

Practica 4 - Benchmarks

Phoronix

1. Instalamos phoronix con el comando:

```
Archivo Editar Ver Buscar Terminal Ayuda
juanka1995@juanka1995-Laptop ~ $ sudo apt-get install phoronix-test-suite
```

2. Listamos los test disponibles e instalamos 2 de ellos.

```
juanka1995@juanka1995-Laptop ~ $ phoronix-test-suite list-available-tests

Phoronix Test Suite v7.6.0
Available Tests

pts/aio-stress          - AIO-Stress          Disk
pts/apache              - Apache Benchmark    System
pts/apitest             - APITest             Graphics
pts/apitrace            - APITrace            Graphics
pts/arrayfire           - ArrayFire           Processor
pts/askap               - ASKAP tConvolveCuda Graphics
pts/asmfish             - asmFish             Processor
pts/battery-power-usage - Battery Power Usage System
pts/bioshock-infinite   - BioShock Infinite   Graphics
pts/blake2              - BLAKE2              Processor
```

3. Añadimos el **cachebench** (testea el rendimiento de la memoria cache del procesador y del ancho de banda) y lo lanzamos.

```
Phoronix Test Suite v7.6.0

To Install: pts/cachebench-1.1.0

Determining File Requirements .....
Searching Download Caches .....

1 Test To Install
  1 File To Download [0.08MB]
  1MB Of Disk Space Is Needed

pts/cachebench-1.1.0:
  Test Installation 1 of 1
  1 File Needed [0.08 MB]
  Downloading: llcbench-20170104.tar.gz
  Downloading .....
  Installation Size: 0.5 MB
  Installing Test @ 16:37:38

CacheBench:
pts/cachebench-1.1.0
Processor Test Configuration
  1: Read
  2: Write
  3: Read / Modify / Write
  4: Test All Options
  Test: █
```

4. Elegimos el test de **READ** y le decimos que queremos guardar los resultados del test.

```
CacheBench:
pts/cachebench-1.1.0
Processor Test Configuration
  1: Read
  2: Write
  3: Read / Modify / Write
  4: Test All Options
Test: 1

System Information

PROCESSOR:      Intel Core i5-4210U @ 2.70GHz
Core Count:    2
Thread Count:  4
Extensions:    SSE 4.2 + AVX2 + AVX + RDRAND + FSGSBASE
Cache Size:    3072 KB
Microcode:     0x20
Scaling Driver: intel_pstate powersave

GRAPHICS:       Intel Haswell Mobile 1536MB (1000MHz)
OpenGL:        4.5 Mesa 17.0.7
Display Driver: intel 2.99.917
Screen:        1366x768

MOTHERBOARD:   Acer ZOR0_BH
Memory:        8192MB
Chipset:        Intel Haswell-ULT DRAM
Network:       Realtek RTL8111/8168/8411 + Qualcomm Atheros Device 0042

DISK:          120GB KINGSTON SA400S3
File-System:   ext4
Mount Options: data=ordered errors=remount-ro relatime rw
Disk Scheduler: CFQ

OPERATING SYSTEM: LinuxMint 18.2
Kernel:        4.10.0-42-generic (x86_64)
Desktop:        Cinnamon 3.4.6
Compiler:      GCC 5.4.0 20160609 + Clang 3.8.0-2ubuntu4

Would you like to save these test results (Y/n):
```

```
Would you like to save these test results (Y/n)y
Enter a name to save these results under: readCacheBenchTest
Enter a unique name to describe this test run / configuration: Test de lectura de cachebench a fecha 14/12/2017
```

5. Esperamos a que realice el test.

```
CacheBench:
pts/cachebench-1.1.0 [Test: Read]
Test 1 of 1
Estimated Trial Run Count: 3
Estimated Time To Completion: 7 Minutes [16:50 CET]
Started Run 1 @ 16:43:55
Started Run 2 @ 16:46:01
Started Run 3 @ 16:48:07
```

```
juanka1995@juanka1995-Laptop ~
Archivo Editar Ver Buscar Terminal Ayuda
top - 16:45:52 up 56 min, 1 user, load average: 1.04, 0.67, 0.56
Tareas: 245 total, 2 ejecutar, 243 hibernar, 0 detener, 0 zombie
%Cpu(s): 26,5 usuario, 0,6 sist, 0,0 adecuado, 72,6 inact, 0,2 en espera, 0,
KiB Mem : 8097448 total, 3820292 free, 2188424 used, 2088732 buff/cache
KiB Swap: 7999484 total, 7999484 free, 0 used. 5354112 avail Mem

  PID  USUARIO  PR  NI  VIRT  RES  SHR  S  %CPU  %MEM  HORA+  ORDEN
18035  juanka1+  20   0  20756 17608 1156 R 100,0  0,2  1:55.58  cachebench
1683  juanka1+  20   0 2193524 321036 66648 S  4,3  4,0  3:20.63  cinnamon
1229  root      20   0  391100 80664 65824 S  1,3  1,0  1:26.98  Xorg
```

6. Mostramos los resultados obtenidos en nuestro navegador.


```
CacheBench:
pts/cachebench-1.1.0 [Test: Read]
Test 1 of 1
Estimated Trial Run Count: 3
Estimated Time To Completion: 7 Minutes [16:50 CET] Read
Started Run 1 @ 16:43:55
Started Run 2 @ 16:46:01
Started Run 3 @ 16:48:07

Test: Read:
2212.3406872857
2223.8947532857
2227.968774

Average: 2221.40 MB/s
Deviation: 0.36%

OpenBenchmarking.org Dynamic Comparison:
MB/s > Higher Is Better
Test de lectura de cachebench a fecha 14/12/2017 .. 2221.40 |====
pts/processor ..... 2295.98 |====
Result Perspective: https://openbenchmarking.org/result/1705261-S
```

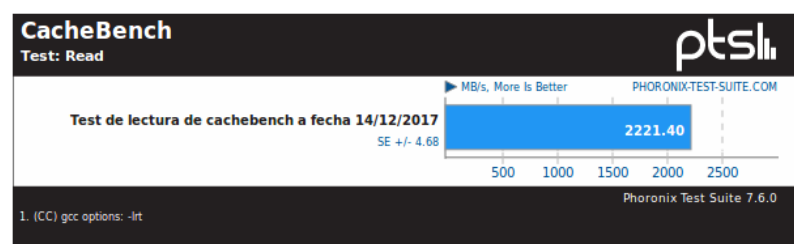
Results Overview

readCacheBenchTest	
	Test de lectura de cachebench a fecha 14/12/2017
cachebench: Test: Read	2221.40
Standard Error	4.68
Standard Deviation	0.36%
PHORONIX-TEST-SUITE.COM	

Test Results

CacheBench

Test: Read



7. Añadimos el **sudokut** (testea el rendimiento del procesador para resolver 100 sudokus) y lo lanzamos.

```
juanka1995@juanka1995-Laptop ~ $ phoronix-test-suite benchmark pts/sudokut
```

8. Elegimos el nombre que le pondremos al resultado del test

```
Would you like to save these test results (Y/n): y

Recently Saved Test Results:
- readcachebenchmark [Today]

Enter a name to save these results under: sudokuT
Enter a unique name to describe this test run / configuration: Test de resolver 100 sudokus
```

9. Esperamos a que termine y mostramos el resultado en nuestro navegador

```
SudokuT 0.4:
pts/sudokuT-1.0.0
Test 1 of 1
Estimated Trial Run Count: 3
Estimated Time To Completion: 5 Minutes [17:08 CET]
Started Run 1 @ 17:03:45
Started Run 2 @ 17:04:02
Started Run 3 @ 17:04:20

Total Time:
16.767526865005
16.814805984497
17.167697906494

Average: 16.92 Seconds
Deviation: 1.29%

Do you want to view the results in your web browser (Y/n):
```

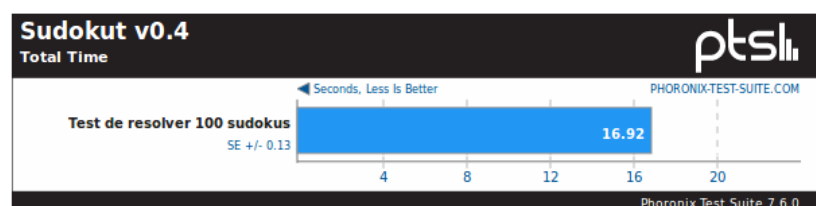
Results Overview

sudokuT	
pts	Test de resolver 100 sudokus
sudokuT: Total Time	16.92
Standard Error	0.13
Standard Deviation	1.29%
PHORONIX-TEST-SUITE.COM	

Test Results

SudokuT

Total Time



Apache Benchmark

1. Instalamos **apache2** en el anfitrión.

```
juanka1995@juanka1995-Laptop ~ $ sudo apt-get install apache2
```

2. Comprobamos que **apache2.service** esta corriendo en el anfitrión y en la maquina virtual de Ubuntu Server.
3. Ejecutamos el **ab** para hacer un test de peticiones a un servidor HTTP, desde la maquina anfitrión hacia la maquina virtual.

```
juanka1995@juanka1995-Laptop ~ $ ab -n 10000 -c 3 http://192.168.56.105/
This is ApacheBench, Version 2.3 <$Revision: 1706008 $>
Copyright 1996 Adam Twiss, Zeus Technology Ltd, http://www.zeustech.net/
Licensed to The Apache Software Foundation, http://www.apache.org/

Benchmarking 192.168.56.105 (be patient)
Completed 1000 requests
Completed 2000 requests
Completed 3000 requests
Completed 4000 requests
Completed 5000 requests
Completed 6000 requests
Completed 7000 requests
Completed 8000 requests
Completed 9000 requests
Completed 10000 requests
Finished 10000 requests

Server Software:      Apache/2.4.18
Server Hostname:      192.168.56.105
Server Port:          80

Document Path:        /
Document Length:       11321 bytes --> tamaño de web devuelta

Concurrency Level:     3
Time taken for tests:   4.337 seconds
Complete requests:      10000
Failed requests:         0
Total transferred:      115950000 bytes
HTML transferred:       113210000 bytes
Requests per second:    2305.56 [#/sec] (mean)
Time per request:       1.301 [ms] (mean)
Time per request:       0.434 [ms] (mean, across all concurrent requests)
Transfer rate:          26106.37 [Kbytes/sec] received

Connection Times (ms)
      min   mean[+/-sd] median   max
Connect:    0      0   0.4      0      7
Processing:  0      1   0.9      1     20
Waiting:    0      1   0.5      1     17
Total:      1      1   1.0      1     20

Percentage of the requests served within a certain time (ms)
 50%      1
 66%      1
 75%      1
 80%      1
 90%      2
 95%      3
 98%      5
 99%      6

porcentaje
```

JMeter

1. Lo primero que debemos hacer es instalar Java 8 ya que JMeter es una aplicación implementada en Java.

```
juanka1995@juanka1995-Laptop ~ $ sudo apt-get install openjdk-8-jre
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
openjdk-8-jre ya está en su versión más reciente (8u151-b12-0ubuntu0.16.04.2).
0 actualizados, 0 nuevos se instalarán, 0 para eliminar y 0 no actualizados.
juanka1995@juanka1995-Laptop ~ $
```

2. Después descargamos el paquete de instalación de la aplicación, lo descomprimos y seguidamente lo ejecutamos.

```
juanka1995@juanka1995-Laptop ~ $ wget http://apache.uvigo.es//jmeter/binaries/apache-jmeter-3.3
.tgz
--2017-12-21 05:46:40-- http://apache.uvigo.es//jmeter/binaries/apache-jmeter-3.3.tgz
Resolviendo apache.uvigo.es (apache.uvigo.es)... 193.146.32.74, 2001:720:1214:4200::74
Conectando con apache.uvigo.es (apache.uvigo.es)[193.146.32.74]:80... conectado.
Petición HTTP enviada, esperando respuesta... 200 OK
Longitud: 51622068 (49M) [application/x-gzip]
Grabando a: "apache-jmeter-3.3.tgz"

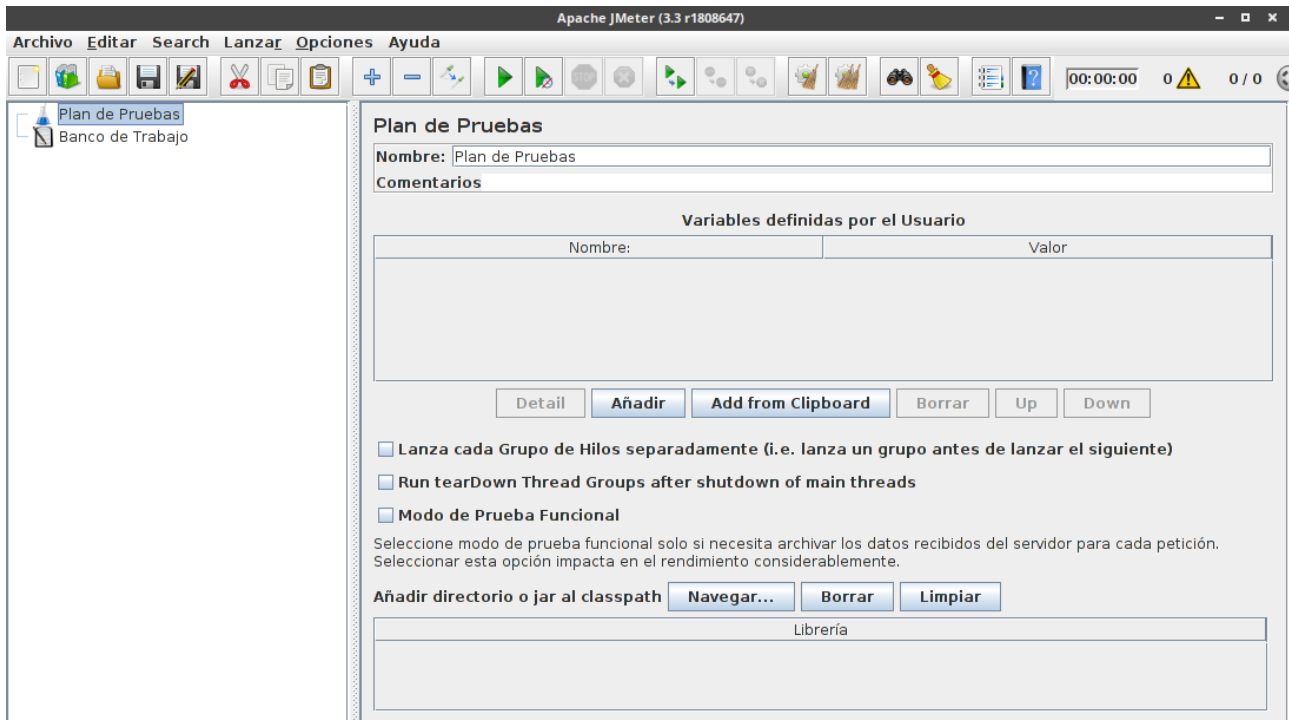
apache-jmeter-3.3.tgz 100%[=====>] 49,23M 5,64MB/s in 14s

2017-12-21 05:46:54 (3,64 MB/s) - "apache-jmeter-3.3.tgz" guardado [51622068/51622068]

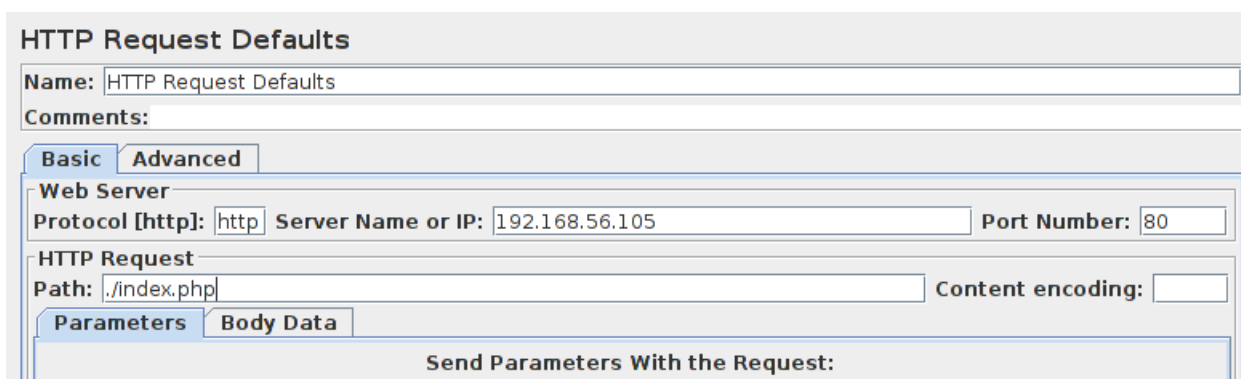
juanka1995@juanka1995-Laptop ~ $ tar -xvf apache-jmeter-3.3.tgz
apache-jmeter-3.3/bin/report-template/sbadmin2-1.0.7/
apache-jmeter-3.3/bin/report-template/sbadmin2-1.0.7/bower_components/
apache-jmeter-3.3/bin/report-template/sbadmin2-1.0.7/bower_components/bootstrap/
apache-jmeter-3.3/bin/report-template/sbadmin2-1.0.7/bower_components/bootstrap-social/
apache-jmeter-3.3/bin/report-template/sbadmin2-1.0.7/bower_components/bootstrap/dist/
apache-jmeter-3.3/bin/report-template/sbadmin2-1.0.7/bower_components/bootstrap/dist/css/
apache-jmeter-3.3/bin/report-template/sbadmin2-1.0.7/bower_components/bootstrap/dist/fonts/
apache-jmeter-3.3/bin/report-template/sbadmin2-1.0.7/bower_components/bootstrap/dist/js/
apache-jmeter-3.3/bin/report-template/sbadmin2-1.0.7/bower_components/bootstrap/fonts/
apache-jmeter-3.3/bin/report-template/sbadmin2-1.0.7/bower_components/bootstrap/grunt/
```

3. Y ahora ya podemos ejecutarlo. Vamos a la carpeta y ejecutamos jmeter.

```
juanka1995@juanka1995-Laptop ~ $ cd apache-jmeter-3.3/bin/  
juanka1995@juanka1995-Laptop ~/apache-jmeter-3.3/bin $ ./jmeter
```



4. Ahora vamos a crear un test desde 0, para ello click derecho sobre **Test Plan/Add/Config Element/HTTP Request Defaults**. Ahora definimos, el protocolo, la ip, la ruta y el puerto.



5. Después añadimos el gestor de hebras **Test Plan/Add/Threads (Users)/Thread Group**. Lo configuramos de la siguiente forma.

Thread Group

Name: Thread Group

Comments:

Action to be taken after a Sampler error

☒ Continue ☐ Start Next Thread Loop ☐ Stop Thread ☐ Stop Test ☐ Stop Test Now

Thread Properties

Number of Threads (users): 100

Ramp-Up Period (in seconds): 0

Loop Count: ☐ Forever 1

☐ Delay Thread creation until needed

☐ Scheduler

Scheduler Configuration

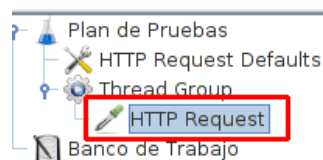
Duration (seconds)

Startup delay (seconds)

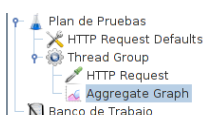
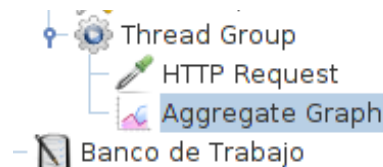
Start Time 2017/12/21 06:14:41

End Time 2017/12/21 06:14:41

6. Sobre este último, hacemos click derecho en **Thread Group/Add/Sampler/HTTP Request**.



7. Por ultimo añadimos **Test plan/Add/Listener/Aggregate Graph**. Y ya podemos pulsar el botón de play y esperar a que se ejecute el test



Aggregate Graph

Name: Aggregate Graph

Comments:

Write results to file / Read from file

Filename Browse...

Log/Display Only: ☐ Errors ☐ Success

Label	# Samples	Average	Median	90% Line	95% Line	99% Line	Min	Max	Error %	Throughput	Response Time
HTTP Requ...	100	6	3	16	20	23	1	23	0,00%	529,1/sec	
Total	100	6	3	16	20	23	1	23	0,00%	529,1/sec	

Settings Graph

Display Graph

Save Graph Save Table Data