

Ingeniería de Servidores (2014-2015)
GRADO EN INGENIERÍA INFORMÁTICA
UNIVERSIDAD DE GRANADA

Memoria Práctica 4

Manuel Castilla Gallardo

8 de diciembre de 2014

Índice

1. Cuestión 1: Instale la aplicación. ¿Qué comando permite listar los benchmarks disponibles?	4
2. Cuestión 2 : De los parámetros que le podemos pasar al comando ¿Qué significa -c 30 ? ¿y -n 1000?	6
3. Cuestión 3 : Ejecute ab contra a las tres máquinas virtuales (desde el SO anfitrión a las máquina virtuales de la red local) una a una (arrancadas por separado) y muestre las estadísticas. ¿Cuál es la que proporciona mejores resultados? Fíjese en el número de bytes transferidos, ¿es igual para cada máquina?	7
4. Cuestión 4 : Instale y siga el tutorial en http://jmeter.apache.org/usermanual/build-web-test-plan.html realizando capturas de pantalla y comentándolas. En vez de usar la web de jmeter, haga el experimento usando alguna de sus máquinas virtuales (Puede hacer una página sencilla, usar las páginas de phpmyadmin, instalar un CMS, etc.).	12
5. Cuestión 5 : Programe un benchmark usando el lenguaje que desee. El benchmark debe incluir:	19

Índice de figuras

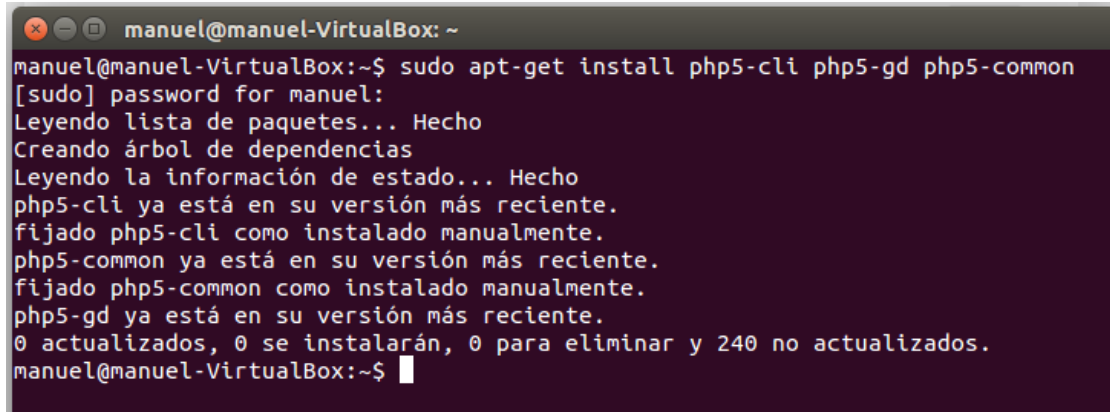
1.1. Descargamos el repositorio	4
1.2. Instalamos PHORONIX SUITE	5
1.3. Listado de benchmarks disponibles	5
2.1. Captura de ejecución del comando, test al monitor Nagios	6
3.1. Captura de las dirección IP Centos	7
3.2. Pagina por defecto Apache	8
3.3. Imposible acceder desde Ubuntu a la pagina por defecto Apache de Centos	8
3.4. Acceso por SSH a Centos Correcto	9
3.5. ab windows Server	10
3.6. Archivo resultados Windows Server	10
3.7. ab Ubuntu Server	11
3.8. Archivo resultados Ubuntu Server	11
4.1. Descarga del archivo apache-jmeter-2.12.tgz	12
4.2. Descomprimos el archivo apache-jmeter-2.12.tgz	13
4.3. Carpeta con el programa Jmeter	13
4.4. Programa Jmeter	14
4.5. URL apache maquina virtual 10.0.2.15	14
4.6. Nombre plan de prueba	15

4.7. Añadir grupo de usuarios	15
4.8. Propiedades de la prueba	16
4.9. Añadir configuración por defecto HTTP	16
4.10. Añadimos la dirección del servidor 10.0.2.15	17
4.11. Añadimos petición HTTP	17
4.12. Ruta index.html	18
4.13. Resultados en Árbol	18
4.14. Resultados de la prueba	19
5.1. CmpBasesDeDatos.sh	20
5.2. mysql.sql	20
5.3. mongo.gm	21
5.4. Resultado de ejecución CmpBasesDeDatos.sh	21

1. Cuestión 1: Instale la aplicación. ¿Qué comando permite listar los benchmarks disponibles?

Instalación¹

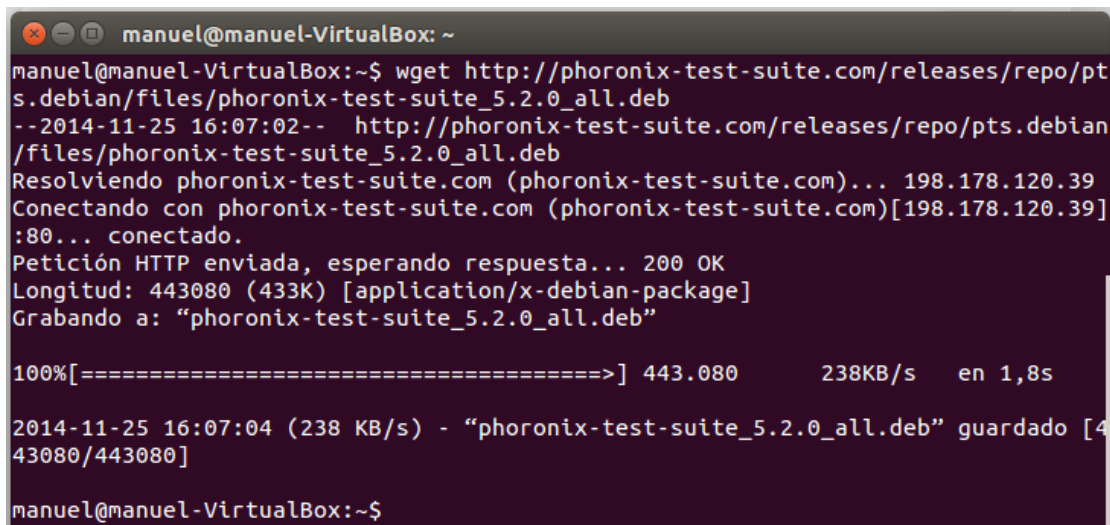
```
sudo apt-get install php5-cli php5-gd php5-common
```



```
manuel@manuel-VirtualBox: ~
manuel@manuel-VirtualBox:~$ sudo apt-get install php5-cli php5-gd php5-common
[sudo] password for manuel:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
php5-cli ya está en su versión más reciente.
fijado php5-cli como instalado manualmente.
php5-common ya está en su versión más reciente.
fijado php5-common como instalado manualmente.
php5-gd ya está en su versión más reciente.
0 actualizados, 0 se instalarán, 0 para eliminar y 240 no actualizados.
manuel@manuel-VirtualBox:~$
```

Instalación de dependencias

```
wget http://phoronix-test-suite.com/releases/repo/pts.debian/files/phoronix-
test-suite_5.2.0_all.deb
```



```
manuel@manuel-VirtualBox: ~
manuel@manuel-VirtualBox:~$ wget http://phoronix-test-suite.com/releases/repo/pt
s.debian/files/phoronix-test-suite_5.2.0_all.deb
--2014-11-25 16:07:02-- http://phoronix-test-suite.com/releases/repo/pts.debian
/files/phoronix-test-suite_5.2.0_all.deb
Resolviendo phoronix-test-suite.com (phoronix-test-suite.com)... 198.178.120.39
Conectando con phoronix-test-suite.com (phoronix-test-suite.com)[198.178.120.39]
:80... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 443080 (433K) [application/x-debian-package]
Grabando a: "phoronix-test-suite_5.2.0_all.deb"

100%[=====>] 443.080      238KB/s   en 1,8s

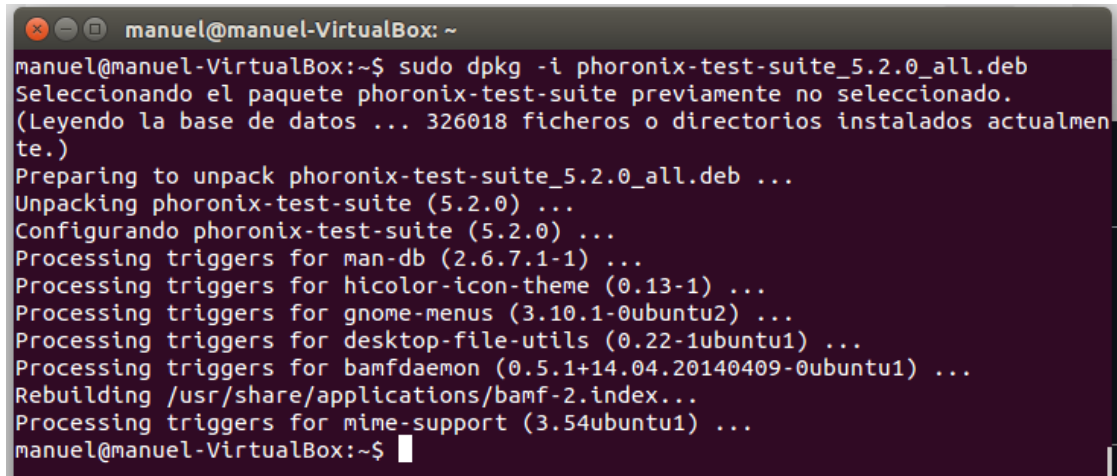
2014-11-25 16:07:04 (238 KB/s) - "phoronix-test-suite_5.2.0_all.deb" guardado [4
43080/443080]

manuel@manuel-VirtualBox:~$
```

Figura 1.1: Descargamos el repositorio

¹<http://www.sysads.co.uk/2014/06/install-phoronix-test-suite-5-2-ubuntu-14-04/>

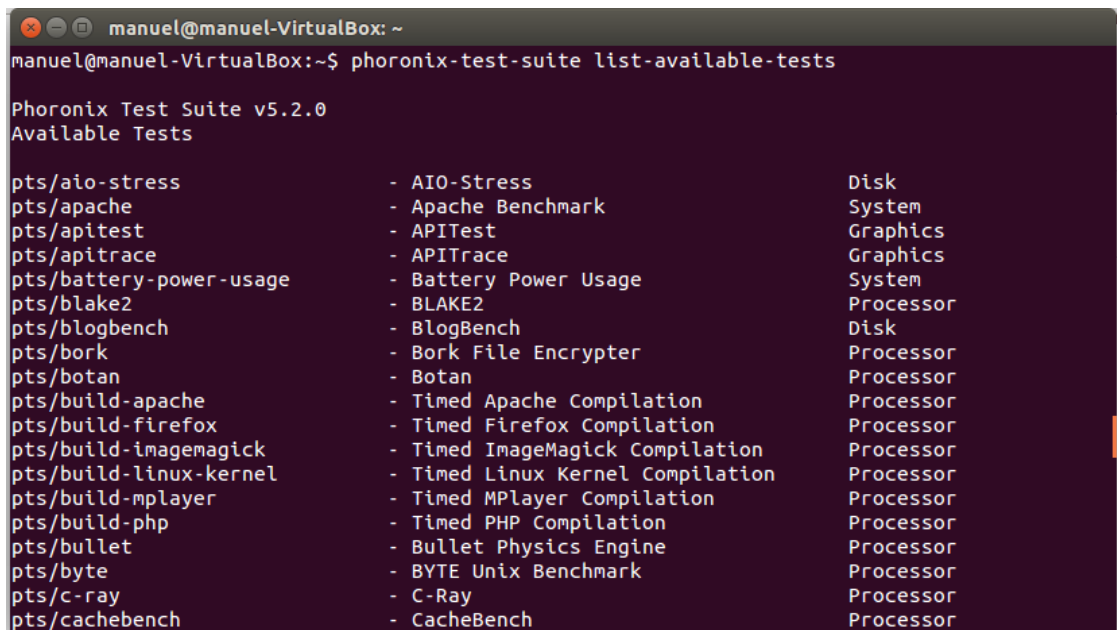
```
sudo dpkg -i phoronix-test-suite_5.2.0_all.deb
```



```
manuel@manuel-VirtualBox: ~  
manuel@manuel-VirtualBox:~$ sudo dpkg -i phoronix-test-suite_5.2.0_all.deb  
Seleccionando el paquete phoronix-test-suite previamente no seleccionado.  
(Leyendo la base de datos ... 326018 ficheros o directorios instalados actualmen  
te.)  
Preparing to unpack phoronix-test-suite_5.2.0_all.deb ...  
Unpacking phoronix-test-suite (5.2.0) ...  
Configurando phoronix-test-suite (5.2.0) ...  
Processing triggers for man-db (2.6.7.1-1) ...  
Processing triggers for hicolor-icon-theme (0.13-1) ...  
Processing triggers for gnome-menus (3.10.1-0ubuntu2) ...  
Processing triggers for desktop-file-utils (0.22-1ubuntu1) ...  
Processing triggers for bamfdaemon (0.5.1+14.04.20140409-0ubuntu1) ...  
Rebuilding /usr/share/applications/bamf-2.index...  
Processing triggers for mime-support (3.54ubuntu1) ...  
manuel@manuel-VirtualBox:~$
```

Figura 1.2: Instalamos PHORONIX SUITE

Para listar los benchmarks disponibles: **phoronix-test-suite list-available-tests**²



```
manuel@manuel-VirtualBox:~$ phoronix-test-suite list-available-tests  
  
Phoronix Test Suite v5.2.0  
Available Tests  
  
pts/aio-stress          - AIO-Stress          Disk  
pts/apache             - Apache Benchmark    System  
pts/apitest            - APITest             Graphics  
pts/apitrace           - APITrace            Graphics  
pts/battery-power-usage - Battery Power Usage System  
pts/blake2              - BLAKE2              Processor  
pts/blogbench          - BlogBench           Disk  
pts/bork               - Bork File Encrypter Processor  
pts/botan              - Botan              Processor  
pts/build-apache        - Timed Apache Compilation Processor  
pts/build-firefox       - Timed Firefox Compilation Processor  
pts/build-imagemagick   - Timed ImageMagick Compilation Processor  
pts/build-linux-kernel  - Timed Linux Kernel Compilation Processor  
pts/build-mplayer       - Timed MPlayer Compilation Processor  
pts/build-php           - Timed PHP Compilation Processor  
pts/bullet             - Bullet Physics Engine Processor  
pts/byte               - BYTE Unix Benchmark Processor  
pts/c-ray              - C-Ray               Processor  
pts/cachebench          - CacheBench          Processor
```

Figura 1.3: Listado de benchmarks disponibles

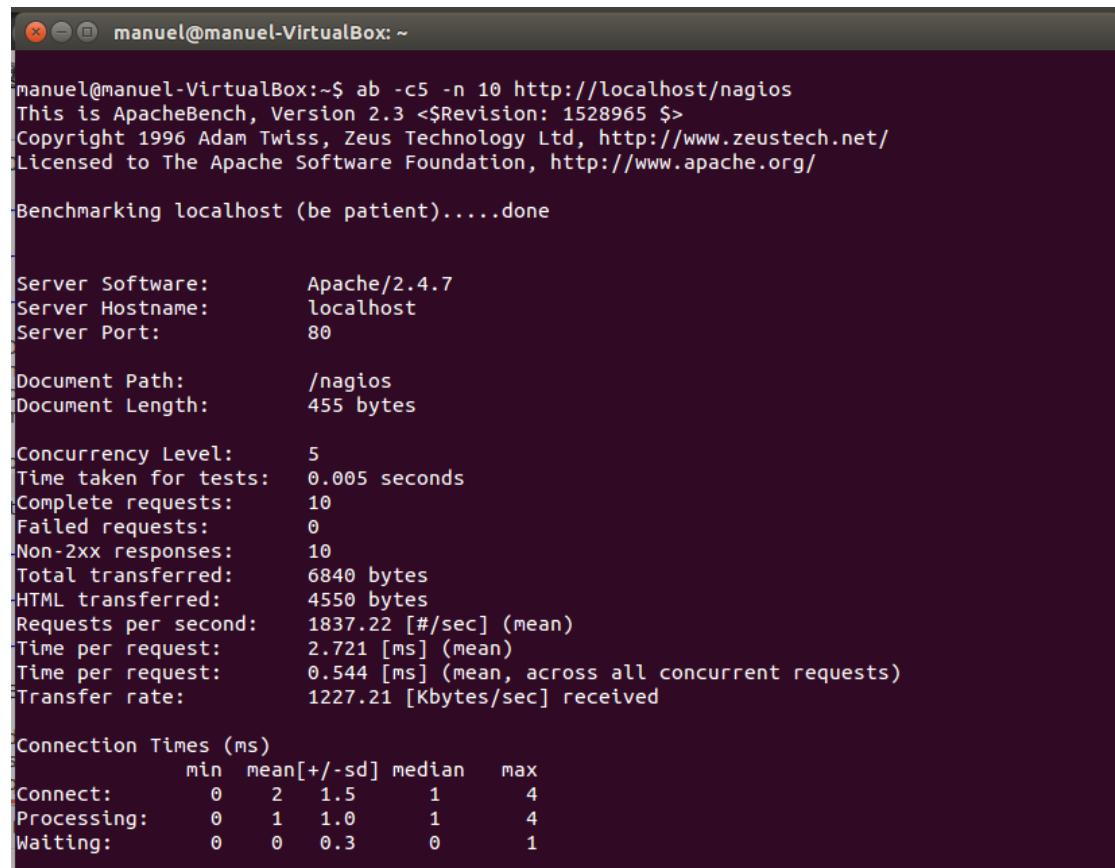
²<http://linux.die.net/man/1/phoronix-test-suite>

2. Cuestión 2 : De los parámetros que le podemos pasar al comando ¿Qué significa -c 30 ? ¿y -n 1000?

-c 30 se van hacer 30 peticiones concurrentemente en el test.³

-n 1000 se hacen 1000 peticiones por cada sesión.

ab -c5 -n 10 http://localhost/nagios



```
manuel@manuel-VirtualBox: ~  
manuel@manuel-VirtualBox:~$ ab -c5 -n 10 http://localhost/nagios  
This is ApacheBench, Version 2.3 <$Revision: 1528965 $>  
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/  
Licensed to The Apache Software Foundation, http://www.apache.org/  
  
Benchmarking localhost (be patient).....done  
  
Server Software:      Apache/2.4.7  
Server Hostname:      localhost  
Server Port:          80  
  
Document Path:        /nagios  
Document Length:      455 bytes  
  
Concurrency Level:    5  
Time taken for tests:  0.005 seconds  
Complete requests:    10  
Failed requests:      0  
Non-2xx responses:    10  
Total transferred:    6840 bytes  
HTML transferred:     4550 bytes  
Requests per second:  1837.22 [#/sec] (mean)  
Time per request:      2.721 [ms] (mean)  
Time per request:      0.544 [ms] (mean, across all concurrent requests)  
Transfer rate:         1227.21 [Kbytes/sec] received  
  
Connection Times (ms)  
            min  mean[+/-sd] median   max  
Connect:     0    2    1.5      1     4  
Processing:   0    1    1.0      1     4  
Waiting:     0    0    0.3      0     1
```

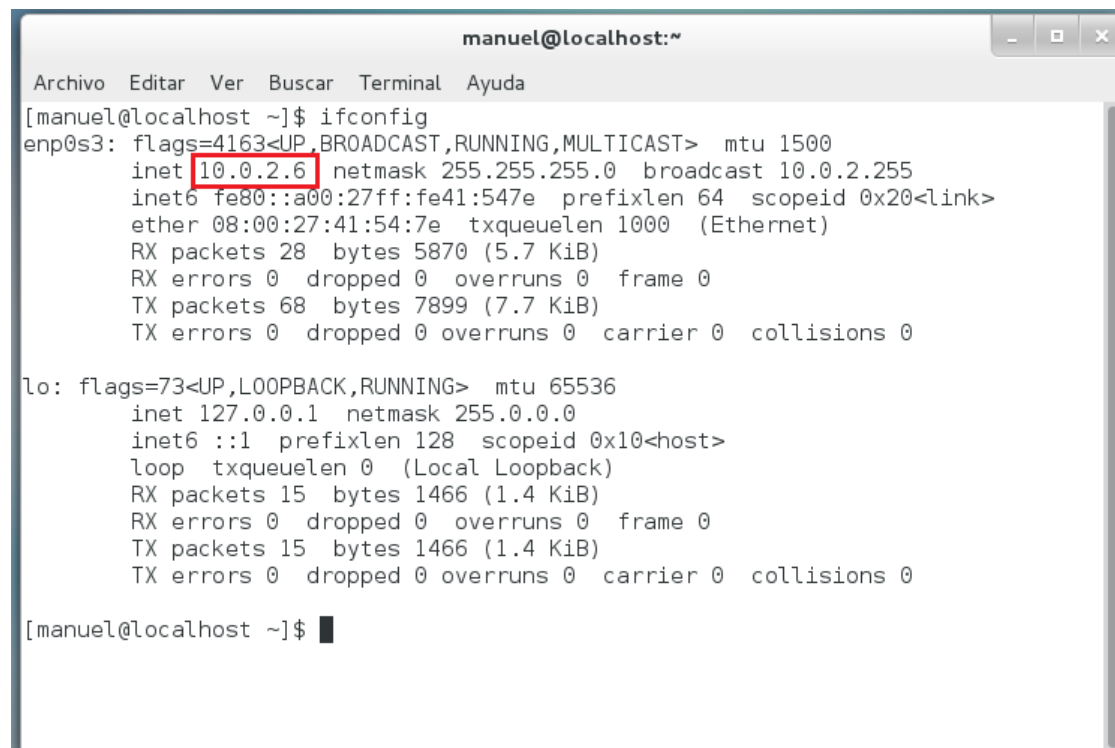
Figura 2.1: Captura de ejecución del comando, test al monitor Nagios

³<http://linux.die.net/man/1/phoronix-test-suite>

3. **Cuestión 3 : Ejecute ab contra a las tres máquinas virtuales (desde el SO anfitrión a las máquina virtuales de la red local) una a una (arrancadas por separado) y muestre las estadísticas. ¿Cuál es la que proporciona mejores resultados? Fíjese en el número de bytes transferidos, ¿es igual para cada máquina?**

Como tengo instalado las maquinas virtuales en Windows 7, la prueba del comando ab la he realizado entre una maquina virtual con Ubuntu y las otras por separado(Windows Server, Ubuntu Server). No he podido hacer la prueba ha **Centos**, no he conseguido acceder desde la red local, por algún motivo que desconozco no deja conexiones entrantes a la pagina por defecto de apache. Tengo ping entre las dos maquinas y también puedo conectarme por ssh.

Comprobaciones realizadas para probar en Centos: Vemos dirección ip de la maquina Centos.



```
manuel@localhost:~  
Archivo Editar Ver Buscar Terminal Ayuda  
[manuel@localhost ~]$ ifconfig  
enp0s3: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500  
    inet 10.0.2.6 netmask 255.255.255.0 broadcast 10.0.2.255  
    inet6 fe80::a00:27ff:fe41:547e prefixlen 64 scopeid 0x20<link>  
    ether 08:00:27:41:54:7e txqueuelen 1000 (Ethernet)  
    RX packets 28 bytes 5870 (5.7 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 68 bytes 7899 (7.7 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536  
    inet 127.0.0.1 netmask 255.0.0.0  
    inet6 ::1 prefixlen 128 scopeid 0x10<host>  
    loop txqueuelen 0 (Local Loopback)  
    RX packets 15 bytes 1466 (1.4 KiB)  
    RX errors 0 dropped 0 overruns 0 frame 0  
    TX packets 15 bytes 1466 (1.4 KiB)  
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0  
  
[manuel@localhost ~]$
```

Figura 3.1: Captura de las dirección IP Centos

En Centos compruebo que el servicio Apache está habilitado y puedo acceder desde el navegador, con la ip 10.0.2.6(localhost).

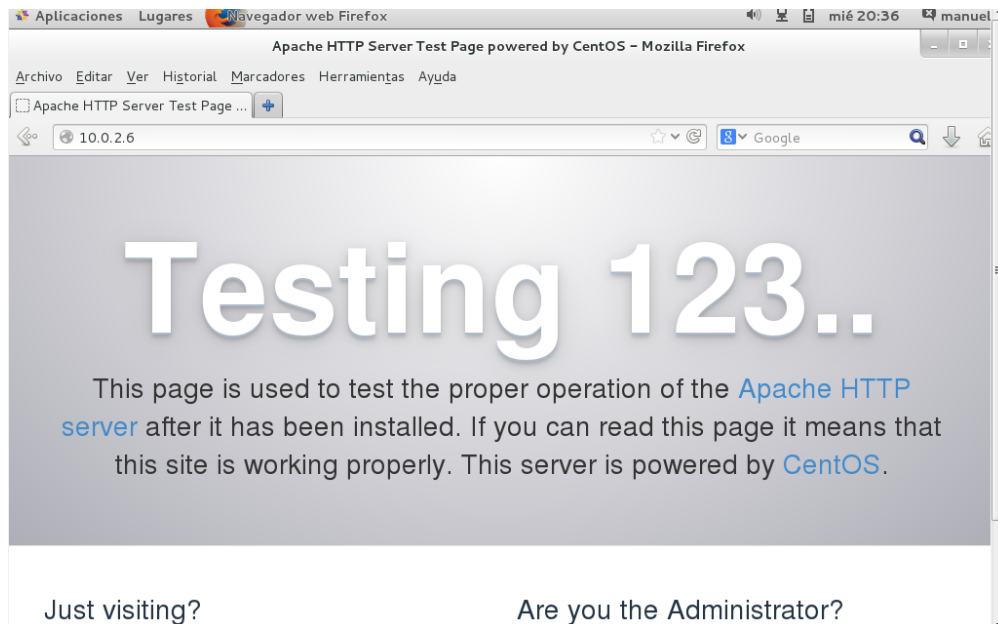


Figura 3.2: Pagina por defecto Apache

Intento acceder desde Ubuntu pero no se puede.

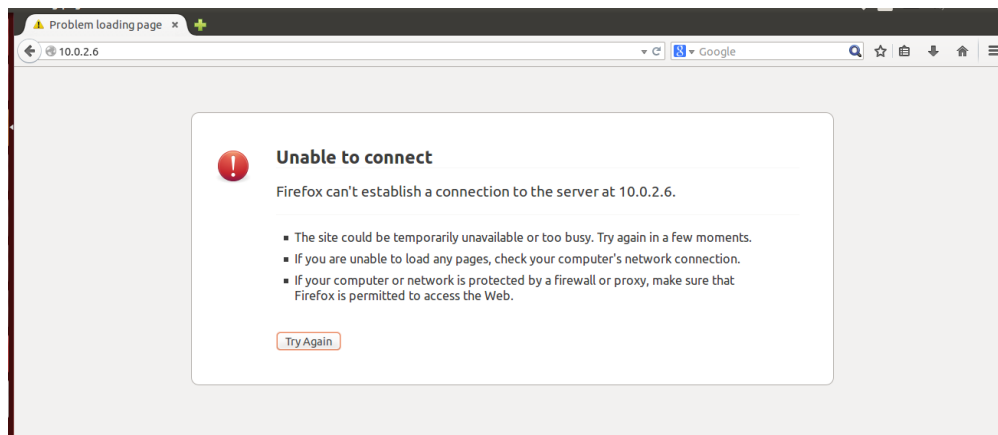


Figura 3.3: Imposible acceder desde Ubuntu a la pagina por defecto Apache de Centos

Para comprobar que puedo acceder al Centos, me conecto por SSH y funciona perfectamente.

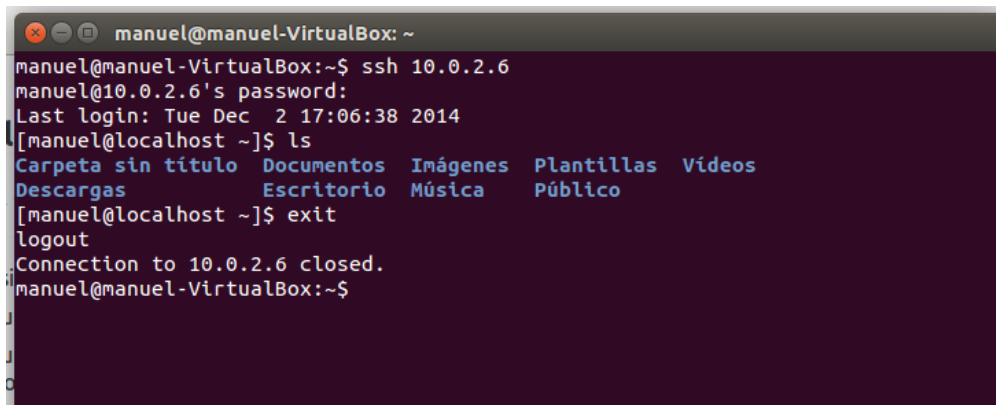
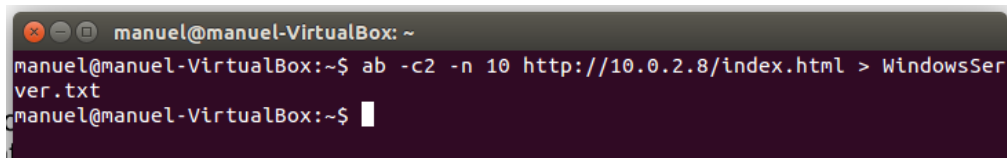
A terminal window titled 'manuel@manuel-VirtualBox: ~' showing a successful SSH connection. The user runs 'ssh 10.0.2.6', enters a password, and is greeted with 'Last login: Tue Dec 2 17:06:38 2014'. The prompt changes to '[manuel@localhost ~]\$'. The user runs 'ls', displaying a directory listing with files like 'Carpeta sin titulo', 'Documentos', 'Imágenes', 'Plantillas', 'Videos', 'Descargas', 'Escritorio', 'Música', and 'Público'. The user then runs 'exit', resulting in a 'logout' message and 'Connection to 10.0.2.6 closed.' The prompt returns to 'manuel@manuel-VirtualBox:~\$'.

Figura 3.4: Acceso por SSH a Centos Correcto

Prueba ab con Windows Server redirigiendo la salida a un archivo



```
manuel@manuel-VirtualBox: ~  
manuel@manuel-VirtualBox:~$ ab -c2 -n 10 http://10.0.2.8/index.html > WindowsServer.txt  
manuel@manuel-VirtualBox:~$
```

Figura 3.5: ab windows Server

Archivo Windows_Server.txt

Archivo: /home/manuel/WindowsServer.txt Página 1 de 1

This is ApacheBench, Version 2.3 <\$Revision: 1528965 \$>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking 10.0.2.8 (be patient).....done

Server Software: Microsoft-IIS/7.5
Server Hostname: 10.0.2.8
Server Port: 80

Document Path: /index.html
Document Length: 1282 bytes

Concurrency Level: 2
Time taken for tests: 0.011 seconds
Complete requests: 10
Failed requests: 0
Non-2xx responses: 10
Total transferred: 14380 bytes
HTML transferred: 12820 bytes
Requests per second: 913.33 [#/sec] (mean)
Time per request: 2.190 [ms] (mean)
Time per request: 1.095 [ms] (mean, across all concurrent requests)
Transfer rate: 1282.58 [Kbytes/sec] received

Connection Times (ms)

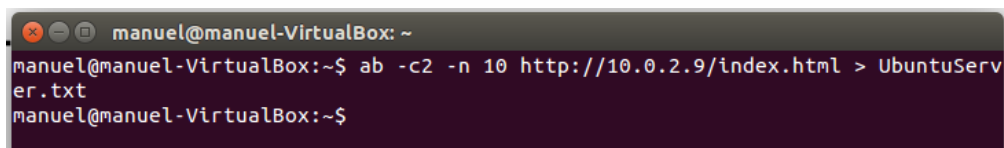
	min	mean[+/-sd]	median	max
Connect:	0	1 0.5	1	2
Processing:	1	1 0.8	1	3
Waiting:	0	1 0.8	1	3
Total:	1	2 0.9	2	3

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

50%	2
66%	2
75%	3
80%	3
90%	3
95%	3
98%	3
99%	3
100%	3 (longest request)

Figura 3.6: Archivo resultados Windows Server

Prueba ab Ubuntu Server redirigiendo la salida a un archivo



```
manuel@manuel-VirtualBox: ~$ ab -c2 -n 10 http://10.0.2.9/index.html > UbuntuServer.txt
manuel@manuel-VirtualBox: ~$
```

Figura 3.7: ab Ubuntu Server

Archivo UbuntuServer.txt

Archivo: /home/manuel/UbuntuServer.txt Página 1 de 1

This is ApacheBench, Version 2.3 <\$Revision: 1528965 \$>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking 10.0.2.9 (be patient).....done

Server Software: Apache/2.4.7
Server Hostname: 10.0.2.9
Server Port: 80

Document Path: /index.html
Document Length: 11510 bytes

Concurrency Level: 2
Time taken for tests: 0.083 seconds
Complete requests: 10
Failed requests: 0
Total transferred: 117830 bytes
HTML transferred: 115100 bytes
Requests per second: 120.53 [#/sec] (mean)
Time per request: 16.594 [ms] (mean)
Time per request: 8.297 [ms] (mean, across all concurrent requests)
Transfer rate: 1386.88 [Kbytes/sec] received

Connection Times (ms)

	min	mean[+/-sd]	median	max
Connect:	0	0 0.2	0	1
Processing:	0	16 30.7	1	74
Waiting:	0	11 20.5	1	50
Total:	1	16 30.9	1	75

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

50%	1
66%	4
75%	5
80%	75
90%	75
95%	75
98%	75
99%	75
100%	75 (longest request)

Figura 3.8: Archivo resultados Ubuntu Server

La que proporciona mejores resultados es la de Ubuntu Server con un ratio de transferencia de 1386.88 Kbytes/sec, mientras que Windows Server tiene un ratio de transferencia de 1282.58 Kbytes/sec.

El número total de Bytes transferidos es distinto, en Windows Server hay un total de 14380 bytes y en Ubuntu Server hay un total de 117830 bytes.

4. Cuestión 4 : Instale y siga el tutorial en <http://jmeter.apache.org/usermanual/build-web-test-plan.html> realizando capturas de pantalla y comentándolas. En vez de usar la web de jmeter, haga el experimento usando alguna de sus máquinas virtuales (Puede hacer una página sencilla, usar las páginas de phpmyadmin, instalar un CMS, etc.).

Nos descargamos de la pag web Jmeter y seguimos el tutorial ⁴ ⁵

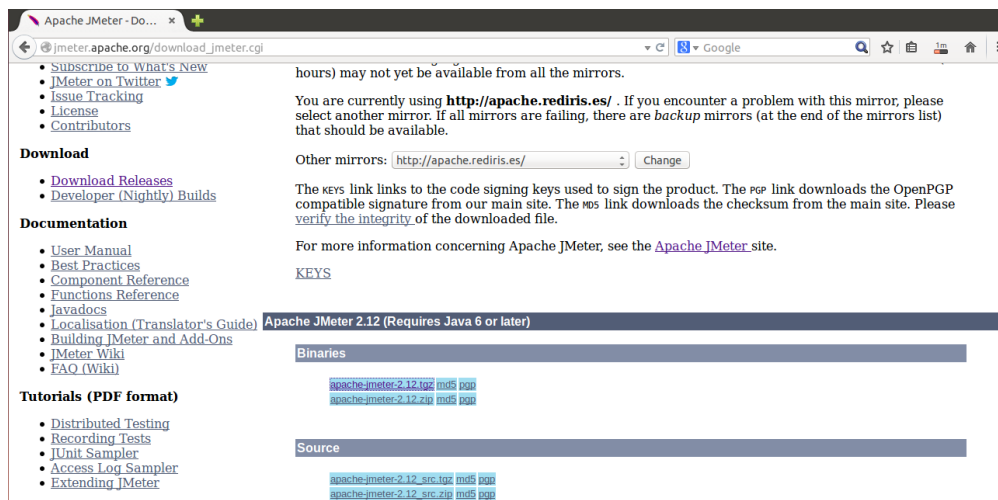


Figura 4.1: Descarga del archivo apache-jmeter-2.12.tgz

⁴http://jmeter.apache.org/download_jmeter.cgi

⁵<http://jmeter.apache.org/usermanual/build-web-test-plan.html>

Lo descomprimos en la ruta que deseemos

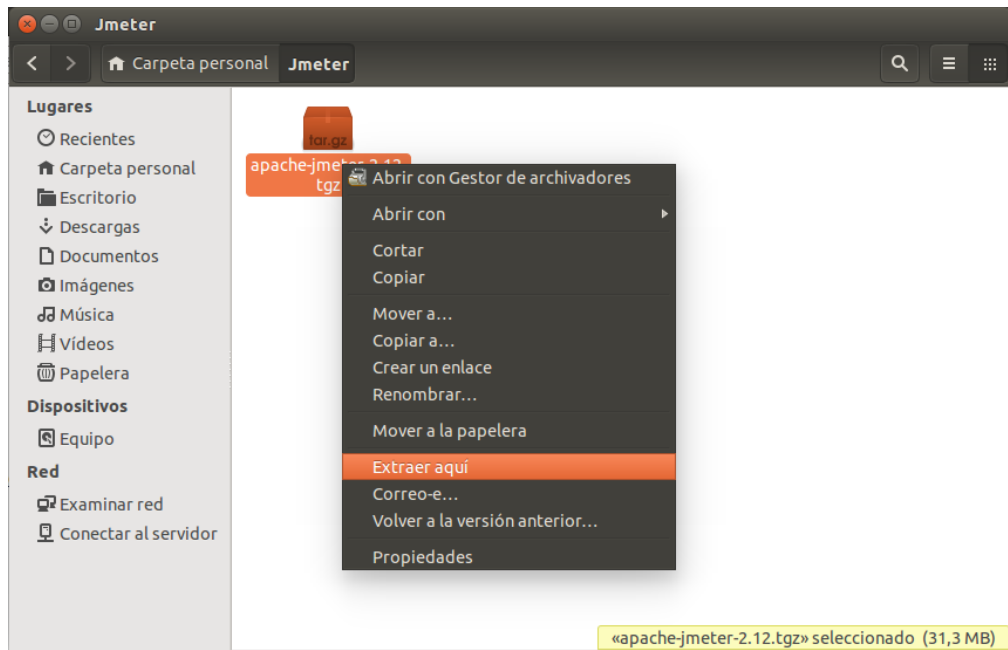


Figura 4.2: Descomprimos el archivo apache-jmeter-2.12.tgz

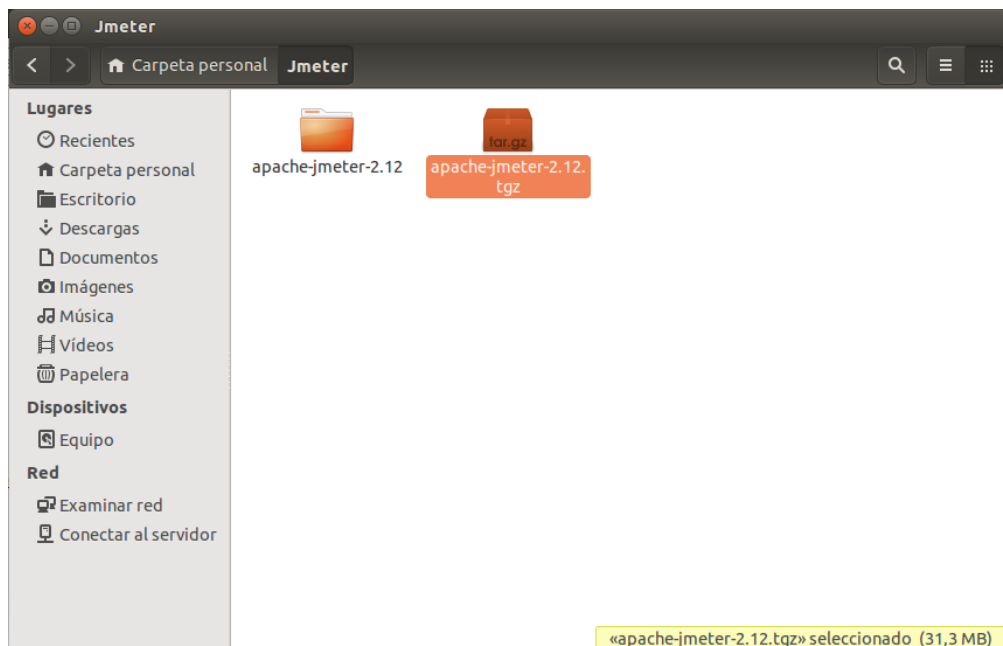


Figura 4.3: Carpeta con el programa Jmeter

Ejecutamos el programa pinchando en el archivo ApacheJMeter.jar, necesitamos tener instalado java para su ejecución.

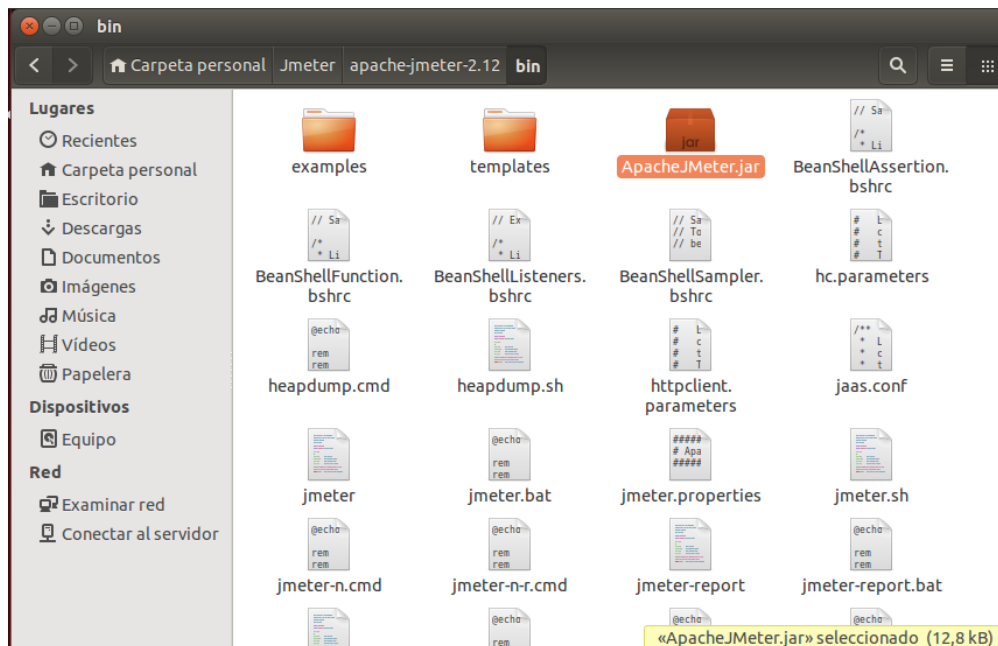


Figura 4.4: Programa Jmeter

URL a la que vamos hacer el test, es la web por defecto de apache de una de las maquinas virtuales.

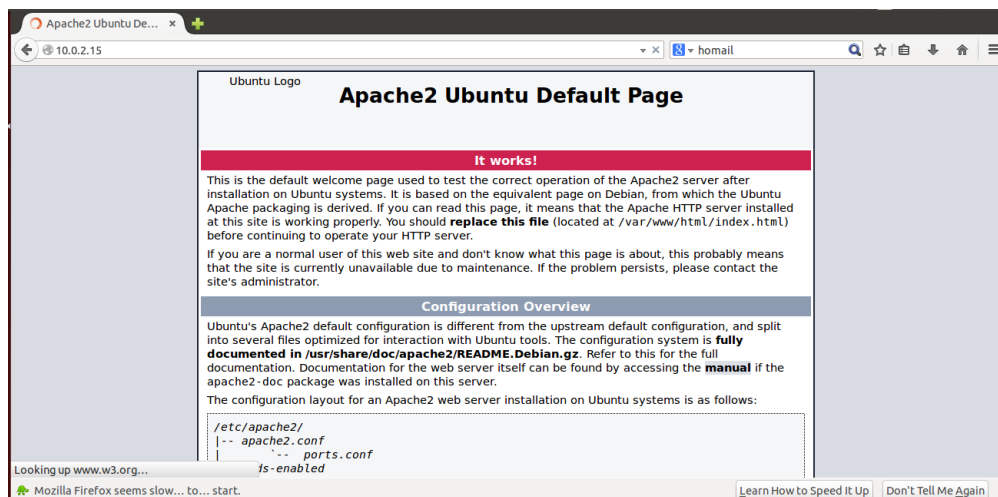


Figura 4.5: URL apache maquina virtual 10.0.2.15

Una vez abierto el programa ponemos el nombre al nuevo plan de pruebas

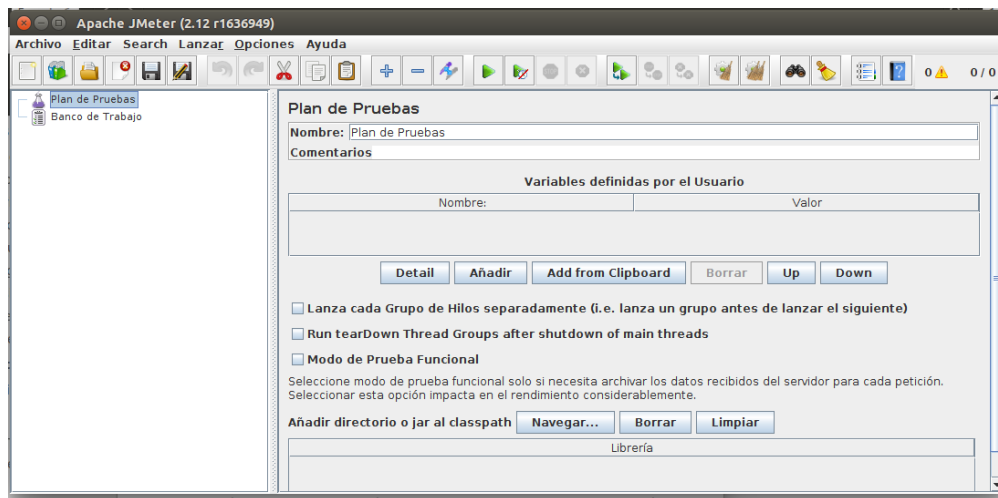


Figura 4.6: Nombre plan de prueba

Añadimos los usuarios que van a realizar las peticiones

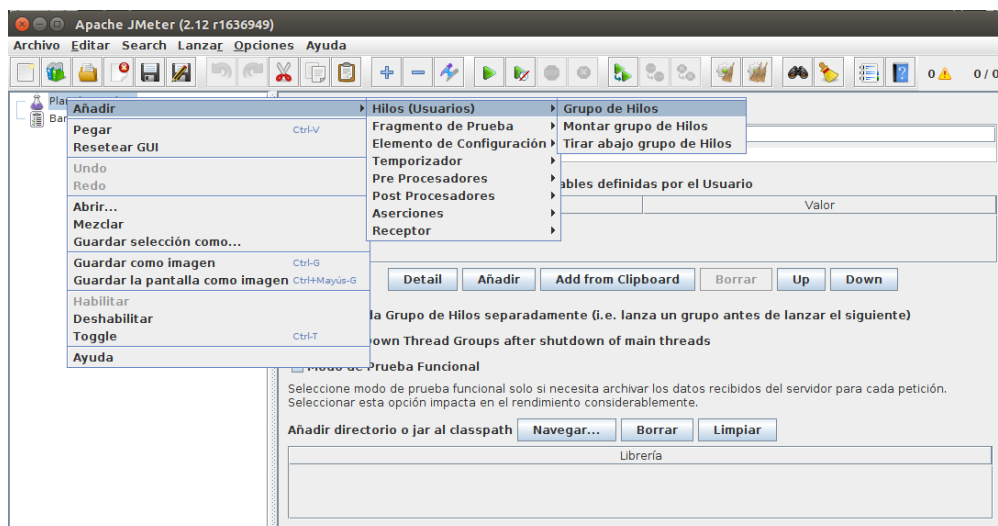


Figura 4.7: Añadir grupo de usuarios

Numero de peticiones = 5, tiempo de retraso entre peticiones = 1 segundo, numero de veces que se repite la prueba = 2.

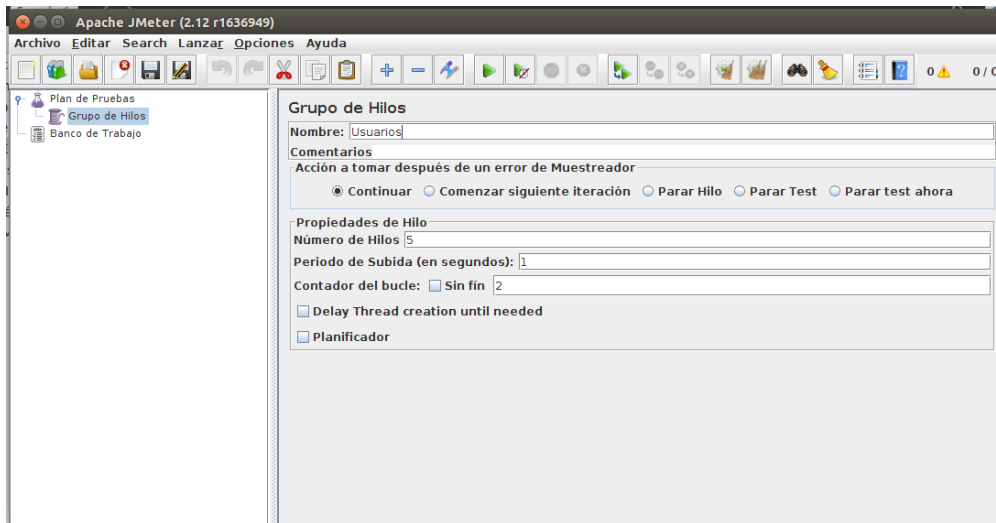


Figura 4.8: Propiedades de la prueba

Añadimos la configuración por defecto para las peticiones HTTP

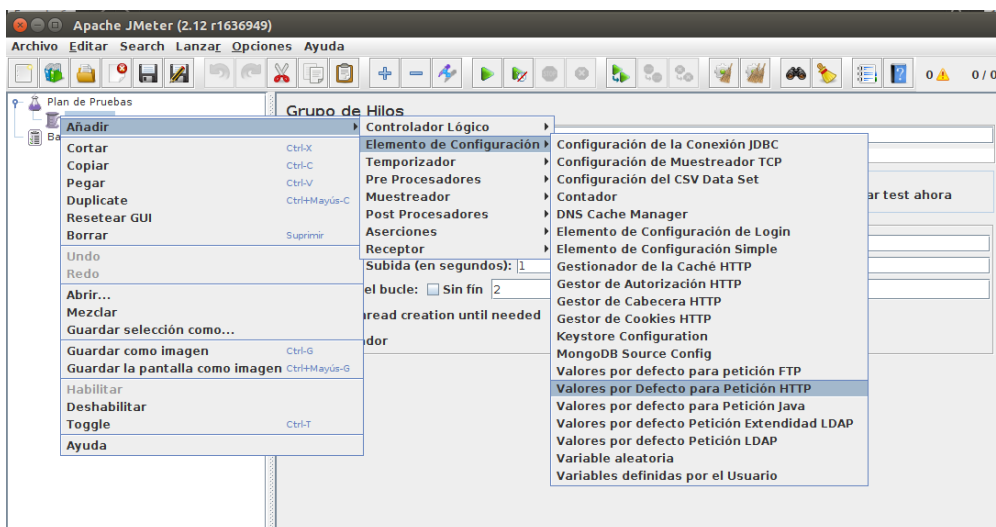


Figura 4.9: Añadir configuración por defecto HTTP

Se añade la dirección IP del servidor en este caso 10.0.2.15

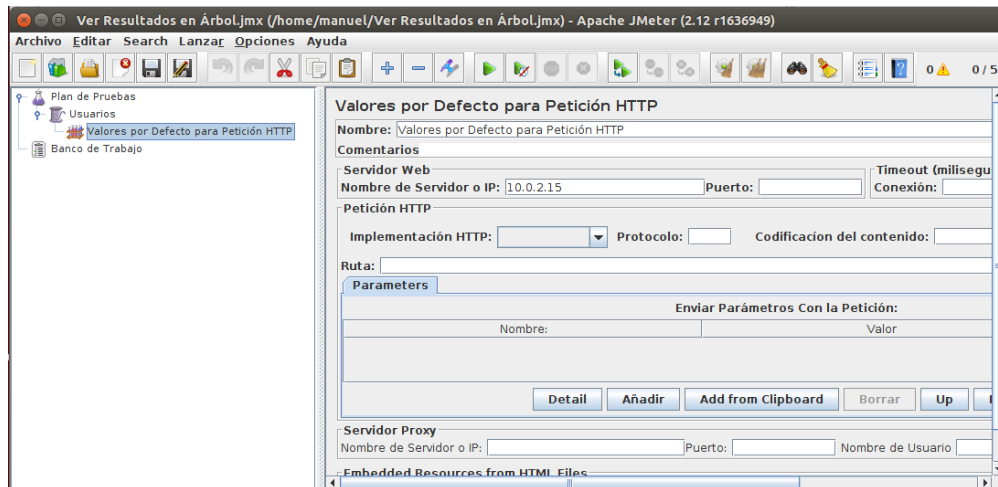


Figura 4.10: Añadimos la dirección del servidor 10.0.2.15

Añadimos la petición HTTP a la que vamos a realizar la prueba

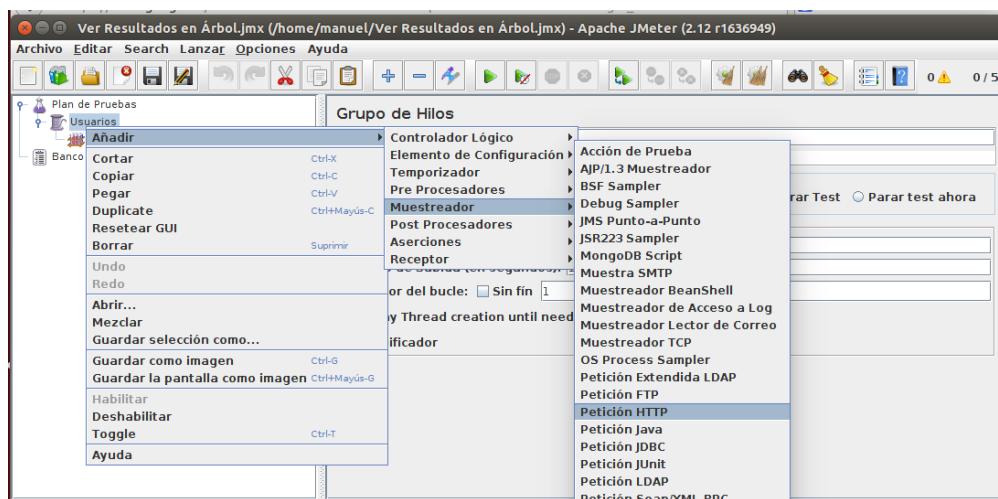


Figura 4.11: Añadimos petición HTTP

Ponemos la ruta de la petición **index.html**

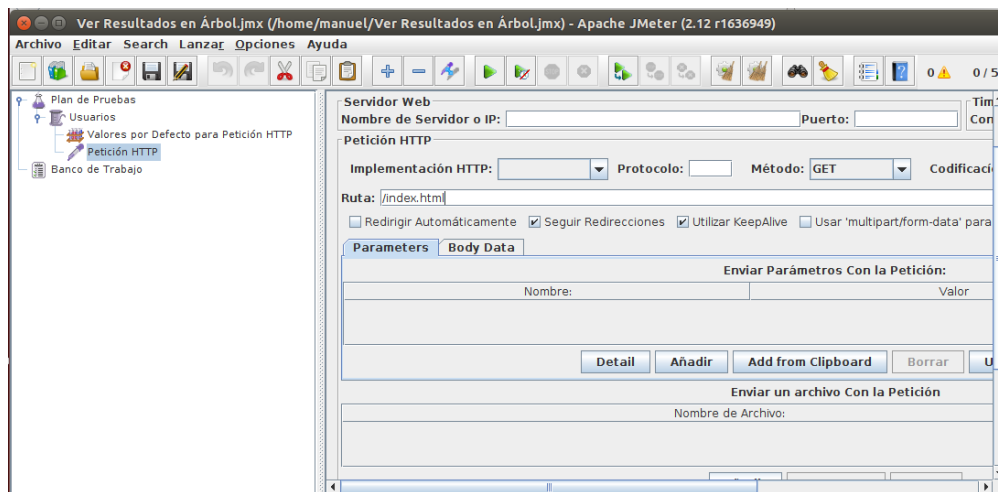


Figura 4.12: Ruta index.html

Para ver los resultados de la prueba añadimos un listener en el plan de pruebas que estamos realizando, he usado resultados en árbol porque el gráfico no se veía bien, tampoco mostraba el resultado correcto de la prueba.

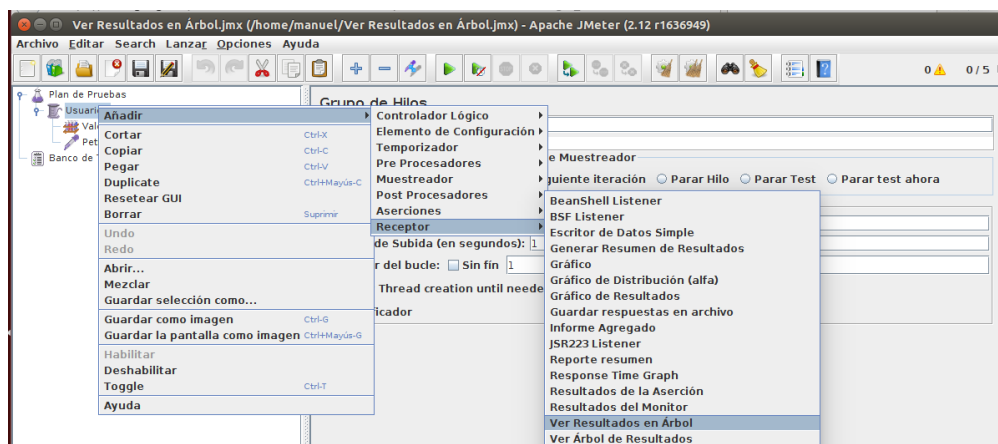
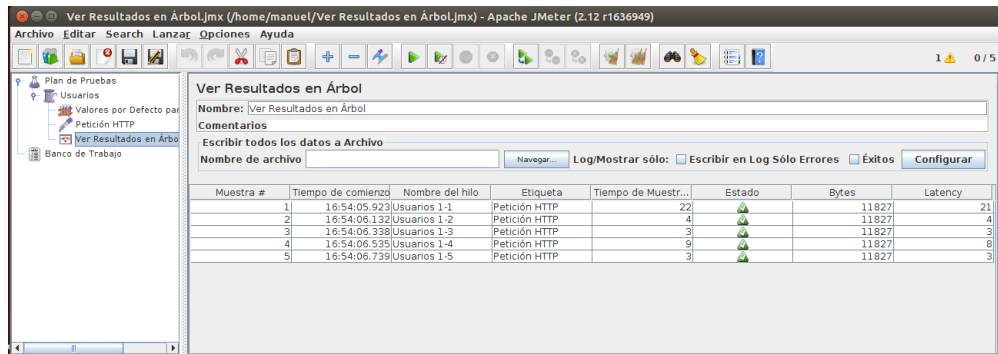


Figura 4.13: Resultados en Árbol

Damos al botón **Arrancar**(el botón play de color verde)para realizar la prueba obteniendo los resultados, en ella aparecen los tiempos de cada muestra en ms, bytes,etc.



The screenshot shows the Apache JMeter interface with the 'Ver Resultados en Árbol' window open. The window displays a table of test results for a 'Petición HTTP' test. The table has columns for 'Muestra #', 'Tiempo de comienzo', 'Nombre del hilo', 'Etiqueta', 'Tiempo de Muestr...', 'Estado', 'Bytes', and 'Latency'. The data shows five samples, all with a status of 'Éxito' (Success) and a latency of 21ms.

Muestra #	Tiempo de comienzo	Nombre del hilo	Etiqueta	Tiempo de Muestr...	Estado	Bytes	Latency
1	16:54:05.923	Usuarios 1-1	Petición HTTP	22	Éxito	11827	21
2	16:54:06.132	Usuarios 1-2	Petición HTTP	4	Éxito	11827	4
3	16:54:06.338	Usuarios 1-3	Petición HTTP	3	Éxito	11827	3
4	16:54:06.535	Usuarios 1-4	Petición HTTP	9	Éxito	11827	9
5	16:54:06.739	Usuarios 1-5	Petición HTTP	3	Éxito	11827	3

Figura 4.14: Resultados de la prueba

5. Cuestión 5 : Programe un benchmark usando el lenguaje que desee. El benchmark debe incluir:

- 1) Objetivo del benchmark
- 2) Métricas (unidades, variables, puntuaciones, etc.)
- 3) Instrucciones para su uso
- 4) Ejemplo de uso analizando los resultados

El objetivo de este benchmark es comparar el tiempo de la inserción de 5000 elementos en una tabla mysql y una tabla en mongo. ⁶ ⁷

El benchmark nos devuelve el tiempo transcurrido en el formato minutos:segundos.

Para su uso ha deben estar instalados MySql y Mongo, ejecutando el script **CmpBasesDeDatos.sh** obtendremos los tiempos.

Este script tiene dependencia de dos archivos, **mysql.sql** y **mongo.mg**, han de estar en la misma ruta que **CmpBasesDeDatos.sh**

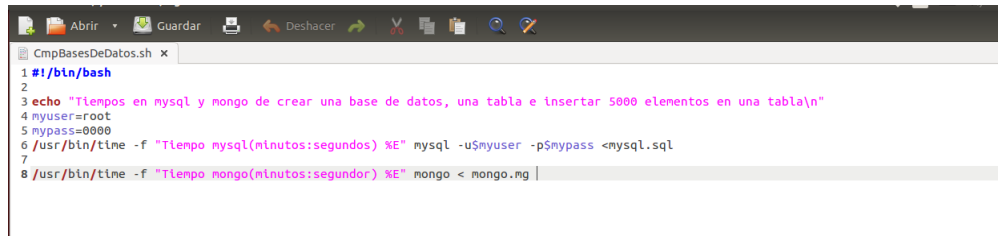
⁶<https://www.digitalocean.com/community/tutorials/a-basic-mysql-tutorial>

⁷<http://docs.mongodb.org/manual/reference/sql-comparison/>

Contenido script CmpBasesDeDatos.sh⁸

Dentro del script se ejecuta el comando mysql al cual se le redirigen todas las sentencias a ejecutar con el archivo mysql.sql, igualmente se ejecuta el comando mongo al que se redirigirle el archivo mongo.mg con las sentencias.

Con el comando /usr/bin/time -f obtenemos los tiempos en la ejecución de cada una de las bases de datos, en formato minutos:segundos.

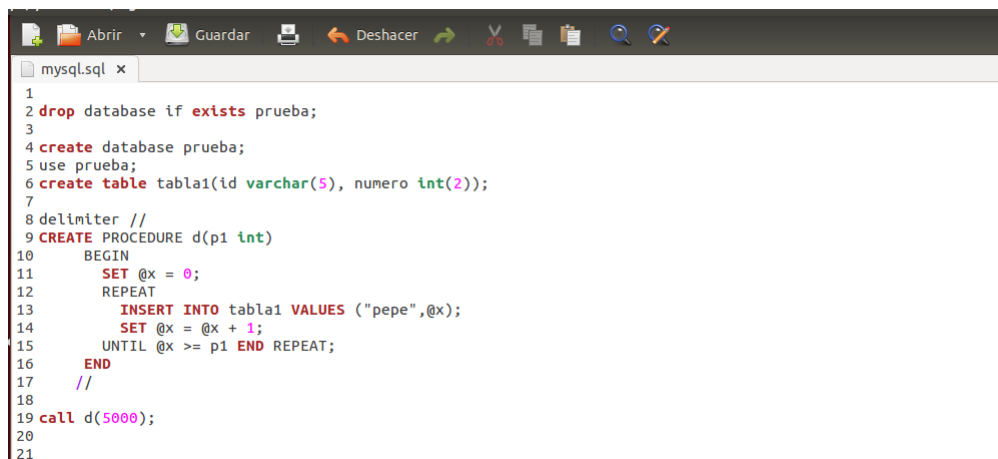
A screenshot of a text editor window titled 'CmpBasesDeDatos.sh'. The script content is as follows:

```
1 #!/bin/bash
2
3 echo "Tiempos en mysql y mongo de crear una base de datos, una tabla e insertar 5000 elementos en una tabla"
4 myuser=root
5 mypass=0000
6 /usr/bin/time -f "Tiempo mysql(minutos:segundos) %E" mysql -u$myuser -p$mypass <mysql.sql
7
8 /usr/bin/time -f "Tiempo mongo(minutos:segundos) %E" mongo < mongo.mg
```

Figura 5.1: CmpBasesDeDatos.sh

Contenido del archivo mysql.sql⁹

En este archivo se encuentran todas las sentencias que se ejecutan en mysql para la inserción de los 5000 elementos.

A screenshot of a text editor window titled 'mysql.sql'. The script content is as follows:

```
1
2 drop database if exists prueba;
3
4 create database prueba;
5 use prueba;
6 create table tabla1(id varchar(5), numero int(2));
7
8 delimiter //
9 CREATE PROCEDURE d(p1 int)
10 BEGIN
11     SET @x = 0;
12     REPEAT
13         INSERT INTO tabla1 VALUES ("pepe",@x);
14         SET @x = @x + 1;
15     UNTIL @x >= p1 END REPEAT;
16 END
17 //
18
19 call d(5000);
20
21
```

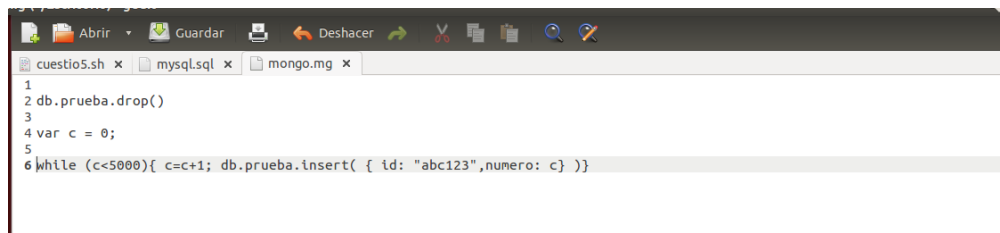
Figura 5.2: mysql.sql

⁸<http://plagatux.es/2010/02/bash-script-midiendo-el-tiempo-de-ejecucion-de-programas/>

⁹<http://dev.mysql.com/doc/refman/5.0/en/repeat.html>

Contenido del archivo mongo.mg¹⁰

En este archivo se encuentran las sentencias que se ejecutarán en mongo para la inserción de los 5000 elementos,

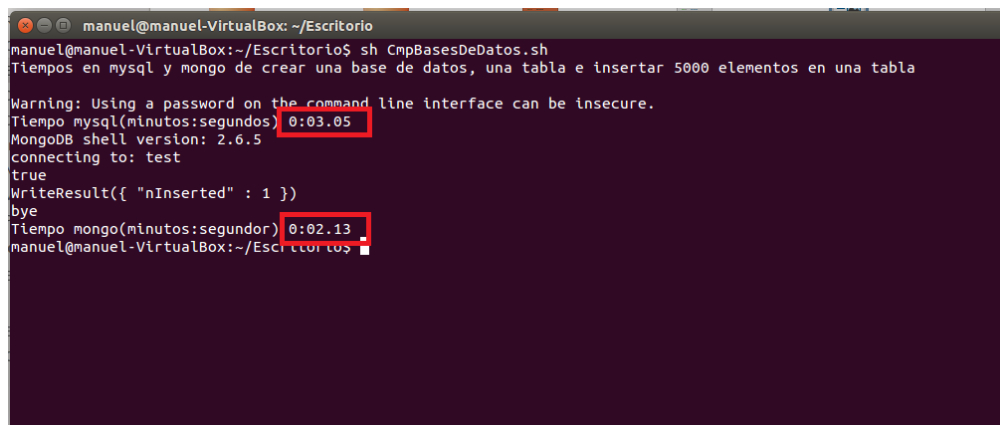


```
1
2 db.prueba.drop()
3
4 var c = 0;
5
6 while (c<5000){ c=c+1; db.prueba.insert( { id: "abc123",numero: c} )}
```

Figura 5.3: mongo.gm

Ejemplo con resultados

En la figura 5.4 vemos como los tiempos obtenidos en mongo son mejores que los de mysql, mongo ha tardado 0.92 segundos menos en insertar los 5000 elementos. La base de datos mongodb es mas eficiente en tiempo que la base de datos mysql.



```
manuel@manuel-VirtualBox: ~/Escritorio
manuel@manuel-VirtualBox:~/Escritorio$ sh CmpBasesDeDatos.sh
Tiempos en mysql y mongo de crear una base de datos, una tabla e insertar 5000 elementos en una tabla

Warning: Using a password on the command line interface can be insecure.
Tiempo mysql(minutos:segundos) 0:03.05
MongoDB shell version: 2.6.5
connecting to: test
true
WriteResult({ "nInserted" : 1 })
bye
Tiempo mongo(minutos:segundor) 0:02.13
manuel@manuel-VirtualBox:~/Escritorio$
```

Figura 5.4: Resultado de ejecución CmpBasesDeDatos.sh

¹⁰<http://docs.mongodb.org/manual/tutorial/iterate-a-cursor/>