

INGENIERÍA DE SERVIDORES

Guión de la Práctica 4

Benchmarks

Curso 2016-2017



GUIÓN DE PRÁCTICAS

Práctica 4

ÍNDICE

1.-OBJETIVOS MÍNIMOS.....	3
2.-BENCHMARKS POPULARES.....	4
2.1.-PHORONIX SUITE.....	4
2.2.-BENCHMARKS Y TESTS DE ESTRÉS PARA WEBS.....	4
2.3.-BENCHMARKS PARA WINDOWS.....	5
2.3.1 SISoftware SANDRA.....	5
2.3.2 AIDA64 (ANTIGUO EVEREST).....	5
2.4.-MÁS BENCHMARKS.....	6
3.-NORMATIVA.....	6

Índice de Figuras



GUIÓN DE PRÁCTICAS

Práctica 4

TÍTULO DE LA PRÁCTICA:

Benchmarks

Centro: Escuela Técnica Superior de Ingeniería
Informática y Telecomunicación

Asignatura: Ingeniería de Servidores

Profesores: Alberto Guillén Perales
David Palomar Sáez
Héctor Pomares Cintas
Gonzalo Ruiz García

Grupos: A, B y C

1.- Objetivos mínimos

- 1) Conocer varios benchmarks para diferentes servicios.
- 2) Saber comparar distintos servicios en base a benchmarks.
- 3) Ser capaz de definir y proponer un benchmark



GUIÓN DE PRÁCTICAS

Práctica 4

2.- Benchmarks populares

2.1.- PHORONIX SUITE

Es una plataforma que permite ejecutar un conjunto de benchmarks bajo la agrupación openbenchmarking.org. La aplicación puede instalarse a través de los gestores de paquetes ya vistos en los guiones anteriores.

Una vez que conoce los benchmarks, puede pasar a seleccionar algunos de ellos e instalarlos (no todos los listados son los que están instalados).

Cuestión 1: Seleccione, instale y ejecute uno, comente los resultados.
Atención: no es lo mismo un benchmark que una suite, instale un benchmark.

2.2.- BENCHMARKS Y TESTS DE ESTRÉS PARA WEBS

2.2.1 Apache benchmark

Dentro de los benchmarks más populares para servidores web podemos encontrar el Apache Benchmark (comando `ab`).

Cuestión 2: De los parámetros que le podemos pasar al comando ¿Qué significa `-c 5` ? ¿y `-n 100`? Monitoree la ejecución de `ab` contra alguna máquina (cualquiera) ¿cuántas “tareas” crea `ab` en el cliente?

Cuestión 3: Ejecute `ab` contra a las tres máquinas virtuales (desde el SO anfitrión a las máquinas virtuales de la red local) una a una (arrancadas por separado). ¿Cuál es la que proporciona mejores resultados? Muestre y coméntelos. (Use como máquina de referencia Ubuntu Server para la comparativa).

2.2.2 Gatling

En vez de benchmark, Gatling(<http://gatling-tool.org/>) se define como una herramienta de estrés (Stress tool) de código abierto. El concepto de test de estrés es llevar al sistema al máximo de sus posibilidades con el fin de analizar la fiabilidad, estabilidad y tolerancia a fallos. Los tests (las situaciones de estrés) se programan en Scala.



GUIÓN DE PRÁCTICAS

Práctica 4

Cuestión opcional 1: ¿Qué es Scala? Instale Gatling y pruebe los escenarios por defecto.

2.2.3 Jmeter

Este software (<http://jmeter.apache.org/>) se autodefine como una aplicación “...*designed to load test functional behavior and measure performance*...”, es decir, no establecen ningún tecnicismo concreto.

Ha sido comparado por la empresa Flood.io con Gatling obteniendo la conclusión de que ambos tienen un comportamiento y capacidades similares (<https://flood.io/blog/11-benchmarking-jmeter-and-gatling>).

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 sus máquinas virtuales ¿coincide con los resultados de ab?

2.3.- BENCHMARKS PARA WINDOWS

2.3.1 SISoftware SANDRA

Este programa (<http://www.sisoftware.net/>) es comercial y permite testear y medir varios aspectos del *hardware* del equipo y hacer diagnósticos del estado del sistema y monitorizarlo.

2.3.2 AIDA64 (ANTIGUO EVEREST)

Es otro programa que nos permite monitorizar varios elementos del computador así como ejecutar benchmarks para llevar los componentes al límite.

Puede descargar una versión de prueba gratuita durante 30 días en <http://www.aida64.com/downloads>

2.4.- MÁS BENCHMARKS...

Hay muchos más (p.ej. Para DNSs: **NameBench** y **GRC's DNS Benchmark**) pero lo más interesante es que puede programar uno para analizar algún parámetro concreto



GUIÓN DE PRÁCTICAS

Práctica 4

<http://sourceforge.net/directory/os:linux/?q=benchmark>

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.

Tenga en cuenta que puede comparar varios gestores de BD, lenguajes de programación web (tiempos de ejecución, gestión de memoria, ...), duración de la batería, servidor DNS, etc. . Alternativamente, puede descargar alguno de algún repositorio en github y modificarlo según sus necesidades.

3.- NORMATIVA

La indicada en la guía de prácticas.