# be used to implement more advanced command definitions or
# command arguments AND the don't blame nrpe directive in this
# config file is set to '1'. This poses a potential security risk, so
# make sure you read the SECURITY file before doing this.

### NISC SYSTEM METRICS ###
#command[check_users]=/usr/local/nagios/libexec/check_users $ARG1$S
#command[check_load]=/usr/local/nagios/libexec/check_load $ARG1$S
#command[check_disk]=/usr/local/nagios/libexec/check_disk $ARG1$S
#command[check_swap]=/usr/local/nagios/libexec/check_swap $ARG1$S
#command[check_cpu_stats]=/usr/local/nagios/libexec/check_cpu_stats.sh $ARG1$S
#command[check_mem]=/usr/local/nagios/libexec/custom_check_mem -n $ARG1$S

### GENERIC SERVICES ###
#command[check_int_service]=sudo /usr/local/nagios/libexec/check_int_service $ARG1$S
#command[check_services]=/usr/local/nagios/libexec/check_services -p $ARG1$S

## Ver ayuda      Guardar      Buscar      Cortar Texto      Justificar      Posición      Pág. ant.      Pág. sig.      Pŕv. linea      Siguiente      Marcar texto      Sangrar txt      Desangr. txt      Deshaber
## Salir        Leer fich.    Reemplazar    Pegar txt     Ortografía     Ir a línea     Pág. sig.      Últ. linea     A llave      Copiar txt      Desangr. txt      Rehacer
```

Reiniciamos el servicio nrpe en la máquina cliente y cuando se actualicen los datos estarán correctos.

The screenshot shows the Nagios Core web interface. At the top, there are three status summary boxes: 'Current Network Status' (Up: 3, Down: 0, Unreachable: 0, Pending: 0), 'Host Status Totals' (Up: 15, Down: 2, Unreachable: 1, Pending: 3), and 'Service Status Totals' (Ok: 6, Warning: 21, Unknown: 1, Critical: 3, Pending: 0). Below these are two more status boxes: 'All Problems' (0) and 'All Types' (21).

The main content area is titled 'Service Status Details For All Hosts'. It lists services for three hosts: 'linux', 'localhost', and 'win10'. The table columns include Host, Service, Status, Last Check, Duration, Attempts, and Status Information.

Host	Service	Status	Last Check	Duration	Attempts	Status Information
linux	Carga actual sistema	WARNING	05-30-2021 16:19:05	0d 0h 0m 25s	3/3	WARNING - load average: 0.15, 0.18, 0.14
	Chequeo ping	WARNING	05-30-2021 16:17:58	0d 0h 1m 32s	1/3	PING WARNING - Packet loss = 60%, RTA = 1.03 ms
	Disco Duro	OK	05-30-2021 16:18:58	0d 0h 26m 46s	1/3	DISK OK - free space: /var/tmp 1723 MB (22.95% inode=47%)
	Swap	OK	05-30-2021 16:18:58	0d 0h 9m 5s	1/3	SWAP OK - 100% free (2044 MB out of 2045 MB)
	Uptime	OK	05-30-2021 16:18:58	0d 0h 15m 6s	1/3	Uptime OK: 0 day(s) 1 hour(s) 9 minute(s)
	Usuarios activos	OK	05-30-2021 16:18:58	0d 0h 26m 46s	1/3	USERS OK - 1 users currently logged in
localhost	Current Load	OK	05-30-2021 16:17:58	1d 0h 17m 2s	1/4	OK - load average: 0.51, 0.47, 0.30
	Current Users	OK	05-30-2021 16:17:58	1d 0h 16m 24s	1/4	USERS OK - 1 users currently logged in
	HTTP	OK	05-30-2021 16:17:58	1d 0h 15m 47s	1/4	HTTP OK: HTTP/1.1 200 OK - 11595 bytes in 0.025 second response time
	PING	OK	05-30-2021 16:17:58	1d 0h 15m 9s	1/4	PING OK - Packet loss = 0%, RTA = 0.06 ms
	Root Partition	OK	05-30-2021 16:17:58	1d 0h 14m 32s	1/4	DISK OK - free space: / 2465 MB (32.83% inode=54%)
	SSH	CRITICAL	05-30-2021 16:17:58	0d 6h 49m 31s	4/4	connect to address 127.0.0.1 port 22: Conexión rechazada
win10	Swap Usage	OK	05-30-2021 16:17:58	0d 6h 52m 31s	1/4	SWAP OK - 92% free (1870 MB out of 2045 MB)
	Total Processes	OK	05-30-2021 16:17:58	0d 6h 52m 31s	1/4	PROCS OK - 67 processes with STATE = RZSDT
	C1 Drive Space	OK	05-30-2021 16:17:58	0d 5h 34m 41s	1/3	c - total: 471.15 Gb - used: 153.22 Gb (32%) - free 322.93 Gb (68%)
	CPU Load	OK	05-30-2021 16:17:58	0d 5h 33m 2s	1/3	CPU Load 9% (5 min average)
	Explorer	CRITICAL	05-30-2021 16:17:58	0d 5h 30m 4s	3/3	Explorer.exe: not running
	Memory Usage	CRITICAL	05-30-2021 16:17:58	0d 0h 15m 4s	1/3	Memory usage: total:9134.07 MB - used: 6323.28 MB (91%) - free: 810.80 MB (9%)
NSClient++ Version	OK	05-30-2021 16:17:58	0d 5h 51m 28s	1/3	NSClient++ 0.5.2.35 2018-01-28	
Uptime	OK	05-30-2021 16:17:58	0d 5h 30m 56s	1/3	System Uptime - 0 day(s) 3 hour(s) 37 minute(s)	
W3SVC	UNKNOWN	05-30-2021 16:17:58	0d 6h 40m 11s	3/3	Failed to open service W3SVC: 424: El servicio especificado no existe como servicio instalado.	

Results 1 - 21 of 21 Matching Services

Así nos queda nuestro Nagios Core una vez añadidos los dos clientes, los errores que quedan son debido a la potencia de mi PC y tener dos máquinas virtuales encendidas, o no tener los servicios iniciados, como el Explorer.

Una vez hecha la instalación y configuración básica, pasamos a ver ciertos usos que se le puede dar a Nagios o ciertas configuraciones interesantes.

-Monitoreo de un router o switch con nagios:

También es posible monitorear nuestro router o switch para ver ya sea el tráfico de la red, la pérdida de paquetes o el estado de los puertos.

Desgraciadamente, los routers que mandan las empresas de telefonía a nuestras casas no se pueden modificar porque son de solo lectura y en el caso del mío, un Livebox, tiene el SNMP(Simple Network Management Protocol), protocolo que se usa al monitorear las redes, inhabilitado por seguridad. Así que solo podremos monitorear el ping. De todas maneras vamos a ver los pasos que hay que seguir.

Primero de todo se activa el SNMP en el router o switch, una vez hecho esto, en nuestro servidor Nagios, nos vamos al archivo de ejemplo switch.cfg que se encuentra en la carpeta /usr/local/nagios/etc/objects ,en este archivo, tal como vemos, tenemos los parametros de define service donde, en el primero pondremos el nombre, alias e IP de nuestro router.

```

root@migue-VirtualBox:/usr/local/nagios/etc/objects# nano 2.5.3                                     Archivo: switch.cfg
#####
# Create a service to PING to switch
define service {
    use generic-service
    host_name orange-casal
    service_description PING
    check_command check_ping!200.0,20%1600.0,60%
    check_interval 2
    retry_interval 1
}

# Monitor uptime via SNMP
define service {
    use generic-service
    host_name orange-casal
    service_description Uptime
    check_command check_snmp!-C public -o sysUpTime.0
}

# Monitor Port 1 status via SNMP
define service {
    use generic-service
    host_name linksys-srw224p
    service_description Port 1 Link Status
    check_command check_snmp!-C public -o ifOperStatus.1 -r 1 -M RFC1213-MIB
}

# Monitor bandwidth via MRTG logs
define service {
    use generic-service
    host_name linksys-srw224p
    service_description Port 1 Bandwidth Usage
    check_command check_local_mrtgtrafi/var/lib/mrtg/192.168.1.253_1.log!AVG!1000000,1000000!5000000,5000000!10
}

```

En los siguientes servicios, simplemente cambiaremos el nombre del host que viene por defecto por el nuestro y la IP que viene de ejemplo por la nuestra. Si hay algunos servicios que no vamos a monitorizar, los comentamos, como en mi caso todos los relacionados con el SNMP, otro que puede que no se use es el de MRTG(*Multi Router Traffic Grapher*), una herramienta para el ancho de banda, que también usa SNMP. Salimos y guardamos.

Después en el archivo de configuración de Nagios /usr/local/nagios/etc/ descomentamos la linea del archivo switch.cfg

```

# You can specify individual object config files as shown below:
cfg_file=/usr/local/nagios/etc/objects/commands.cfg
cfg_file=/usr/local/nagios/etc/objects/contact.cfg
cfg_file=/usr/local/nagios/etc/objects/timers.cfg
cfg_file=/usr/local/nagios/etc/objects/templates.cfg
cfg_file=/usr/local/nagios/etc/objects/linux.cfg
# Definitions for monitoring the local (linux) host
cfg_file=/usr/local/nagios/etc/objects/localhost.cfg

# Definitions for monitoring a Windows machine
cfg_file=/usr/local/nagios/etc/objects/windows.cfg

# Definitions for monitoring a router/switch
#cfg_file=/usr/local/nagios/etc/objects/switch.cfg

```

Una vez hecho esto, reiniciamos y ya tenemos el router monitorizado.

The screenshot shows the Nagios web interface at <http://192.168.1.101/nagios/>. The main page displays current network status, host status totals, and service status totals. A detailed table follows, listing services for hosts like 'linux_'. The table columns include Host, Service, Status, Last Check, Duration, Attempt, and Status Information. A blue highlight covers the 'orange-case1' host row.

Host	Service	Status	Last Check	Duration	Attempt	Status Information
linux_	Carga actual sistema	CRITICAL	06-01-2021 19:33:33	1d 9h 56m 42s	3/3	Nodo1: ssh: no (0x0) stderr: connect to address 192.168.1.50 port 5666: No route to host
	Chequeo ping	CRITICAL	06-01-2021 19:33:33	0d 0h 24m 58s	3/3	CRITICAL: Host Unreachable (192.168.1.50)
	Disco Duro	CRITICAL	06-01-2021 19:33:33	0d 1h 24m 4s	3/3	(No output on stdio) stderr: connect to address 192.168.1.50 port 5666: No route to host
	Swap	CRITICAL	06-01-2021 19:33:33	0d 1h 22m 4s	3/3	(No output on stdio) stderr: connect to address 192.168.1.50 port 5666: No route to host
	Uptime	CRITICAL	06-01-2021 19:33:33	0d 1h 22m 4s	3/3	(No output on stdio) stderr: connect to address 192.168.1.50 port 5666: No route to host
	Usuarios activos	CRITICAL	06-01-2021 19:33:33	0d 1h 22m 4s	3/3	(No output on stdio) stderr: connect to address 192.168.1.50 port 5666: No route to host
localhost	Current Load	OK	06-01-2021 19:32:42	1d 1h 41m 43s	1/4	OK - load average: 0.63, 0.67, 0.54
	Current Users	OK	06-01-2021 19:32:42	1d 1h 41m 5s	1/4	USERS OK - 1 users currently logged in
	HTTP	OK	06-01-2021 19:32:42	1d 1h 40m 28s	1/4	HTTP OK: HTTP/1.1 200 OK - 1195 bytes in 0.027 second response time
	PING	OK	06-01-2021 19:32:42	1d 1h 39m 50s	1/4	PING OK - Packet loss = 0%, RTA = 0.05 ms
	Root Partition	OK	06-01-2021 19:32:42	1d 1h 39m 13s	1/4	DISK OK - free space: / 2463 MB (32.80% inode=54%)
	SSH	CRITICAL	06-01-2021 19:32:42	0d 9h 14m 12s	4/4	connect to address 127.0.0.1 and port 22: Conexión rechazada
	Swap Usage	OK	06-01-2021 19:32:42	0d 1h 7m 12s	1/4	SWAP OK - 90% free (1827 MB out of 2045 MB)
orange-case1	PING	OK	06-01-2021 19:33:48	0d 8h 17m 22s	1/3	PROCS OK: 64 processes with STATE = R/S/Z/D
	CPU Load	OK	06-01-2021 19:27:42	0d 6h 58m 22s	1/3	PING OK - Packet loss = 0%, RTA = 0.98 ms
	Explorer	OK	06-01-2021 19:27:42	0d 6h 55m 42s	1/3	c: total: 4761.5 Gb - used: 154.77 Gb (33%) - free 321.38 Gb (67%)
	Memory Usage	CRITICAL	06-01-2021 19:27:42	0d 6h 54m 45s	3/3	CPU Load 9% (5 min average)
	NSClient+ Version	WARNING	06-01-2021 19:21:42	0d 7h 16m 9s	1/3	Explorer.exe: not running
	Uptime	OK	06-01-2021 19:27:42	0d 6h 55m 37s	1/3	Memory usage: total 9073.67 MB - used: 7584.65 MB (84%) - free: 1489.02 MB (16%)
	W3SVC	UNKNOWN	06-01-2021 19:27:42	0d 8h 4m 52s	3/3	NSClient+: 0.5.2.35 2018-01-28

-Creación de grupos de servicios y de hosts:

Con Nagios podemos agrupar hosts y servicios, por ejemplo todos los discos duros de los equipos o los equipos de una sala, como en nuestro ejemplo práctico. Para crear grupos de hosts, simplemente nos vamos al archivo de configuración, ya sea de windows o de linux y definimos un nuevo grupo con define group. Para hacer más fácil los ejemplos añadiremos otro equipo para monitorear aunque no exista y no se monitoree.

```
define hostgroup {
    hostgroup_name           sala1          ;
    alias                     'Equipos de la sala 1';
}
```

Luego los equipos que queramos, nos vamos al archivo de configuración donde estén definidos y los añadimos con hostgroups.

También es posible añadir los equipos en define hostgroup añadiendo members debajo de alias y poniendo el nombre de los servidores. De igual forma en los servicios.

```
define host {
    use                      windows-server;
    host_name                windows1_sala1
    alias                    'Equipo 1 de la Sala 1';
    address                  192.168.1.10;
    hostgroups               equipos,sala1;
}
```

Para los servicios, es exactamente igual, solo cambian los nombres como podemos observar:

```
define servicegroup{
    servicegroup_name          discos_duros
    alias                      Discos Duros de nuestros equipos
}
```

```
define service {
    use                         generic-service
    host_name                   win10,window1_sala1
    service_description          C:\ Drive Space
    check_command                check_nt!USEDISKSPACE!-l c -w 80 -c 90
    servicegroups                discos_duros
}
```

Una vez hemos acabado, reiniciamos Nagios y en la interfaz web tenemos las pestañas de grupos, tanto de los servicios como de los hosts como vemos:

Service Overview For All Host Groups								
No server (equipos)			Linux Servers (linux-servers)			Equipos de la sala 1 (sala1)		
Host	Status	Services	Host	Status	Services	Host	Status	Services
win10	UP	5 OK 1 UNKNOWN 1 CRITICAL	localhost	UP	7 OK 1 CRITICAL	linux_	DOWN	6 CRITICAL
window1_sala1	DOWN	1 CRITICAL 3 PENDING				win10	UP	5 OK 1 UNKNOWN 1 CRITICAL
						window1_sala1	DOWN	1 CRITICAL 3 PENDING
Network Switches (switches)								
Host	Status	Services	Actions					
orange-casa1	UP	1 OK						

Service Overview For All Service Groups

Discos Duros de nuestros equipos (discos_duros)			
Host	Status	Services	Actions
linux_	UP	1 CRITICAL	
win10	UP	1 OK	
windows1_sala1	DOWN	1 PENDING	

-Edición del tiempo de monitoreo y de reintento de un host o servicio:

En las configuraciones de cada host y servicio en la carpeta /usr/local/nagios/etc//objects, ya sea linux o windows, y en cada servicio y host que queramos monitorear en un tiempo mas reducido o más amplio de los estándar, que son 5 minutos, ponemos el comando check_interval y los minutos que queramos. Al ponerlo en el host, no quiere decir que se aplique a todos los servicios de este, solo se aplica al propio host.

```
define service{
    use generic-service
    host_name linux_
    check_interval 1
    service_description Carga actual sistema
    check_command check_nrpe!check_load
}
```

Otra versión de este comando es el retry_interval que es igual, pero solo utiliza ese tiempo, si el servicio o host ha dado error crítico.

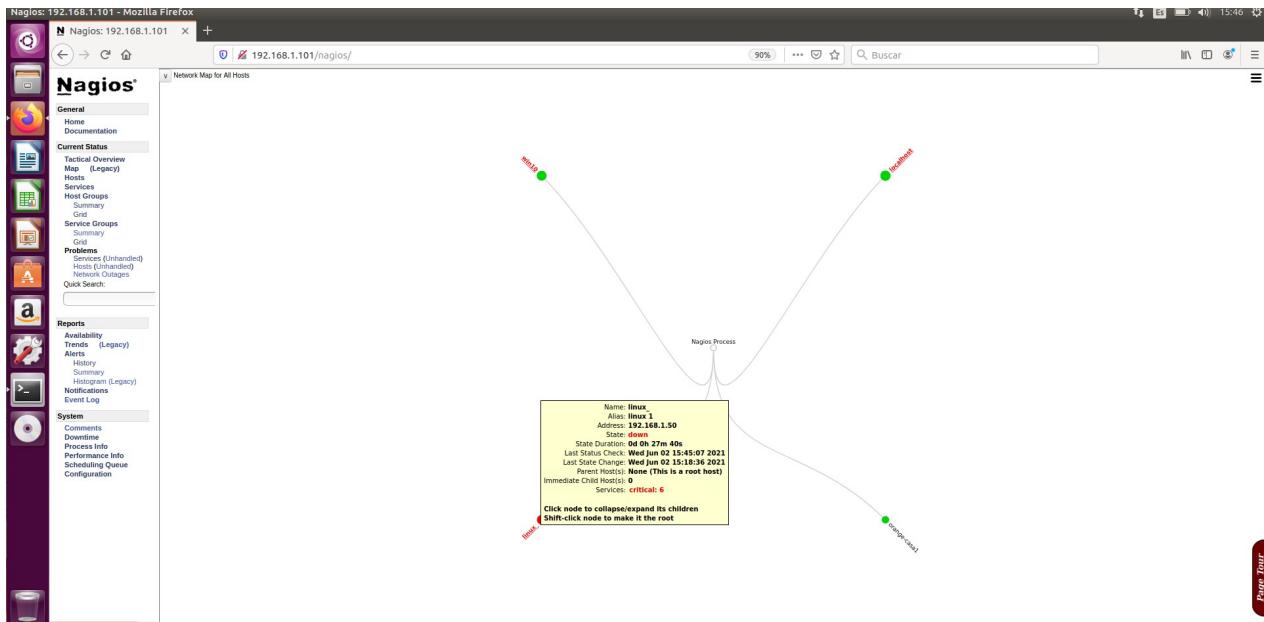
-Interfaz web de Nagios:

El objetivo de este apartado es familiarizarse con la interfaz web de Nagios y ver cada apartado y las posibilidades que nos ofrece.

Ya hemos visto por encima el apartado de servicios, donde se ve cada máquina con sus respectivos servicios y la página de grupos.

A parte de estas también tenemos la pestaña de hosts donde aparece cada máquina y su estado. Esta pestaña, junto con la de los grupos de servicios y hosts, pueden no parecer muy útiles, ya que nos dan la misma información que las otras, pero en una empresa donde se estén monitoreando cientos de máquinas sí que sirve.

Además tenemos la pestaña del mapa de red, donde aparece la estructura de la red y veremos gráficamente si hay alguna máquina que depende de otra. Pasando el ratón por encima de cada máquina vemos su estado, su último chequeo, el próximo y más información.



Por último tenemos la página de configuración, donde podemos ver las configuraciones activas de los hosts, los grupos de ambos tipos, los servicios...

Select Type of Config Data You Wish To View

Object Type:

Hosts

- Hosts
- Host Dependencies
- Host Escalations
- Host Groups
- Services
- Service Groups
- Service Dependencies
- Service Escalations
- Contacts
- Contact Groups
- Timeperiods
- Commands
- Command Expansion

Aquí podemos ver el apartado de hosts, donde vemos todas las opciones que hemos elegido y aprendido en los ficheros de configuración, como las通知aciones o el intervalo de chequeo.

Hosts																											
Host Name	Alias/Description	Address	Importance (Host)	Importance (Host + Services)	Parent Hosts	Max. Check Attempts	Check Interval	Retry Interval	Host Check Command	Check Period	Obsess Over	Enable Active Checks	Enable Passive Checks	Check Freshness	Freshness Threshold	Default Contacts/Groups	Notification Interval	First Notification Delay	Notification Options	Notification Period	Event Handler	Enable Event Handler	Stalking Options	Enable Flap Detection	Low Flap Threshold	High Flap Threshold	Flap Detection Options
linux_1	linux 1	192.168.1.50	0	0		10	0m 1s	0m 1s	check-host-alive	24x7	Yes	Yes	No	Auto-determined value	admins	2h 0m 0s	0h 0m 0s	Down, Unreachable, Recovery	workhours	Yes	None	Yes	Program-wide value	Program-wide value	Up, Down, Unreachable		
localhost	localhost	127.0.0.1	0	0		10	0m 5m	0m 1s	check-host-alive	24x7	Yes	Yes	No	Auto-determined value	admins	2h 0m 0s	0h 0m 0s	Down, Unreachable, Recovery	workhours	Yes	None	Yes	Program-wide value	Program-wide value	Up, Down, Unreachable		
orange_casa1	Router de Orange Cesa	192.168.1.1	0	0		10	0m 5m	0m 1s	check-host-alive	24x7	Yes	Yes	No	Auto-determined value	admins	0h 30m 0s	0h 0m 0s	Down, Recovery	24x7	Yes	None	Yes	Program-wide value	Program-wide value	Up, Down, Unreachable		
win10	Portatil win10	192.168.1.2	0	0		10	0m 5m	0m 1s	check-host-alive	24x7	Yes	Yes	No	Auto-determined value	admins	0h 30m 0s	0h 0m 0s	Down, Recovery	24x7	Yes	None	Yes	Program-wide value	Program-wide value	Up, Down, Unreachable		

-Como activar las notificaciones por correo de Nagios:

En mi opinión, esta es una parte muy importante de Nagios, desde cualquier lugar y con nuestro correo electrónico, seremos avisados si se cae un servicio o un equipo, lo cuál es muy útil, ya que sabremos cuando tenemos que arreglar algo sin tener Nagios abierto.

Primero de todo iremos en la carpeta /usr/local/nagios/etc/objects al archivo contacts.cfg, allí cambiaremos el correo por defecto por el nuestro donde queremos enviarlas.

```
GNU nano 2.5.3                               Archivo: /usr/local/nagios/etc/objects/contacts.cfg
#####
# Just one contact defined by default - the Nagios admin (that's you)
# This contact definition inherits a lot of default values from the
# 'generic-contact' template which is defined elsewhere.

define contact {
    contact_name      nagiosadmin          ; Short name of user
    use               generic-contact        ; Inherit default values from generic-contact template (defined above)
    alias             Nagios Admin          ; Full name of user
    email             nagios@localhost ; <<***** CHANGE THIS TO YOUR EMAIL ADDRESS *****>>
}

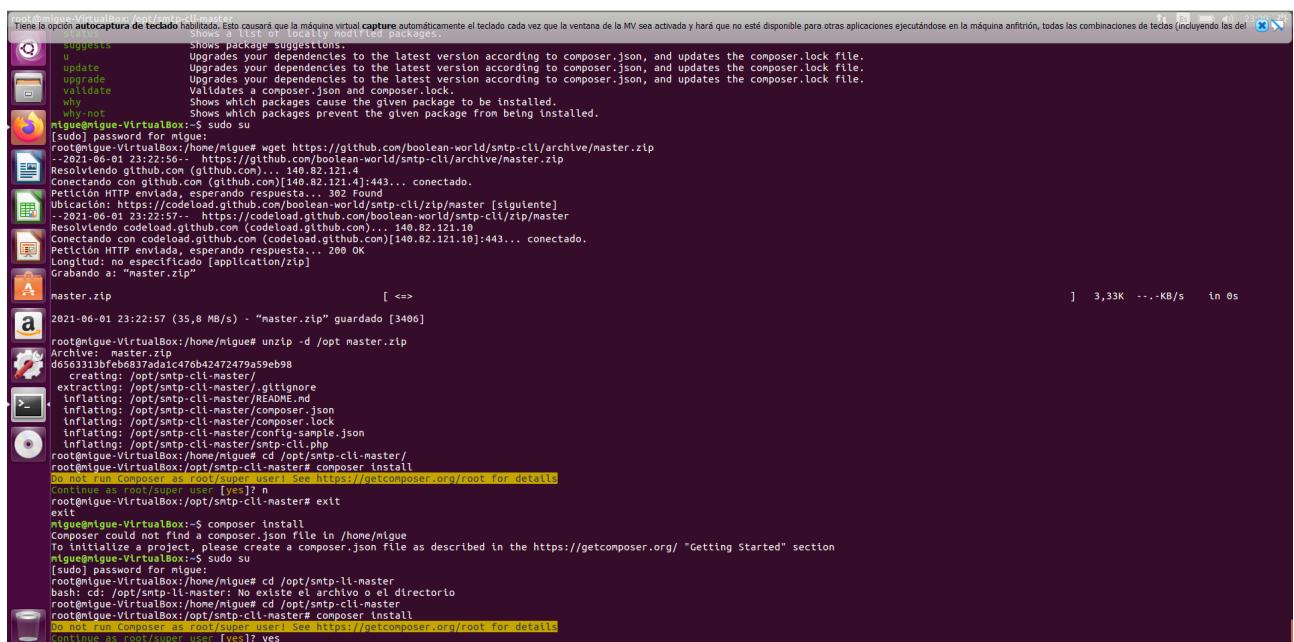
#
# CONTACT GROUPS
#
#####

# We only have one contact in this simple configuration file, so there is
# no need to create more than one contact group.

define contactgroup {
    contactgroup_name   admins
    alias              Nagios Administrators
    members            nagiosadmin
}
```

También tenemos la opción de añadir grupos de contacto con define contactgroup, donde podremos, si tenemos varios contactos, agruparlos. Ahora activaremos el correo SMTP para permitir enviarnos mensajes desde nuestro servidor, para ello, descargaremos el programa composer, además de unas utilidades de php. Utilizaremos este comando [apt install php php-cli php-gd php-curl php-zip php-intl php-mbstring php-xml php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"]. Algunas utilidades probablemente ya estarán instaladas pero así nos aseguraremos. Ahora lo instalamos, indicando el directorio y el nombre [php composer-setup.php --install-dir=/usr/local/bin –filename=composer].

Podemos comprobar con el comando composer. El siguiente paso es descargar e instalar el cliente SMTP. Para ello usamos un archivo del repositorio Github [wget https://github.com/boolean-world/smtp-cli/archive/master.zip] y descomprimimos con tar -xzvf /opt master.zip. Ahora en la carpeta /opt/mysqli-master instalamos composer con composer install.



```

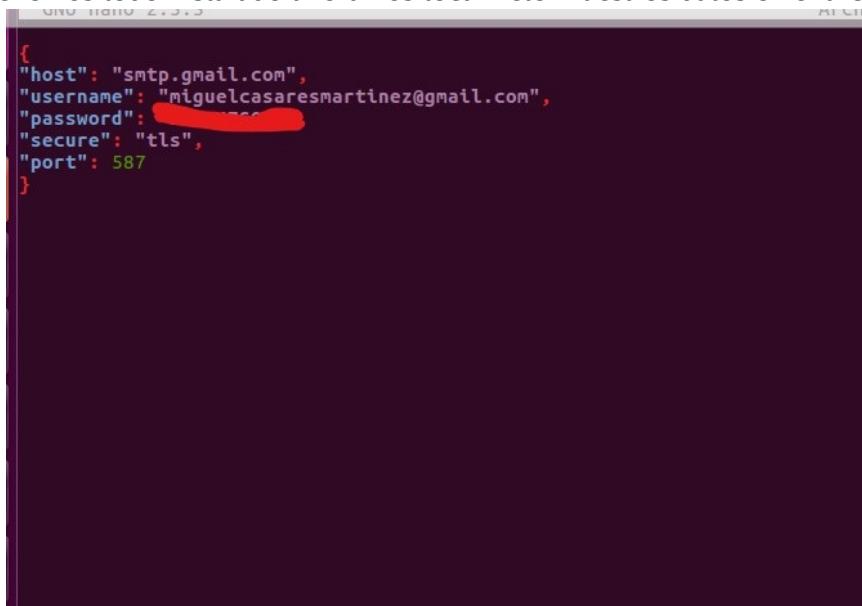
root@miguel-VirtualBox:~# apt install php php-cli php-gd php-curl php-zip php-intl php-mbstring php-xml php -r "copy('https://getcomposer.org/installer', 'composer-setup.php');"
Tiene la opción autocaptura de teclado habilitada. Esto causará que la máquina virtual capture automáticamente el teclado cada vez que la ventana de la MV sea activada y hará que no esté disponible para otras aplicaciones ejecutándose en la máquina anfitrión, todas las combinaciones de teclas (incluyendo las de escape) se enviarán directamente al teclado de la MV.
Sugges...
Shows package suggestions.
u          Upgrades your dependencies to the latest version according to composer.json, and updates the composer.lock file.
update     Upgrades your dependencies to the latest version according to composer.json, and updates the composer.lock file.
upgrade   Upgrades your dependencies to the latest version according to composer.json, and updates the composer.lock file.
validate   Validates composer.json and composer.lock.
why       Shows which packages cause the given package to be installed.
why-not   Shows which packages prevent the given package from being installed.
miguel@igue-VirtualBox:~$ sudo su
[Su contraseña]: 
root@miguel-VirtualBox:~# wget https://github.com/boolean-world/smtp-cli/archive/master.zip
--2021-06-01 23:22:56-- https://github.com/boolean-world/smtp-cli/archive/master.zip
Resolving github.com (github.com)... 140.82.121.4
Conectando con github.com (github.com)[140.82.121.4]:443... conectado.
Petición HTTP enviada, esperando respuesta... 302 Found
Ubicación: https://codeload.github.com/boolean-world/smtp-cli/zip/master [siguiente]
--2021-06-01 23:22:57-- https://codeload.github.com/boolean-world/smtp-cli/zip/master
Resolviendo codeload.github.com (codeload.github.com)... 140.82.121.10
Conectando con codeload.github.com (codeload.github.com)[140.82.121.10]:443... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: no especificado [application/zip]
Grabando a: "master.zip"

A                         master.zip
[ <> ] 3,33K  ---KB/s  in 0s

a 2021-06-01 23:22:57 (35,8 MB/s) - "master.zip" guardado [3406]
root@miguel-VirtualBox:~/home/miguel# unzip -d /opt master.zip
Archive: master.zip
d6563313bfe6837adac1476b42472479a59eb9b
  creating: /opt/smtp-cli-master/
  extracting: /opt/smtp-cli-master/.gitignore
  inflating: /opt/smtp-cli-master/README.ad
  inflating: /opt/smtp-cli-master/composer.json
  inflating: /opt/smtp-cli-master/composer.lock
  inflating: /opt/smtp-cli-master/config-sample.json
  inflating: /opt/smtp-cli-master/smtp.php
root@miguel-VirtualBox:~/home/miguel# cd /opt/smtp-cli-master/
root@miguel-VirtualBox:/opt/smtp-cli-master# composer install
Do not run Composer as root/super user! See https://getcomposer.org/root for details
Continuar as root/super user [yes]? n
root@miguel-VirtualBox:/opt/smtp-cli-master# exit
exit
root@miguel-VirtualBox:~$ composer install
Composer could not find a composer.json file in /home/miguel
To initialize a project, please create a composer.json file as described in the https://getcomposer.org/ "Getting Started" section
miguel@igue-VirtualBox:~$ sudo su
[su] password: 
root@miguel-VirtualBox:~/home/miguel# cd /opt/smtp-li-master
bash: cd: /opt/smtp-li-master: No existe el archivo o el directorio
root@miguel-VirtualBox:~/home/miguel# cd /opt/smtp-li-master
root@miguel-VirtualBox:/opt/smtp-li-master# composer install
Do not run Composer as root/super user! See https://getcomposer.org/root for details
Continuar as root/super user [yes]? yes

```

Ya tenemos todo instalado ahora nos toca meter nuestros datos en el archivo config.json.



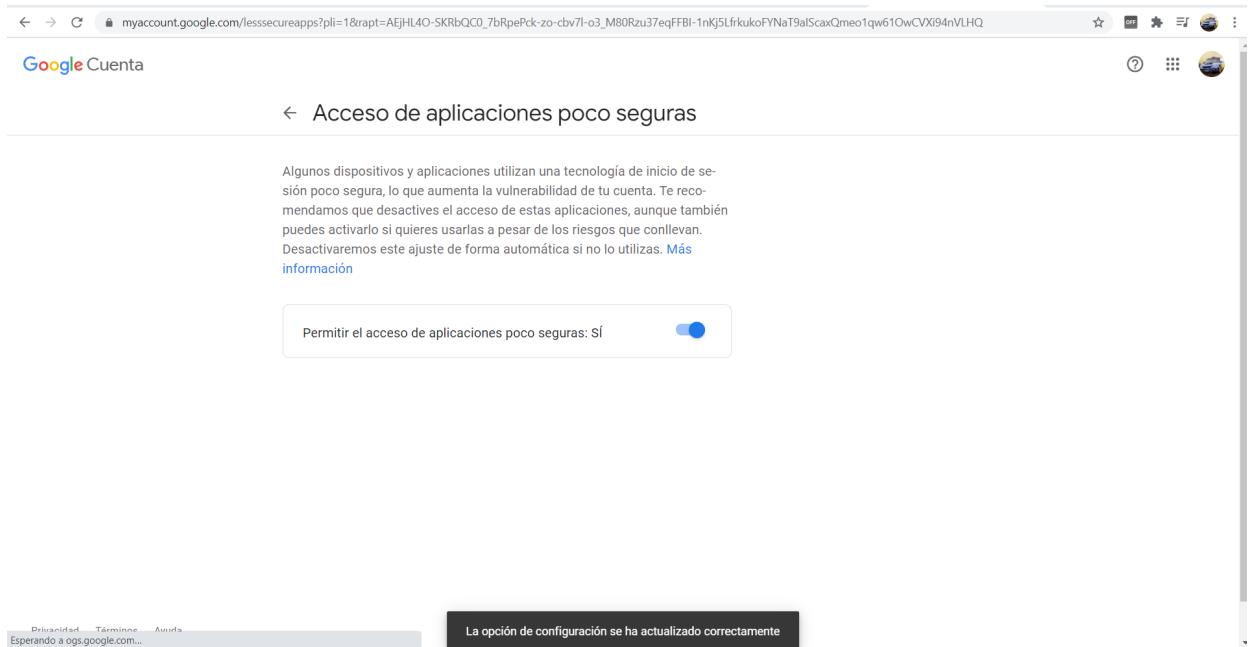
```

{
  "host": "smtp.gmail.com",
  "username": "miguelcasaresmartinez@gmail.com",
  "password": "REDACTED",
  "secure": "tls",
  "port": 587
}

```

Una vez que tenemos el SMTP configurado nos toca “conectarlo” con nuestro Nagios, para ello nos vamos al archivo /usr/local/nagios/etc/objects/commands.cfg, allí indicamos la nueva ruta del SMTP, en nuestro caso /opt/smtp-cli-master/smtp-cli.php.

Como última configuración, permitiremos en nuestro correo el acceso a aplicaciones poco seguras en nuestra cuenta de Google.



Reiniciamos Nagios y podemos comprobar que está todo correctamente en la web en configuración en el apartado contacts.

Una vez que hayamos acabado, tiraremos queriendo algún servicio, para que nos llegue la notificación y como podemos ver en la captura, nos llega nuestro correo del SMTP de Nagios con nuestro error, en que equipo está, la IP, la fecha y una descripción.

Como forma opcional, se pueden editar las notificaciones para que nos lleguen solo en las horas que nosotros queramos o solo de cierto tipo de problemas, para ello nos dirigimos hacia el archivo templates.cfg.

Todo esto se hará en el apartado define contact.

Para editar las horas que queremos que nos envíen notificaciones, como podemos ver tenemos dos apartados, para servicios y hosts, en los que tenemos varias opciones, que podemos ver en la configuración web de nagios en el apartado timeperiods.

Time Periods

Name	Alias/Description	Exclusions	Days/Dates	Times
24x7	24 Hours A Day, 7 Days A Week		sunday monday tuesday wednesday thursday friday saturday	00:00:00 - 24:00:00 00:00:00 - 24:00:00 00:00:00 - 24:00:00 00:00:00 - 24:00:00 00:00:00 - 24:00:00 00:00:00 - 24:00:00 00:00:00 - 24:00:00
24x7_sans_holidays	24x7 Sans Holidays		december 25 july 4 january 1 thursday 4 november monday 1 september monday -1 may sunday monday tuesday wednesday thursday friday saturday	00:00:00 - 00:00:00 00:00:00 - 00:00:00 00:00:00 - 00:00:00 00:00:00 - 00:00:00 00:00:00 - 00:00:00 00:00:00 - 00:00:00 00:00:00 - 24:00:00 00:00:00 - 24:00:00 00:00:00 - 24:00:00 00:00:00 - 24:00:00 00:00:00 - 24:00:00 00:00:00 - 24:00:00 00:00:00 - 24:00:00
none	No Time Is A Good Time			
us-holidays	U.S. Holidays		january 1 july 4 december 25 monday -1 may monday 1 september thursday 4 november	00:00:00 - 00:00:00 00:00:00 - 00:00:00 00:00:00 - 00:00:00 00:00:00 - 00:00:00 00:00:00 - 00:00:00 00:00:00 - 00:00:00
workhours	Normal Work Hours		monday tuesday wednesday thursday friday	09:00:00 - 17:00:00 09:00:00 - 17:00:00 09:00:00 - 17:00:00 09:00:00 - 17:00:00 09:00:00 - 17:00:00

```

opción autocaputra de teclado habilitada. Esto causará que la máquina virtual capture automáticamente el teclado cada vez que la ventana de la MV sea activada y hará que no esté disponible para otras aplicaciones ejecutándose en la máquina anfitrión, todas las combinaciones de teclas (incluyendo las del teclado numérico) se enviarán directamente a la MV.
Archivo: templates.cfg

#####
# TEMPLATES.CFG - SAMPLE OBJECT TEMPLATES
#
# NOTES: This config file provides you with some example object definition
# templates that are referred by other host, service, contact, etc.
# definitions in other config files.
#
# You don't need to keep these definitions in a separate file from your
# other object definitions. This has been done just to make things
# easier to understand.
#
#####

#####
# CONTACT TEMPLATES
#
#####

# Generic contact definition template
# This is NOT a real contact, just a template!
define contact {
    name generic-contact ; The name of this contact template
    service_notification_period 24x7 ; service notifications can be sent anytime
    host_notification_period 24x7 ; host notifications can be sent anytime
    service_notification_options w,u,c,r,f,s ; send notifications for all service states, flapping events, and scheduled downtime events
    host_notification_options d,u,r,f,s ; send notifications for all host states, flapping events, and scheduled downtime events
    service_notification_commands notify-service-by-email ; send service notifications via email
    host_notification_commands notify-host-by-email ; send host notifications via email
    register 0 ; DON'T REGISTER THIS DEFINITION - ITS NOT A REAL CONTACT, JUST A TEMPLATE!
}

#####
# HOST TEMPLATES
#
#####

# Generic host definition template
# This is NOT a real host, just a template!
define host {
    Ver ayuda Guardar Buscar Cortar Texto Justificar Posición Pág. ant. Prt. linea Siguiente Marcar texto Sangrar txt Deshacer
    Salir Leer fich. Reemplazar Pegar txt Ortografía If a linea Pág. sig. Ult. linea A llave Copiar txt Desang. txt Rehacer
}

```

Para editar el tipo de notificaciones que queremos que nos envien, tenemos en el caso de los servicios unas siglas que corresponden a: Unknown,Warning,Critical,Recovery,Flapping,Downtime.

Y en el caso de los hosts a Down, Unreachable, Recovery, Flapping, Downtime.

Simplemente pondremos el tipo de notificaciones que queramos recibir.

Además podemos, una vez que tenemos SMTP configurado, añadir más de un correo por si hubiera varios encargados de llevar Nagios, para ello, lo añadimos en el archivo contacts.cfg, poniendo además al grupo que queremos que pertenezca, ya que podemos añadir varios grupos en el archivo templates.cfg y con diferentes tipos de notificaciones y de horario.

También tendremos que añadir al grupo de admins el nuevo contacto creado.

```
Miguel-VirtualBox /usr/local/nagios/etc/objects
GNU nano 2.5.3                                     Archivo: contacts.cfg                                         17:16
Modificado

define contact {
    contact_name      nagiosadmin          ; Short name of user
    use               generic-contact       ; Inherit default values from generic-contact template (defined above)
    alias             Nagios Admin         ; Full name of user
    email             miguelcasaresmartinez@gmail.com ; <***** CHANGE THIS TO YOUR EMAIL ADDRESS *****
}

define contact {
    contact_name      Migue                ; Short name of user
    use               generic-contact2      ; Inherit default values from generic-contact template (defined above)
    alias             Miguel Casares        ; Full name of user
    email             miguelcasares2001@gmail.com ; <***** CHANGE THIS TO YOUR EMAIL ADDRESS *****
}

#####
# CONTACT GROUPS
#
#####

# We only have one contact in this simple configuration file, so there is
# no need to create more than one contact group.

define contactgroup {
    contactgroup_name   admins
    alias              Nagios Administrators
    members            nagiosadmin,Migue
}
```

Ver ayuda Ver Guardar Buscar Cortar Texto Justificar Posición Pág. ant. Pri. linea Siguiente Marcar texto Sangrar txt Deshacer Salir Leer fich. Reemplazar Pegar txt Ortografía Ir a línea Pág. stg. Últ. linea A llave Copiar txt Desang. txt Rehacer

No se nos debe olvidar activar el acceso a aplicaciones poco seguras de Google como hemos hecho con la otra cuenta.

Aquí en el archivo templates.cfg ponemos diferentes tipos de contacto.

```
define contact {
    name           generic-contact          ; The name of this contact template
    service_notification_period 24x7          ; service notifications can be sent anytime
    host_notification_period 24x7            ; host notifications can be sent anytime
    service_notification_options w,u,c,r,f,s ; send notifications for all service states, flapping events, and scheduled downtime events
    host_notification_options d,u,r,f,s     ; send notifications for all host states, flapping events, and scheduled downtime events
    service_notification_commands notify-service-by-email ; send service notifications via email
    host_notification_commands notify-host-by-email ; send host notifications via email
    register        0                          ; DON'T REGISTER THIS DEFINITION - ITS NOT A REAL CONTACT, JUST A TEMPLATE!
}

define contact {
    name           generic-contact2         ; The name of this contact template
    service_notification_period 24x7          ; service notifications can be sent anytime
    host_notification_period 24x7            ; host notifications can be sent anytime
    service_notification_options w,r,c        ; send notifications for all service states, flapping events, and scheduled downtime events
    host_notification_options d              ; send notifications for all host states, flapping events, and scheduled downtime events
    service_notification_commands notify-service-by-email ; send service notifications via email
    host_notification_commands notify-host-by-email ; send host notifications via email
    register        0                          ; DON'T REGISTER THIS DEFINITION - ITS NOT A REAL CONTACT, JUST A TEMPLATE!
}
```

También tendremos la posibilidad de crear un nuevo grupo de contacto para que solo controle unas máquinas en específico, se ve mejor con un ejemplo:

Este sería el archivo contacts.cfg:

```
define contactgroup {
    contactgroup_name      admins
    alias                  Nagios Administrators
    members                nagiosadmin
}

define contactgroup {
    contactgroup_name      trabajadores
    alias                  Trabajadores no administradores
    members                Migue
}
```

Y este el templates.cfg:

```
define host {
    name          linux-server
    use           generic-host
    check_period  24x7
    check_interval 5
    retry_interval 1
    max_check_attempts 10
    check_command  check-host-alive
    notification_period workhours
    ; The name of this host template
    ; This template inherits values from the generic-host template
    ; By default, Linux hosts are checked round the clock
    ; Actively check the host every 5 minutes
    ; Schedule host check retries at 1 minute intervals
    ; Check each Linux host 10 times (max)
    ; Default command to check Linux hosts
    ; Linux admins hate to be woken up, so we only notify during the day
    ; Note: notification_period variable is being overridden from
    ; the value that is inherited from the generic-host template!
    ; Resend notifications every 2 hours
    ; Only send notifications for specific host states
    ; Notifications get sent to the admins by default
    ; DON'T REGISTER THIS DEFINITION - ITS NOT A REAL HOST, JUST A TEMPLATE!

    notification_interval 120
    notification_options d,u,r
    contact_groups admins
    register         0
}

# Windows host definition template
# This is NOT a real host, just a template!
define host {
    name          windows-server
    use           generic-host
    check_period  24x7
    check_interval 5
    retry_interval 1
    max_check_attempts 10
    check_command  check-host-alive
    notification_period 24x7
    ; The name of this host template
    ; Inherit default values from the generic-host template
    ; By default, Windows servers are monitored round the clock
    ; Actively check the host every 5 minutes
    ; Schedule host check retries at 1 minute intervals
    ; Check each server 10 times (max)
    ; Default command to check if servers are "alive"
    ; Send notification out at any time - day or night
    ; Resend notifications every 30 minutes
    ; Only send notifications for specific host states
    ; Notifications get sent to the admins by default
    ; Host groups that Windows servers should be a member of
    ; DON'T REGISTER THIS - ITS JUST A TEMPLATE!

    notification_interval 30
    notification_options d,r
    contact_groups admins,trabajadores
    hostgroups        windows-servers
    register         0
}

# We define a generic printer template that can
# be used for most printers we monitor
define host {
```

At the bottom of the terminal window, there is a menu bar with options like Ver ayuda, Guardar, Buscar, Cortar Texto, Justificar, Posición, Pág. ant., Pág. sgt., Pág. linea, Ult. linea, A llave, Marcar texto, Sangrar txt, Deshacer, Salir, Leer fich., Reemplazar, Pegar txt, Ortografía, Ir a línea, Copiar txt, Desangr. txt, and Rehacer.

En el ejemplo que acabamos de crear, nos llegarían las notificaciones en el grupo admin de todas las máquinas, y en el grupo trabajadores solo las notificaciones de las máquinas tipo windows-server. Esto no tiene que ver con el tipo de contacto, para eso, que lo hemos explicado antes. Esto es para que un grupo de personas pueda controlar ciertas máquinas en concreto. Cuando terminemos, como siempre, reiniciamos Nagios.

Aquí vemos la comprobación del mensaje de correo.

-Conclusiones:

Un sistema de monitoreo permite ahorrar costes, una administración mucho más flexible y fácil. Acceder a todo tipo de datos (como el uso de la CPU o la memoria) y al histórico de estos y mejorar la seguridad. Además permite reducir los tiempos de resolución de errores y prevenirlas.

Dependiendo de la empresa que necesite el sistema se elegirá una herramienta u otra, pueden ser OpenSource o no, y hay muchos como Zabbix, Cacti, PandoraFMS o Nagios.

Un sistema que no es difícil de instalar y podemos conseguir una monitorización de equipos es Nagios Core. La primera configuración de un equipo no es fácil, pero una vez que te acostumbres a los ficheros de configuración es más fácil, ya al agregar equipos, se hace de forma parecida.

La herramienta en sí me ha gustado, puede que se haga en falta algún gráfico en tiempo real, aunque se pueda refrescar cada poco tiempo, hay que tener también en cuenta que la herramienta es gratuita.

Ayudándonos de páginas como stack over flow o comandos como journalctl -xe podemos encontrar posibles errores al instalar y configurar este tipo de herramientas, como es en nuestro caso Nagios Core, además esta herramienta en concreto tiene buenos foros y comunidad para poder solucionarlos.

-Bibliografía:

[Monitoreo de Redes: 16 mejores herramientas de monitorización de redes \(pandorafms.com\)](#) 17-4-2021

[12 Software de monitoreo de red para pequeñas y empresas \(geekflare.com\)](#) 12-4-2021

[Nagios Network Analyzer. Netflow Analysis and Monitoring](#) 14-4-2021

[Cacti® - The Complete RRDTool-based Graphing Solution](#) 14-4-2021

[Zabbix :: The Enterprise-Class Open Source Network Monitoring Solution](#) 14-4-2021

[Nagios - Wikipedia, la enciclopedia libre](#) 14-4-2021

[Monitorización de Sistemas Informáticos: ventajas, procedimientos e implementación \(pandorafms.com\)](#) 14-4-2021

[REDISEÑO E IMPLEMENTACIÓN DEL SISTEMA DE MONITOREO DE LA RED.pdf \(ucatolica.edu.co\)](#) 28-4-2021

[Plugins para Nagios: instalación de extensiones - IONOS](#) 5-5-2021

[Posts containing 'nagios' - Stack Overflow](#) 5-5-2021

[Nagios - Guía rápida \(hebergementwebs.com\)](#) 5-5-2021

[Monitoreo de la red con Nagios – INTRO | Sysadmins de Cuba](#) 15-5-2021

<https://library.nagios.com/library/products/nagios-core/documentation/> (de donde he sacado mucha de la información para instalar y configurar Nagios) 20-5-2021 a 30-5-2021

Miguel Casares Martínez

Administración de Sistemas Informáticos en Red

Tutor FCT: Miguel Ángel Moreno Garrido

I.E.S Zaidín Vergeles - Granada

Email:mcasares2710@ieszaidinvergeles.org

Teléfono: 625261063