

Práctica 6. Pruebas de rendimiento y monitorización en red

En esta práctica, haremos pruebas de rendimiento sencillas sobre dos servidores web con Apache y Nginx. Después, usaremos el monitor de sistemas en red Zabbix para monitorizar los servidores. Utilizaremos varias máquinas virtuales.

Contenido:

[Preparación del entorno \(10 min.\)](#)

[Pruebas de rendimiento \(40 min.\)](#)

[Monitorización de sistemas en red \(40 min.\)](#)

[Instalación de Zabbix](#)

[Descubrimiento automático y pruebas web](#)

Preparación del entorno (10 min.)

En esta práctica usaremos cuatro máquinas (`cliente`, `apache`, `nginx` y `zabbix`).

Una vez importada la imagen `ECO.ova`, en *Configuración* → *Red*, habilita el *Adaptador 2* (`eth1`), que estará conectado a una red interna llamada `intnet`.

Realiza tres clonaciones enlazadas generando nuevas direcciones MAC para todos los interfaces de red. Establece el nombre correspondiente a cada máquina. Inicia las máquinas.

Configura la máquina `cliente`:

```
$ sudo ip addr add 192.168.0.1/24 dev eth1
$ sudo ip link set dev eth1 up
$ sudo apt-get update
$ sudo apt-get install apache2-utils
```

Configura la máquina `apache`:

```
$ sudo ip addr add 192.168.0.2/24 dev eth1
$ sudo ip link set dev eth1 up
$ sudo apt-get update
$ sudo apt-get install apache2
```

Configura la máquina `nginx`:

```
$ sudo ip addr add 192.168.0.3/24 dev eth1
$ sudo ip link set dev eth1 up
$ sudo apt-get update
$ sudo apt-get install nginx
$ sudo service nginx start
```

Configura la máquina `zabbix`:

```
$ sudo ip addr add 192.168.0.4/24 dev eth1
$ sudo ip link set dev eth1 up
```

Pruebas de rendimiento (40 min.)

Apache Bench es una herramienta para realizar pruebas de rendimiento sencillas en servidores web.

Copia el contenido del fichero `/var/www/index.html` de `apache` en el fichero `/usr/share/nginx/www/index.html` de `nginx`.

```
apache -> /var/www/index.html
```

```
<html><body><h1>It works!</h1>
<p>This is the default web page for this server.</p>
<p>The web server software is running but no content has been added, yet.</p>
</body></html>
```

```
nginx -> /usr/share/nginx/www/index.html
```

```
<html>
<head>
<title>Welcome to nginx!</title>
</head>
<body bgcolor="white" text="black">
<center><h1>Welcome to nginx!</h1></center>
</body>
</html>
```

Usaremos Apache Bench (`ab`) en `cliente` para realizar pruebas de rendimiento en `apache` y `nginx` con un fichero estático simple (`index.html`). Por ejemplo, la siguiente orden realiza 1.000 peticiones con 25 peticiones concurrentes:

```
$ ab -n 1000 -c 25 -r http://192.168.0.2/
```

Realiza pruebas de carga consistentes en 10.000 peticiones con diferentes niveles de concurrencia (por ejemplo, con 1, 25, 50, 100, 250 y 500). Observa la productividad media (*Requests per second*) y el tiempo de respuesta (*Time per request*) para cada nivel de concurrencia. Si algún resultado no es convincente, repite la prueba.

Crea dos tablas (o, mejor, dos gráficos), una con la productividad y otra con el tiempo de respuesta, comparando Apache y Nginx. Escribe un análisis de los resultados.

Para practicar más:

- Usa páginas dinámicas (ej. PHP, acceso a bases de datos...) y peticiones más complejas (parámetros, método POST, cabeceras, autenticación, *cookies*...).
- Usa otras herramientas de pruebas de rendimiento (HTTPPerf, Siege, jMeter...).
- Haz pruebas con una aplicación real (OpenCart, Drupal, Wordpress, Joomla...).

DOCUMENTACION.

<https://www.datadoghq.com/blog/apachebench/>

Apache

```
ab -n 10000 -c 1 -r http://192.168.0.2/
```

```
Server Software:      Apache/2.2.22
Server Hostname:      192.168.0.2
Server Port:          80

Document Path:        /
Document Length:      177 bytes

Concurrency Level:    1
Time taken for tests:  3.648 seconds
Complete requests:    10000
Failed requests:      0
Write errors:         0
Total transferred:    4530000 bytes
HTML transferred:     1770000 bytes
Requests per second:  2741.50 [#/sec] (mean)
Time per request:     0.365 [ms] (mean)
Time per request:     0.365 [ms] (mean, across all concurrent requests)
Transfer rate:        1212.79 [Kbytes/sec] received
```

Connection Times (ms)

	min	mean[+/-sd]	median	max
Connect:	0	0 0.1	0	2
Processing:	0	0 0.2	0	6
Waiting:	0	0 0.2	0	6
Total:	0	0 0.2	0	7

Percentage of the requests served within a certain time (ms)

50%	0
66%	0
75%	0
80%	0
90%	0
95%	1
98%	1
99%	2
100%	7 (longest request)

```
usuario@debian:~$
```

```
ab -n 10000 -c 25 -r http://192.168.0.2/
```

```
Server Software:      Apache/2.2.22
Server Hostname:      192.168.0.2
Server Port:          80

Document Path:        /
Document Length:      177 bytes

Concurrency Level:     25
Time taken for tests:  2.066 seconds
Complete requests:     10000
Failed requests:        0
Write errors:           0
Total transferred:     4530000 bytes
HTML transferred:      1770000 bytes
Requests per second:   4839.98 [#/sec] (mean)
Time per request:       5.165 [ms] (mean)
Time per request:      0.207 [ms] (mean, across all concurrent requests)
Transfer rate:          2141.12 [Kbytes/sec] received
```

Connection Times (ms)

	min	mean[+/-sd]	median	max
Connect:	0	0 0.2	0	6
Processing:	1	5 0.7	5	33
Waiting:	1	5 0.7	5	33
Total:	1	5 0.7	5	33

Percentage of the requests served within a certain time (ms)

50%	5
66%	5
75%	5
80%	5
90%	6
95%	6
98%	7
99%	7
100%	33 (longest request)

```
usuario@debian:~$
```

```
ab -n 10000 -c 50 -r http://192.168.0.2/
```

```
Server Software:      Apache/2.2.22
Server Hostname:      192.168.0.2
Server Port:          80

Document Path:        /
Document Length:      177 bytes

Concurrency Level:     50
Time taken for tests:  2.043 seconds
Complete requests:     10000
Failed requests:       0
Write errors:          0
Total transferred:     4530000 bytes
HTML transferred:     1770000 bytes
Requests per second:   4895.24 [#/sec] (mean)
Time per request:      10.214 [ms] (mean)
Time per request:      0.204 [ms] (mean, across all concurrent requests)
Transfer rate:         2165.57 [Kbytes/sec] received
```

Connection Times (ms)

	min	mean[+/-sd]	median	max
Connect:	0	0 0.1	0	2
Processing:	2	10 1.8	10	69
Waiting:	2	10 1.7	10	50
Total:	3	10 1.8	10	69

Percentage of the requests served within a certain time (ms)

50%	10
66%	10
75%	11
80%	11
90%	11
95%	12
98%	13
99%	13
100%	69 (longest request)

```
usuario@debian:~$
```

```
ab -n 10000 -c 100 -r http://192.168.0.2/
```

```
Server Software:      Apache/2.2.22
Server Hostname:      192.168.0.2
Server Port:          80

Document Path:        /
Document Length:      177 bytes

Concurrency Level:     100
Time taken for tests:  2.053 seconds
Complete requests:     10000
Failed requests:       0
Write errors:          0
Total transferred:     4530000 bytes
HTML transferred:     1770000 bytes
Requests per second:   4871.76 [#/sec] (mean)
Time per request:      20.526 [ms] (mean)
Time per request:      0.205 [ms] (mean, across all concurrent requests)
Transfer rate:         2155.18 [Kbytes/sec] received
```

Connection Times (ms)

	min	mean[+/-sd]	median	max
Connect:	0	0 0.4	0	5
Processing:	3	20 2.5	20	80
Waiting:	3	20 2.5	20	80
Total:	8	20 2.4	20	80

Percentage of the requests served within a certain time (ms)

50%	20
66%	21
75%	21
80%	21
90%	22
95%	23
98%	24
99%	25
100%	80 (longest request)

```
usuario@debian:~$
```

```
ab -n 10000 -c 250 -r http://192.168.0.2/
```

```
Server Software:      Apache/2.2.22
Server Hostname:      192.168.0.2
Server Port:          80

Document Path:        /
Document Length:      177 bytes

Concurrency Level:     250
Time taken for tests:  3.089 seconds
Complete requests:     10000
Failed requests:       0
Write errors:          0
Total transferred:     4530000 bytes
HTML transferred:     1770000 bytes
Requests per second:   3237.52 [#/sec] (mean)
Time per request:      77.220 [ms] (mean)
Time per request:      0.309 [ms] (mean, across all concurrent requests)
Transfer rate:         1432.22 [Kbytes/sec] received

Connection Times (ms)
              min    mean[+/-sd] median    max
Connect:        0      3  53.7      0    1000
Processing:      6     46 209.7     26    3074
Waiting:        6     46 209.7     26    3074
Total:         15     50 216.9     26    3081

Percentage of the requests served within a certain time (ms)
 50%    26
 66%    27
 75%    28
 80%    28
 90%    29
 95%    30
 98%    35
 99%   1028
100%   3081 (longest request)
usuario@debian:~$
```

```
ab -n 10000 -c 500 -r http://192.168.0.2/
```

```
Server Software:      Apache/2.2.22
Server Hostname:      192.168.0.2
Server Port:          80

Document Path:        /
Document Length:      177 bytes

Concurrency Level:     500
Time taken for tests:  6.375 seconds
Complete requests:     10000
Failed requests:       3
    (Connect: 0, Receive: 1, Length: 1, Exceptions: 1)
Write errors:          0
Total transferred:     4529547 bytes
HTML transferred:      1769823 bytes
Requests per second:   1568.70 [#/sec] (mean)
Time per request:      318.735 [ms] (mean)
Time per request:      0.637 [ms] (mean, across all concurrent requests)
Transfer rate:         693.90 [Kbytes/sec] received

Connection Times (ms)
      min   mean[+/-sd] median   max
Connect:    0    14 115.6      0   1000
Processing:  6   135 676.4     27   6350
Waiting:    0   134 673.5     27   6350
Total:      16   149 700.0     27   6364

Percentage of the requests served within a certain time (ms)
 50%    27
 66%    28
 75%    28
 80%    29
 90%    30
 95%    35
 98%   3093
 99%   3101
100%   6364 (longest request)
usuario@debian:~$
```


Nginx

```
ab -n 10000 -c 1 -r http://192.168.0.3/
```

```
Server Software:      nginx/1.2.1
Server Hostname:      192.168.0.3
Server Port:          80

Document Path:        /
Document Length:      177 bytes

Concurrency Level:     1
Time taken for tests:  2.586 seconds
Complete requests:     10000
Failed requests:       0
Write errors:          0
Total transferred:     3870000 bytes
HTML transferred:     1770000 bytes
Requests per second:   3866.43 [#/sec] (mean)
Time per request:      0.259 [ms] (mean)
Time per request:      0.259 [ms] (mean, across all concurrent requests)
Transfer rate:         1461.24 [Kbytes/sec] received

Connection Times (ms)
              min    mean[+/-sd] median    max
Connect:        0      0   0.0      0      2
Processing:      0      0   0.2      0     15
Waiting:         0      0   0.2      0     15
Total:           0      0   0.2      0     15

Percentage of the requests served within a certain time (ms)
 50%      0
 66%      0
 75%      0
 80%      0
 90%      0
 95%      0
 98%      0
 99%      0
100%     15 (longest request)
usuario@debian:~$
```

```
ab -n 10000 -c 25 -r http://192.168.0.3/
```

```
Server Software:      nginx/1.2.1
Server Hostname:      192.168.0.3
Server Port:          80

Document Path:        /
Document Length:      177 bytes

Concurrency Level:     25
Time taken for tests:  1.183 seconds
Complete requests:     10000
Failed requests:       0
Write errors:          0
Total transferred:     3870000 bytes
HTML transferred:     1770000 bytes
Requests per second:   8454.56 [#/sec] (mean)
Time per request:      2.957 [ms] (mean)
Time per request:      0.118 [ms] (mean, across all concurrent requests)
Transfer rate:         3195.23 [Kbytes/sec] received

Connection Times (ms)
              min    mean[+/-sd] median    max
Connect:        0      0   0.3      0      5
Processing:      0      3   0.6      3     11
Waiting:         0      3   0.6      3     11
Total:           1      3   0.5      3     11

Percentage of the requests served within a certain time (ms)
 50%      3
 66%      3
 75%      3
 80%      3
 90%      3
 95%      4
 98%      4
 99%      4
100%     11 (longest request)
usuario@debian:~$
```

```
ab -n 10000 -c 50 -r http://192.168.0.3/
```

```
Server Software:      nginx/1.2.1
Server Hostname:      192.168.0.3
Server Port:          80

Document Path:        /
Document Length:      177 bytes

Concurrency Level:     50
Time taken for tests:  1.177 seconds
Complete requests:     10000
Failed requests:       0
Write errors:          0
Total transferred:     3870000 bytes
HTML transferred:      1770000 bytes
Requests per second:   8496.96 [#/sec] (mean)
Time per request:      5.884 [ms] (mean)
Time per request:      0.118 [ms] (mean, across all concurrent requests)
Transfer rate:         3211.25 [Kbytes/sec] received
```

Connection Times (ms)

	min	mean[+/-sd]	median	max
Connect:	0	0 0.5	0	6
Processing:	1	6 1.1	6	18
Waiting:	1	5 1.1	5	18
Total:	3	6 0.9	6	18

Percentage of the requests served within a certain time (ms)

50%	6
66%	6
75%	6
80%	6
90%	7
95%	7
98%	9
99%	10
100%	18 (longest request)

```
usuario@debian:~$ █
```

```
ab -n 10000 -c 100 -r http://192.168.0.3/
```

```
Server Software:      nginx/1.2.1
Server Hostname:      192.168.0.3
Server Port:          80

Document Path:        /
Document Length:      177 bytes

Concurrency Level:    100
Time taken for tests:  1.352 seconds
Complete requests:    10000
Failed requests:      0
Write errors:         0
Total transferred:    3870000 bytes
HTML transferred:     1770000 bytes
Requests per second:  7396.58 [#/sec] (mean)
Time per request:     13.520 [ms] (mean)
Time per request:     0.135 [ms] (mean, across all concurrent requests)
Transfer rate:        2795.39 [Kbytes/sec] received
```

Connection Times (ms)

	min	mean[+/-sd]	median	max
Connect:	0	1 1.5	0	13
Processing:	2	13 3.1	12	31
Waiting:	1	12 3.0	12	30
Total:	5	13 3.2	12	31

Percentage of the requests served within a certain time (ms)

50%	12
66%	13
75%	14
80%	15
90%	18
95%	20
98%	24
99%	26
100%	31 (longest request)

```
usuario@debian:~$ █
```

```
ab -n 10000 -c 250 -r http://192.168.0.3/
```

```
Server Software:      nginx/1.2.1
Server Hostname:      192.168.0.3
Server Port:          80

Document Path:        /
Document Length:      177 bytes

Concurrency Level:    250
Time taken for tests:  1.671 seconds
Complete requests:    10000
Failed requests:      0
Write errors:         0
Total transferred:    3870000 bytes
HTML transferred:     1770000 bytes
Requests per second:  5983.62 [#/sec] (mean)
Time per request:     41.781 [ms] (mean)
Time per request:     0.167 [ms] (mean, across all concurrent requests)
Transfer rate:        2261.39 [Kbytes/sec] received
```

Connection Times (ms)

	min	mean[+/-sd]	median	max
Connect:	0	12 106.0	0	1001
Processing:	7	20 48.6	16	628
Waiting:	5	20 48.6	16	628
Total:	12	33 144.7	17	1626

Percentage of the requests served within a certain time (ms)

50%	17
66%	17
75%	18
80%	18
90%	19
95%	21
98%	31
99%	1015
100%	1626 (longest request)

```
usuario@debian:~$
```

```
ab -n 10000 -c 500 -r http://192.168.0.3/
```

```
Server Software:      nginx/1.2.1
Server Hostname:      192.168.0.3
Server Port:          80

Document Path:        /
Document Length:      177 bytes

Concurrency Level:     500
Time taken for tests:  1.691 seconds
Complete requests:     10000
Failed requests:       0
Write errors:          0
Total transferred:     3870000 bytes
HTML transferred:     1770000 bytes
Requests per second:   5915.21 [#/sec] (mean)
Time per request:      84.528 [ms] (mean)
Time per request:      0.169 [ms] (mean, across all concurrent requests)
Transfer rate:         2235.53 [Kbytes/sec] received
```

Connection Times (ms)

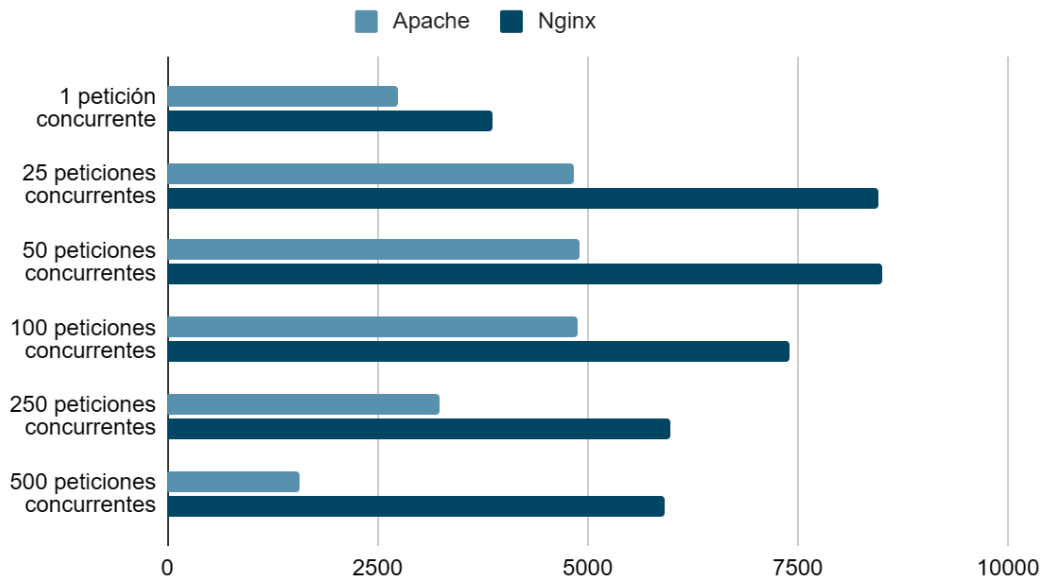
	min	mean[+/-sd]	median	max
Connect:	0	31 170.3	0	1005
Processing:	9	25 66.9	17	643
Waiting:	6	25 67.0	16	643
Total:	10	56 218.2	17	1631

Percentage of the requests served within a certain time (ms)

50%	17
66%	18
75%	19
80%	19
90%	21
95%	45
98%	1022
99%	1625
100%	1631 (longest request)

```
usuario@debian:~$
```

Productividad (peticiones/segundo)



Se puede observar que en todos los casos nginx tiene una productividad mucho mayor, siendo capaz de manejar un número de peticiones por segundo mucho mayor. Además contra mayor número de peticiones concurrentes hay se hace más notable la diferencia entre apache y nginx.

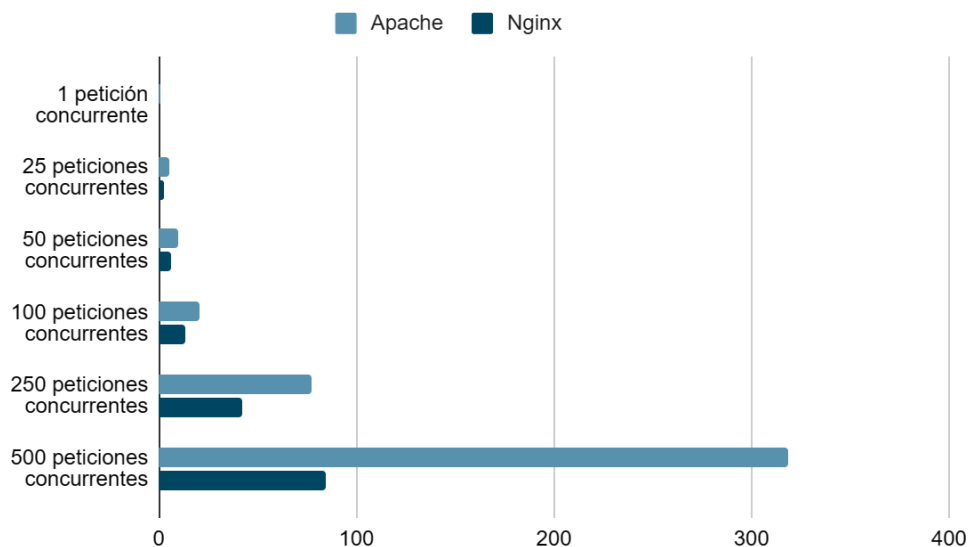
Diferencia entre los dos campos de time per request (ejemplo para entenderlo)

Assume that we make **3** requests with **3** concurrent connections. The **Time taken for tests** will be **90ms** and each request are 40ms, 50ms, 30ms. So what's the value of these two **Time per request**?

- Time per request (mean) = $(40 + 50 + 30) / 3 = 40\text{ms}$
- Time per request (mean, across all concurrent requests) = $90 / 3 = 30\text{ms}$

Para hacer el gráfico he cogido el (mean):

Tiempo de respuesta (ms)



En este grafo se puede ver como cuando solo hay una petición concurrente tanto apache como nginx tienen un tiempo constante. Sin embargo, a medida que se van aumentando el número de peticiones concurrentes, apache tiene un tiempo de respuesta mucho mayor que nginx.

Monitorización de sistemas en red (40 min.)

Instalación de Zabbix

Zabbix (www.zabbix.com) es una herramienta gratuita y de código abierto diseñada para la monitorización en tiempo real de millones de métricas recogidas desde decenas de miles de servidores, máquinas virtuales y dispositivos de red.

Configura el repositorio de Zabbix en apache, nginx y zabbix:

```
$ wget
http://repo.zabbix.com/zabbix/3.4/debian/pool/main/z/zabbix
release/zabbix-release_3.4-1+wheezy_all.deb
```

El enlace viene mal, este es el correcto y AÑADIR SUDO

http://repo.zabbix.com/zabbix/3.4/debian/pool/main/z/zabbix-release/zabbix-release_3.4-1%2Bwheezy_all.deb

```
$ sudo dpkg -i zabbix-release_3.4-1+wheezy_all.deb
```

```
sudo dpkg -i zabbix-release_3.4-1+jessie_all.deb
```

```
$ sudo apt-get update
```

Instala el agente de Zabbix en apache y nginx:

```
$ sudo apt-get install zabbix-agent
```

En el fichero /etc/zabbix/zabbix_agentd.conf, cambia la siguiente

opción: Server=192.168.0.4

Finalmente, reinicia el agente de Zabbix:

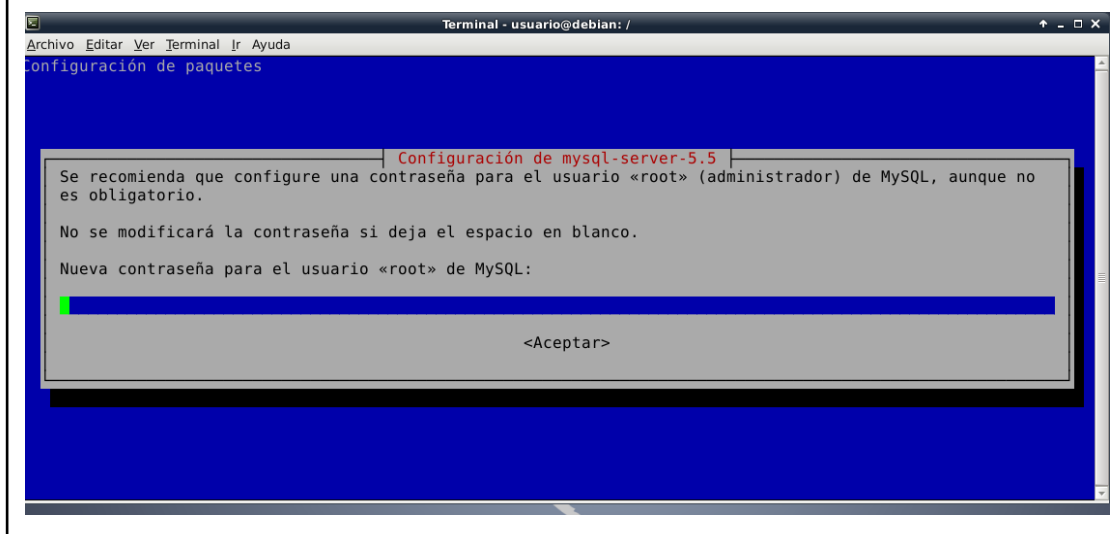
```
$ sudo service zabbix-agent restart
```

```
usuario@debian:~$ sudo nano /etc/zabbix/zabbix_agentd.conf
usuario@debian:~$ sudo service zabbix-agent restart
[ ok ] zabbix_agentd stopping...done.
[ ok ] zabbix_agentd starting...done.
usuario@debian:~$
```


Instala el servidor y el interfaz *web* de Zabbix en zabbix:

```
$ sudo apt-get install apache2 zabbix-server-mysql  
zabbix frontend-php php5-mysql
```

pssw: zabbix



NOTA: zabbix-frontend-php

Establece “zabbix” como contraseña del usuario root de MySQL cuando se solicite.

Configura el servidor de Zabbix en zabbix:

```
$ sudo mysql -uroot -pzabbix << EOF  
create database zabbix character set utf8 collate  
utf8_bin; grant all privileges on zabbix.* to  
zabbix@localhost identified by 'zabbix';  
quit  
EOF
```

He tenido que volver a crearla
Nombre DB: zabbix3

```
Enter password:
ERROR 1007 (HY000) at line 1: Can't create database 'zabbix2'; database exists
usuario@debian:~$ sudo mysql -u root -p zabbix << EOF
create database zabbix3 character set utf8 collate utf8_bin;
grant all privileges on zabbix.* to zabbix@localhost identified by 'zabbix3';
quit
EOF

Enter password:
usuario@debian:~$
```

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> h
-> help
-> \help

For information about MySQL products and services, visit:
  http://www.mysql.com/
For developer information, including the MySQL Reference Manual, visit:
  http://dev.mysql.com/
To buy MySQL Enterprise support, training, or other products, visit:
  https://shop.mysql.com/

List of all MySQL commands:
Note that all text commands must be first on line and end with ';'
?      (\?) Synonym for 'help'.
clear  (\c) Clear the current input statement.
connect (\r) Reconnect to the server. Optional arguments are db and host.
delimiter (\d) Set statement delimiter.
edit    (\e) Edit command with $EDITOR.
ego     (\G) Send command to mysql server, display result vertically.
exit    (\q) Exit mysql. Same as quit.
go      (\g) Send command to mysql server.
help    (\h) Display this help.
nopager (\n) Disable pager, print to stdout.
noteee  (\t) Don't write into outfile.
pager   (\P) Set PAGER [to pager]. Print the query results via PAGER.
print    (\p) Print current command.
prompt  (\R) Change your mysql prompt.
quit    (\q) Quit mysql.
rehash  (\#) Rebuild completion hash.
source  (\.) Execute an SQL script file. Takes a file name as an argument.
status  (\s) Get status information from the server.
system  (\!) Execute a system shell command.
tee      (\T) Set outfile [to outfile]. Append everything into given outfile.
use      (\u) Use another database. Takes database name as argument.
charset (\C) Switch to another charset. Might be needed for processing binlog with multi-byte charsets.
warnings (\W) Show warnings after every statement.
nowarning (\w) Don't show warnings after every statement.

For server side help, type 'help contents'

->
```

He tenido que borrar la db -> var/lib/mysql/zabbix

```
Terminal - usuario@debian: /var/lib
Archivo Editar Ver Terminal Ir Ayuda
total 28696
-rwx----- 5 mysql mysql      4096 mar 24 11:06 .
-rwxr-xr-x 40 root  root      4096 mar 24 11:05 ..
-rw-r--r-- 1 root  root         0 mar 24 11:05 debian-5.5.flag
-rw-rw---- 1 mysql mysql 18874368 mar 24 11:05 ibdata1
-rw-rw---- 1 mysql mysql 5242880 mar 24 11:05 ib_logfile0
-rw-rw---- 1 mysql mysql 5242880 mar 24 11:05 ib_logfile1
-rwx----- 2 mysql root       4096 mar 24 11:05 mysql
-rw----- 1 root  root         6 mar 24 11:05 mysql_upgrade_info
-rwx----- 2 mysql mysql      4096 mar 24 11:05 performance_schema
-rwx----- 2 mysql mysql      4096 mar 24 11:06 zabbix
root@debian:/var/lib/mysql# rmdir zabbix/
rmdir: fallo al borrar «zabbix/»: El directorio no está vacío
root@debian:/var/lib/mysql# cd zabbix/
root@debian:/var/lib/mysql/zabbix# ls
db.opt
root@debian:/var/lib/mysql/zabbix# rm db.opt
root@debian:/var/lib/mysql/zabbix# cd -
/var/lib/mysql
root@debian:/var/lib/mysql# rmdir zabbix/
root@debian:/var/lib/mysql# ls
debian-5.5.flag  ib_logfile0  mysql          performance_schema
ibdata1          ib_logfile1  mysql_upgrade_info
root@debian:/var/lib/mysql#
```

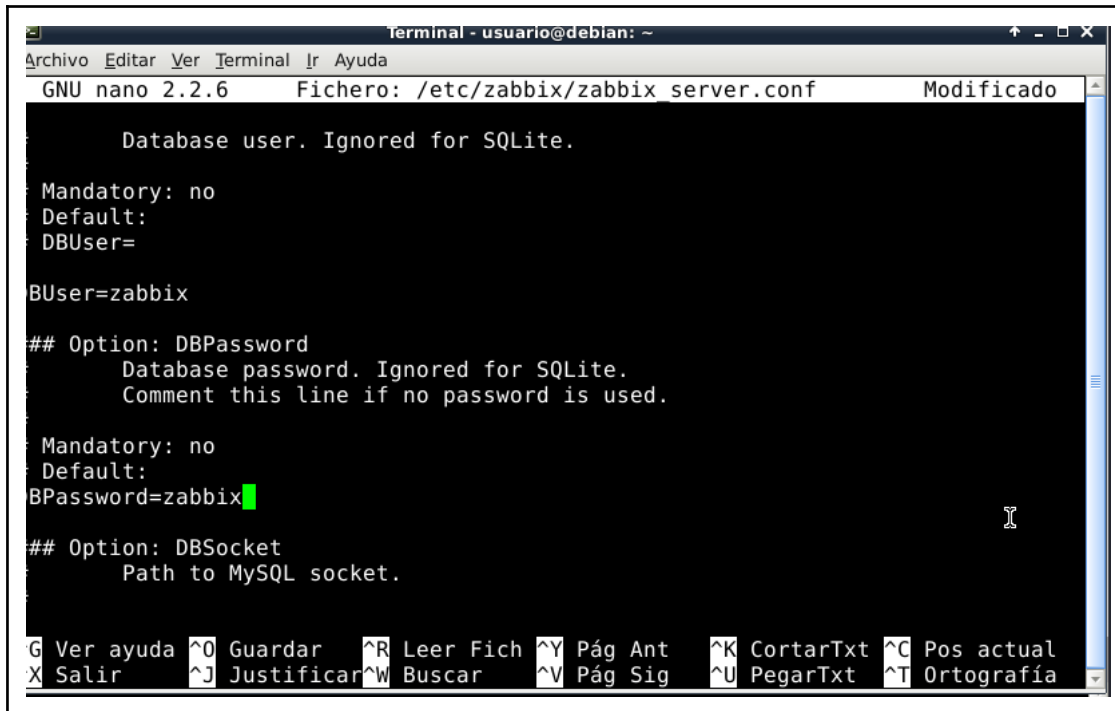
zcat /usr/share/zabbix-server-mysql/schema.sql.gz | mysql -uzabbix -p zabbix

```
Terminal - usuario@debian: /var/lib
Archivo Editar Ver Terminal Ir Ayuda
debian-5.5.flag  ib_logfile0  mysql          performance_schema
ibdata1          ib_logfile1  mysql_upgrade_info  zabbix
root@debian:/var/lib/mysql# cd zabbix/
root@debian:/var/lib/mysql/zabbix# ls
acknowledges.frm      host_discovery.frm      opmessage_usr.frm
actions.frm            host_inventory.frm      optemplate.frm
alerts.frm             hostmacro.frm           profiles.frm
applications.frm       hosts.frm               proxy_autoreg_host.frm
application_template.frm hosts_groups.frm         proxy_dhistory.frm
auditlog_details.frm   hosts_templates.frm     proxy_history.frm
auditlog.frm           housekeeper.frm         regexps.frm
autoreg_host.frm       httpstep.frm            rights.frm
conditions.frm         httpstepitem.frm        screens.frm
config.frm             httpstest.frm           screens_items.frm
db.opt                 httpstestitem.frm       scripts.frm
dbversion.frm          icon_map.frm            service_alarms.frm
dchecks.frm            icon_mapping.frm        services.frm
dhosts.frm             ids.frm                 services_links.frm
drules.frm             images.frm               services_times.frm
dservices.frm          interface_discovery.frm  sessions.frm
escalations.frm        interface.frm            slides.frm
events.frm             item_discovery.frm      slideshows.frm
expressions.frm        items_applications.frm  sysmap_element_url.frm
functions.frm          items.frm                sysmaps_elements.frm
globalmacro.frm        maintenances.frm         sysmaps.frm
globalvars.frm         maintenances_groups.frm  sysmaps_links.frm
graph_discovery.frm    maintenances_hosts.frm  sysmaps_link_triggers.frm
graphs.frm             maintenances_windows.frm sysmap_url.frm
graphs_items.frm       mappings.frm             timeperiods.frm
graph_theme.frm        media.frm                trends.frm
group_discovery.frm    media_type.frm          trends_uint.frm
group_prototype.frm    node_cksum.frm          trigger_depends.frm
groups.frm             nodes.frm                trigger_discovery.frm
history.frm            opcommand.frm           triggers.frm
history_log.frm        opcommand_grp.frm       user_history.frm
history_str.frm        opcommand_hst.frm      users.frm
history_str_sync.frm   opconditions.frm        users_groups.frm
history_sync.frm       operations.frm          usrgroup.frm
history_text.frm       opgroup.frm             valuemaps.frm
history_uint.frm       opmessage.frm
history_uint_sync.frm  opmessage_grp.frm
root@debian:/var/lib/mysql/zabbix#
```

```
$ zcat /usr/share/doc/zabbix-server-mysql/create.sql.gz  
| mysql -uzabbix -pzabbix zabbix
```

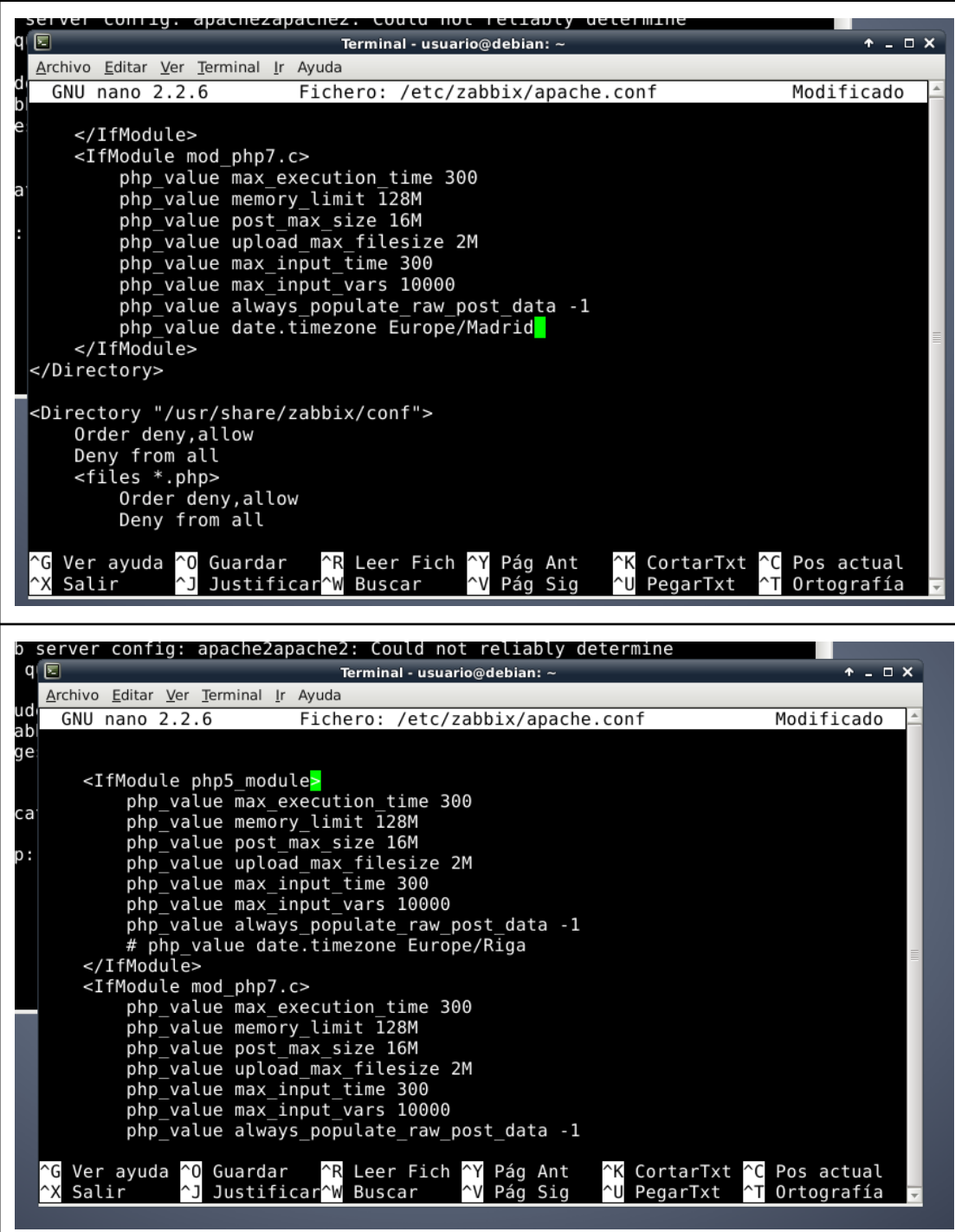
En el fichero `/etc/zabbix/zabbix_server.conf`, establece la siguiente opción (y asegúrate de que no esté comentada):

```
DBPassword=zabbix
```



```
Terminal - usuario@debian: ~  
GNU nano 2.2.6 Fichero: /etc/zabbix/zabbix_server.conf Modificado  
  
Database user. Ignored for SQLite.  
  
Mandatory: no  
Default:  
DBUser=  
  
BUser=zabbix  
  
## Option: DBPassword  
Database password. Ignored for SQLite.  
Comment this line if no password is used.  
  
Mandatory: no  
Default:  
BPassword=zabbix  
  
## Option: DBSocket  
Path to MySQL socket.  
  
G Ver ayuda ^O Guardar ^R Leer Fich ^Y Pág Ant ^K CortarTxt ^C Pos actual  
X Salir ^J Justificar ^W Buscar ^V Pág Sig ^U PegarTxt ^T Ortografía
```

En el fichero `/etc/zabbix/apache.conf`, descomenta la opción `date.timezone` y cámbiala a `Europe/Madrid`. Cambia también la línea con `<IfModule mod_php5.c>` a `<IfModule php5_module>`, o bien, coméntala junto con el correspondiente `</IfModule>`. Finalmente, reinicia Apache y el servidor de Zabbix:



```
server config: apache2apache2: Could not reliably determine
GNU nano 2.2.6      Fichero: /etc/zabbix/apache.conf      Modificado

</IfModule>
<IfModule mod_php7.c>
    php_value max_execution_time 300
    php_value memory_limit 128M
    php_value post_max_size 16M
    php_value upload_max_filesize 2M
    php_value max_input_time 300
    php_value max_input_vars 10000
    php_value always_populate_raw_post_data -1
    php_value date.timezone Europe/Madrid
</IfModule>
</Directory>

<Directory "/usr/share/zabbix/conf">
    Order deny,allow
    Deny from all
    <files *.php>
        Order deny,allow
        Deny from all

^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Pág Ant ^K CortarTxt ^C Pos actual
^X Salir    ^J Justificar ^W Buscar ^V Pág Sig ^U PegarTxt ^T Ortografía

server config: apache2apache2: Could not reliably determine
GNU nano 2.2.6      Fichero: /etc/zabbix/apache.conf      Modificado

<IfModule php5_module>
    php_value max_execution_time 300
    php_value memory_limit 128M
    php_value post_max_size 16M
    php_value upload_max_filesize 2M
    php_value max_input_time 300
    php_value max_input_vars 10000
    php_value always_populate_raw_post_data -1
    # php_value date.timezone Europe/Riga
</IfModule>
<IfModule mod_php7.c>
    php_value max_execution_time 300
    php_value memory_limit 128M
    php_value post_max_size 16M
    php_value upload_max_filesize 2M
    php_value max_input_time 300
    php_value max_input_vars 10000
    php_value always_populate_raw_post_data -1

^G Ver ayuda ^O Guardar ^R Leer Fich ^Y Pág Ant ^K CortarTxt ^C Pos actual
^X Salir    ^J Justificar ^W Buscar ^V Pág Sig ^U PegarTxt ^T Ortografía

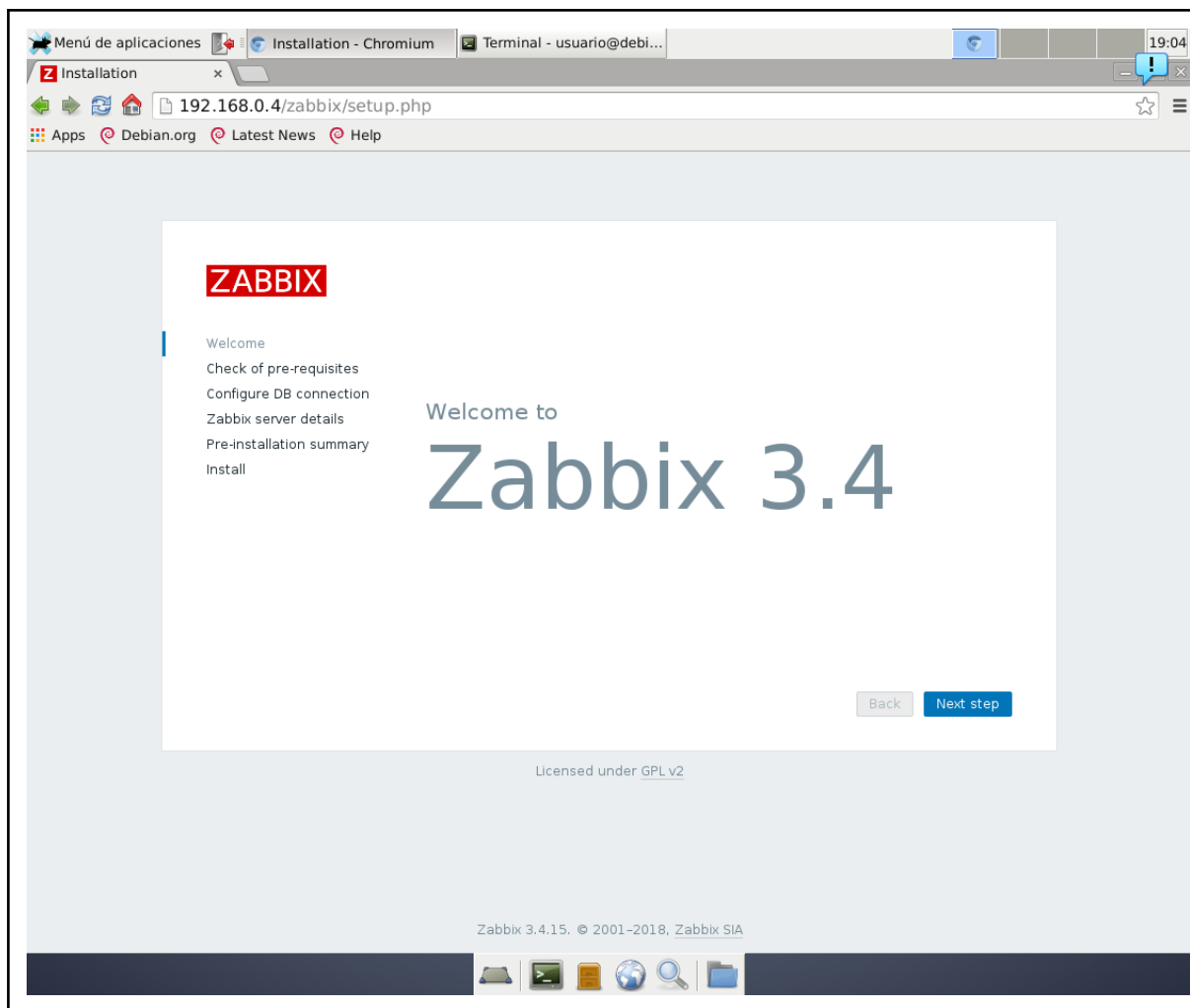
$ sudo service apache2 restart
$ sudo service zabbix-server restart
```

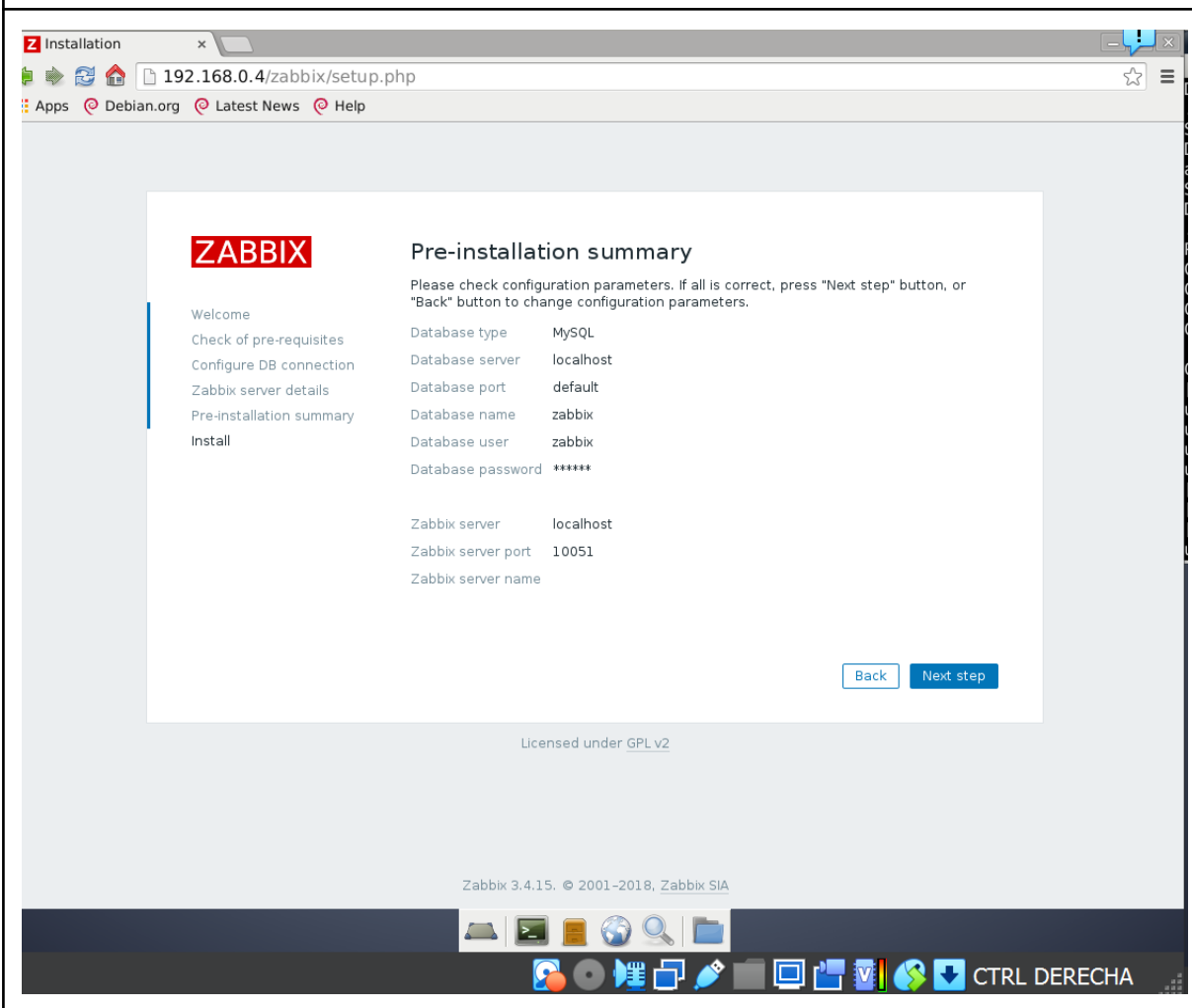
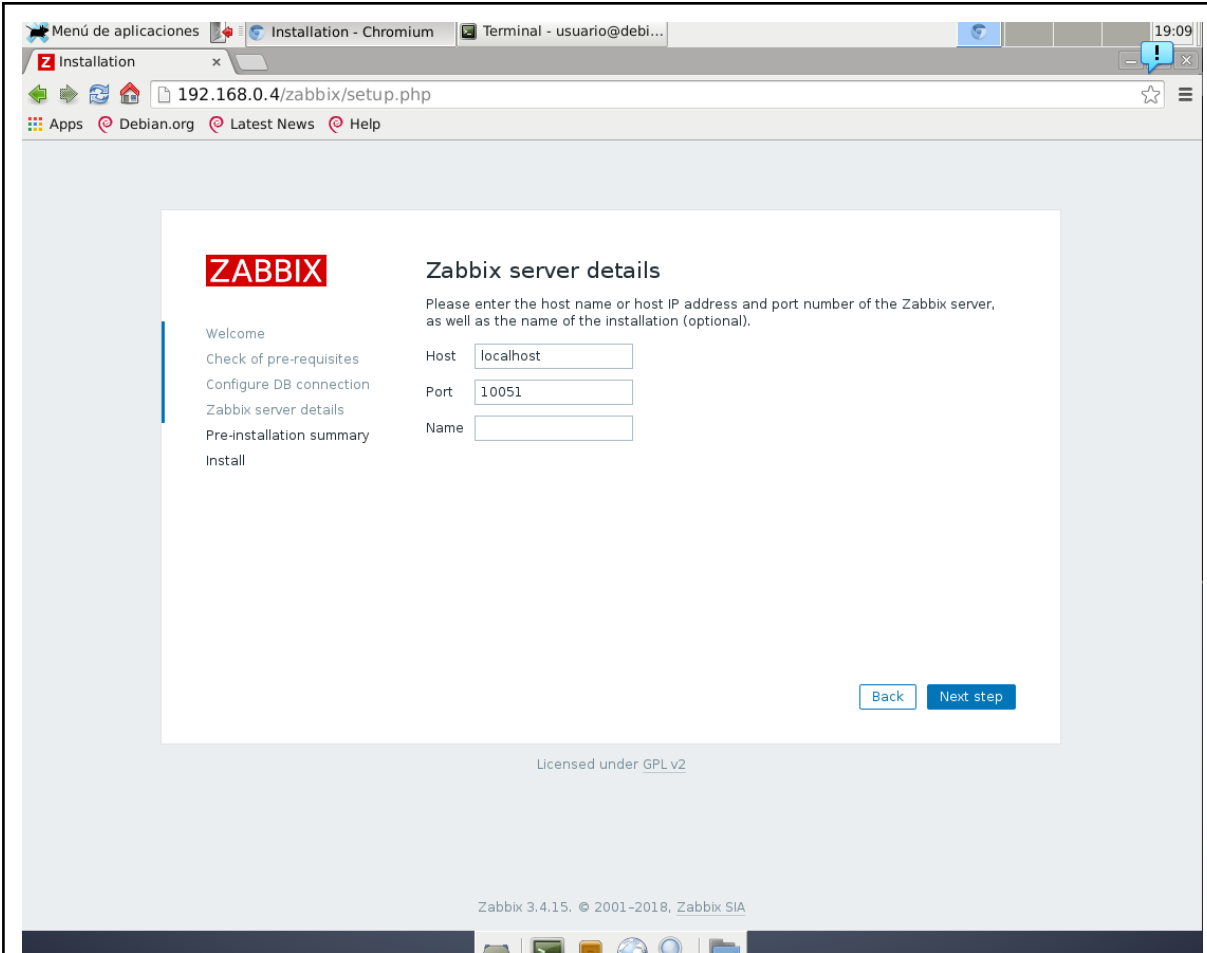
```
Terminal - usuario@debian: ~
Archivo Editar Ver Terminal Ir Ayuda
usuario@debian:~$ sudo service apache2 restart
[....] Restarting web server: apache2apache2: Could not reliably determine the s
erver's fully qualified domain name, using 127.0.1.1 for ServerName
... waiting apache2: Could not reliably determine the server's fully qualified
domain name, using 127.0.1.1 for ServerName
. ok
usuario@debian:~$ sudo service zabbix-server restart
[....] Stopping Zabbix server: zabbix_serverNo process in pidfile '/var/run/zabb
ix/zabbix_server.pid' found running; none killed.
. ok
[ ok ] Starting Zabbix server: zabbix_server.
usuario@debian:~$
```

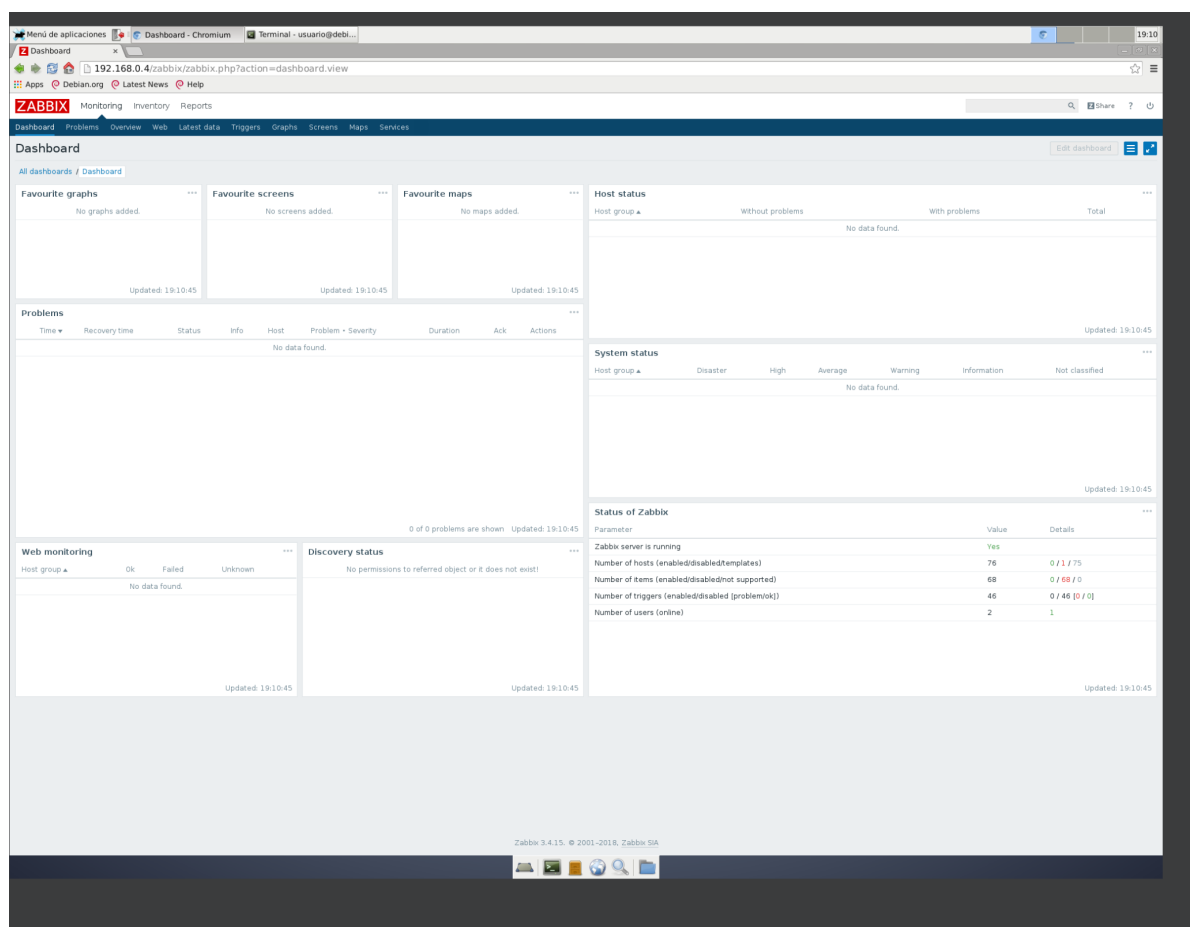
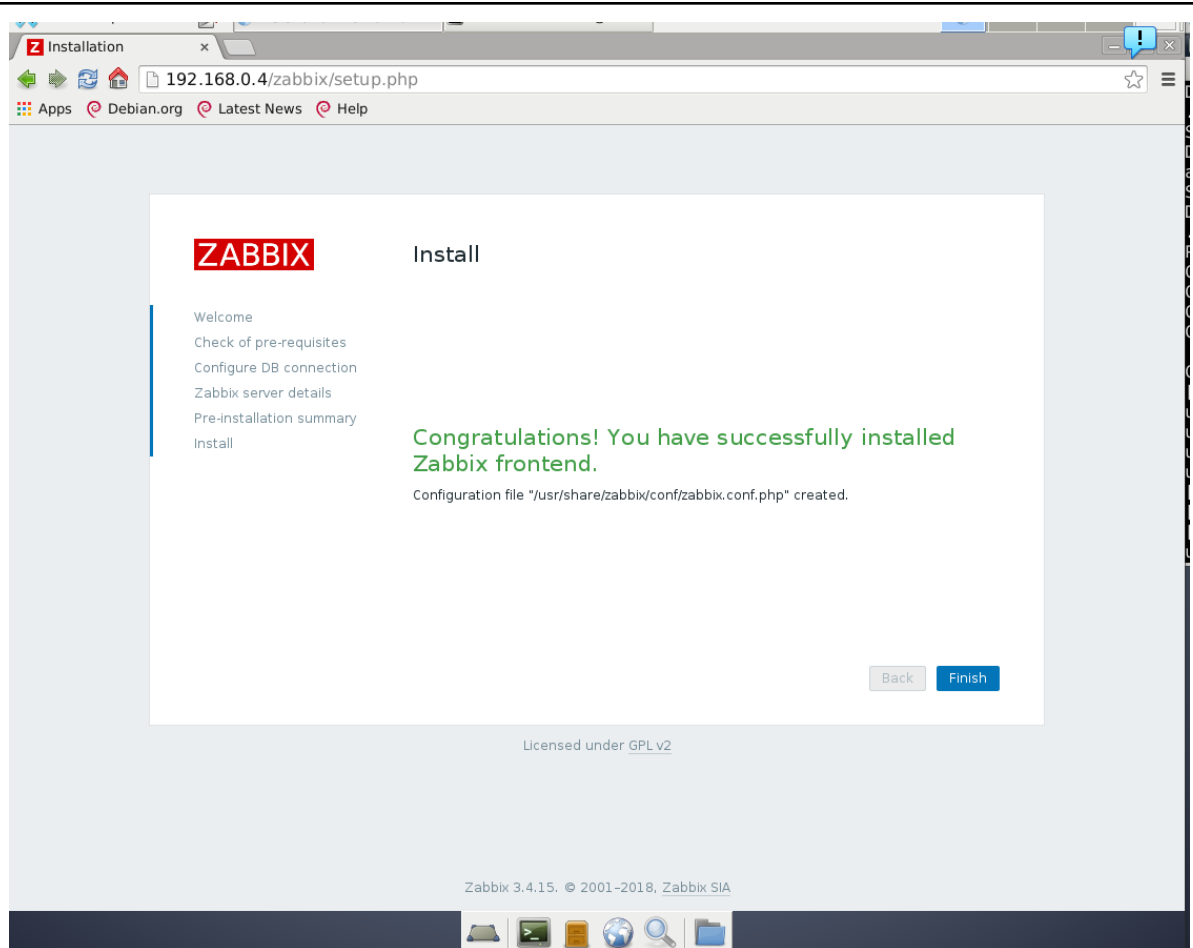
Abre un navegador en cliente y ve a 192.168.0.4/zabbix. Completa la instalación del interfaz web con los valores que aparecen por defecto, asegurándote de que el tipo de base de datos es MySQL y de que el usuario y la contraseña son “zabbix”. Cuando hayas finalizado el proceso de configuración, aparecerá una pantalla de *login*. Entra con el usuario “Admin” y la contraseña “zabbix”.

```
-- 192.168.0.4 ping statistics ---
  packets transmitted, 4 received, 0% packet loss, time 3004ms
tt min/avg/max/mdev = 0.951/1.030/1.064/0.046 ms
suario@debian:~$ ping 192.168.0.4 -c 5
PING 192.168.0.4 (192.168.0.4) 56(84) bytes of data.
 4 bytes from 192.168.0.4: icmp_req=1 ttl=64 time=1.32 ms
 4 bytes from 192.168.0.4: icmp_req=2 ttl=64 time=1.06 ms
 4 bytes from 192.168.0.4: icmp_req=3 ttl=64 time=1.06 ms
 4 bytes from 192.168.0.4: icmp_req=4 ttl=64 time=1.16 ms
 4 bytes from 192.168.0.4: icmp_req=5 ttl=64 time=1.00 ms

-- 192.168.0.4 ping statistics ---
  packets transmitted, 5 received, 0% packet loss, time 4006ms
tt min/avg/max/mdev = 1.003/1.123/1.324/0.112 ms
suario@debian:~$
```







Descubrimiento automático y pruebas web

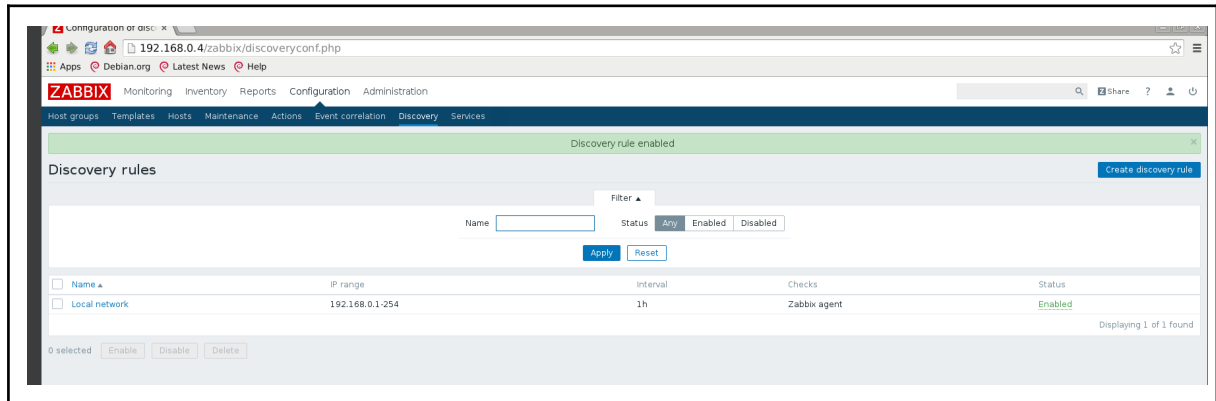
He tenido que cambiar el usuario y contraseña:

user: Admin

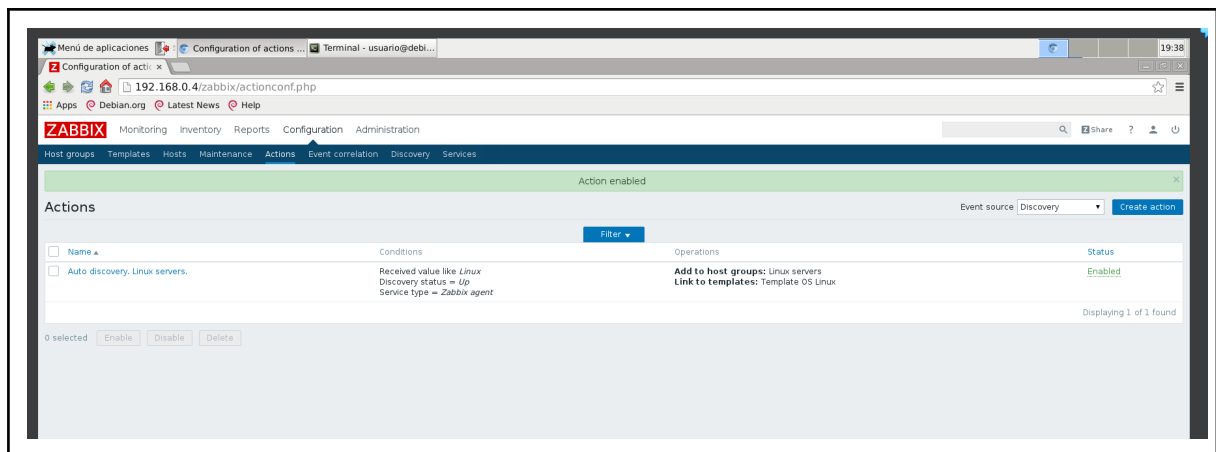
passwd;zabbix

```
mysql> update users set passwd=md5('zabbix') where alias='Admin';
```

En *Configuration* → *Discovery*, habilita la regla *Local network* pulsando en **Disabled**, que cambiará a **Enabled**.

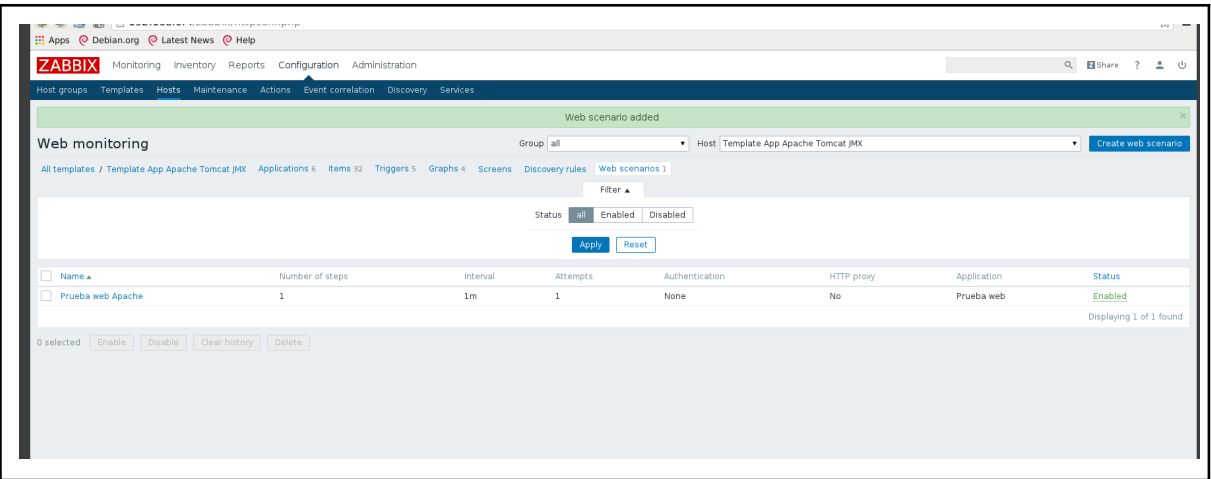
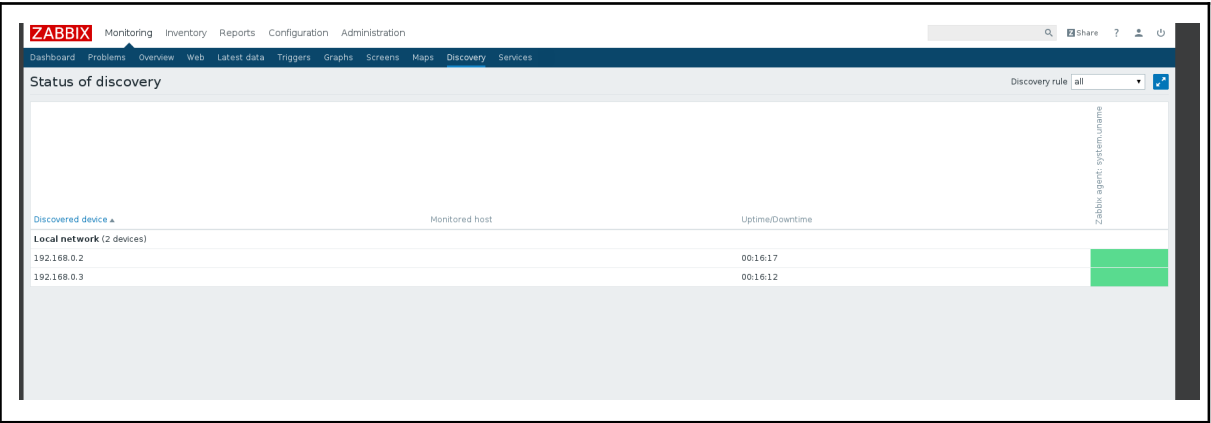


En *Configuration* → *Actions*, selecciona *Discovery* en *Event source* and habilita la acción *Auto discovery*. *Linux servers* pulsando en **Disabled**, que cambiará a **Enabled**.



Comprueba que se descubren automáticamente las máquinas `apache` y `nginx`. Si no, prueba a reducir el intervalo de actualización en la regla *Local network*. También puedes comprobar si hay algún mensaje de error en el fichero de `log /var/log/zabbix/zabbix_server.log`.

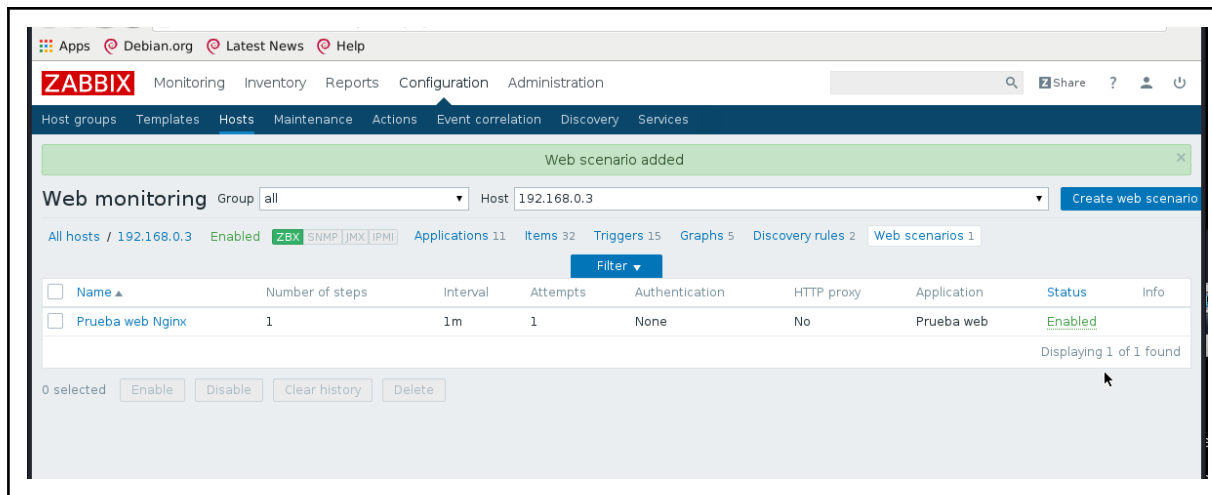
En *Configuration* → *Hosts*, pulsa *Web* en la fila correspondiente a `apache`. Pulsa en *Create scenario*. En el formulario, establece *Name* a “Prueba web Apache”, *New application* a “Prueba web”. Pulsa *Steps* y luego *Add*. En el formulario, establece *Name* a “index” y *URL* a “{HOST.IP}/index.html”. Pulsa *Add* para añadir el paso y *Add* para añadir el escenario.



The screenshot shows the Zabbix web interface. The top navigation bar includes links for Apps, Debian.org, Latest News, and Help. The main menu has options like Monitoring, Inventory, Reports, Configuration, and Administration. The 'Problems' tab is active, displaying a list of problems. A filter panel on the left allows for searching by host groups, hosts, application, triggers, and problem details. A table below the filter shows a single problem: 'Zabbix server' with a duration of 48s and a status of 'No'. A terminal window titled 'Scripts - Chromium' is open, showing the output of a ping command to 127.0.0.1, indicating successful connectivity.

Repite el procedimiento anterior para nginx.

The screenshot shows the Zabbix web interface, specifically the 'Configuration' section. The 'Web monitoring' tab is active, displaying a table of web scenarios. The table has columns for Name, Number of steps, Interval, Attempts, Authentication, HTTP proxy, Application, Status, and Info. A single scenario is listed: 'Prueba web Apache' with 1 step, 1m interval, 1 attempt, and a status of 'Enabled'. The interface also includes a 'Web scenario added' notification and a 'Create web scenario' button.

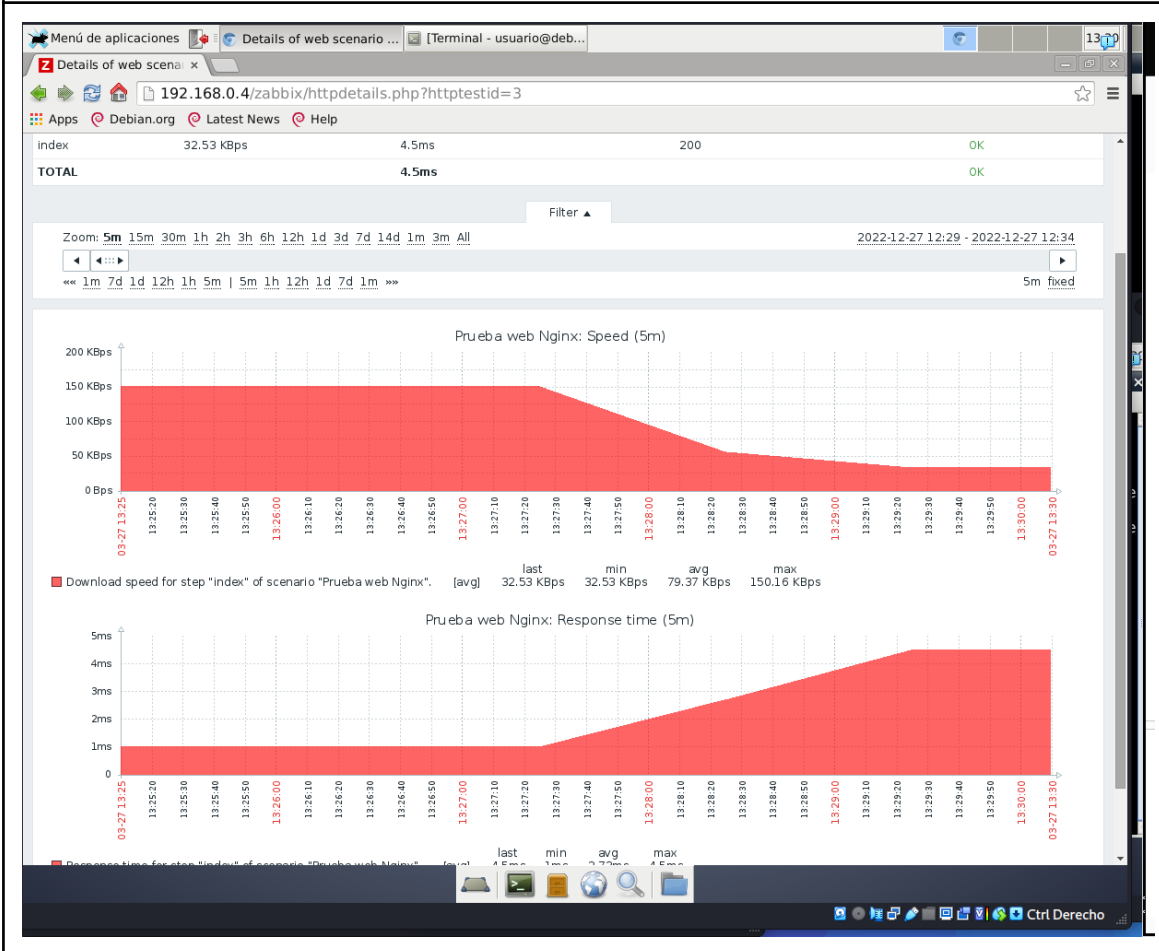
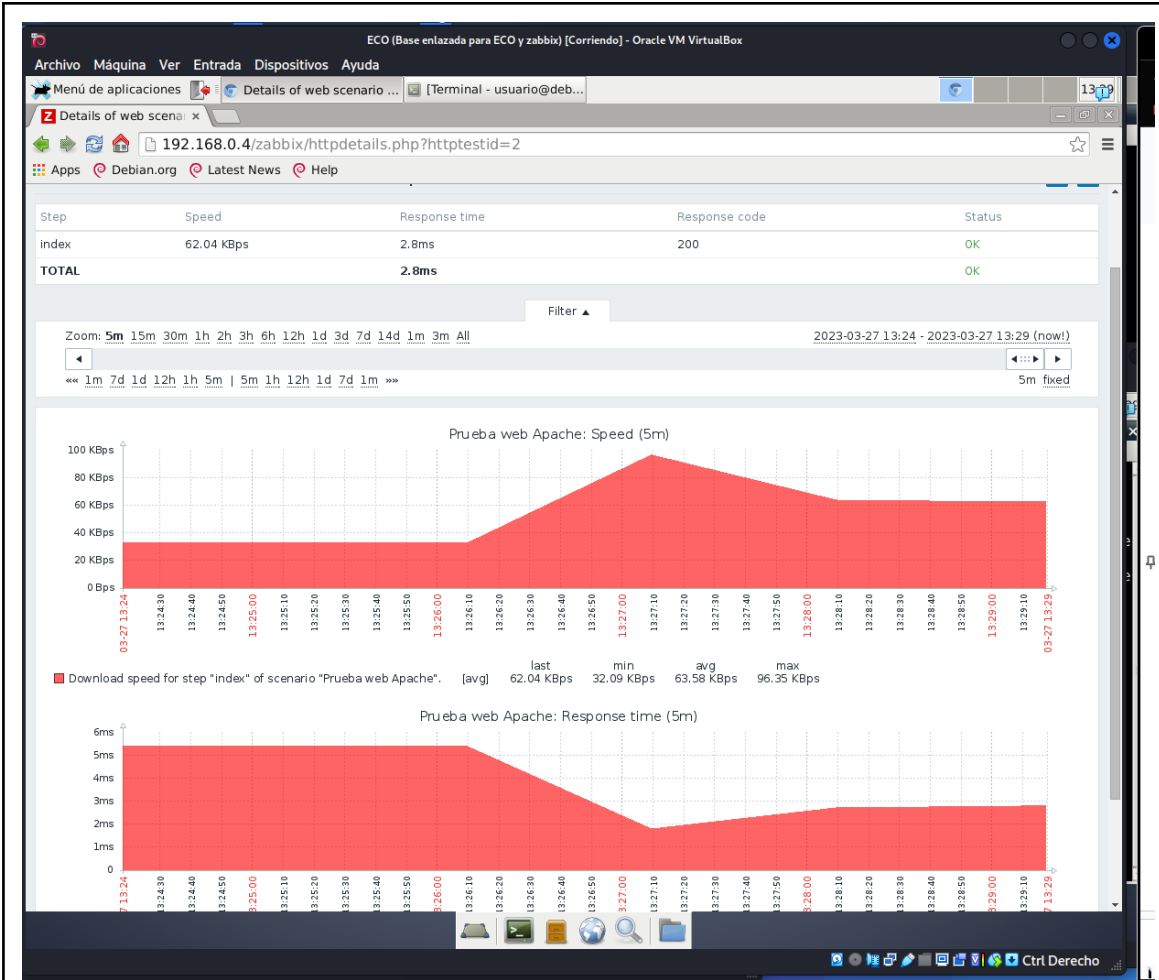


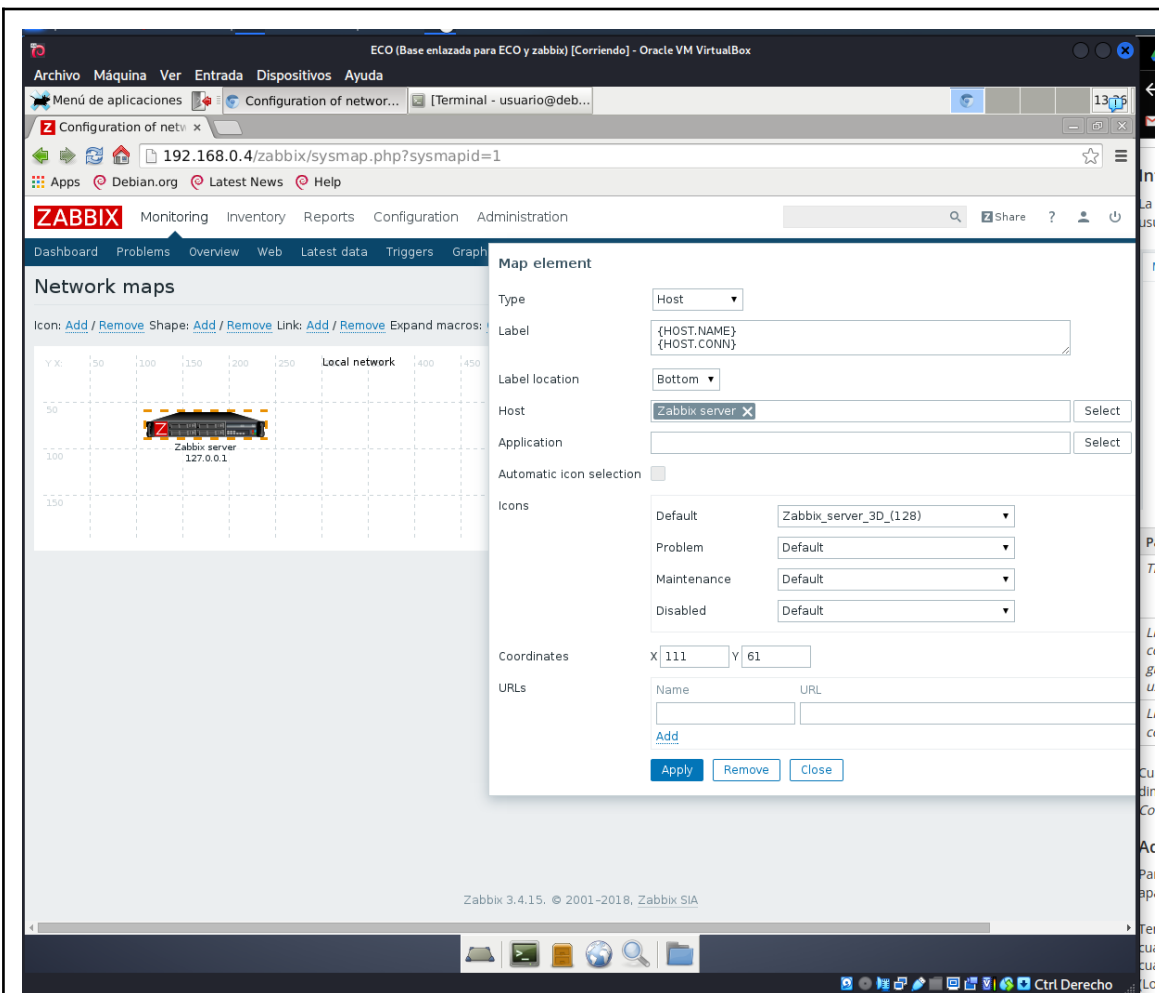
Observa los gráficos de monitorización de los sistemas y los resultados de las pruebas web en *Monitoring* → *Web*.

Copia varios gráficos o pantallas de Zabbix que te parezcan interesantes.

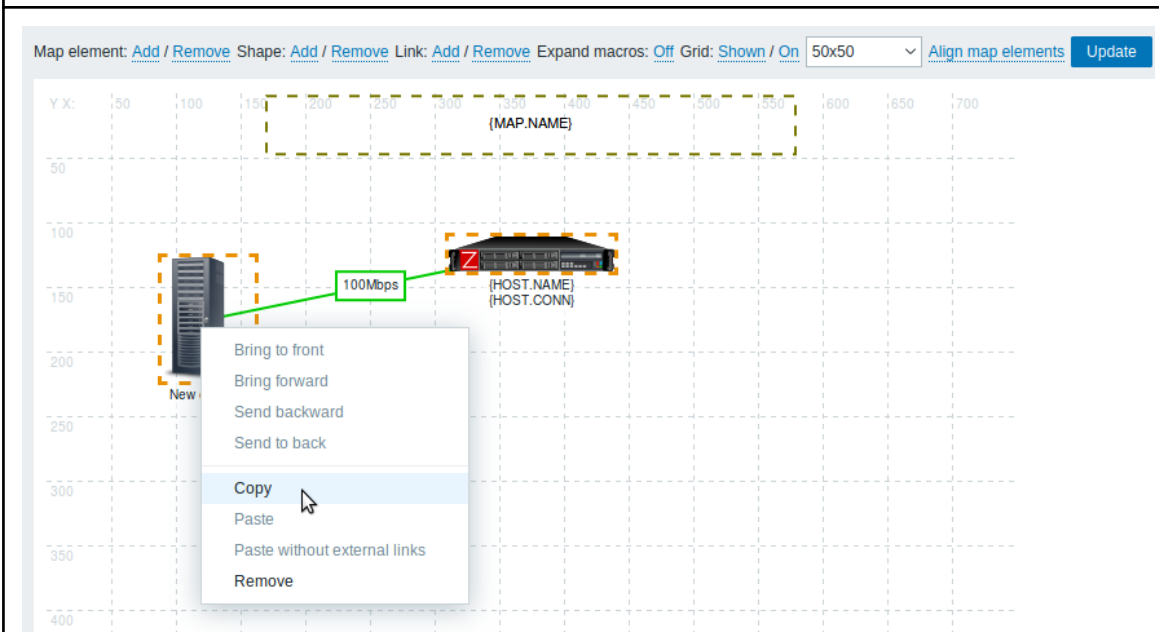
Para practicar más:

- Investiga otras funcionalidades de Zabbix, como la creación de mapas o pantallas.
- Prueba otros monitores de sistemas en red.





Se pueden crear algunos mapas como los de la siguiente imagen que aparece en la documentación.



<https://www.zabbix.com/documentation/6.0/es/manual/config/visualization/maps/map>

