



**Brais Lestón Lucas**

**1º ASIR MP**

# Índice

<b>Introducción.....</b>	<b>3</b>
<b>Definición.....</b>	<b>4</b>
<b>Sistema Operativo e distribución.....</b>	<b>5</b>
<b>Requisitos do sistema.....</b>	<b>6</b>
Memoria e espazo en disco.....	6
Compilador ANSI-C e sistema de compilación.....	6
<b>Configuración Servidor e Cliente.....</b>	<b>7</b>
<b>Instalación Monit.....</b>	<b>9</b>
<b>Configuración Monit.....</b>	<b>10</b>
<b>Definición de compoñentes a monitorizar.....</b>	<b>12</b>
Monitorización de parámetros xenerais do sistema.....	12
Monitorización do sistema de ficheiros.....	14
Monitorización de host.....	15
Monitorización dun enlace de rede.....	16
<b>Agregar servizos de monitoreo.....</b>	<b>19</b>
Instalación e monitorización de MySQL.....	19
Instalación de PHP.....	23
Monitorización de una aplicación python.....	25
Configuración avisos correo electrónico.....	29
Monitorización de servicio Apache.....	33
<b>Configuración equipo cliente para monitorización.....</b>	<b>35</b>
Monitorización de system e filesystem.....	36
Monitorización de SSH.....	37
Monitorización de MySQL.....	39
Monitorización de Apache.....	40
<b>Configuración para monitorización cliente-servidor</b>	
<b>.....</b>	<b>41</b>

<b>Conclusión.....</b>	<b>45</b>
<b>Webgrafía.....</b>	<b>48</b>

## ***Introducción***

Para a documentación deste traballo procederase en primeiro lugar a facer unha breve descripción do que é o sistema monit así como os requisitos e pasos que son necesarios para a súa correcta configuración.

## ***Definición***

Monit é unha ferramenta de software libre para monitorización de procesos de sistema Unix e Linux, tales como procesos, arquivos, directorios e sistemas de arquivo. Con este sistema, o estado do sistema pode ser visto directamente dende o intérprete de comandos ou dende o servidor web HTTP(S) nativo; resulta posible tamén realizar con él un mantemento e reparación automáticos así como a execución de accións causais en caso de erro.

## ***Sistema Operativo e distribución***

O sistema operativo usado para este caso é Linux, usando a distribución Ubuntu, na súa versión Escritorio(Desktop) e na súa versión Servidor, ambos na última edición sendo a 22.04.2.

## ***Requisitos do sistema***

### **Memoria e espazo en disco.**

Requírese un mínimo de 1 MB de RAM e arredor de 500 KB de espazo posible, sendo esto ampliable en cantidade en función dos requisitos que se necesiten monitorizar con este sistema.

### **Compilador ANSI-C e sistema de compilación.**

Será preciso empregar un compliador ANSI-C99 para a correcta compilación de Monit, resultando incluso recomendable usar o compilador GNU C (GCC) da Free Software Foundation (FSF). Ademáis, a súa ruta deberá conter ferramentas de compilación básicas como make.

## **Configuración Servidor e Cliente**

En ambas máquinas usaremos unha Red NAT 172.20.0.0/16, e un espazo de disco de 50GB.

Crearemos mínimamente 4 particións, /, /boot, /home e unha partición swap.

Unha vez feita a instalación do servidor, accederemos ao ficheiro /etc/hosts para configurar IP e nome do dominio, así como nome significativo do equipo.

Previo a configuración da rede na máquina, instalaremos a utilidade de rede co comando:

```
sudo apt install net-tools.
```

A continuación editamos o arquivo, con permisos de root, /etc/netplan/00-installer-config.yaml, neste arquivo ao usar a linguaxe yaml, deberemos de ter especial coidado, pois é un tipo de linguaxe que só admite espazo e non tabulador, e que está a ser usado cada vez máis debido a que facilita a lexibilidade e a capacidade de estrutura e permite o seu uso con todos as linguaxes de programación.

Unha vez editado o ficheiro, executamos o comando:

```
netplan try
```

para comprobar que non existen erros no mesmo, e unha vez confirmado realizamos o comando:

```
netplan apply
```

para aplicar a configuración de rede estática seleccionada.



```
root@servidor:/# cat /etc/netplan/00-installer-config.yaml
# This is the network config written by 'subiquity'
network:
  ethernets:
    enp0s3:
      addresses:
        - 172.20.0.10/16
      dhcp4: false
      routes:
        - to: default
          via: 172.20.0.1
      nameservers:
        addresses:
          - 8.8.8.8
          - 8.8.4.4
  version: 2
root@servidor:/#
```

Figura 1: Exemplo de configuración en servidor

## ***Instalación Monit***

De maneira determinada, a ferramenta Monit non se atopa dispoñible nos repositorios base do sistema, para instalala, faremolo facilmente usando o seguinte comando:

```
sudo apt install monit.
```

## Configuración Monit

A súa configuración resulta sinxela, pois os ficheiros están creados para ser fáciles de leer e para que os estudantes o entendan máis facilmente, xa que ademais atópase configurado para monitorizar os servizos en execución cada 2 minutos e mantén os rexistros no ficheiro `/var/log/monit`.

Monit ten a súa interfaz web que se executa no porto 2812 mediante o servidor web. Para habilitar a interfaz web, debemos de realizar cambios no ficheiro de configuración de monit, sendo o ficheiro de configuración principal de monit `/etc/monit/monitrc`, abrindo para a súa edición co seguinte comando:

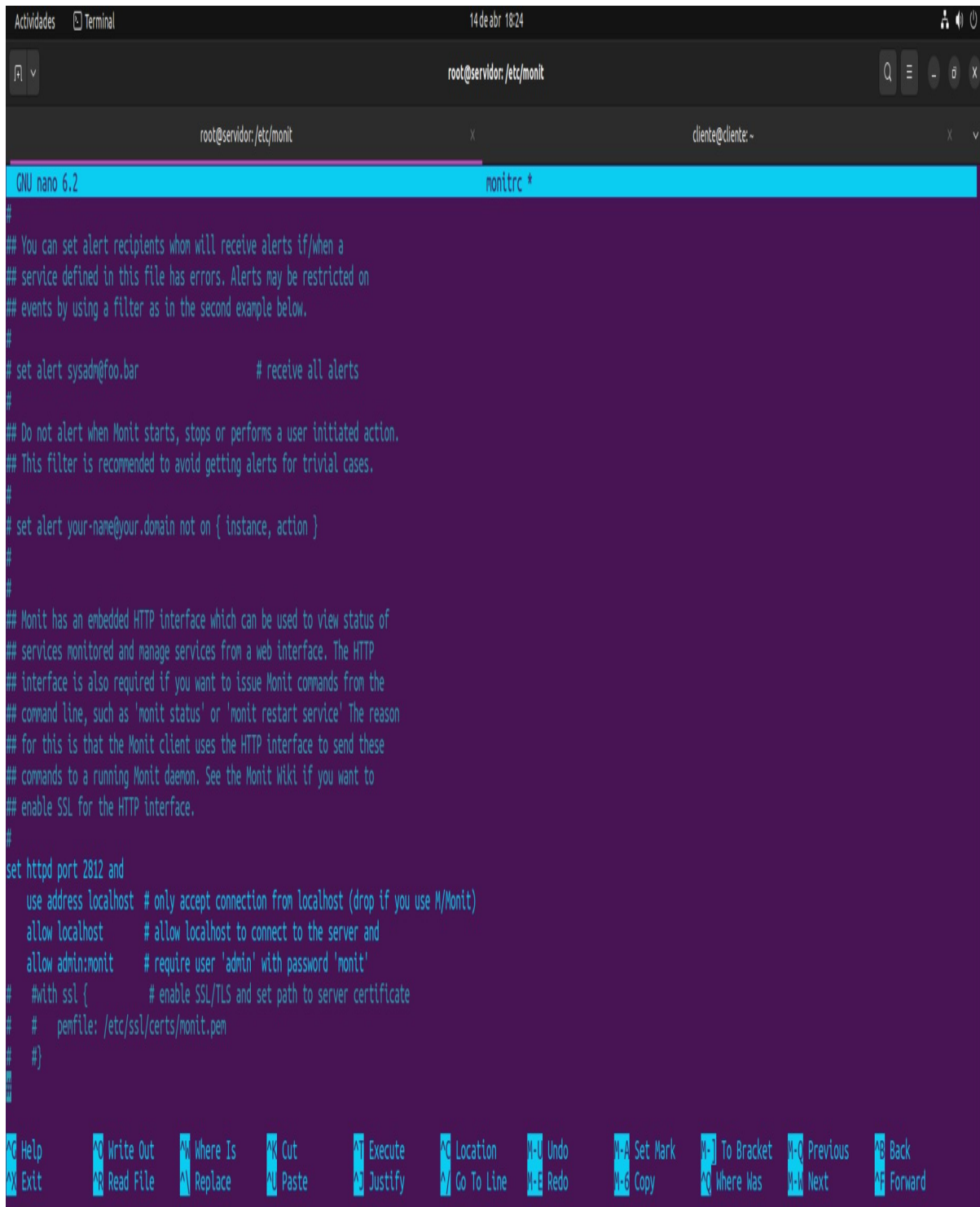
```
sudo nano /etc/monit/monitrc
```

A continuación, descomentamos a seguinte sección e agregamos a dirección IP ou o nome de dominio do seu servidor, permitindo segundo que se configure un maior ou menor nivel de seguridade, isto debido a que si só descomentamos non alteraremos os contrasinais que ven por defecto, pero o poder ser editado, podemos nós establecer o nivel de seguridade que se considere máis axeitado, dependendo a situación.

Unha vez configurado, debemos iniciar o servizo monit para volver a cargar os axustes de configuración, isto farémolo co comando:

```
sudo /etc/init.d/monit start
```

Ahora poderemos acceder á interfaz web de monit navegando a `"http://IP_SERVIDOR:2812"`. Despois ingresarase co nome de usuario e contrasinal que definiramos no ficheiro de configuración.



```
GNU nano 6.2 monitrc *
#
## You can set alert recipients whom will receive alerts if/when a
## service defined in this file has errors. Alerts may be restricted on
## events by using a filter as in the second example below.
#
# set alert sysadm@foo.bar          # receive all alerts
#
## Do not alert when Monit starts, stops or performs a user initiated action.
## This filter is recommended to avoid getting alerts for trivial cases.
#
# set alert your-name@your.domain not on { instance, action }
#
#
## Monit has an embedded HTTP interface which can be used to view status of
## services monitored and manage services from a web interface. The HTTP
## interface is also required if you want to issue Monit commands from the
## command line, such as 'monit status' or 'monit restart service'. The reason
## for this is that the Monit client uses the HTTP interface to send these
## commands to a running Monit daemon. See the Monit Wiki if you want to
## enable SSL for the HTTP interface.
#
set httpd port 2812 and
  use address localhost # only accept connection from localhost (drop if you use W/Monit)
  allow localhost      # allow localhost to connect to the server and
  allow admin:monit    # require user 'admin' with password 'monit'
#
#with ssl {           # enable SSL/TLS and set path to server certificate
#  # pemfile: /etc/ssl/certs/monit.pem
#}
#
```

Figura 2: Configuración interfaz web Monit

## **Definición de compoñentes a monitorizar**

Aínda que como vimos ata agora, é posible definir as directivas de monitorización directamente no ficheiro `monitrc`, pero debemos de ter en conta que a efectos organizativos e aclaratorios é máis recomendable definir as directivas en ficheiros dentro do directorio `/etc/monit/conf.d`.

A continuación vamos a crear varios ficheiros de texto nos que se incorporarán directivas de monitorización específicas para distintos casos de uso.

### **Monitorización de parámetros xenerais do sistema**

Para este paso en primeiro lugar crearemos o ficheiro co seguinte comando:

```
sudo nano /etc/monit/conf.d/system.monit.
```

Dentro deste ficheiro incorporamos as seguintes directivas:

```
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
```

A linguaxe de definición de directivas de monitorización é claro e intuitivo dabondo, utilizando de xeito asiduo a estrutura condicional `if`.

Tras editar o ficheiro reiniciamos o servizo co comando:

```
sudo systemctl status monit.service
```

```

root@servidor:/etc/monit/conf.d# ls
apache.monit filesystem.monit host.monit mysql.monit network.monit php-fpm.monit python-webapp.monit remote-client.monit system.monit
root@servidor:/etc/monit/conf.d# cat system.monit
check system SHOST
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

alert a22brasil@lessanclenente.net
root@servidor:/etc/monit/conf.d#

```

Figura 3: Configuración ficheiro System

System status

Parameter	Value
Name	servidor
Status	Resource limit matched
Monitoring status	Monitored
Monitoring mode	active
On reboot	start
Load average	[0.03] [0.11] [0.15]
Cpu	0.0%usr 0.0%sys 0.0%nice 0.0%idlewait 0.0%hardirq 0.1%softirq 0.0%steal 0.0%guest 0.0%guestnice
Memory usage	727.6 MB (75.1%)
Swap usage	5.8 MB (0.3%)
Uptime	22m
Boot time	Sun, 07 May 2023 05:51:38
Filedescriptors	1536 [0.0% of 922337203685475807 limit]
Data collected	Sun, 07 May 2023 06:12:43
Swap usage limit	If greater than 25.0% then alert
Memory usage limit	If greater than 75.0% then alert
CPU usage limit	If greater than 95.0% for 10 cycles then alert
Load average (5m)	If greater than 2.0 then alert
Load average (1m)	If greater than 4.0 then alert
Alert mail to	a22brasil@lessanclenente.net
Alert on	All events

[Disable monitoring](#)

Copyright © 2003-2022 [Meehan](#). All rights reserved. [Monit web site](#) [Monit code](#) [Monit docs](#)

Figura 4: Interfaz web Monit con System

## Monitorización do sistema de ficheiros

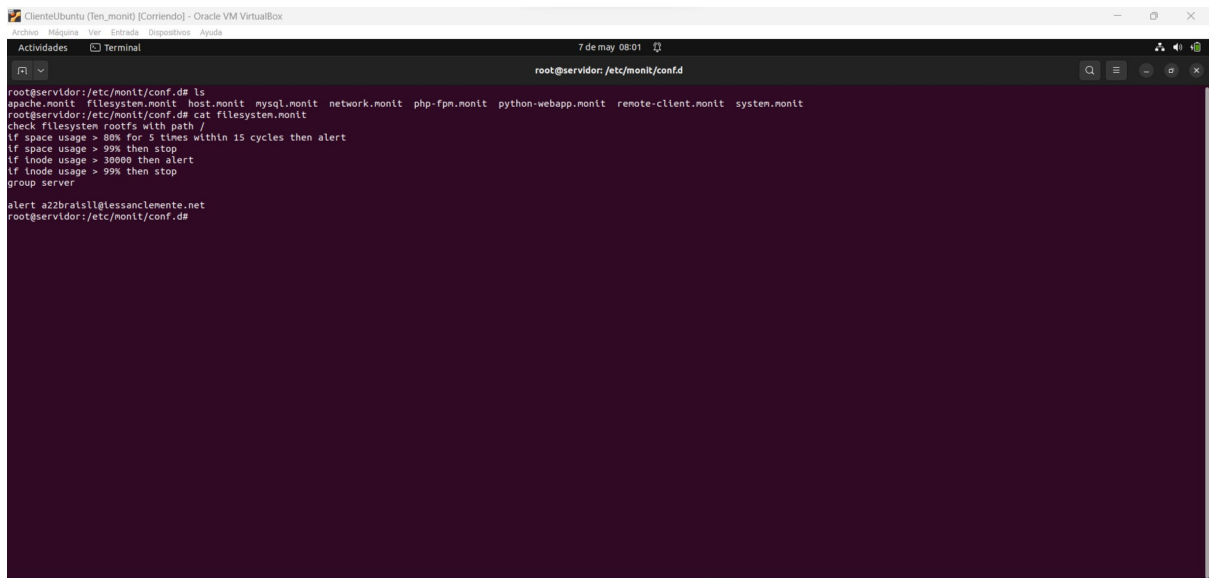
Para esto crearemos un ficheiro `/etc/monit/conf.d/filesystem.monit`.

Dentro del establecemos as seguintes directivas:

```
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 o ficheiro reiniciamos o servizo usando o comando:

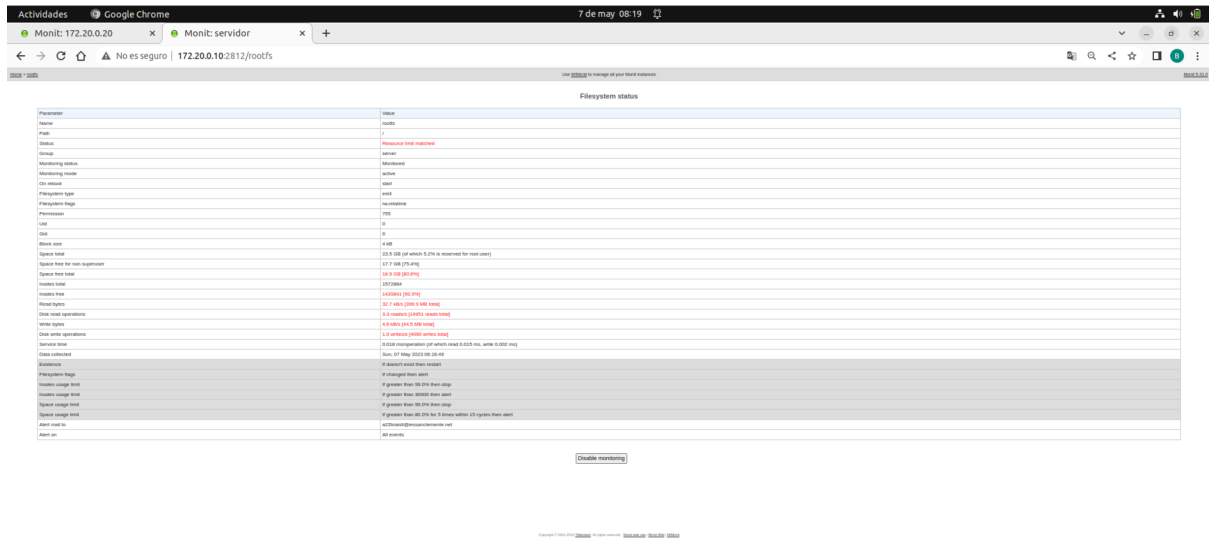
```
sudo systemctl restart monit.service.
```

A screenshot of a terminal window titled 'ClienteUbuntu (Ten\_monit) [Corriendo] - Oracle VM VirtualBox'. The terminal shows the user 'root' at the 'servidor' prompt. The user lists files in '/etc/monit/conf.d', showing 'filesystem.monit'. Then, the user cat's the content of 'filesystem.monit', which displays the configuration for checking the root filesystem on /dev/sda1 for space and inode usage, with alerts and stop actions. The terminal also shows an alert message from 'a22braisl@lessanclente.net' and the user returning to the prompt.

```
root@servidor:/etc/monit/conf.d# ls
apache.monit  filesystem.monit  host.monit  mysql.monit  network.monit  php-fpm.monit  python-webapp.monit  remote-client.monit  system.monit
root@servidor:/etc/monit/conf.d# cat filesystem.monit
check filesystem rootfs with path /
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

alert a22braisl@lessanclente.net
root@servidor:/etc/monit/conf.d#
```

Figura 5: Configuración ficheiro filesystem



*Figura 6: Interfaz web Monit de Filesystem*

## Monitorización de host

Para esto crearemos un ficheiro `/etc/monit/conf.d/host.monit`

Engadimos as seguintes directivas:

```
check host localhost with address 172.20.0.10
if failed ping the alert
if failed port 80 protocol http
then alert.
```

Neste caso faremos un chequeo de host local consistente en varias acci3ns:

- ```
.Execuccion dun ping.
.Chequeo do porto 80 local.
```

Neste caso, en caso de fallo de calquera dos chequeos envíase unha alerta.

Tras editar o ficheiro, reiniciamos o servizo co comando:

```
sudo systemctl restart monit.service
```



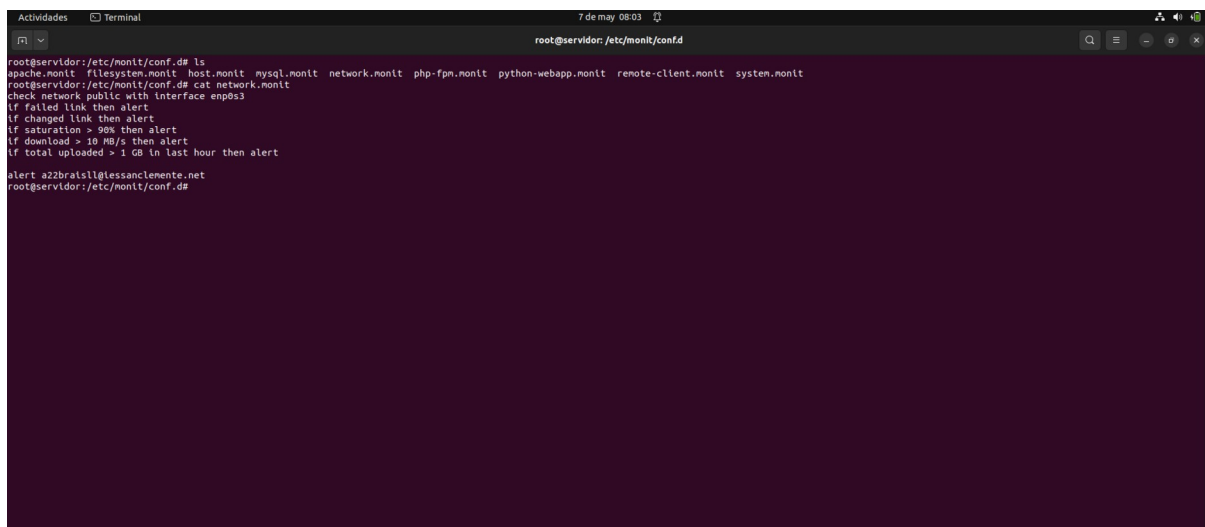


*check network public with interface enp0s3*

*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 hou then alert*

Tras modificar o ficheiro, reiniciamos o servizo, co seguinte comando:

*sudo service monit restart*



```
root@servidor:/etc/monit/conf.d# ls
apache.monit  filesystem.monit  host.monit  mysql.monit  network.monit  php-fpm.monit  python-webapp.monit  remote-client.monit  system.monit
root@servidor:/etc/monit/conf.d# cat 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

alert a22braisl@lessanclamente.net
root@servidor:/etc/monit/conf.d#
```

Figura 9: Configuración ficheiro net

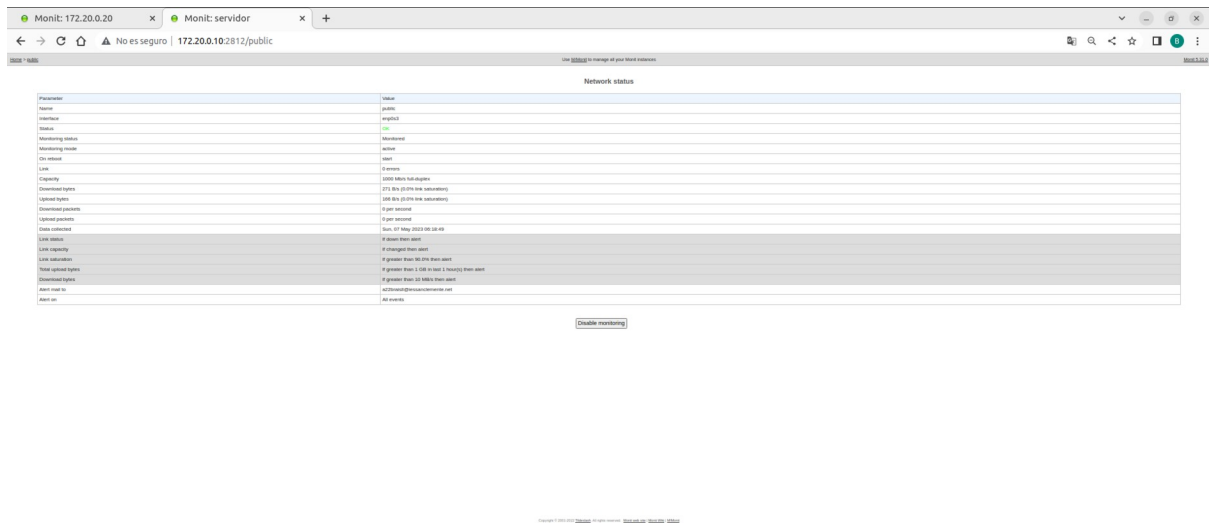


Figura 10: Interfaz web Monit de Net

## **Agregar servicios de monitoreo**

Unha vez que temos correctamente configurada a interfaz web, comezamos a agregar os programas que se desexa monitoroar.

Neste caso vamos a monitoroar un LAMP, que é un acrónimo dos servizos a instalar para permitir unha infraestrutura para servizos de Internet, que o compoñen Linux, Apache2, MySQL/MariaDB e Perl/Python/PHP5.

## **Instalación e monitorización de MySQL**

Para a instalación de MySQL no servidor, executaremos en primeiro lugar o seguinte comando:

```
sudo apt install mysql-server.
```

Unha vez feita a instalación, vamos a protexer o servidor MySQL executando o seguinte comando:

```
sudo mysql_secure_installation
```

Pero debido a cambios nestas últimas versións, poderemos atopar erros ao establecer o contrasinal, polo que antes, de executar o comando de securización, deberemos de seguir os seguintes pasos:

Iniciamos sesión en MySQL como usuario root:

```
sudo mysql -u root -p
```

Cambiamos o método de autenticación do usuario root a "mysql\_native\_password" cun comando ALTER:

```
ALTER USER 'root'@'localhost' IDENTIFIED WITH mysql_native_password BY  
'tu_contraseña_root';  
FLUSH PRIVILEGES;
```

Saimos de MySQL:

exit;

Executamos, agora si, o comando de securización e seguimos os pasos dados.

Isto fará que pasemos por un asistente de preguntas para asegurar o servidor MySQL, polo que iremos seguindo as instrucións que nos vaian saíndo por pantalla.

En primeiro lugar, presionamos "y" para habilitar o complemento de validación de contrasinal, o que nos permitirá establecer unha política estricta de contrasinais para as contas, o que nos atoparemos será o seguinte texto:

*VALIDATE PASSWORD COMPONENT can be used to test passwords and improve security. It checks the strength of password and allows the users to set only those passwords which are secure enough. Would you like to setup VALIDATE PASSWORD component?*

*Press y|Y for Yes, any other key for No: y*

A continuación, teremos que facer unha escolla do nivel de complexidade do contrasinal, donde nos dará 3 opcións e teremos que escoller unha, sendo o seguinte:

*LOW Length >= 8*  
*MEDIUM Length >= 8, numeric, mixed case, and special characters*  
*STRONG Length >= 8, numeric, mixed case, special characters and dictionary file*

*Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG:*

A continuación, introducimos o contrasinal e volvemos a confirmalo, debendo de asegurarnos o nivel de complexidade seleccionado e unha vez feito presionamos "y" para continuar co contrasinal proporcionado, sendo este o último texto que nos apareza:

*Do you wish to continue with the password provided?(Press y|Y for Yes, any other key for No) : y*

A continuación, procedemos a eliminar os usuarios predeterminados do servidor MySQL e a deshabilitar o acceso raíz dende sistemas remotos, eliminando a base de datos de proba do módulo MySQL creado por defecto durante a instalación e vólvese a cargar todos os privilexios para aplicar os cambios feitos de xeito inmediato, a continuación vérase os diálogos de escolla, que nos dará o programa para asegurar os pasos mencionados:

*Remove anonymous users? (Press y|Y for Yes, any other key for No) : y*

*Disallow root login remotely? (Press y|Y for Yes, any other key for No) : y*

*Remove test database and access to it? (Press y|Y for Yes, any other key for No) : y*

*Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y*

Con estes pasos estará debidamente securizado MySQL.

A continuación, crearemos, a modo de exemplo, bases de datos e usuarios no servidor MySQL.

Creamos unha base de datos chamada "XBD":

```
mysql> CREATE DATABASE XBD;
```

Creamos un usuario chamado "alumno" accesible só dende "localhost":

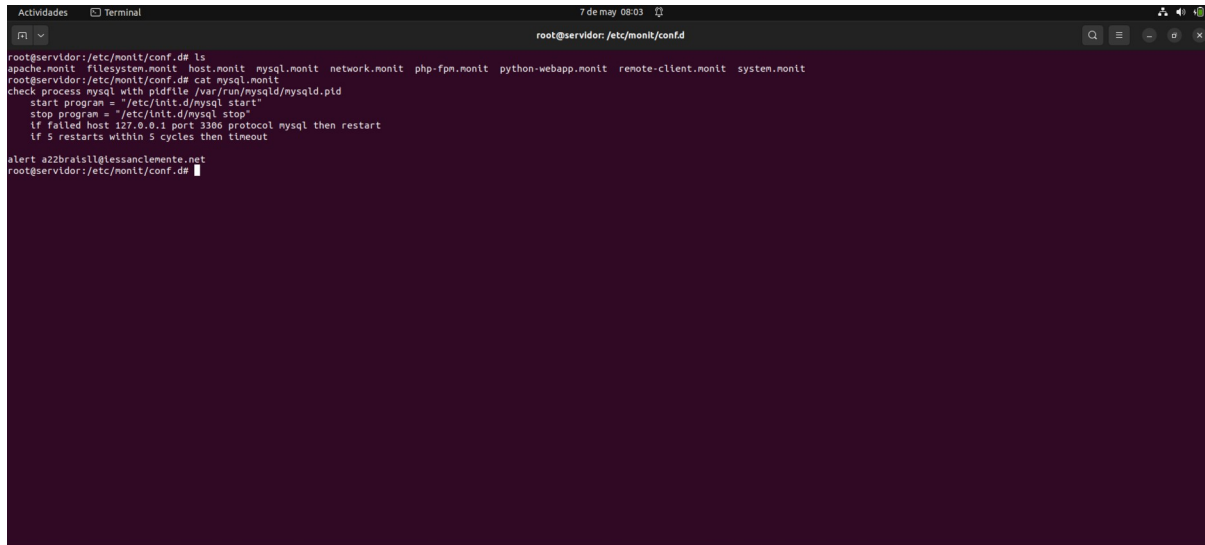
```
CREATE USER 'alumno'@'localhost' IDENTIFIED BY 'Abcd1234.';
```

Otorgamos permisos de base de datos ao usuario:

```
GRANT ALL ON XBD.* to 'usuario'@'localhost';
```

Aplicamos os cambios de permisos en tempo de execución:

```
FLUSH PRIVILEGES;
```



```

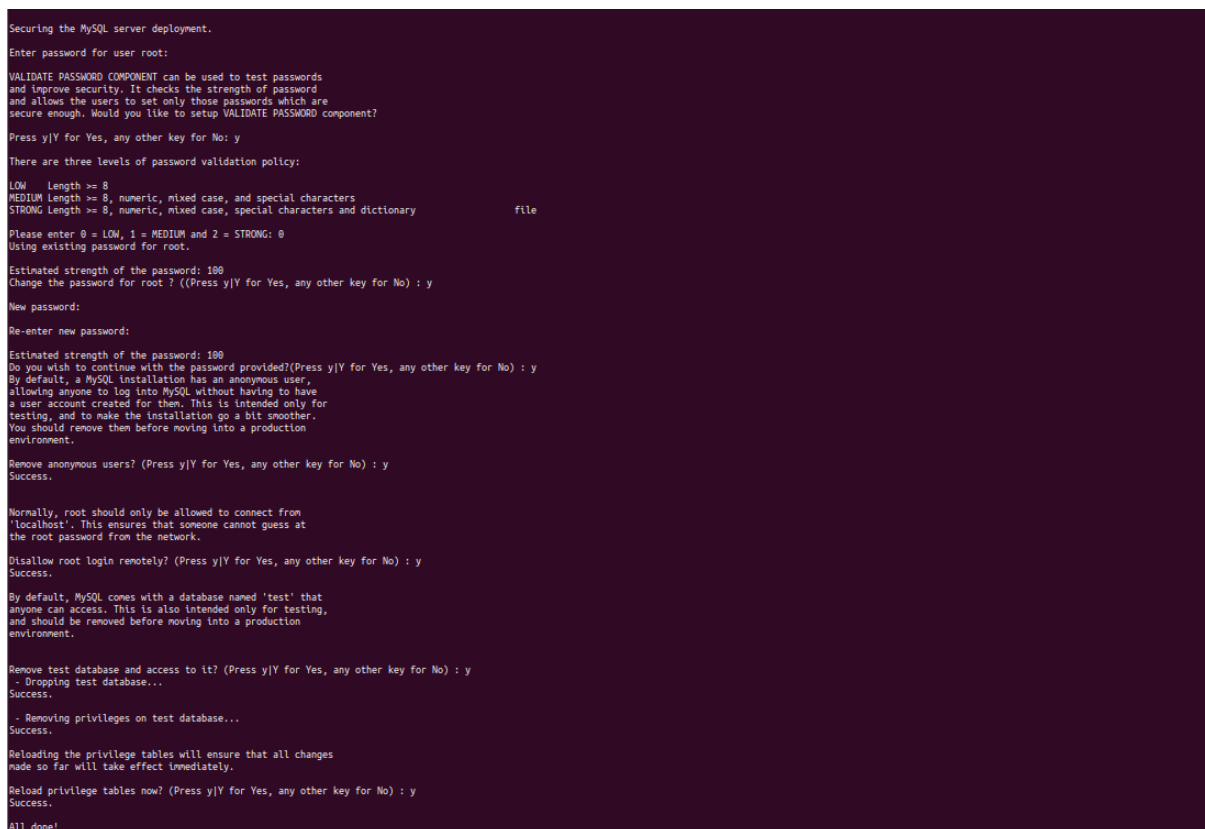
Actividades Terminal 7 de may 08:03
root@servidor: /etc/monit/conf.d

root@servidor:/etc/monit/conf.d# ls
apache.monit  filesystem.monit  host.monit  mysql.monit  network.monit  php-fpm.monit  python-webapp.monit  remote-client.monit  system.monit
root@servidor:/etc/monit/conf.d# cat mysql.monit
check process mysql with ptotile /var/run/mysqld/mysqld.pid
start program = "/etc/init.d/mysql start"
stop program = "/etc/init.d/mysql stop"
if failed host 127.0.0.1 port 3306 protocol mysql then restart
if 5 restarts within 5 cycles then timeout

alert a22braisl@lessanclamente.net
root@servidor:/etc/monit/conf.d#

```

Figura 11: Configuración ficheiro MySQL



```

Securing the MySQL server deployment.

Enter password for user root:

VALIDATE PASSWORD COMPONENT can be used to test passwords
and improve security. It checks the strength of password
and allows the users to set only those passwords which are
secure enough. Would you like to setup VALIDATE PASSWORD component?

Press y|Y for Yes, any other key for No: y

There are three levels of password validation policy:

LOW Length >= 8
MEDIUM Length >= 8, numeric, mixed case, and special characters
STRONG Length >= 8, numeric, mixed case, special characters and dictionary file

Please enter 0 = LOW, 1 = MEDIUM and 2 = STRONG: 0
Using existing password for root.

Estimated strength of the password: 100
Change the password for root ? ((Press y|Y for Yes, any other key for No) : y

New password:
Re-enter new password:

Estimated strength of the password: 100
Do you wish to continue with the password provided?((Press y|Y for Yes, any other key for No) : y
By default, a MySQL installation has an anonymous user,
allowing anyone to log into MySQL without having to have
a user account created for them. This is intended only for
testing, and to make the installation go a bit smoother.
You should remove them before moving into a production
environment.

Remove anonymous users? (Press y|Y for Yes, any other key for No) : y
Success.

Normally, root should only be allowed to connect from
'localhost'. This ensures that someone cannot guess at
the root password from the network.

Disallow root login remotely? (Press y|Y for Yes, any other key for No) : y
Success.

By default, MySQL comes with a database named 'test' that
anyone can access. This is also intended only for testing,
and should be removed before moving into a production
environment.

Remove test database and access to it? (Press y|Y for Yes, any other key for No) : y
- Dropping test database...
Success.
- Removing privileges on test database...
Success.

Reloading the privilege tables will ensure that all changes
made so far will take effect immediately.

Reload privilege tables now? (Press y|Y for Yes, any other key for No) : y
Success.

All done!

```

Figura 12: Securitización de MySQL

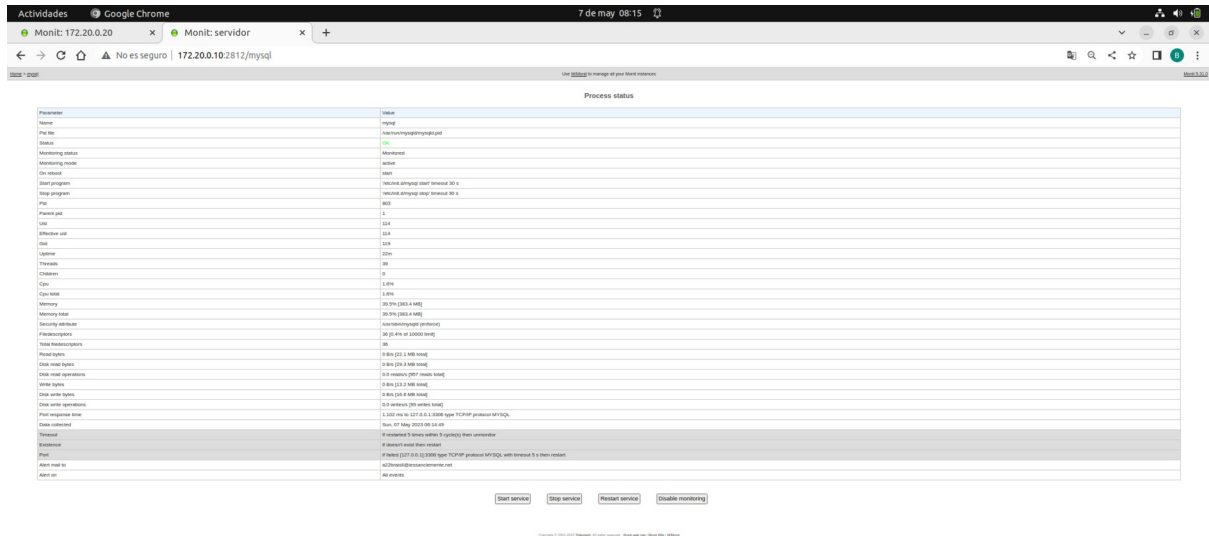


Figura 13: Interfaz web Monit de MySQL

## Instalación de PHP

En primeiro lugar cabe definir que é PHP, sendo este unha linguaxe de programación de código aberto do lado do servidor que se emprega principalmente para crear páxinas web dinámicas.

Para a súa instalación, tanto de PHP e PHP-FPM, empregaremos o comando :

```
sudo apt install php php-fpm.
```

Acto seguido, crearemos un ficheiro de configuración para Monit no directorio `/etc/monit/conf.d/` con extensión `.monit`, empregando para isto o seguinte comando:

```
sudo nano /etc/monit/conf.d/php-fpm.monit
```

E engadimos as seguintes liñas ao ficheiro:

```
check process php-fpm with pidfile /run/php/php8.1-fpm.pid
group php-fpm
start program = "/etc/init.d/php8.1-fpm start"
stop program = "/etc/init.d/php8.1-fpm stop"
if failed unixsocket /run/php/php8.1-fpm.sock then restart
```



*if 5 restarts within 5 cycles then timeout*

Gardamos o ficheiro e saímos.

Por último para aplicar os cambios reinicamos Monit executando o comando:

*sudo systemctl restart monit.*

```

root@servidor:/etc/monit/conf.d# ls
apache.monit filesystem.monit host.monit mysql.monit network.monit php-fpm.monit python-webapp.monit remote-client.monit system.monit
root@servidor:/etc/monit/conf.d# cat php-fpm.monit
check process php-fpm with pidfile /run/php/php8.1-fpm.pid
group phpfpm
start program = "/etc/init.d/php8.1-fpm start"
stop program = "/etc/init.d/php8.1-fpm stop"
if failed unixsocket /run/php/php8.1-fpm.sock then restart
if 5 restarts within 5 cycles then timeout

alert a22braisl@essanclamente.net
root@servidor:/etc/monit/conf.d#

```

Figura 14: Configuración ficheiro PHP

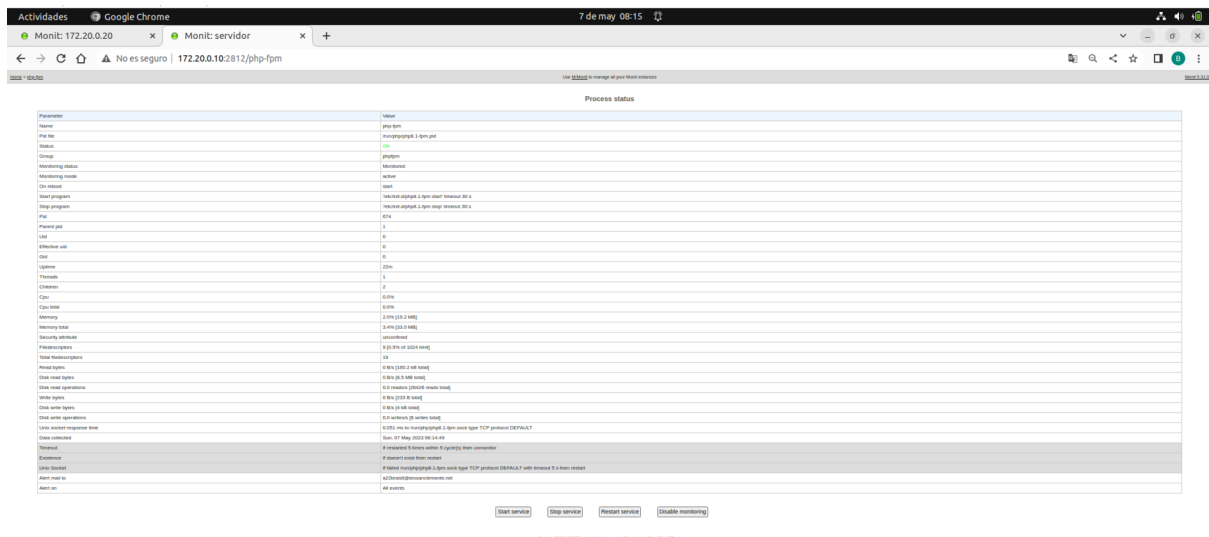


Figura 15: Interfaz web Monit de PHP

## Monitorización de una aplicación python

Para monitorizar una aplicación Python con Monit, primeiro imos necesitar executar a nosa aplicación Python como un servizo do sistema. Despois, configuraremos Monit para supervisalo, para isto seguiremos os seguintes pasos:

Primeiro, crearemos unha aplicación de Python, na ruta '/home/servidor/' co comando

```
sudo nano app.py
```

neste caso o que fara será iniciar un servidor web no porto 8080, e para esto emplearemos o seguinte código:

```
import http.server
import socketserver

PORT = 8080
Handler = http.server.SimpleHTTPRequestHandler

class CustomHandler(Handler):
    def do_GET(self):
        self.send_response(200)
        self.send_header("Content-type", "text/html")
        self.end_headers()
        self.wfile.write(b"Hola, Mundo!")

with socketserver.TCPServer(("", PORT), CustomHandler) as httpd:
    print("sirviendo en el puerto", PORT)
    httpd.serve_forever()
```

Segundo, crearemos un ficheiro de servizo systemd para a nosa aplicación, e elaboramos un ficheiro chamado 'python-webapp.service' en '/etc/systemd/system/', co seguinte comando:

```
sudo nano /etc/systemd/system/python-webapp.service
```

Terceiro, agregamos o seguinte contido ao ficheiro:

```
[Unit]
Description=My Python Web Application
After=network.target

[Service]
Type=simple
ExecStart=/usr/bin/python3 /home/servidor/app.py
Restart=on-failure

[Install]
WantedBy=multi-user.target
```

Gardamos o ficheiro e saímos do editor.

Cuarto, habilitamos e executamos o servizo python cos seguintes comandos:

```
nano sudo systemctl daemon-reload
nano sudo systemctl enable python-webapp.service
nano sudo systemctl start python-webapp.service
```

Quinto, creamos un ficheiro de configuración no directorio '/etc/monit/conf.d' con extensión '.monit', o cal chamaremos 'python-webapp.monit', e faremolo co seguinte comando:

```
sudo nano /etc/monit/conf.d/python-webapp.monit
```

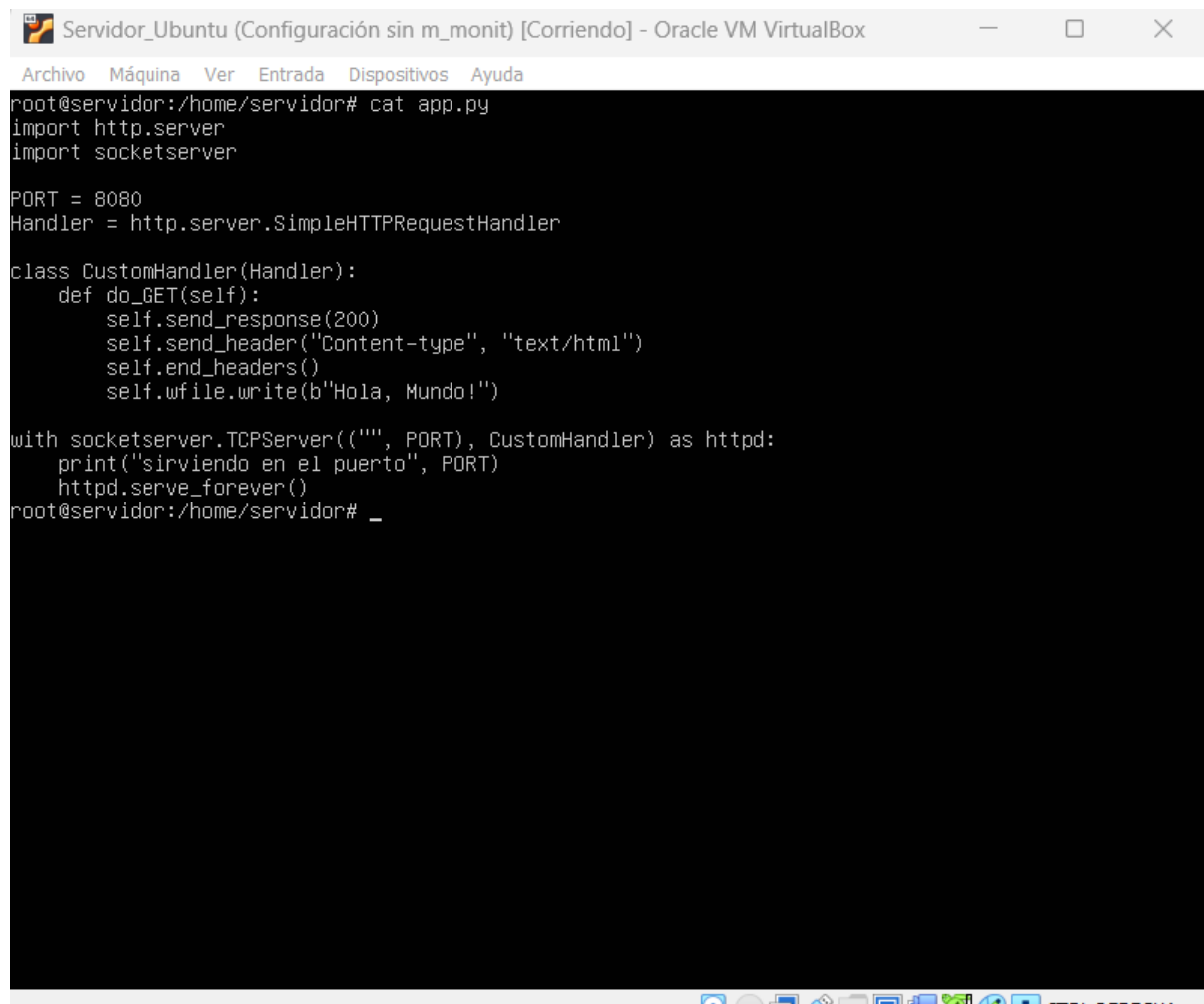
Agregamos as seguintes liñas ao ficheiro:

```
check process python-webapp with matching "app.py"
start program = "/bin/systemctl start python-webapp.service"
stop program = "/bin/systemctl stop python-webapp.service"
if failed host 172.20.0.10 port 8080 protocol http then restart
if 5 restarts within 5 cycles then timeout
```

Gardamos o ficheiro e saímos do editor.

Sexto, reiniciamos Monit para aplicar os cambios, co comando:

```
sudo systemctl restart monit
```



The screenshot shows a terminal window titled "Servidor\_Ubuntu (Configuración sin m\_monit) [Corriendo] - Oracle VM VirtualBox". The terminal displays the following Python code:

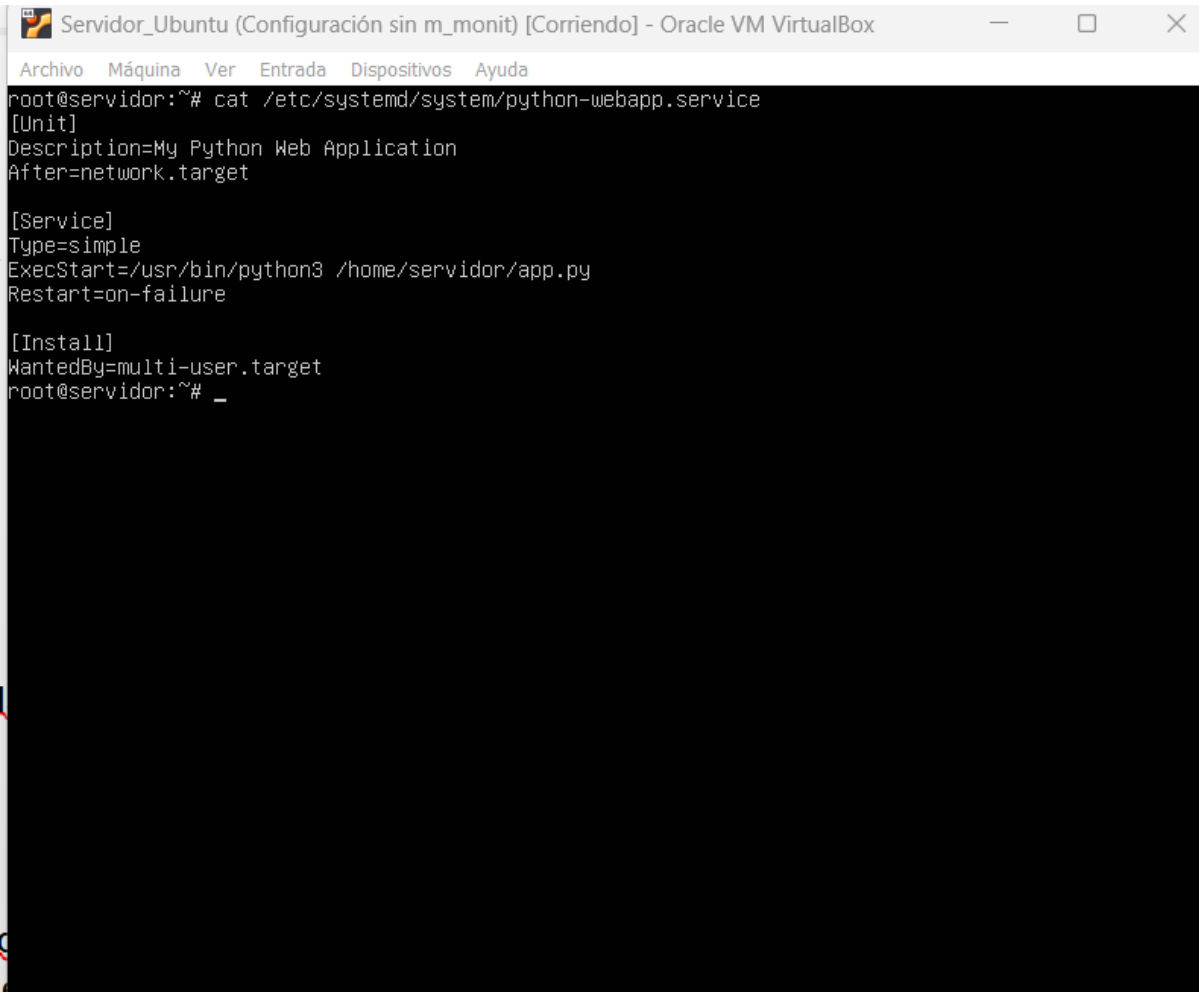
```
root@servidor:/home/servidor# cat app.py
import http.server
import socketserver

PORT = 8080
Handler = http.server.SimpleHTTPRequestHandler

class CustomHandler(Handler):
    def do_GET(self):
        self.send_response(200)
        self.send_header("Content-type", "text/html")
        self.end_headers()
        self.wfile.write(b"Hola, Mundo!")

with socketserver.TCPServer(("", PORT), CustomHandler) as httpd:
    print("sirviendo en el puerto", PORT)
    httpd.serve_forever()
root@servidor:/home/servidor# _
```

Figura 16: Programa de Python



```
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
root@servidor:~# cat /etc/systemd/system/python-webapp.service
[Unit]
Description=My Python Web Application
After=network.target

[Service]
Type=simple
ExecStart=/usr/bin/python3 /home/servidor/app.py
Restart=on-failure

[Install]
WantedBy=multi-user.target
root@servidor:~# _
```

Figura 17: Ficheiro python de sistema

```

root@servidor:/etc/monit/conf.d# ls
apache.monit  filesystem.monit  host.monit  mysql.monit  network.monit  php-fpm.monit  python-webapp.monit  remote-client.monit  system.monit
root@servidor:/etc/monit/conf.d# cat python-webapp.monit
check process python-webapp with matching "app.py"
start program = "/bin/systemctl start python-webapp.service"
stop program = "/bin/systemctl stop python-webapp.service"
if failed host 172.20.0.10 port 8080 protocol http then restart
if 5 restarts within 5 cycles then timeout

alert a2brazil@essanclenets.net
root@servidor:/etc/monit/conf.d#

```

Figura 18: Configuración ficheiro Python

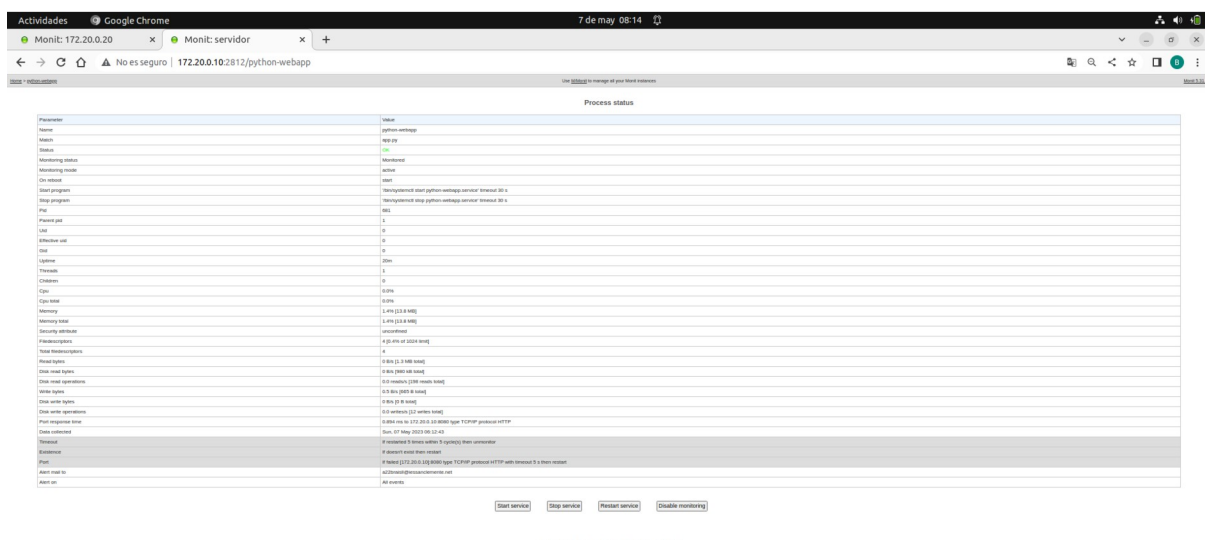


Figura 19: Interfaz web Monit de Python

## Configuración avisos correo electrónico

Para configurar Monit para que envíe alertas por correo electrónico cando detecte un fallo nun servizo, seguiremos os seguintes pasos:

Primeiro, debemos instalar un servidor de correo, como Postfix, para enviar correos electrónicos dende o noso sistema, e faremolo co seguinte comando:

```
sudo apt install postfix
```

Durante a instalación, pedírase que se configure o tipo de correo, para o que escolleremos "Sitio de Internet" e proporcionaremos un nome de dominio completo para o sistema, como no noso caso "gmail.com".

Segundo, abriremos o ficheiro de configuración principal de Monit nun editor de texto, co seguinte comando:

```
sudo nano /etc/monit/monitrc
```

Buscaremos a liña que comeza con 'set alert'. Se se atopa comentada (ten un '#' ao comezo da liña), descomentarase e remplazamos a dirección de correo electrónico predeterminada ca nosa dirección de correo electrónico 'a22braisl@iesanclemente.net'.

```
set alert a22braisl@iessanclemente.net
```

A continuación, buscamos a sección de configuración do correo no ficheiro 'monitrc', deberíamos de ver algo como esto:

```
# set mailserver mail.example.com port 25  
#   username "myuser" password "mypassword"  
#   using tlsv12  
#   with timeout 30 seconds
```

Descomentamos e configuramos a sección 'set mailserver' cos detalles do noso servidor de correo sainte (STMP), que debería de quedar semellante ao seguinte exemplo:

```
set mailserver smtp.gmail.com port 587  
  username "a22braisl@iessanclemente.net" password "your-email-password"  
  using tlsv1  
  with timeout 30 seconds
```

Gardamos o ficheiro e saímos do editor.

Terceiro, para configurar as alertas por correo electrónico nun servizo específico, iremos ao directorio '/etc/monit/conf.d', que é onde se atopan todos os servizos que estamos a monitorizar neste caso, e abrindo co editor que escollamos, engadiremos a seguinte liña ao final da configuración do servizo:

```
alert a22braisl@iesanclemente.net
```

Esta liña indicará a Monit que envíe alertas por correo electrónico á dirección de correo electrónico especificada cando se detecte un fallo neste servizo.

Gardamos o ficheiro e saímos do editor.

Cuanto, reiniciamos o servizo Monit para aplicar os cambios co seguinte comando:

```
sudo systemctl restart monit
```

Deste xeito, Monit está configurado para enviar alertas por correo electrónico a 'a22braisl@iesanclemente.net' cando detecte un fallo nos servizos os que engadimos a liña en cuestión.



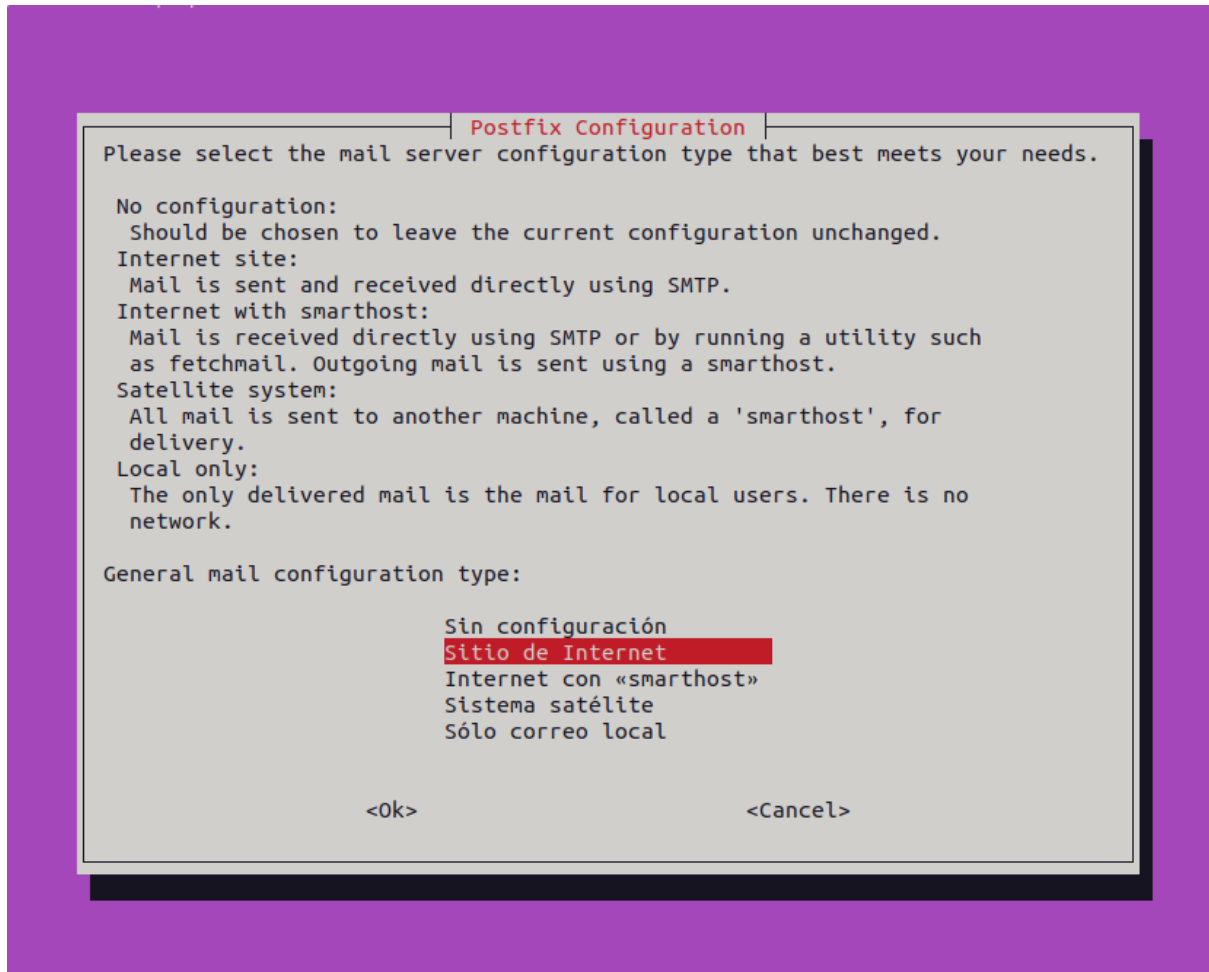


Figura 20: Postfix

## Monitorización de servicio Apache

En primeiro lugar, antes da instalación de apache deberemos de actualizar o sistema, deberemos de executar o comando *sudo apt update*, para facer unha correcta actualización dos paquetes do sistema.

Acto seguido, executaremos o seguinte para a instalación de apache:

```
Sudo apt install apache2
```

Continuando, co formato de traballo que estamos a facer neste equipo, vamos a crear un ficheiro de configuración para o sistema, co seguinte comando:

```
Sudo nano /etc/monit/conf.d/apache.monit
```

Estableceremos as seguintes configuracións no ficheiro:

```
Check process apache2 with pidfile /run/apache2/apache2.pid
Start program = "/etc/init.d/apache2 start"
Stop program = "/etc/init.d/apache2 stop"
If failed 127.0.0.1 port 80 protocol http then restart
If 5 restarts within 5 cycles then timeout
```

Gardamos e saímos do editor.

Para aplicar os cambios, reiniciamos o servizo Monit co seguinte comando:

```
Sudo systemctl restart monit.
```



## Configuración equipo cliente para monitorización

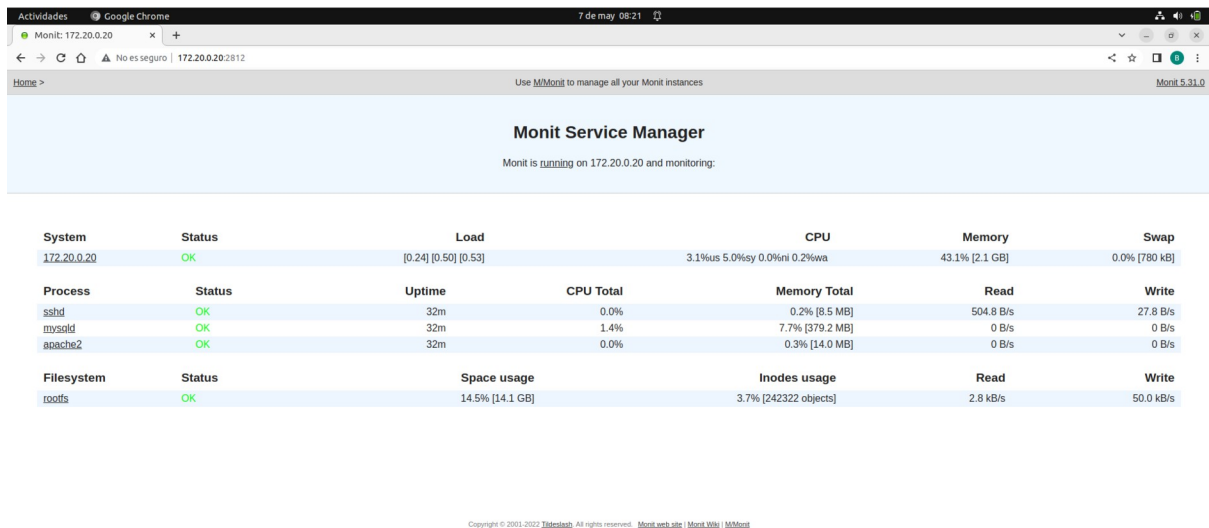
Para este caso no equipo cliente vamos a monitorizar, os procesos internos do propio equipo así como os procesos de SSH, MySQL e Apache2.

No caso da configuración da rede, neste caso poderemos facelo a través da interfaz gráfica do SO, polo que non será necesario redundar na explicación do primeiro equipo.

No apartado da habilitación da interfaz web deberemos de habilitar as liñas correspondentes, indicadas no primeiro equipo, ca salvedade, de que na liña *allow* indicaremos a IP do equipo servidor, para permitir acceder dende o servidor aos procesos de monitorización do cliente.

A diferencia do primeiro caso, o que faremos é configurar os servicios internos do sistema dentro do ficheiro principal de configuración de Monit, mentras que os servicios externos, fánzanse coma no servidor, para poder demostrar que se pode facer en ambos sitios, sen problema de ningún tipo.

Para poder facer unha monitorización, deberemos de instalar Monit no equipo cliente, o cal faremos seguindo os mesmos pasos descritos en primeira instancia.



| System      | Status | Load                 | CPU                         | Memory         | Swap          |
|-------------|--------|----------------------|-----------------------------|----------------|---------------|
| 172.20.0.20 | OK     | [0.24] [0.50] [0.53] | 3.1%us 5.0%sy 0.0%ni 0.2%wa | 43.1% [2.1 GB] | 0.0% [780 kB] |

| Process | Status | Uptime | CPU Total | Memory Total    | Read      | Write    |
|---------|--------|--------|-----------|-----------------|-----------|----------|
| sshd    | OK     | 32m    | 0.0%      | 0.2% [8.5 MB]   | 504.8 B/s | 27.8 B/s |
| mysqld  | OK     | 32m    | 1.4%      | 7.7% [379.2 MB] | 0 B/s     | 0 B/s    |
| apache2 | OK     | 32m    | 0.0%      | 0.3% [14.0 MB]  | 0 B/s     | 0 B/s    |

| Filesystem | Status | Space usage     | Inodes usage          | Read     | Write     |
|------------|--------|-----------------|-----------------------|----------|-----------|
| rootfs     | OK     | 14.5% [14.1 GB] | 3.7% [242322 objects] | 2.8 kB/s | 50.0 kB/s |

Copyright © 2003-2022 [Iddeslab](#). All rights reserved. [Monit web site](#) | [Monit Wiki](#) | [MAMonit](#)

Figura 23: Interfaz web Monit cliente

## Monitorización de system e filesystem

Neste caso debemos de editar o ficheiro `/etc/monit/monitrc`, co editor da nosa preferencia.

Ao final do ficheiro, podemos editar engadindo as seguintes liñas neste caso:

Para a configuración do system:

*Check system 172.20.0.20*

*If cpu usage (user) > 80% for 2 cycles then alert*

*If cpu usage (system) > 80% for 2 cycles then alert*

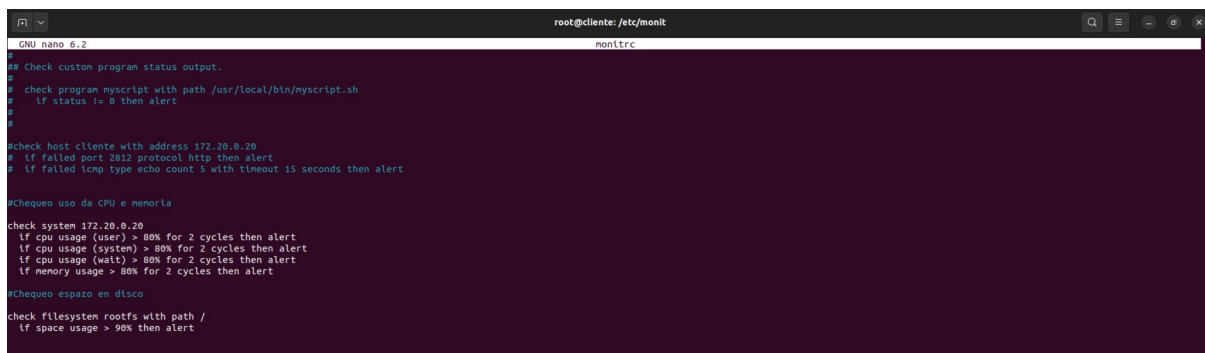
*If cpu usage (wait) > 80% for 2 cycles then alert*

*If memory usage > 80% for 2 cycles then alert*

Para a configuración de filesystem:

*Check filesystem rootfs with path /*

*If space usage > 90% then alert*



```
GNU nano 6.2 monitrc
# Check custom program status output.
#
# check program myscript with path /usr/local/bin/myscript.sh
#   if status != 0 then alert
#
#check host cliente with address 172.20.0.20
#   if failed port 2812 protocol http then alert
#   if failed icmp type echo count 5 with timeout 15 seconds then alert

#Chequeo uso da CPU e memoria
check system 172.20.0.20
  if cpu usage (user) > 80% for 2 cycles then alert
  if cpu usage (system) > 80% for 2 cycles then alert
  if cpu usage (wait) > 80% for 2 cycles then alert
  if memory usage > 80% for 2 cycles then alert

#Chequeo espazo en disco
check filesystem rootfs with path /
  if space usage > 90% then alert
```

Figura 24: Configuración ficheiro monitrc con system e filesystem

## Monitorización de SSH

Neste caso, deberemos en primeiro lugar de facer a instalación do servidor SSH, a través do seguinte comando:

```
sudo apt install openssh-server
```

Unha vez descargado, crearemos un ficheiro co seguinte comando en `/etc/monit/conf.d/ssh.d.monit`:

```
sudo nano /etc/monit/conf.d/ssh.monit
```

Engadimos as seguintes directivas para a súa correcta configuración:

```
check process sshd with pidfile /var/run/sshd.pid
```

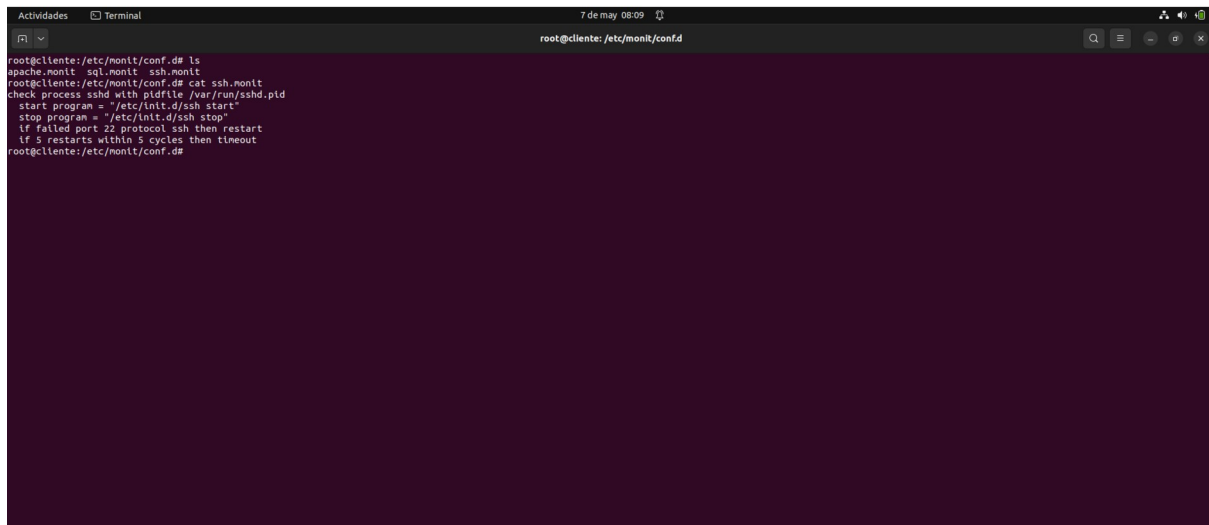
```
start program = "/etc/init.d/ssh start"
```

```
stop program = "/etc/init.d/ssh stop"
```

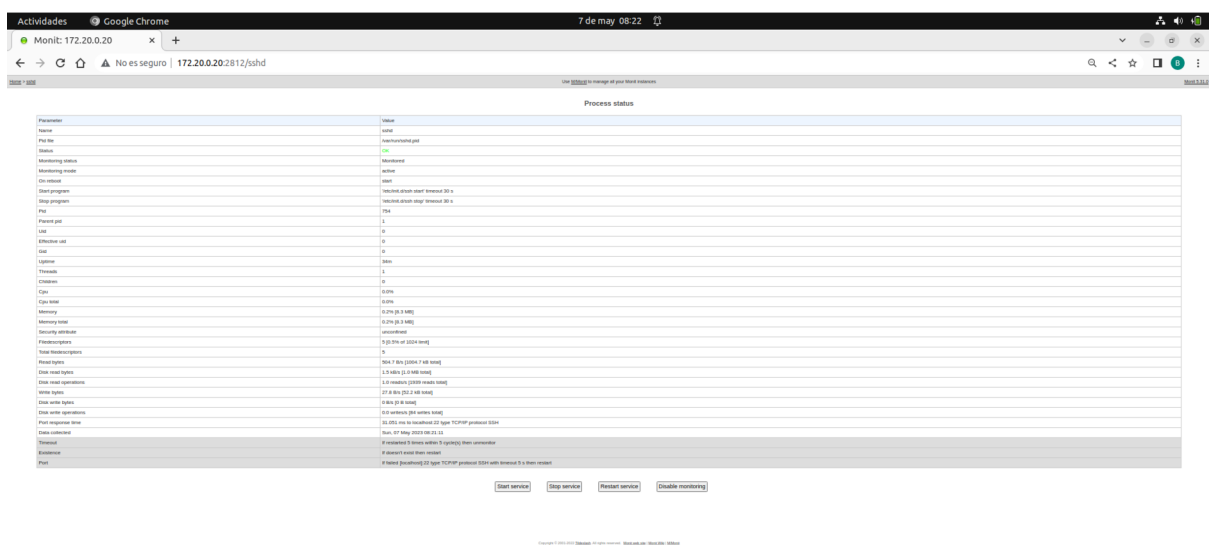
```
if failed port 22 protocol ssh then restart
```

```
if 5 restarts within 5 cycles then timeout
```

Cerramos o editor, e reiniciamos o servizo de Monit, co comando `sudo systemctl restart monit.service`.



*Figura 25: Configuração ficheiro SSH*



*Figura 26: Interfaz web Monit con SSH*

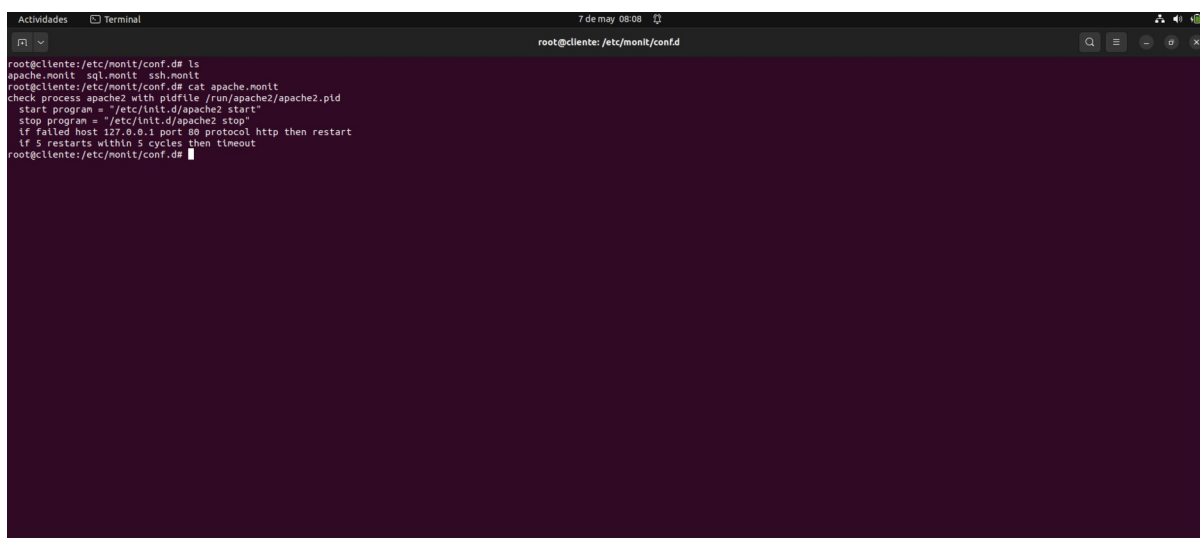
Neste caso, faise remisión á instalación e configuración feita no servidor, pois salvo uns cambios nas directivas de monitorización, os pasos a seguir son practicamente o mesmo.



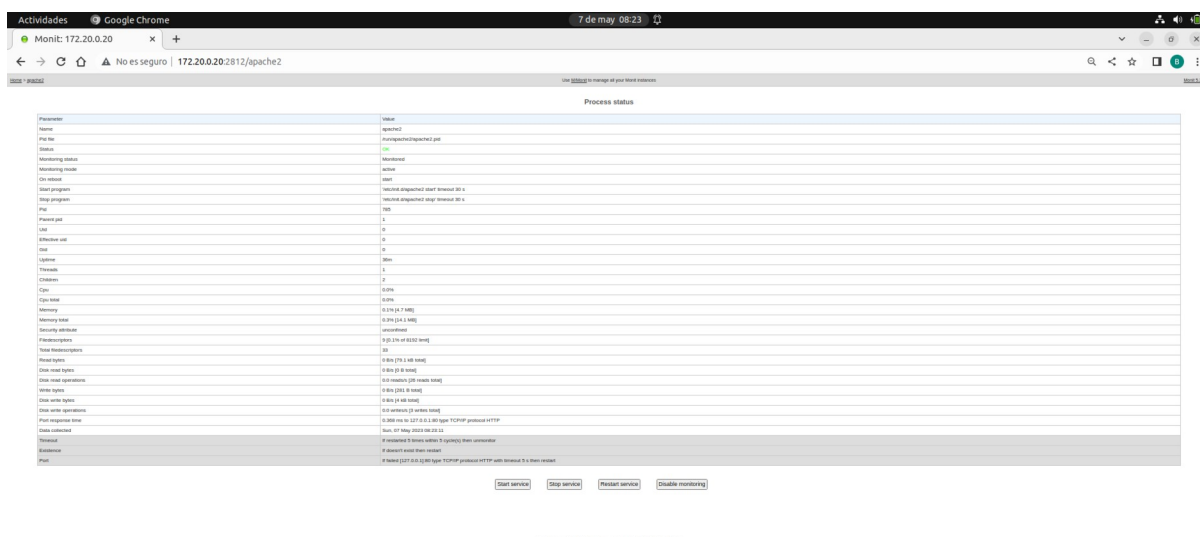


## Monitorización de Apache

Igual que no caso de MySQL, recoméndase a consulta da instalación no servidor pois a orde das instrucións da configuración é a mesma, tendo a pequena variación das directivas de monitorización do ficheiro, pero eso debe ser o administrador encada caso, quen realice a configuración adecuada.



*Figura 29: Configuração ficheiro Apache*



*Figura 30: Interfaz web Monit de Apache*

## Configuración para monitorización cliente-servidor

Neste caso, o que faremos é explicar como conseguir facer unha monitorización de sistema dun equipo cliente dende un servidor, pese a que a este “cliente”, no se atope unido a un servidor, como tal se pode entender, a través dun dominio.

En primeiro lugar o que faremos será a creación dun ficheiro, seguindo a estrutura deste caso, co seguinte comando: `sudo nano /etc/monit/conf.d/remote-client.monit`.

O cal engadiremos as seguintes liñas:

```
check host cliente with address 172.20.0.20
if failed port 2812 protocol http
  with timeout 15 seconds
  then alert
```

```
check program sshd_status with path "/usr/local/bin/check_service.sh sshd"
if status == 0 then alert
```

```
check program mysqld_status with path "/usr/local/bin/check_service.sh mysqld"
if status == 0 then alert
```

```
check program apache2_status with path "/usr/local/bin/check_service.sh apache2"
if status == 0 then alert
```

Cerramos o editor e aplicamos os cambios, reiniciando o servizo Monit, co seguinte comando: `sudo systemctl restart monit.service`.

Con isto xa estaría configurado e dende a interfaz web de Monit do servidor, vains ser posible acceder a ver e administrar os servizos do cliente.

Sin embargo, é necesario facer mención a que deberemos de ter en conta un par de cousas:

En primeiro lugar a URL que empregaremos no ficheiro, xa que si propocionamos algún dato mal, o servidor, non será capaz de chegar ata o equipo cliente.

Neste caso empregamos a seguinte URL:

[http://admin:monit%23@172.20.0.20:2812/\\_status?format=xml](http://admin:monit%23@172.20.0.20:2812/_status?format=xml)

Neste caso “%23” é a codificación URL para o carácter #.

Dentro de ese XML que se xenera deberemos de comprobar que o status non sexa igual a 0, porque o 0 na codificación de Monit significa que o servizo está correcto.

En segundo lugar, cando se realizou a configuración daba erro constantemente e eso debía-se a ese 0, porque pese a non funcionar para o sistema si estaba correcto, polo que foi necesario axustar a cadea de búsqueda na configuración para que busque un “status” distinto de 0, polo que a solución máis pausable foi usar un script de shell que busque todas as posibles cadeas de “status” distinto de 0 e devolva un código de estado 0 (o que indica que o servizo non está funcionando adecuadamente), e o script usado foi o seguinte:

```
#!/bin/bash
```

```
output=$(curl -s -u admin:monit http://172.20.0.20:2812/_status?format=xml)
service=$1
```

```
for status in {1..255}
do
    if [[ $output == *"$service"*status="$status"* ]]
    then
        exit 0
    fi
done

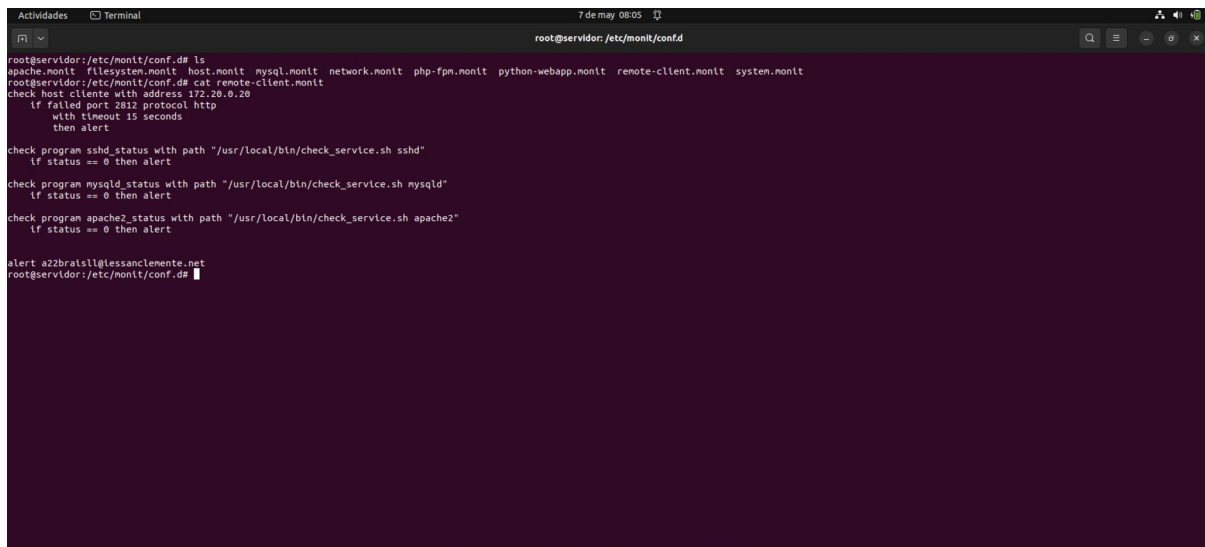
exit 1
```

Con este script tomamos o nome dun servizo como argumento, obtendo a saída XML de Monit no cliente, e busca unha cadea que indique que o servizo non está funcionando adecuadamente. Se atopa unha destas cadeas, devolve un código de estado 0. Se non atopa ningunha destas cadeas, devolve un código de estado 1, o que indica que o servizo está funcionando correctamente.

Despois gárdase este script nun ficheiro, como por exemplo “/usr/local/bin/check\_service.sh”, e damoslle permisos de execución co comando:

```
sudo chmod +x /usr/local/bin/check_service.sh.
```

Por este motivo, é polo que no ficheiro de configuración veremos este script dentro das directivas, con esa configuración, Monit no servidor xerará unha alerta e marcará o servico como “failed” se o script devolve un código 0, o que indica que o servico non está funcionando correctamente no cliente. Cando devolva un código de estado 1, o que indica que está a funciona correctamente, non se xerará ningunha alarma, e amosarase o servico como “OK” na interfaz web de Monit.



```
Actividades Terminal 7 de may 06:05
root@servidor: /etc/monit/conf.d
root@servidor:/etc/monit/conf.d# ls
apache.monit  fllesystem.monit  host.monit  mysql.monit  network.monit  php-fpm.monit  python-webapp.monit  remote-client.monit  system.monit
root@servidor:/etc/monit/conf.d# cat remote-client.monit
check host cliente with address 172.20.0.20
    if failed port 2812 protocol http
        with timeout 15 seconds
        then alert

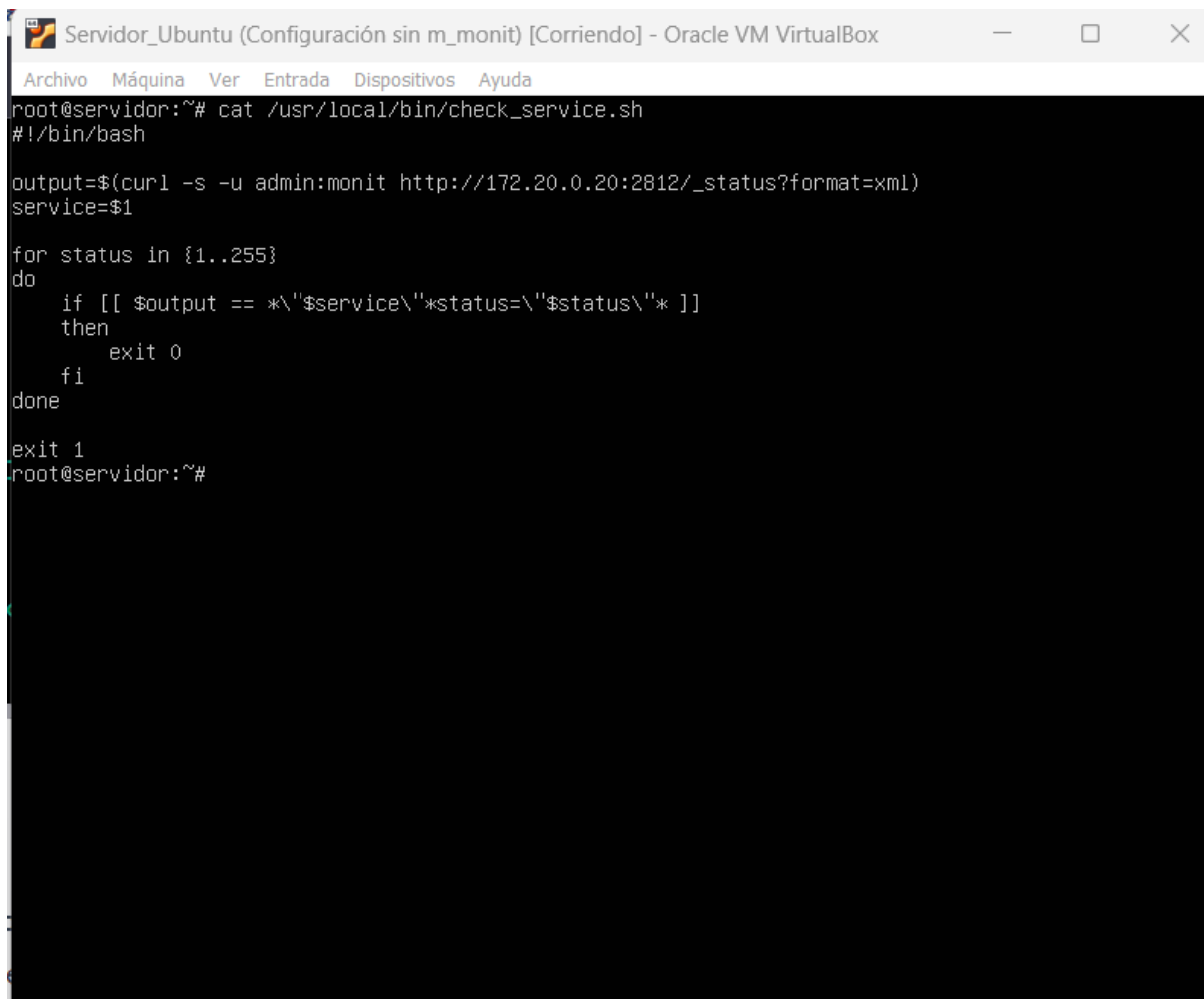
check program sshd_status with path "/usr/local/bin/check_service.sh sshd"
    if status == 0 then alert

check program mysqld_status with path "/usr/local/bin/check_service.sh mysqld"
    if status == 0 then alert

check program apache2_status with path "/usr/local/bin/check_service.sh apache2"
    if status == 0 then alert

alert a22braisl@lessanclente.net
root@servidor:/etc/monit/conf.d#
```

Figura 31: Configuración ficheiro remote-client



```

Servidor_Ubuntu (Configuración sin m_monit) [Corriendo] - Oracle VM VirtualBox
Archivo  Máquina  Ver  Entrada  Dispositivos  Ayuda
root@servidor:~# cat /usr/local/bin/check_service.sh
#!/bin/bash

output=$(curl -s -u admin:monit http://172.20.0.20:2812/_status?format=xml)
service=$1

for status in {1..255}
do
    if [[ $output == *"$service\ "*status="$status\ "* ]]
    then
        exit 0
    fi
done

exit 1
root@servidor:~#
```

*Figura 32: Script*

## Conclusión

Como conclusión, amosarase como é a vista da interfaz web de Monit no servidor que amosa tamén os servicios instalados no equipo cliente.

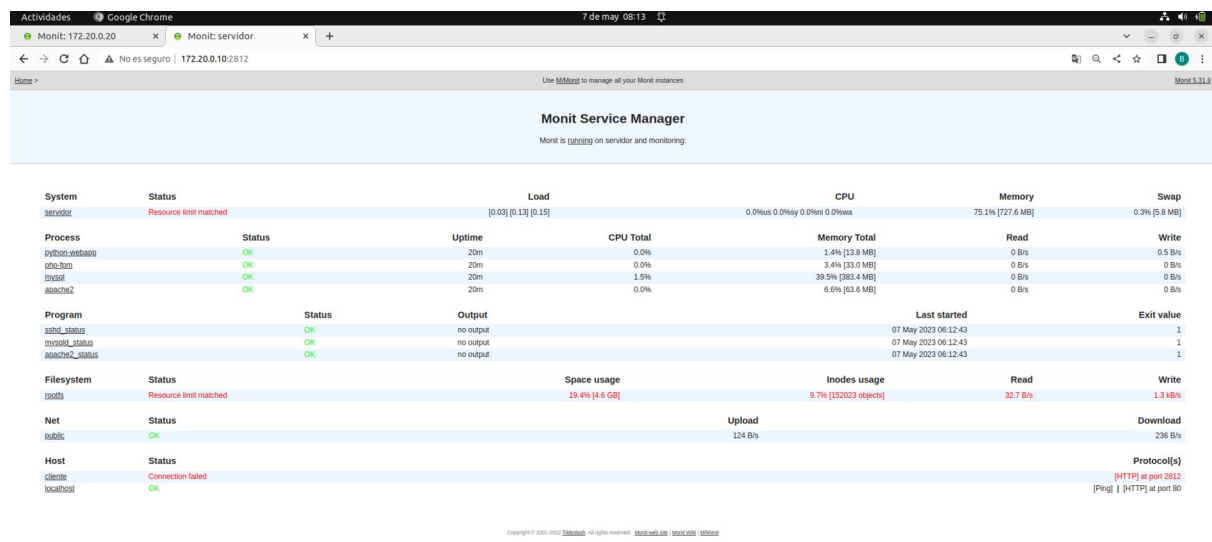


Figura 33: Interfaz web Monit servidor con servicios cliente

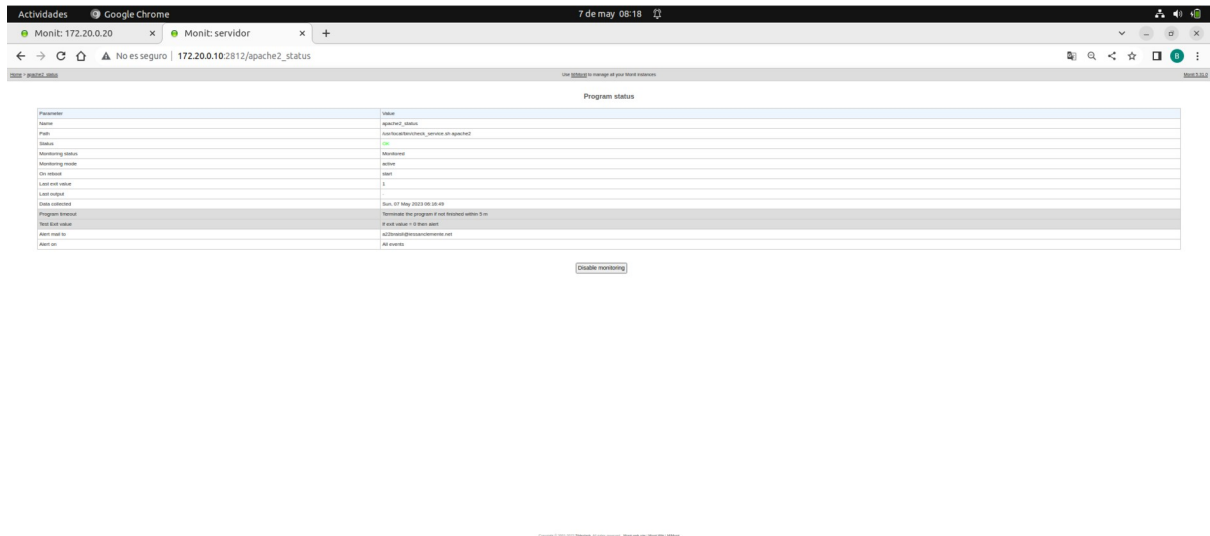


Figura 34: Interfaz web Monit servidor de servicio apache en cliente

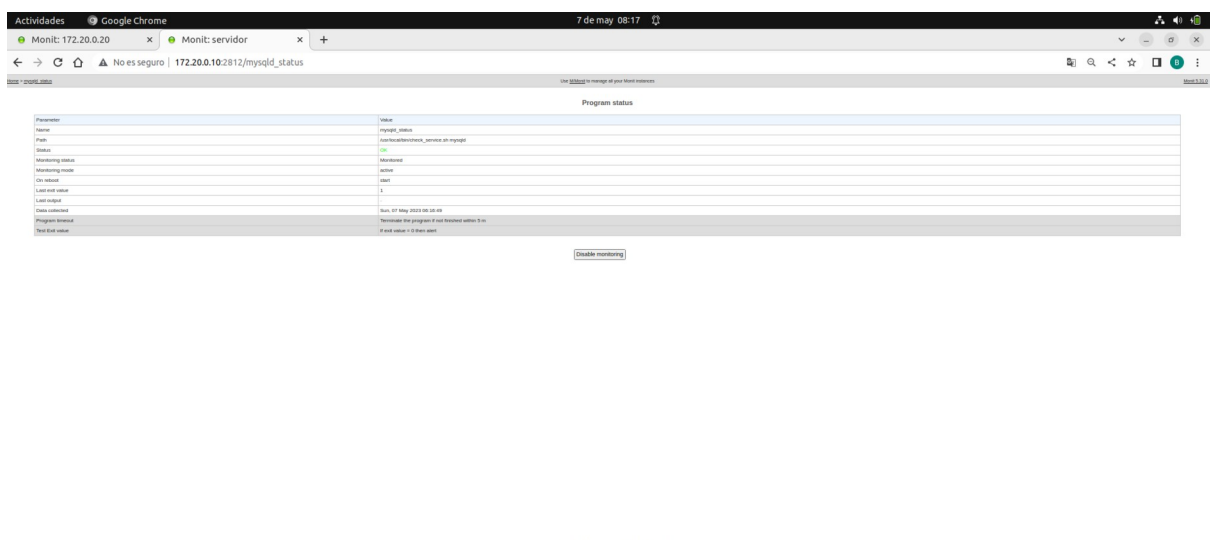


Figura 35: Interfaz web Monit servidor con servicio MySQL de cliente

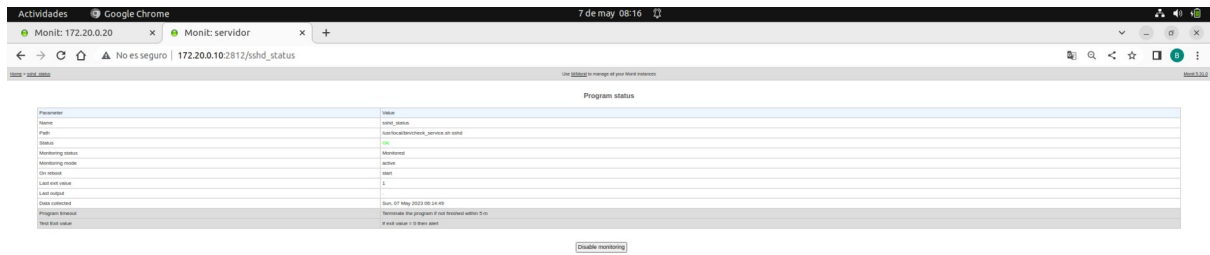


Figura 36: Interfaz web Monit servidor con servicio SSH de cliente



## **Webgrafía**

<https://es.wikipedia.org/wiki/Monit>

<https://jcastaneda.com/servidores/configurar-ip-estatica-en-ubuntu-server-22-04-jammy-jellyfish/>

<https://www.ochobitshacenunbyte.com/2014/11/13/como-instalar-lamp-en-debian-7/>

<https://voidnull.es/monitoriza-tu-servidor-con-monit-en-gnu-linux/>

<https://comoinstalar.me/como-instalar-phpmyadmin-en-ubuntu-22-04-lts/>

<https://www.linuxadictos.com/instalar-php-7-2-ubuntu-debian.html>

<https://ubuntu.tutorials24x7.com/blog/configure-monit-for-mysql-apache-nginx>

<https://noviello.it/es/como-instalar-el-servidor-mysql-en-ubuntu-22-04/>

<https://www.javierleal.com/ubuntu/monit-para-monitorizar-los-servicios-de-un-servidor-p5624?reload=266017>