Índice

[Herramienta de test de rendimiento jMeter. 2](#_Toc510458708)

[Creando un caso de prueba. 2](#_Toc510458709)

[Grabando un guion (script). 3](#_Toc510458710)

[Ejecutando los scripts. 5](#_Toc510458711)

[Informes de rendimiento. 6](#_Toc510458712)

[Ejecutando los test. 7](#_Toc510458713)

[Diagnóstico. 7](#_Toc510458714)

Las pruebas de rendimiento se refieren a una serie de pruebas que se llevan a cabo para

determinar cómo funciona un sistema en términos de estabilidad y capacidad de respuesta cuando es enfrentado con una carga de trabajo particular. Por estabilidad nos referimos a la capacidad del sistema para realizar sin fallas; por receptividad nos referimos a su capacidad para responder a

solicitudes en un tiempo razonable.

Antes de comenzar con las pruebas de rendimiento es necesario conocer dos conceptos clave:

* Caso de prueba: es una prueba que emula a un usuario que realiza un caso de uso. Los casos de prueba se implementan usando scripts que emulan las interacciones que un usuario realiza con un sistema cuando está ejecutando un caso de uso. Los casos de prueba también tienen los llamados informes de rendimiento, que básicamente miden el rendimiento del sistema en términos de casos de uso completados por unidad de tiempo.
* Test suite: es una colección de casos de prueba.

Para cada caso de uso:

* Escribe un caso de prueba.
* Ejecútelo repetidamente aumentando gradualmente la carga de trabajo. Aumentando la carga de trabajo significa que simulas un número monótonamente creciente de usuarios que están usando tu sistema. No empiece con 1000 usuarios.
* Anote la carga de trabajo máxima. La carga de trabajo máxima es el número máximo de usuarios simultáneos que su sistema admite con respecto a un caso de uso particular mientras sigue respondiendo y no se cuelga.
* Como resultado, obtendrá un cuadro en el que documentará la carga de trabajo máxima que su sistema puede manejar según cada caso de uso.

Cuando estén todos los casos de uso probado el rendimiento calcule la carga de trabajo máxima que su sistema puede manejar. Esta cantidad es el mínimo que haya entre todos los casos de uso probados, es decir tenemos por ejemplo 3 casos de uso, el primero soporta 100 usuarios, el segundo 250 y el tercero 49 entonces la carga máxima soportada es 49 usuarios.

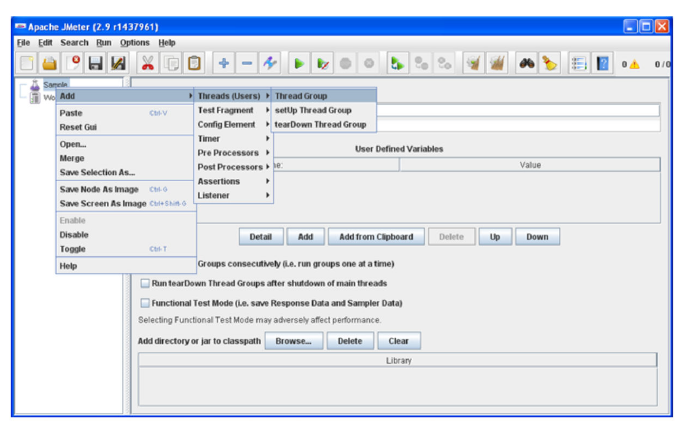
Puede usar el entorno de desarrollo para los test funcionales y el entorno de preproducción para llevar a cabo los test de rendimiento.

Vamos a usar Apache jMeter para las pruebas de rendimiento. Nos ayuda a crear pruebas de rendimiento y ejecutarlas fácilmente.

# Herramienta de test de rendimiento jMeter.

## Creando un caso de prueba.

Una vez abierta la herramienta tendremos que cambiar el nombre del test. Debemos crear un llamado grupo de hilos, el grupo de hilos nos ayudará a simular una cantidad de usuarios simultáneos que están ejecutando casos de uso típicos en nuestro sistema.

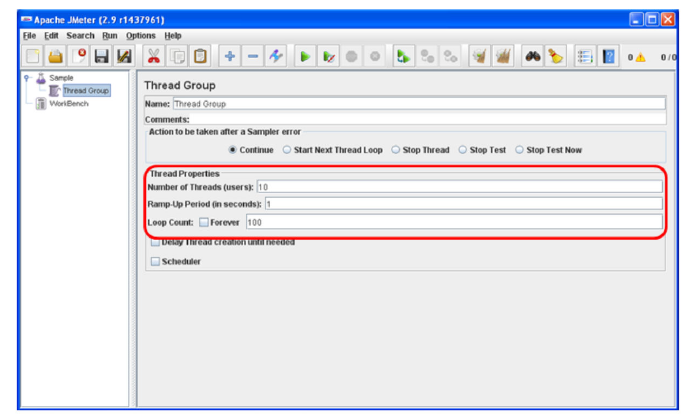


La configuración mínima que debe realizar es cambiar el número de subprocesos,

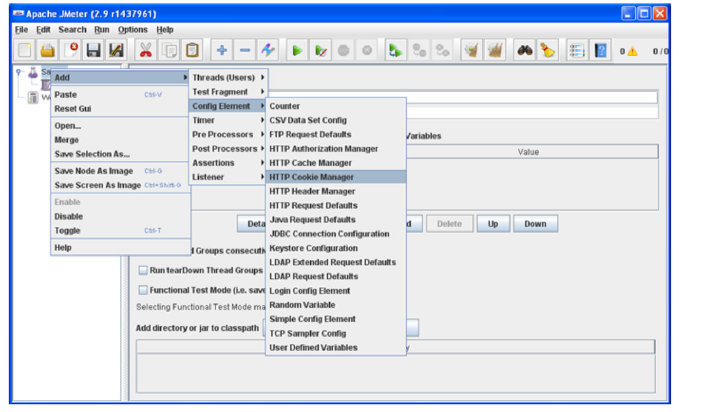
que indica el número de usuarios que se simularán y el recuento de bucles, que

indica cuántas veces cada hilo ejecutará el caso de uso con el que vamos

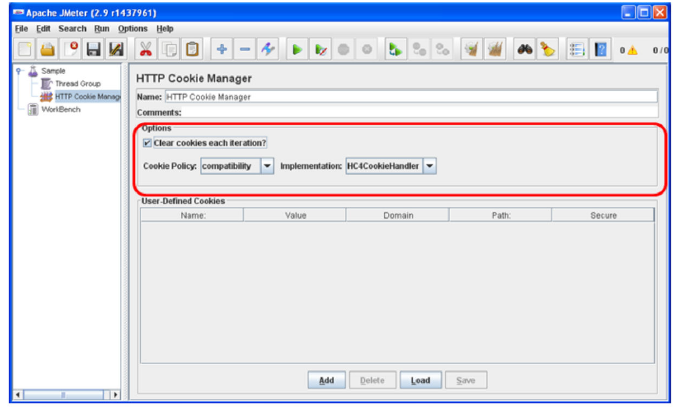
para probar nuestro sistema.



También necesitamos agregar un administrador de cookies HTTP a nuestro punto de referencia.



Compruebe que desea que las cookies se borren en cada iteración, establecer la política de cookies en "compatibilidad" y establecer la implementación de "HC4CookieHandler".



## Grabando un guion (script).

Ahora tenemos que explorar cómo grabar una secuencia de comandos.

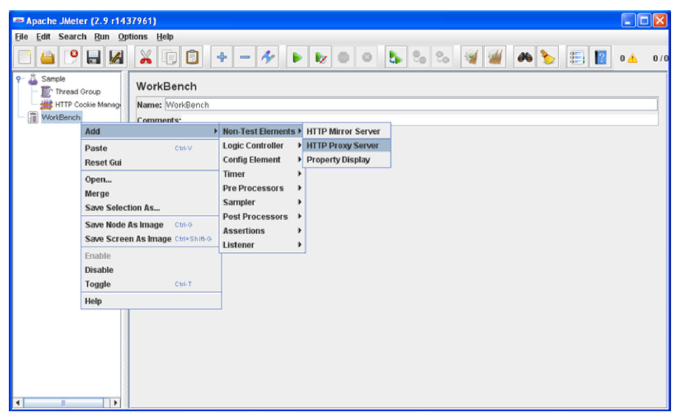
Nosotros necesitamos agregar scripts a nuestros casos de prueba para que jMeter pueda reproducirlos y simular un número de usuarios que están interactuando con nuestro sistema.

La elaboración manual de tales scripts no es nada fácil. Más que eso: es muy propenso a errores. Afortunadamente, jMeter puede grabarlos automáticamente mientras interactúa con sus sistemas. jMeter usa proxies para grabar sus guiones.

Conexiones proxy. Tal conexión pasa por un proxy, que es un dispositivo que actúa como intermediario entre su navegador y el el sitio real al que está enviando una solicitud.

Tener un proxy tiene una serie de ventajas para el sistema que ya se conocen.

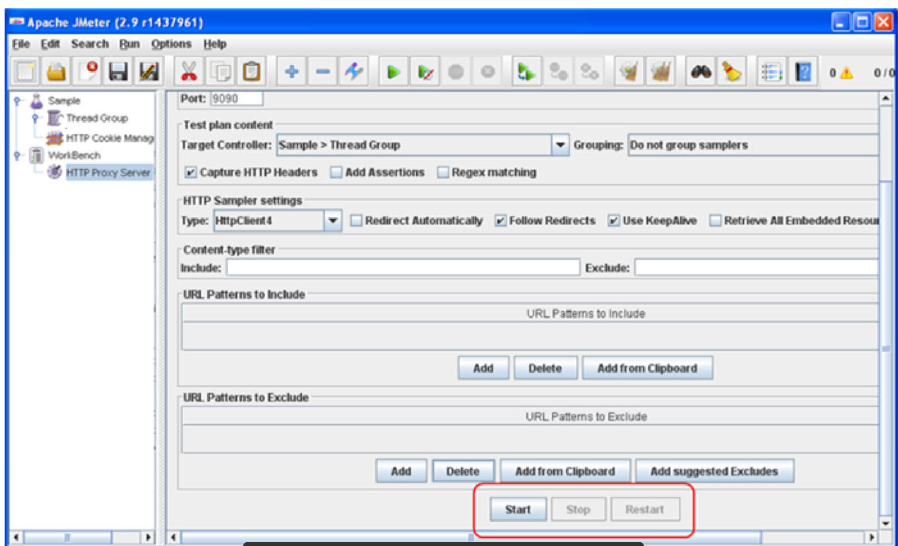
jMeter será nuestro proxy. Para convertir jMeter en un proxy, debe hacer clic derecho en "Workbench" y agregar un HTTP servidor proxy.



Solo necesita cambiar el puerto a 9090 para evitar confusiones con el puerto HTTP que usamos en configuraciones de desarrollo.   
El siguiente paso es cambiar el proxy de tu sistema operativo. Para hacerlo, debes abrir tu configuración de red de área local haciendo clic en Inicio> Panel de control> Red e Internet Conexiones> Opciones de Internet> Conexiones> Configuración de LAN. Obtendrás un diálogo, donde debe verificar que su servidor proxy esté en la dirección "localhost" y el puerto "9090". Por favor, no olvide verificar que desea eludir el servidor proxy para direcciones locales.

En este punto si abrimos el navegador y accedemos a acme.com debemos de obtener un error porque establecimos el proxy, pero no lo iniciamos. Si probamos con “localhost” si debe funcionar puesto que se salta el proxy.

Ahora nos dirigimos a jMeter y arrancamos el proxy y una vez arrancado debe funcionar ingresando [www.acme.com](http://www.acme.com).

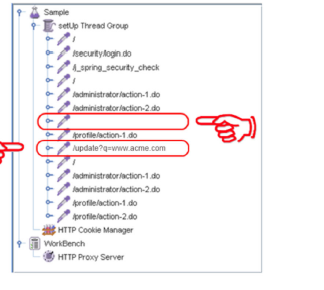


## Ejecutando los scripts.

Desafortunadamente, los scripts que usted graba generalmente no están listos para ser ejecutados. antes de eso, necesita eliminar entradas espurias y agregar retrasos humanos sensibles.

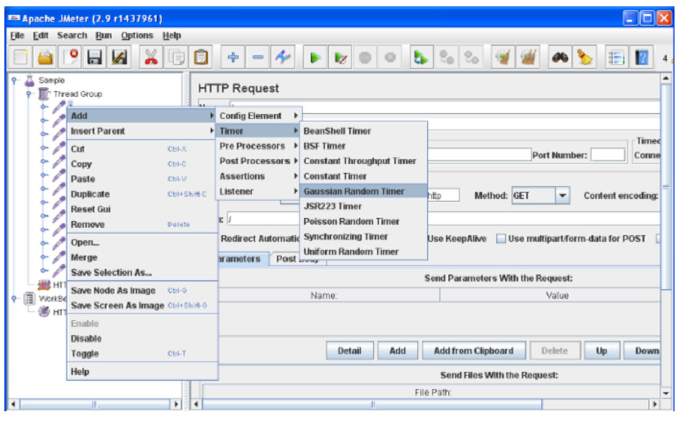
Debes verificar que cada entrada en su secuencia de comandos corresponde a su sistema de información web; es decir, debes compruebe que no haya entradas falsas.

Las entradas falsas son entradas como las que hemos resaltado en esta imagen. Nota que hay una entrada sin una ruta y una entrada adicional que corresponde a una ruta que no existe en nuestro sistema. Ejemplo en la siguiente imagen.

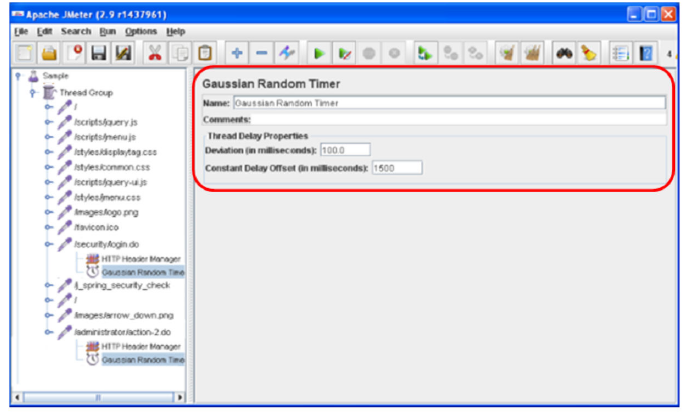


Las solicitudes falsas se mandan a Google. Las primeras veces, Google le permite recuperar el servicio escribiendo una captcha; pero si persiste, Google definitivamente prohibirá que su dirección IP tenga acceso a sus servicios. Así que será mejor que revises tu script y elimines las entradas que involucran servidores de otras empresas.

Las interacciones de un humano con un sistema tienen retrasos, si ejecutamos un script en este no existen retrasos por tanto debemos de introducirlos nosotros mismos. Para introducir un retraso en sus scripts, necesita agregar temporizadores a sus entradas. Haga clic en las entradas que representan las interacciones humanas y seleccione "Agregar> Temporizador". Ahí hay una variedad de temporizadores disponibles, pero recomendamos que use Gaussian temporizadores.



La configuración de un temporizador gaussiano es muy fácil. Recomendamos que establezca la desviación en 100 milisegundos y el desplazamiento a 1500 milisegundos.



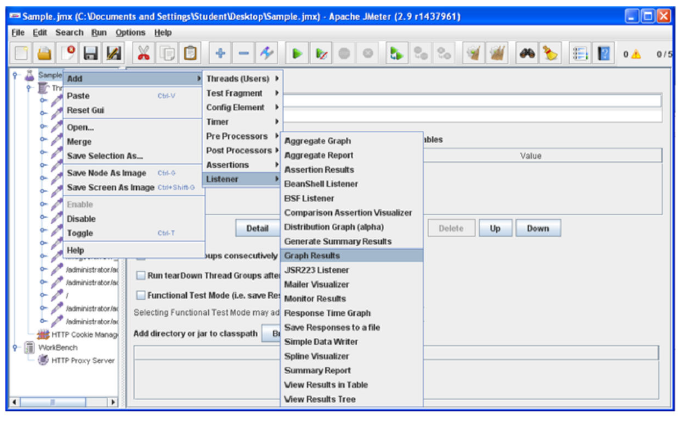
Tenga en cuenta que solo debe retrasar las entradas relacionadas con las interacciones del usuario.

## Informes de rendimiento.

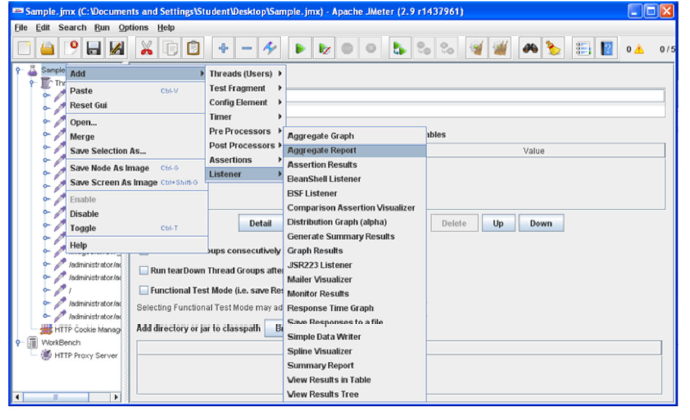
Nos centraremos en informes de gráficos e informes agregados.

Los informes se presentan por medio de los llamados oyentes. Un oyente es un objeto que escucha

a la ejecución de sus casos de prueba, recopila medidas y crea informes. Por favor añadir un oyente "Resultados de gráficos" para su caso de prueba para que pueda tener un informe gráfico.



Necesitamos un informe agregado para poder inspeccionar los detalles. Para agregar un informe, haga clic en "Agregar> Oyente" y selecciona "Agregar informe".



## Ejecutando los test.

jMeter proporciona el típico botón triángulo verde para comenzar a ejecutar un punto de referencia, un rojo botón de detener para detenerlo, y un botón para borrar los informes. Es importante que borre sus informes cada vez que reinicie un caso de prueba.

# Diagnóstico.

El diagnóstico consiste en encontrar la razón por la cual el sistema no puede manejar más carga de trabajo.

Comúnmente un sistema no puede manejar suficiente carga de trabajo porque hay defectos (errores) que resultan en fallas (errores), o hay un cuello de botella.

Recuerde que hay una columna llamada "Error%" en los informes agregados. Mide el número de errores HTTP que jMeter atrapa. jMeter cuenta un error cuando obtiene un HTTP

código de respuesta en la serie 4xx o 5xx.

Primero, tú podrías haber cometido un error con respecto a una URL, que podría no existir en el servidor (HTTP código de error 404) o el cliente podría no estar autorizado para recuperarlo (códigos de error HTTP 401 y 403). En segundo lugar, si olvida agregar retrasos humanos sensibles a sus secuencias de comandos, entonces el servidor probablemente responderá con un código de error HTTP 429.

Si obtiene un error HTTP, debe encontrar la razón por la cual ... y solo hay una forma: usted

tiene que buscar los registros de Tomcat y revisarlos para encontrar el problema. Recordar que

hay una carpeta llamada "logs" en el directorio de instalación de Tomcat. Por favor, busque los archivos de registro que corresponden al día en el que estás trabajando y analízalos.

Mientras ejecuta sus casos de prueba, debe verificar que el tiempo de respuesta promedio no sea demasiado alto para la mayoría de los usuarios. Eche un vistazo a una medida llamada "Línea 90%" en su informe agregado.

¿cómo puede descubrir qué componentes de su sistema son

cuellos de botella? La respuesta es muy simple: use un medidor de rendimiento. Abierto shell de un administrador y ejecute el comando "perfmon.exe". Debe agregar el llamado contadores de rendimiento para el medidor de rendimiento para que pueda supervisar su actividad del sistema mientras se ejecuta un caso de prueba de rendimiento. Para agregar contadores de rendimiento, por favor, haga clic en el botón "+" en la barra de botones; Desafortunadamente, no hay opción en el menú principal.

Primero, agregue un contador de rendimiento llamado "% de tiempo del procesador". Este contador mide el porcentaje de tiempo que el procesador gasta para ejecutar un hilo de usuario. Es el principal indicador de actividad del procesador, y muestra el porcentaje promedio de tiempo ocupado observado durante el intervalo de muestra.

Ahora, agregue un contador de rendimiento llamado "Prom. Longitud de la cola del disco ". Mide la

promedio de solicitudes de lectura y escritura que se pusieron en cola para el disco seleccionado

durante el intervalo de muestra.

A continuación, agregue un contador de rendimiento llamado "Páginas / seg". Este contador de rendimiento mide la velocidad a la que se leen o escriben las páginas en el disco.

Finalmente, agregue un contador de rendimiento llamado "Paquetes / seg". Este contador proporciona una estimación precisa del ancho de banda que está entregando la interfaz de red.