Bogotá, 25 de Octubre de 2015

Universidad de los Andes

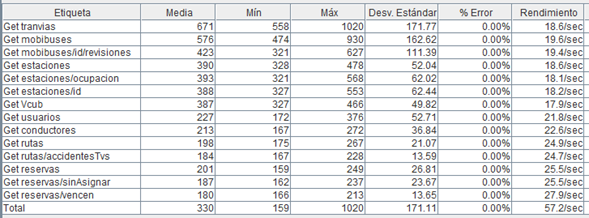
Arquitectura de software 2015-20

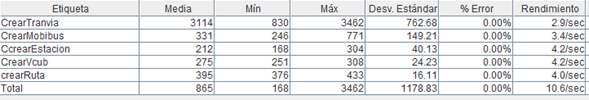
Grupo 1

**Documento de experimentación.**

En la pre entrega de este experimento se probaron los servicios de TBC transportes usando un computador como balanceador de carga y otros dos como servidores de la aplicación destinados a manejar las solicitudes que se generan a partir de los diferentes usuarios como lo son los vehículos, los clientes y el administrador del sistema. Además, las bases de datos estaban alojadas en la plataforma de heroku, esto generó limitaciones puesto que se contaba con un plan gratuito de servicios.

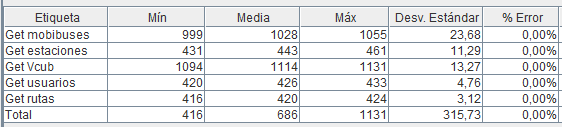
Se presenta a continuación se presentan los resultados obtenidos en la pre entrega del experimento, en donde se consumen los servicios crear y get de la aplicación. El valor esperado del tiempo de respuesta debe ser a lo sumo 1 segundo.

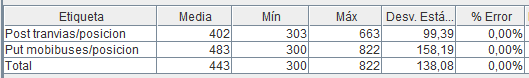




Se puede ver que todas las pruebas salvo crear tranvía cumplen con el requerimiento de calidad de rendimiento. Ahora bien, para la prueba definitiva de este experimento se agregó un componente de seguridad, el cual inevitablemente ha de afectar el comportamiento. Para garantizar la seguridad se implementó el protocolo https, el cual se encarga de cifrar los mensajes que viajan entre la capa de la aplicación y la red, para evitar así espionaje a nivel de red. Además, se realiza una autenticación intercambiando tokens entre el cliente y el servidor. Dichos tokens se encuentran cifrados en la base de datos para garantizar confidencialidad y control de acceso sobre quién puede consumir los recursos y consultar información importante.

A continuación se presentan los resultados de las pruebas tras haber implementado la capa de seguridad para analizar el efecto que tuvo sobre el rendimiento de la aplicación.



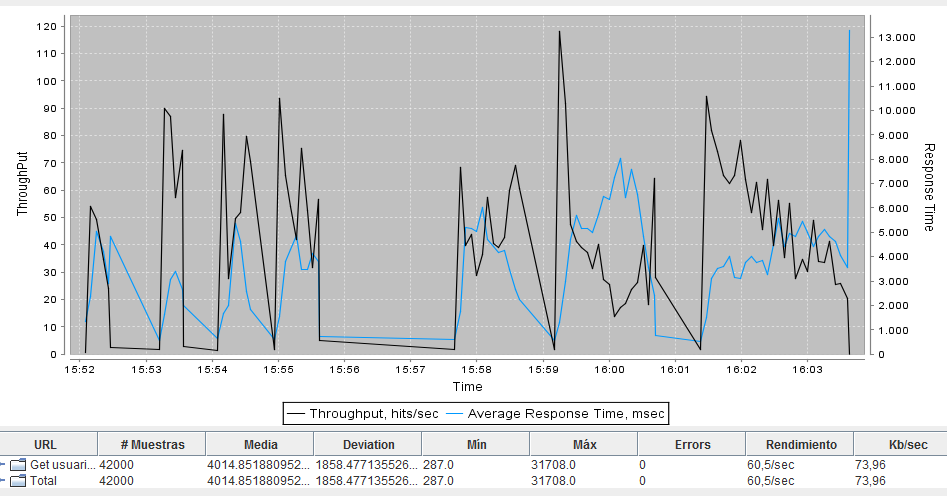


Comparando los resultados de la pre experimentación se hace evidente un incremento en la media de tiempo de respuesta de las peticiones http de tipo get, en el escenario sin seguridad hubo una media de respuesta de 330 milisegundos, y tras implementar el protocolo https, la media fue de 686 milisegundos. Impacto significativo que redujo el rendimiento a aproximadamente la mitad.

El impacto que se generó en el rendimiento es debido a que en la implementación del protocolo https se añade una capa entre la capa de aplicación y red, la cual se encarga de servicios de encriptación, proceso que tiene un alto costo computacional y esto incrementa la latencia de la respuesta del servidor.

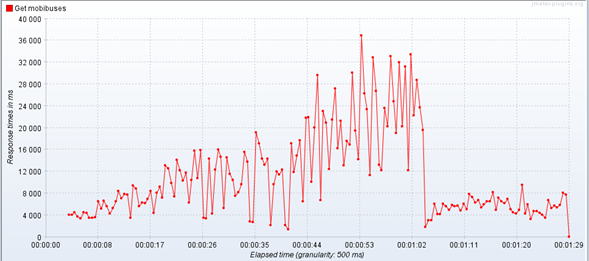
Cabe resaltar que habiendo implementado el protocolo de seguridad la aplicación sigue cumpliendo el escenario de calidad de rendimiento.

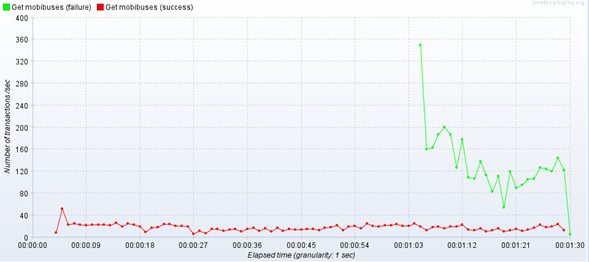
La prueba de escalabilidad se realizó sobre un ambiente de sobrecarga, se enviaron solicitudes https de un servicio get para tratar de ver en qué momento el comportamiento no es el esperado. esta prueba se realizó con 500 hilos y se incrementó el número de bucles, comenzando en 1500, 3500, 5500, hasta 11000 solicitudes. en el gráfico de resultados que se presenta a continuación se nota como el tiempo de respuesta fue elevado( superior a 1 segundo), pero se atendieron todas las solicitudes con 0% de error. lo cual es un buen indicador de escalabilidad, esto se atribuye primero al balanceador de carga de heroku, y segundo a su potente capacidad informática.



en las últimas respuestas, las cuales iban dentro de un paquete de 11000 solicitudes tuvieron una latencia promedio de 13 segundos. En este escenario no se cumple el requerimiento de rendimiento, pero gracias a la capacidad de heroku la escalabilidad no se ve comprometida.

punto importante a comparar con las pruebas realizadas en la pre entrega, en donde hubo una significante cantidad de solicitudes que dejaron de ser atendidas y por ende retornaron un error. Estos resultados se muestran a continuación :





**Conclusión**

La etapa de pruebas funcionales de una aplicación son demaciado importantes, debido a que a partir de las mismas se podrá medir el comportamiento del sistema TBC en terminus de rendimiento y escalabilidad. Una mala configuración del servidor puede influenciar negativamente el uso de la misma.

En este experimento se evaluó la utilización de la herramienta Jmeter, para la etapa de prueba de rendimiento del servidor web de heroku implementado para el sistema. Al tratarse de un sistema al cual acceden multiples usuarios en forma concurrente y se esta recibiendo informacion de los vehiculos periodicamente, es indispensable estar seguros de una buena configuración del servidor donde se procesaran estas solicitudes.

Jairo Bautista

Santiago Beltran

Carlos Rincon

Daniel Torres