

# Monit

## Sumario

- 1 Instalación de monit
- 2 Configuración de Monit
  - ◆ 2.1 Activación de la interfaz http
  - ◆ 2.2 Definición de directivas genéricas
    - ◇ 2.2.1 Período de ejecución de chequeos
    - ◇ 2.2.2 Credenciales smtp para envío de alertas al mail
    - ◇ 2.2.3 Formato del mail enviado
- 3 Definición de componentes a monitorizar
  - ◆ 3.1 Monitorización de parámetros generales del sistema
  - ◆ 3.2 Monitorización de un servicio
  - ◆ 3.3 Monitorización del sistema de archivos
  - ◆ 3.4 Monitorización de host
  - ◆ 3.5 Monitorización de un enlace de red
- 4 Referencias

## Instalación de monit

Veremos el proceso de instalación en un Sistema Debian 9 (Stretch)

Monit solo requiere de la instalación de un paquete

```
apt update
apt install monit
```

## Configuración de Monit

### Activación de la interfaz http

Monit dispone de un servidor http integrado. A través de esta interfaz http puede accederse al estado de los servicios monitorizados por Monit, aunque también se utiliza esta interfaz para el envío de comandos Monit desde la línea de comandos.

Para activar la interfaz http editamos el archivo **/etc/monit/monitrc**

```
vi /etc/monit/monitrc
```

Localizamos la sección comentada

```
# set httpd port 2812 and
# use address localhost # only accept connection from localhost
# allow localhost # allow localhost to connect to the server and
# allow admin:monit # require user 'admin' with password 'monit'
```

Y la editamos para que quede

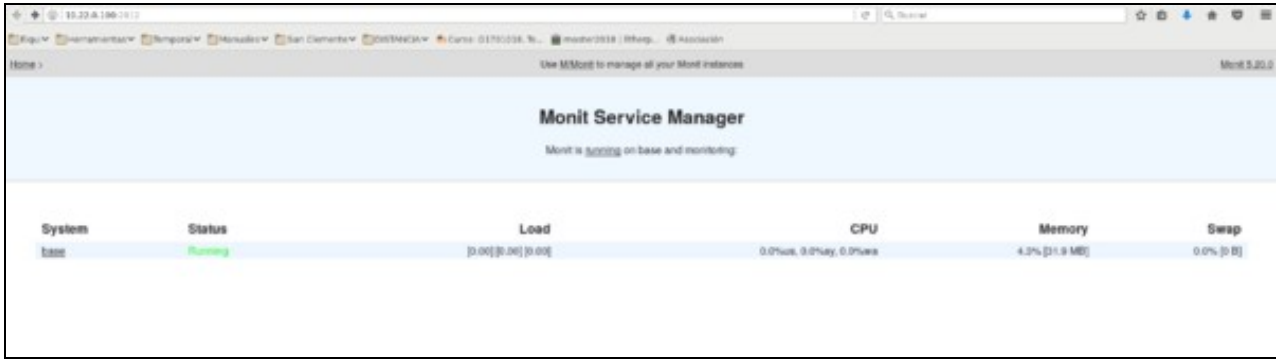
```
set httpd port 2812 and
use address 0.0.0.0
allow 0.0.0.0/0 # permitir todos
allow admin:monit # require user 'admin' with password 'monit'
```

Con esta configuración podremos acceder al servicio http utilizando la URL

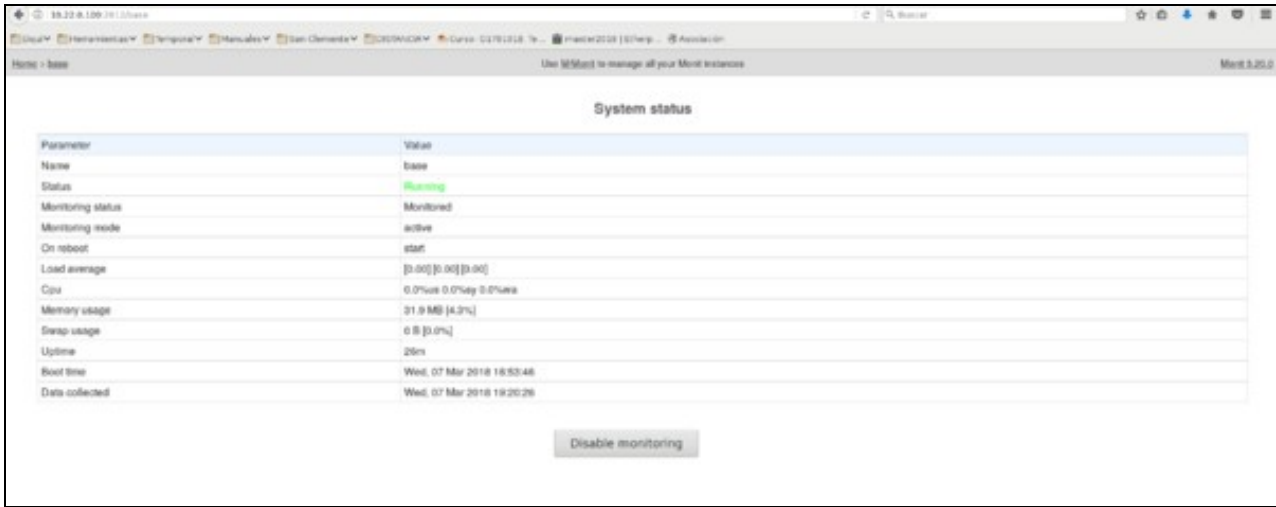
[http://IP\\_Monit\\_Server:2812](http://IP_Monit_Server:2812)

Para acceder a la información de monitorización usaremos (como se define en la configuración anterior)

user: **admin** password: **monit**



Pinchando en el enlace el host accedemos al estado del mismo



Definición de directivas genéricas

Algunas directivas interesantes que pueden ser establecidas en el archivo **monitrc**

Período de ejecución de chequeos

set daemon

```
set daemon 120
```

Indica que se hará un chequeo de los servicios a monitorizar cada 2 minutos, puede ser modificado si necesitamos una periodicidad mayor

Credenciales smtp para envío de alertas al mail

set mailserver

Permite definir un servidor SMTP para el envío de alertas por correo. Por ejemplo podemos establecer la siguiente definición de esa directiva

```
# Configuración de servidor de correo para alertas de Monit
set mailserver smtp.gmail.com port 587
username "monitortest" password "abc123."
using tlsv1
# Configuración del buzón al que se enviarán las alertas
set alert monitortest@gmail.com
```

**NOTA:** Hay que efectuar un par de pasos adicionales en la cuenta de gmail para poder habilitar el envío utilizando este sistema.

ATENCIÓN A LAS URLs



## ← Aplicaciones menos seguras

Algunos dispositivos y aplicaciones utilizan una tecnología de inicio que tu cuenta sea más vulnerable, por lo que te recomendamos que aplicaciones. Si, a pesar del riesgo que ello supone, quieres utilizarla [información](#)

Permitir el acceso de aplicaciones menos seguras: ☒ SÍ



https://accounts.google.com/b/0/DisplayUnlockCaptcha

Eiqui ▾

Herramientas ▾

Temporal ▾

Manuales ▾

San Clemente ▾

DISTANCIA ▾

Curso: C

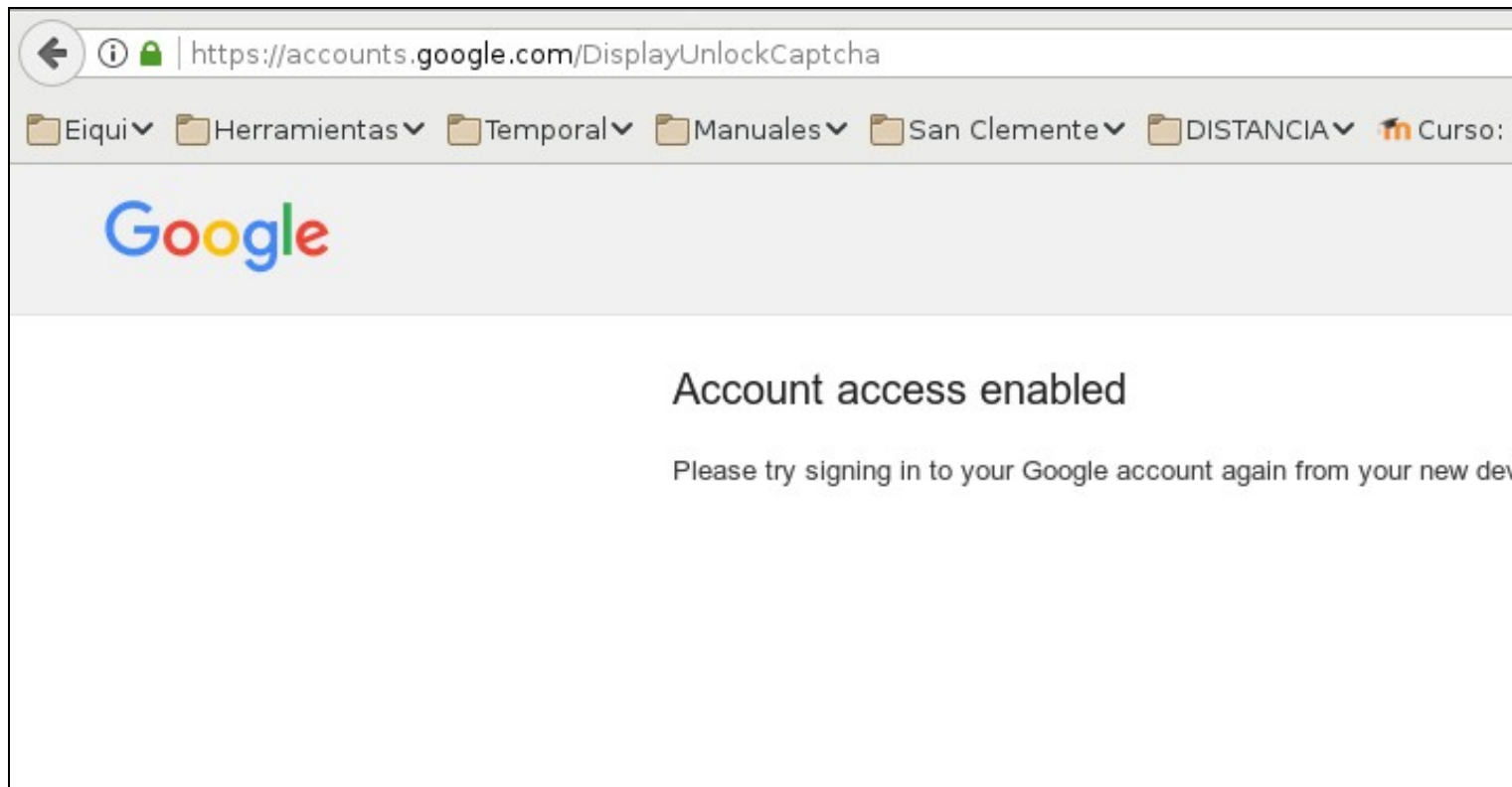


## Allow access to your Google account

As a security precaution, Google may require you to complete this additional security check on your device or application.

To allow access, click the Continue button below.

Continue



#### Formato del mail enviado

##### set mail-format

Esta directiva permite establecer el formato del correo de alerta que se enviará cuando se produzca alguno de los eventos notificados por monit. Por defecto utiliza un formato de plantilla de mail genérico que puede ser modificado con esta directiva

Podemos acceder a los parámetros del **runtime** desde el enlace [Monit is running](#) on base and monitoring de la página principal:

Home > Runtime		Use M:Monit to manage all your Monit instances	Monit 5.20.0
Monit runtime status			
Parameter	Value		
Monit ID	38f0d3a3b175c427b1734f5d07d4b862		
Host	base		
Process id	1267		
Effective user running Monit	root		
Controllfile	/etc/monit/monitrc		
Logfile	/var/log/monit.log		
Pidfile	/run/monit.pid		
State file	/var/lib/monit/state		
Debug	False		
Log	True		
Use syslog	False		
Event queue	base directory /var/lib/monit/events with 100 slots		
Mail server(s)	smtp.gmail.com:587 using SSL/TLS with options (version: fsv1)		
Limit for Send/Expect buffer	256 B		
Limit for file content buffer	512 B		
Limit for HTTP content buffer	1024 kB		
Limit for program output	512 B		
Limit for network timeout	5 s		
Limit for check program timeout	5 m		
Limit for service stop timeout	30 s		
Limit for service start timeout	30 s		
Limit for service restart timeout	30 s		
On reboot	start		

## Definición de componentes a monitorizar

Es posible definir las directivas de monitorización directamente en el archivo `monitrc`. Sin embargo, a efectos organizativos y aclaratorios es más recomendable definir las directivas en archivos dentro del directorio `/etc/monit/conf.d`

A continuación vamos a crear varios archivos de texto en los que se incorporarán directivas de monitorización específicas para distintos casos de uso.

## Monitorización de parámetros generales del sistema

Crearemos un archivo `/etc/monit/conf.d/system.monit`

```
vi /etc/monit/conf.d/system.monit
```

Dentro del archivo incorporaremos las directivas siguientes

```
check system $HOST
if loadavg (1min) > 4 then alert
if loadavg (5min) > 2 then alert
if cpu usage > 95% for 10 cycles then alert
if memory usage > 75% then alert
if swap usage > 25% then alert
```

El lenguaje de definición de directivas de monitorización es bastante claro e intuitivo, utilizando prolijamente la estructura condicional `if`. Ante cualquier duda de sintaxis o semántica acerca de las mismas se recomienda consultar el **Monit Manual** de la sección **Referencias**

Tras editar el fichero reiniciamos el servicio

```
service monit restart
```

## Monitorización de un servicio

Vamos a ver un conjunto de directivas de monitorización para el servicio `apache2`. Si este servicio no está instalado en la máquina lo instalamos previamente

```
apt install -y apache2
```

A continuación crearemos un archivo `/etc/monit/conf.d/apache2.monit`

```
vi /etc/monit/conf.d/apache2.monit
```

Dentro de ese archivo incorporaremos las siguientes directivas de monitorización

```
check process apache with pidfile /var/run/apache2/apache2.pid
start program = "/etc/init.d/apache2 start" with timeout 60 seconds
stop program = "/etc/init.d/apache2 stop"
if cpu > 60% for 2 cycles then alert
if cpu > 80% for 5 cycles then restart
if totalmem > 200.0 MB for 5 cycles then restart
if children > 250 then restart
if loadavg(5min) greater than 10 for 8 cycles then stop
if failed host localhost port 80 protocol http
and request "/index.html"
then restart
# if failed port 443 protocol https with timeout 15 seconds then restart
# if 3 restarts within 5 cycles then unmonitor
group server
```

En este bloque de definición se hace uso de las directivas **start program** y **stop program** las cuales se utilizan en caso de ser necesario ejecutar el reinicio (restart) del servicio. Por ejemplo en la línea

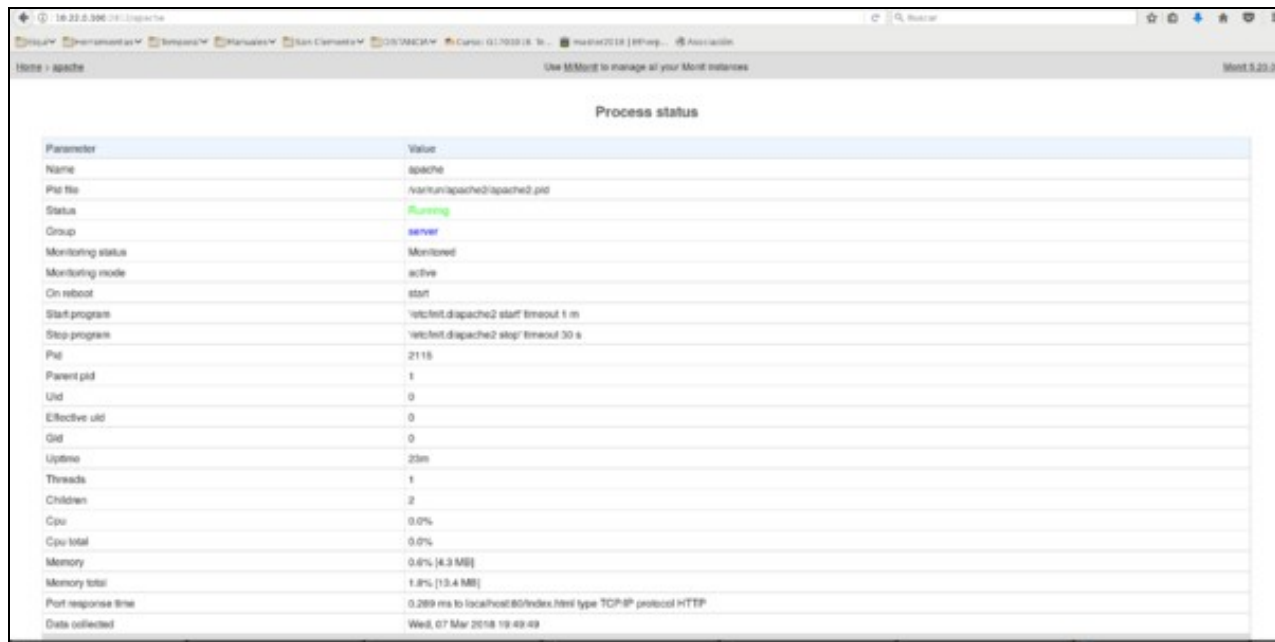
**if totalmem > 200.0 MB for 5 cycles then restart**

... en caso de que la memoria total del proceso monitorizado excediera los 200MB durante 5 períodos de monitorización consecutivos se ejecutaría un reinicio del servicio, para ello se invocaría primero al comando de la directiva **stop program** y a continuación al comando de **start program**

Tras editar el fichero reiniciamos el servicio

```
service monit restart
```

Página de monitorización de Apache;



The screenshot shows the Monit web interface in a browser. The page title is "Process status" and it displays a table with various parameters and their values for the Apache process.

Parameter	Value
Name	apache
Pid file	/var/run/apache2/apache2.pid
Status	Running
Group	server
Monitoring status	Monitored
Monitoring mode	active
On reboot	start
Start program	/etc/init.d/apache2 start timeout 1 m
Stop program	/etc/init.d/apache2 stop timeout 30 s
Pid	2115
Parent pid	1
Uid	0
Effective uid	0
Gid	0
Uptime	25m
Threads	1
Children	2
Cpu	0.0%
Cpu total	0.0%
Memory	0.6% (4.3 MB)
Memory total	1.8% (13.4 MB)
Port response time	0.289 ms to localhost/80/index.html type TCP-IP protocol HTTP
Date collected	Wed, 07 Mar 2018 19:49:49

## Monitorización del sistema de archivos

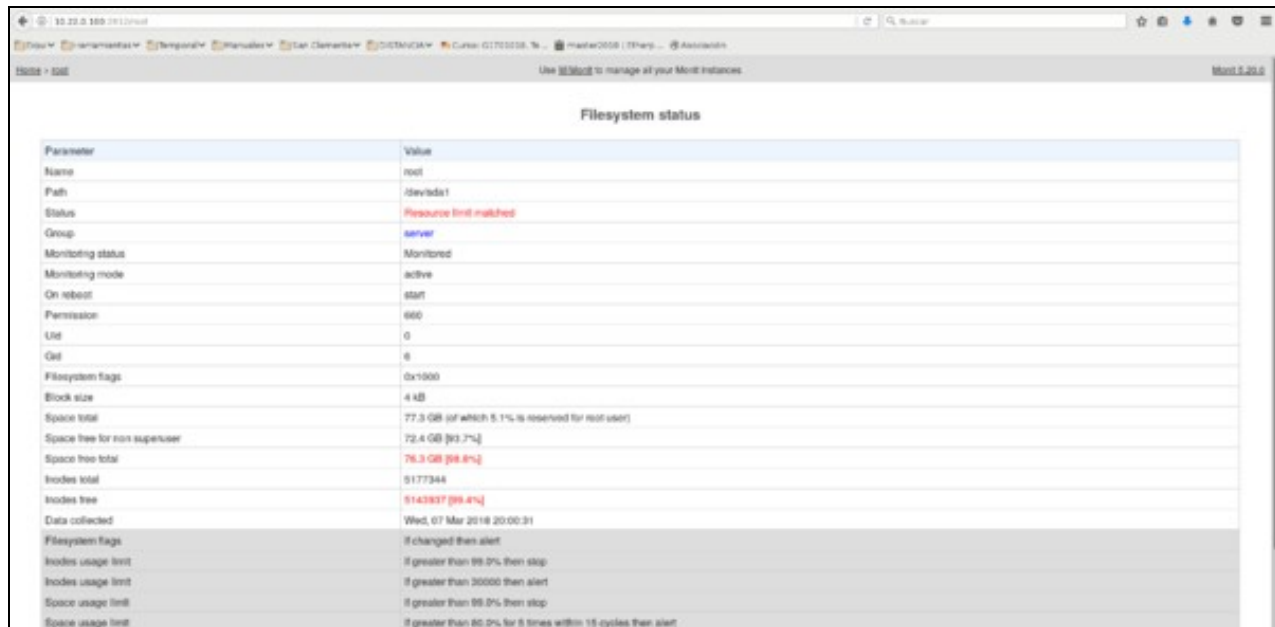
A continuación crearemos un archivo `/etc/monit/conf.d/filesystem.monit`

```
vi /etc/monit/conf.d/filesystem.monit

check filesystem root with path /dev/sda1
if space usage > 80% for 5 times within 15 cycles then alert
if space usage > 99% then stop
if inode usage > 30000 then alert
if inode usage > 99% then stop
group server
```

Tras editar el fichero reiniciamos el servicio

```
service monit restart
```



The screenshot shows the Monit web interface with the title "Filesystem status". It displays a table of parameters and their values for the root filesystem. The status is "Resource limit matched".

Parameter	Value
Name	root
Path	/dev/sda1
Status	Resource limit matched
Group	server
Monitoring status	Monitored
Monitoring mode	active
On reboot	start
Permission	660
Unit	0
Get	6
Filesystem flags	0x1000
Block size	4 KB
Space total	77.3 GB (of which 5.1% is reserved for root user)
Space free for non superuser	72.4 GB [93.7%]
Space free total	76.3 GB [98.6%]
Inodes total	5177344
Inodes free	5143937 [99.4%]
Data collected	Wed, 07 Mar 2018 20:00:31
Filesystem flags	if changed then alert
Inodes usage limit	if greater than 99.0% then stop
Inodes usage limit	if greater than 30000 then alert
Space usage limit	if greater than 99.0% then stop
Space usage limit	if greater than 80.0% for 5 times within 15 cycles then alert

## Monitorización de host

A continuación crearemos un archivo `/etc/monit/conf.d/host.monit`

```
vi /etc/monit/conf.d/host.monit

check host localhost with address 127.0.0.1
if failed ping then alert
if failed port 3306 protocol mysql with timeout 15 seconds then alert
if failed port 80 protocol http
then alert
```

En este caso hacemos un chequeo del host local consistente en varias acciones

- ejecución de un ping
- chequeo del puerto 3306 local (mysql)
- chequeo del puerto 80 local (http)

En caso de fallo de cualquiera de los chequeos se envía una alerta

Tras editar el fichero reiniciamos el servicio

```
service monit restart
```



## Monitorización de un enlace de red

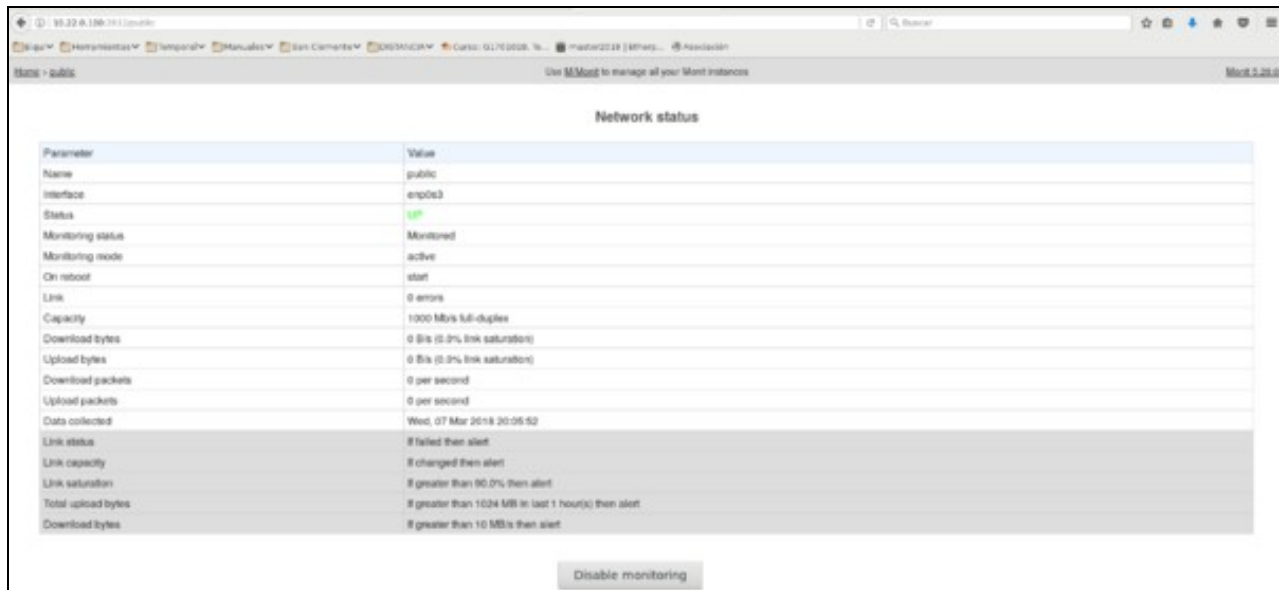
A continuación crearemos un archivo `/etc/monit/conf.d/network.monit`

```
vi /etc/monit/conf.d/network.monit

check network public with interface enp0s3
if failed link then alert
if changed link then alert
if saturation > 90% then alert
if download > 10 MB/s then alert
if total uploaded > 1 GB in last hour then alert
```

Tras editar el fichero reiniciamos el servicio

```
service monit restart
```



Network status

Parameter	Value
Name	public
Interface	enp0s3
Status	UP
Monitoring status	Monitored
Monitoring mode	active
On reboot	start
Link	0 errors
Capacity	1000 Mbps full-duplex
Download bytes	0 B/s (0.0% link saturation)
Upload bytes	0 B/s (0.0% link saturation)
Download packets	0 per second
Upload packets	0 per second
Data collected	Wed, 07 Mar 2018 20:05:52
Link status	If failed then alert
Link capacity	If changed then alert
Link saturation	If greater than 90.0% then alert
Total upload bytes	If greater than 1024 MB in last 1 hour(s) then alert
Download bytes	If greater than 10 MB/s then alert

Disable monitoring

## Referencias

<https://mmonit.com/monit/documentation/monit.html>

Volver

JavierFP 18:42 09 jue 2018 (CET)