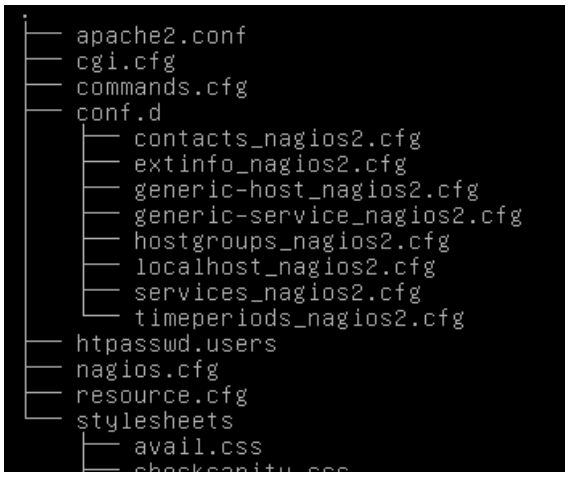
Nagios Core. Archivos de configuración

El propósito de esta artículo es revisar cuales son los archivos de configuración de Nagios y para que se usan. Uno de los problemas habituales cuando empiezas con Nagios es saber donde hay que configurar correctamente los objetos que definamos cuando la realidad es que a Nagios, en general, le da igual donde estén mientras pueda leer los ficheros que los contienen.
Comentaré dos casos habituales: instalación independiente de Nagios (por paquetes de la distro) e instalación integrado en la de OMD.

Estructura en instalación clásica de Nagios Core.

Si echamos un vistazo al directorio etc de una instalación de Nagios por paquetes en Debian 6 veremos algo así (/etc/nagios3).



Quitando el directorio stylsheets que son las hojas de estilo de interface web y no nos interesan empezamos de arriba a abajo:

- **apache2.conf** _ Archivo de configuración de Apache para el site de Nagios. Existe un enlace simbólico a este archivo en el directorio de configuración de Apache (/etc/apache2/conf.d/nagios3.conf). En dicho archivo hace referencia al fichero de passwords para identificarse en el interface de Nagios, htpasswd.users, que contiene las claves de los usuarios (del clásico nagiosadmin). Veremos ese fichero en otra entrada donde hablemos de la gestión de usuarios en Nagios.
- **cgi.cfg** __Archivo de configuración para los cgis que usa el interface web de nagios. Aquí tenemos configuraciones interesantes como **donde localiza el archivo principal de configuración de nagios, nagios.cfg** (valor de cfg_dir), los usuarios que pueden acceder al interface y a qué en concreto y alguna parametrización relacionada con la visualización.
- **commands.cfg** _ Archivo con objetos tipo comando de Nagios. Este si ya contiene definición de objetos.
- **nagios.cfg** Este es el principal archivo de configuración de nagios, de momento nos

interesan dos variables que nos dirán **donde están TODOS los archivos de configuración de OBJETOS que usará Nagios** Core. Editando el fichero las veremos al principio:

```
# Commands definitions
cfg_file=/etc/nagios3/commands.cfg
# Debian also defaults to using the check commands defined by the debian
# nagios-plugins package
cfg_dir=/etc/nagios-plugins/config
# Debian uses by default a configuration directory where nagios3-common,
# other packages and the local admin can dump or link configuration
# files into.
cfg_dir=/etc/nagios3/conf.d
```

En este fichero podemos definir tantas directivas cfg_file y cfg_dir como queramos. La directiva cfg_dir leerá todos los ficheros de ese directorio **y subdirectorios**. Solo contempla los ficheros que **acaban en ".cfg"**. Bastará con renombrar la extensión de un fichero para que no lo tenga en cuenta.

Vemos que también lee los ficheros del directorio configuración de los plugins de Nagios. En esa ubicación nos encontraremos un montón de ficheros cada uno de ellos definiendo el objeto "command" adecuado para llamar al plugin.

- **resources.cfg** Este es un fichero un tanto especial. Contiene macros globales que podremos usar en todo el entorno de Nagios. Se define también con una directiva en el fichero nagios.cfg. He dejado los comentarios ya que son muy clarificadores. En este articulo hablo de este tema.

```
# Commands
# RESOURCE FILE
# This is an optional resource file that contains $USERx$ macro
# definitions. Multiple resource files can be specified by using
# multiple resource_file definitions.  The CGIs will not attempt to
# read the contents of resource files, so information that is
# considered to be sensitive (usernames, passwords, etc) can be
# defined as macros in this file and restrictive permissions (600)
# can be placed on this file.
resource_file=/etc/nagios3/resource.cfg
```

Eso es todo en cuanto a la ubicación de ficheros de configuración de objetos. Los objetos que definamos pueden estar en cualquier fichero. Si es cierto y vemos de hecho que en el directorio "conf.d" hay varios ficheros que parecen separar por nombre en estos la definición de distintos objetos. Se suele hacer de esta manera por "higiene" mental , para que tenga un sentido para nosotros nada más. Podemos usar la forma que más nos convenga.

Estructura en instalación con OMD.

En una instalación de OMD, nuestros ficheros estarán por defecto en /omd/sites/nombre_instancia/etc/nagios

```
root@debian6:/omd/sites/foo/etc/nagios# tree -A
.
├── apache.conf
├── cgi.cfg
├── conf.d
│   ├── check_mk_objects.cfg
│   ├── check_mk_templates.cfg
│   ├── commands.cfg
│   ├── grupos.cfg
│   ├── jmx4perl_nagios.cfg
│   ├── notification_commands.cfg
│   ├── pnp4nagios.cfg
│   ├── templates.cfg
│   └── timeperiods.cfg
├── config.inc.php
├── nagios.cfg
├── nagios.d
│   ├── dependency.cfg
│   ├── eventhandler.cfg
│   ├── flapping.cfg
│   ├── freshness.cfg
│   ├── logging.cfg
│   ├── misc.cfg
│   ├── mk-livestatus.cfg -> ../../mk-livestatus/nagios.cfg
│   ├── obsess.cfg
│   ├── omd.cfg
│   ├── pnp4nagios.cfg -> ../../pnp4nagios/nagios_npcdmod.cfg
│   ├── retention.cfg
│   ├── timing.cfg
│   └── tuning.cfg
├── resource.cfg
└── ssi
    ├── extinfo-header.ssi
    ├── README
    └── status-header.ssi
```

Vemos una estructura similar en cuanto a los ficheros que contiene. Si miramos en su nagios.cfg nos sorprenderá que… ¡está vacío!. Miramos entonces en cgi.cfg que nos tiene que decir con el valor de "main_config" donde está el fichero de configuración de Nagios.

```
main_config_file=/omd/sites/foo/tmp/nagios/nagios.cfg
```

Nos los está mandando a un directorio temporal. La razón de este es que OMD "controla" los aspectos de los ficheros de configuración de Nagios para modificarlos cuando necesite. Si vamos a buscarlo a dicho directorio y vemos las directivas de configuración nos aclarará este tema y veremos que solo hay una entrada cfg_dir que apuna al directorio concreto donde estarán los

ficheros.

```
# This file has been created by OMD out of the following files:
# /omd/sites/foo/etc/nagios/nagios.d/dependency.cfg
# /omd/sites/foo/etc/nagios/nagios.d/eventhandler.cfg
# /omd/sites/foo/etc/nagios/nagios.d/flapping.cfg
…
# Do not edit this file. It will be recreated each time Nagios
# is started or reloaded. Rather change things in the original
# files.
…
# /omd/sites/foo/etc/nagios/nagios.d/omd.cfg:26
cfg_dir=/omd/sites/foo/etc/nagios/conf.d
```

¿Y el directorio nagios.d? En dicho directorio parece que OMD divide el fichero principal de Nagios por secciones / usos y se vale luego de estos para crear su fichero nagios.conf. Resaltamos en el subdirectorio "conf.d" dos archivos importantes:

- **check_mk_templates.cfg** Este archivo contiene las definiciones de objetos (plantillas y otros) en formato Nagios propias de check_mk, en las que se apoya para crear luego todos los objetos de Nagios. No se debe editar.
- **check_mk_objects.cfg** Este archivo contiene los objetos generados en formato Nagios a partir de la definición de estos realizar en el fichero main.mk de check_mk. Este fichero se regenera cada vez que añadimos objetos en check_mk y recargamos la configuración como explicábamos en este artículo. Este lógicamente al ser dinámico no deberiamos a editarlo nunca.
- El **resto de archivos cfg** de este directorio contienen objetos reales de Nagios. Comentamos previamente que check_mk no sustituye completamente todos los objetos de Nagios y algunos es necesario definirlos en los ficheros de este.

Por si quieres empezar a conocer más sobre los objetos en Nagios.