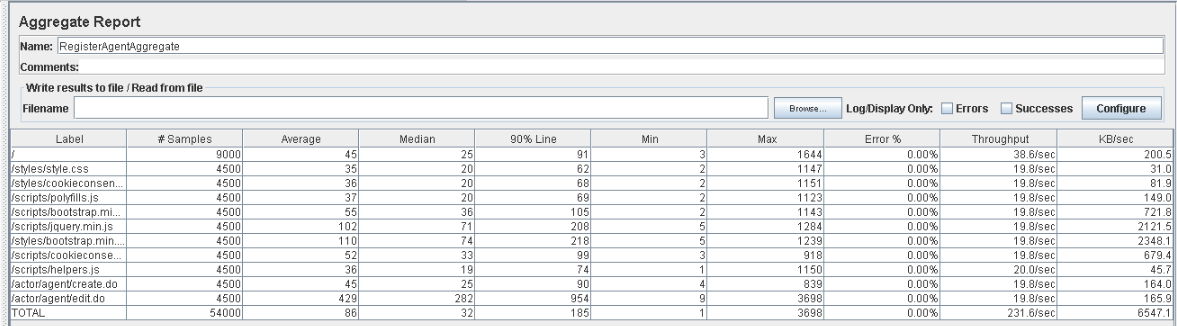
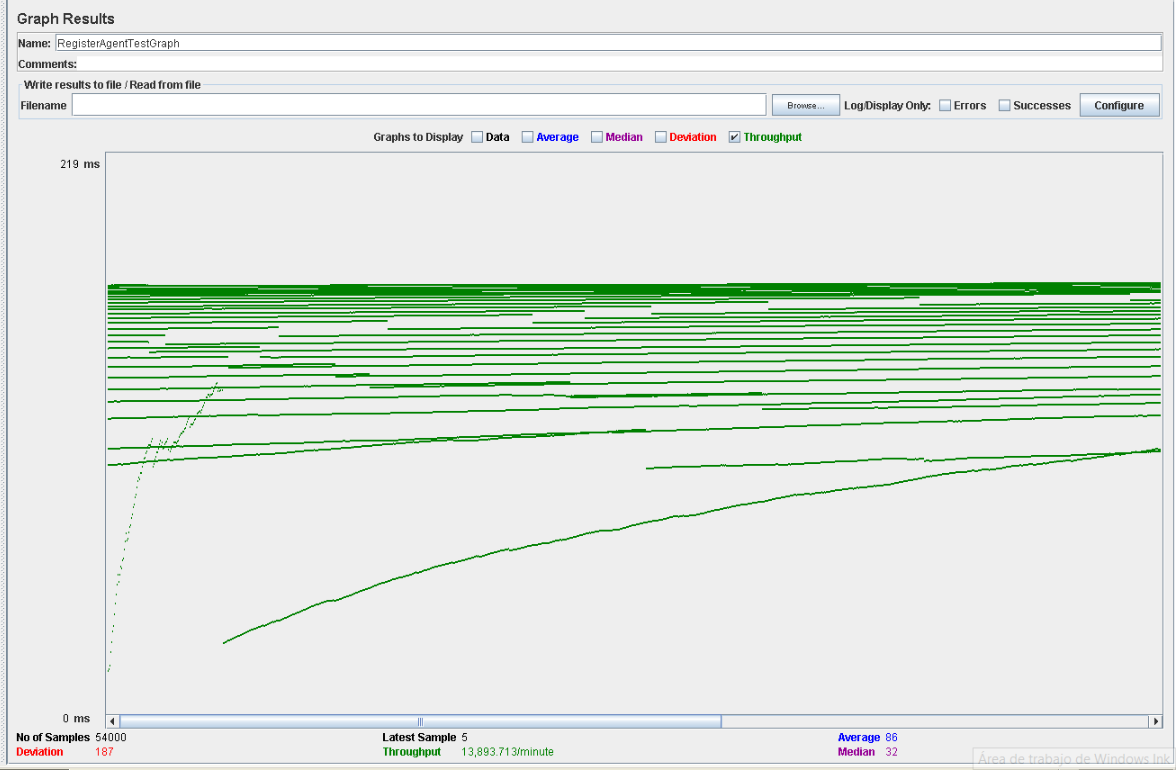
# Caso de uso: registrarse como agente.

En este caso de uso, accedemos a la página, pulsamos en el menú “Create agent”, rellenamos el formulario y guardamos.

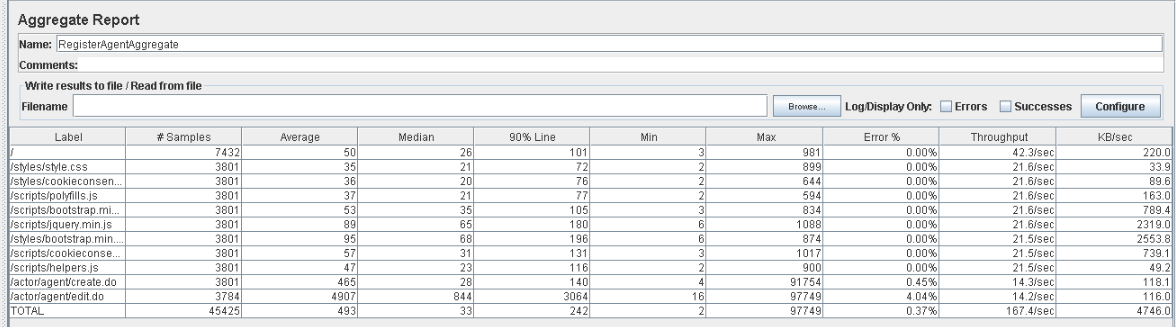
Comenzamos probando con 130 usuarios, 1 ramp-up y 30 acciones cada uno, no obteniendo errores:



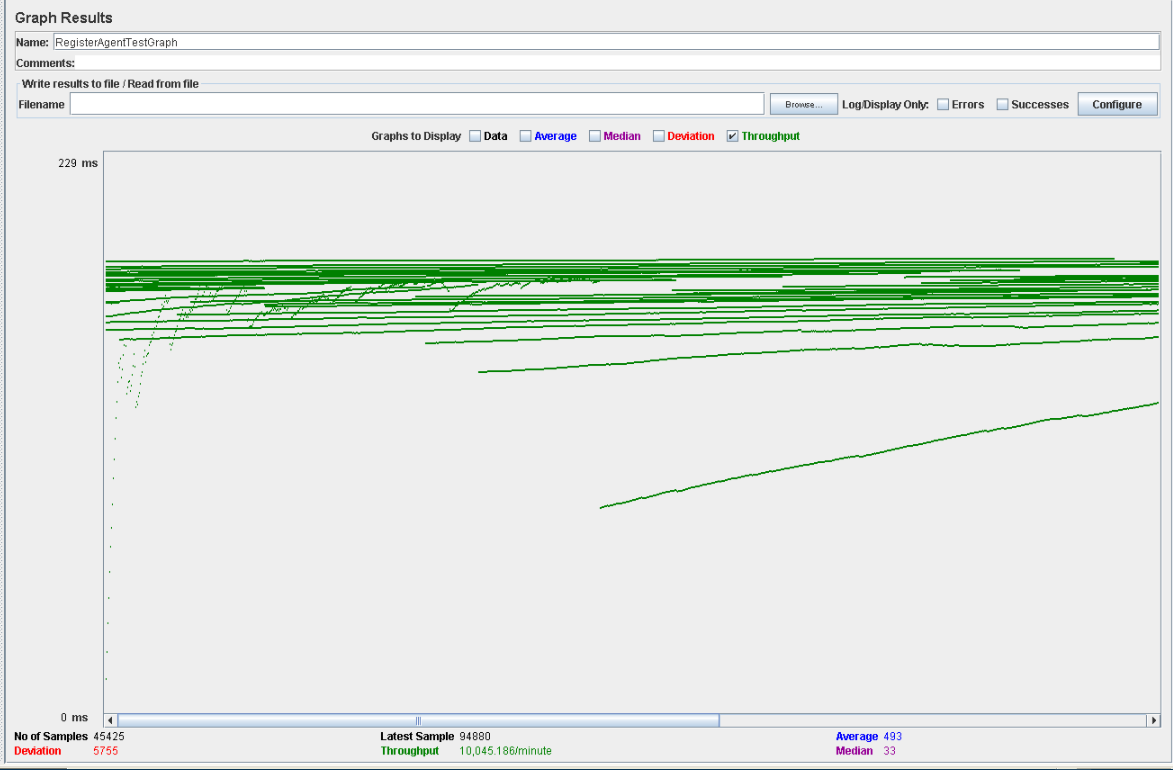


En esta tabla, el rendimiento es de aproximadamente 13.900 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

La configuración en esta ocasión es de 170 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores, pero unos tiempos de espera considerables, los que podemos ver en el Aggregate Report y en el Graph Results:



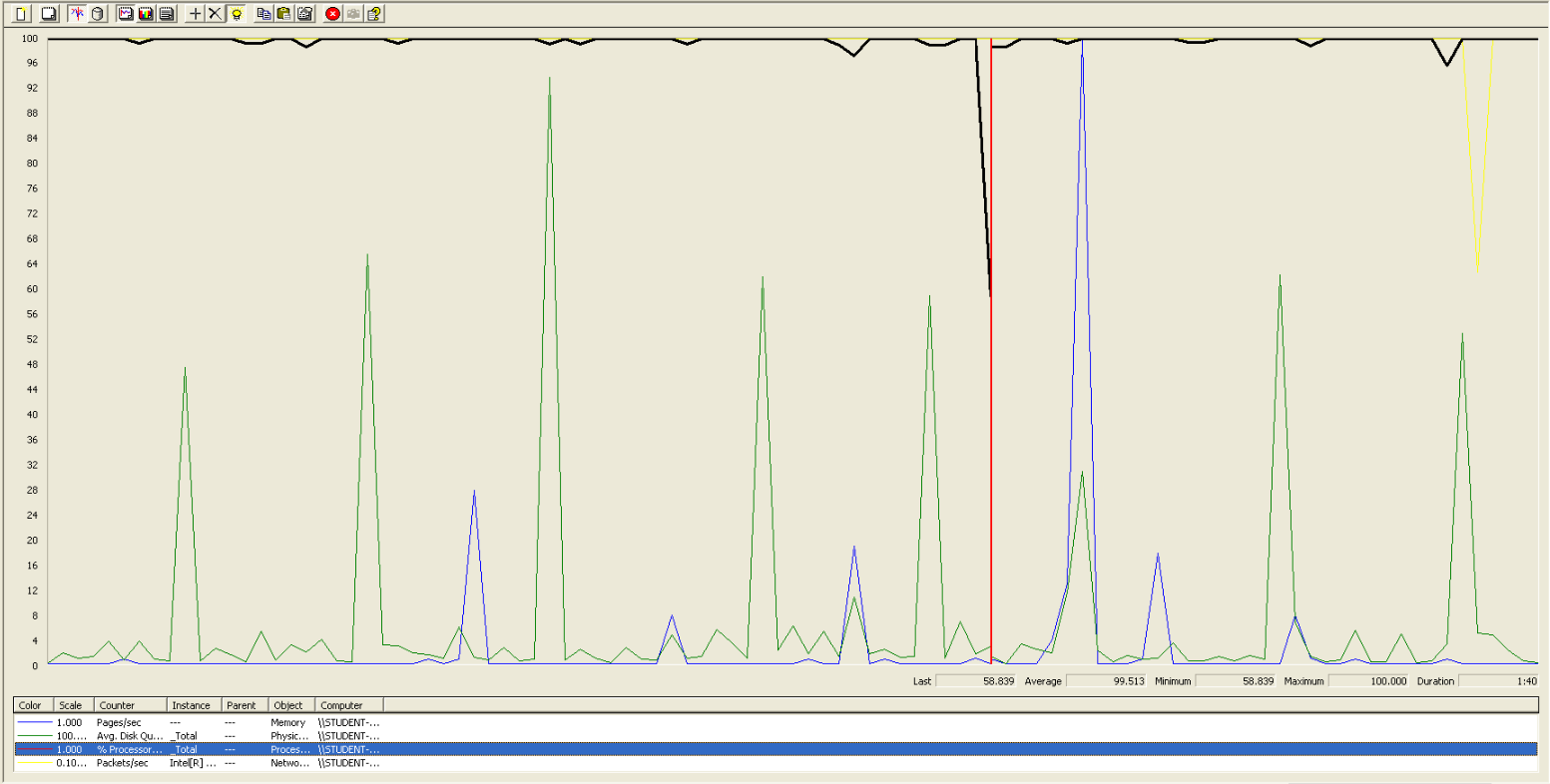
Como podemos observar fijándonos en 90% Line, la dirección /actor/agent/edit.do genera un tiempo de respuesta (3064 ms) mayor a lo establecido (1000 ms).



En esta tabla, el rendimiento es de aproximadamente 10.000 ejecuciones por minuto (desciende al aumentar el número de usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando, y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Ahora estudiaremos a que se deben los tiempos y errores:

Abrimos performance.exe y vemos la CPU tienen porcentajes muy altos, llegando al 100% en ocasiones, por lo que podemos estar ante un problema de CPU, ya que se está utilizando un alto porcentaje de la capacidad del procesador.



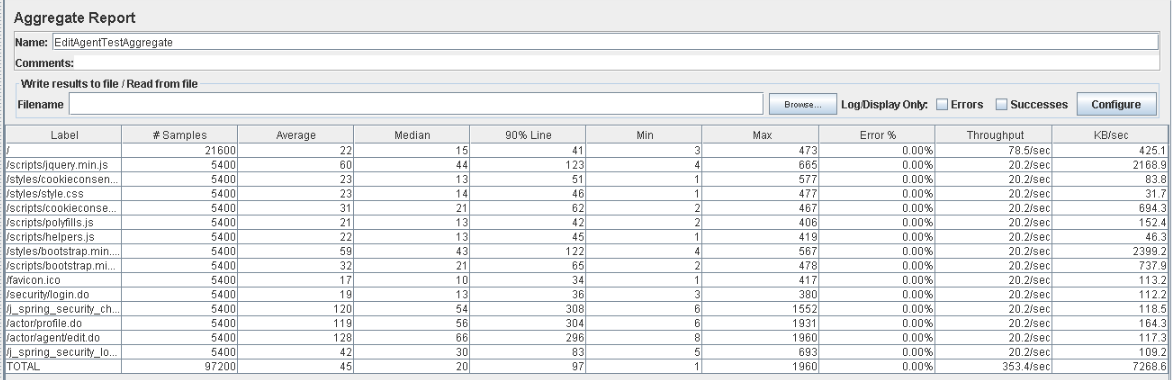
Performance con Processor Time.

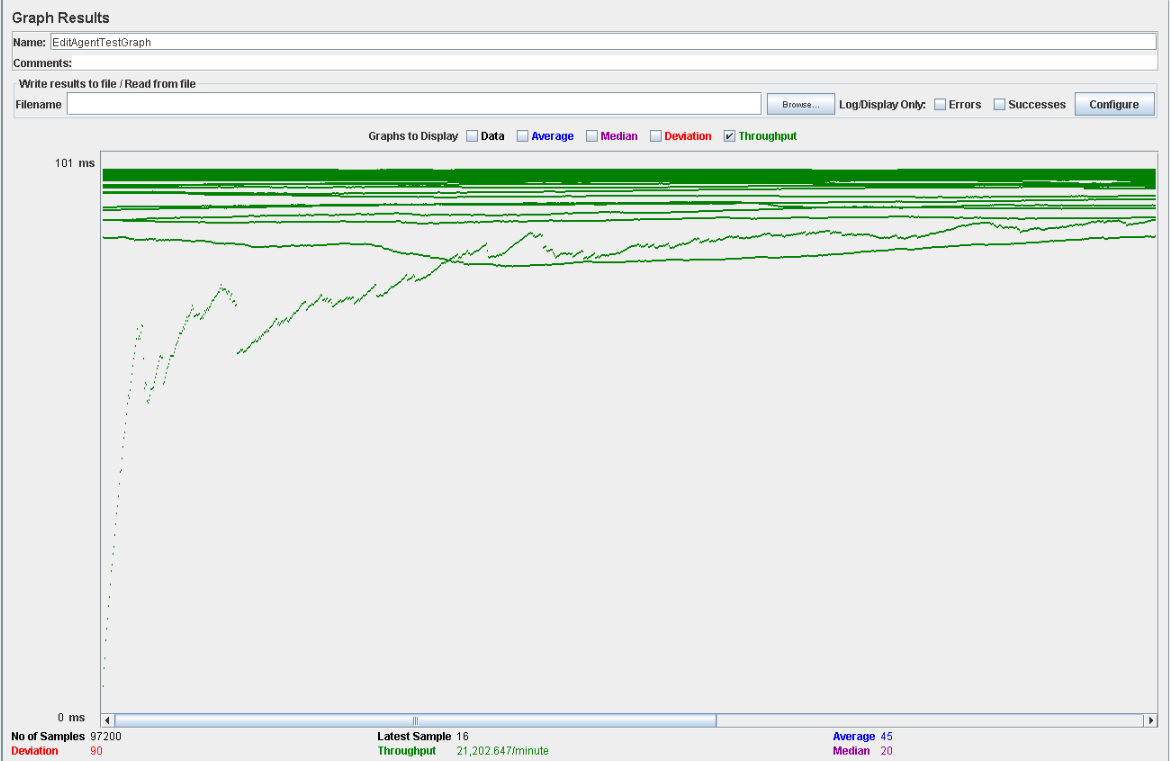
Por lo que, en este caso de uso, los usuarios permitidos serán entre 130 y 140.

# Caso de uso: editar perfil como agente.

En este caso de uso, accedemos a la página, nos autenticamos como agent, accedemos a nuestro perfil, editamos la información que deseemos y nos desautenticamos.

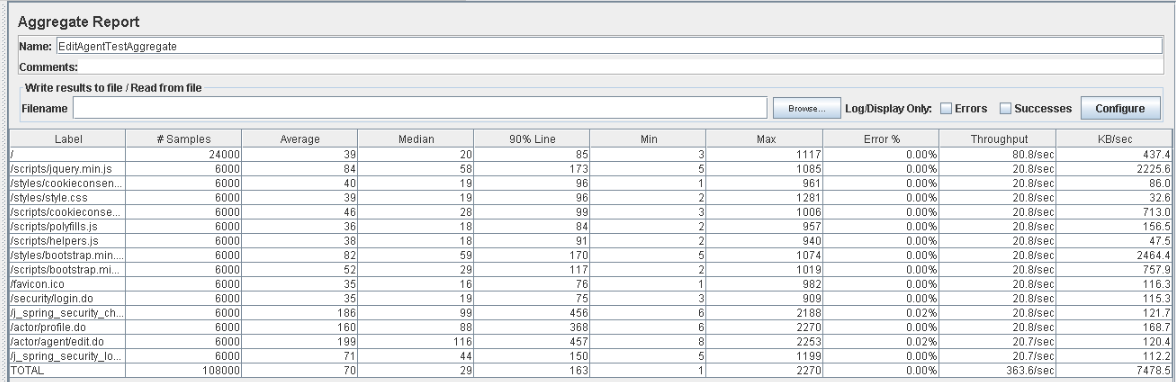
Probamos una configuración 180 usuarios, 1 ramp-up y 30 acciones cada uno, no obteniendo errores ni esperas demasiado altas:



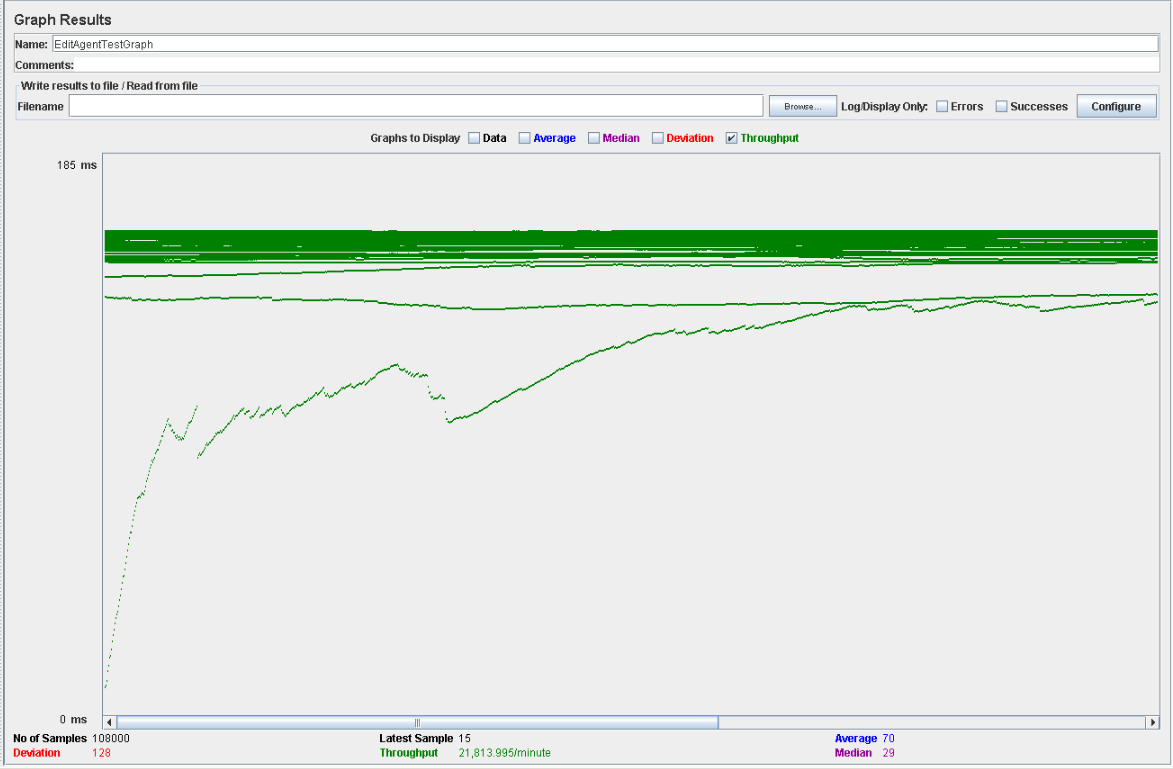


En esta tabla, el rendimiento es de aproximadamente 21.200 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

La configuración en esta ocasión es de 200 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores, pero unos tiempos de espera considerables, los que podemos ver en el Aggregate Report y en el Graph Results:



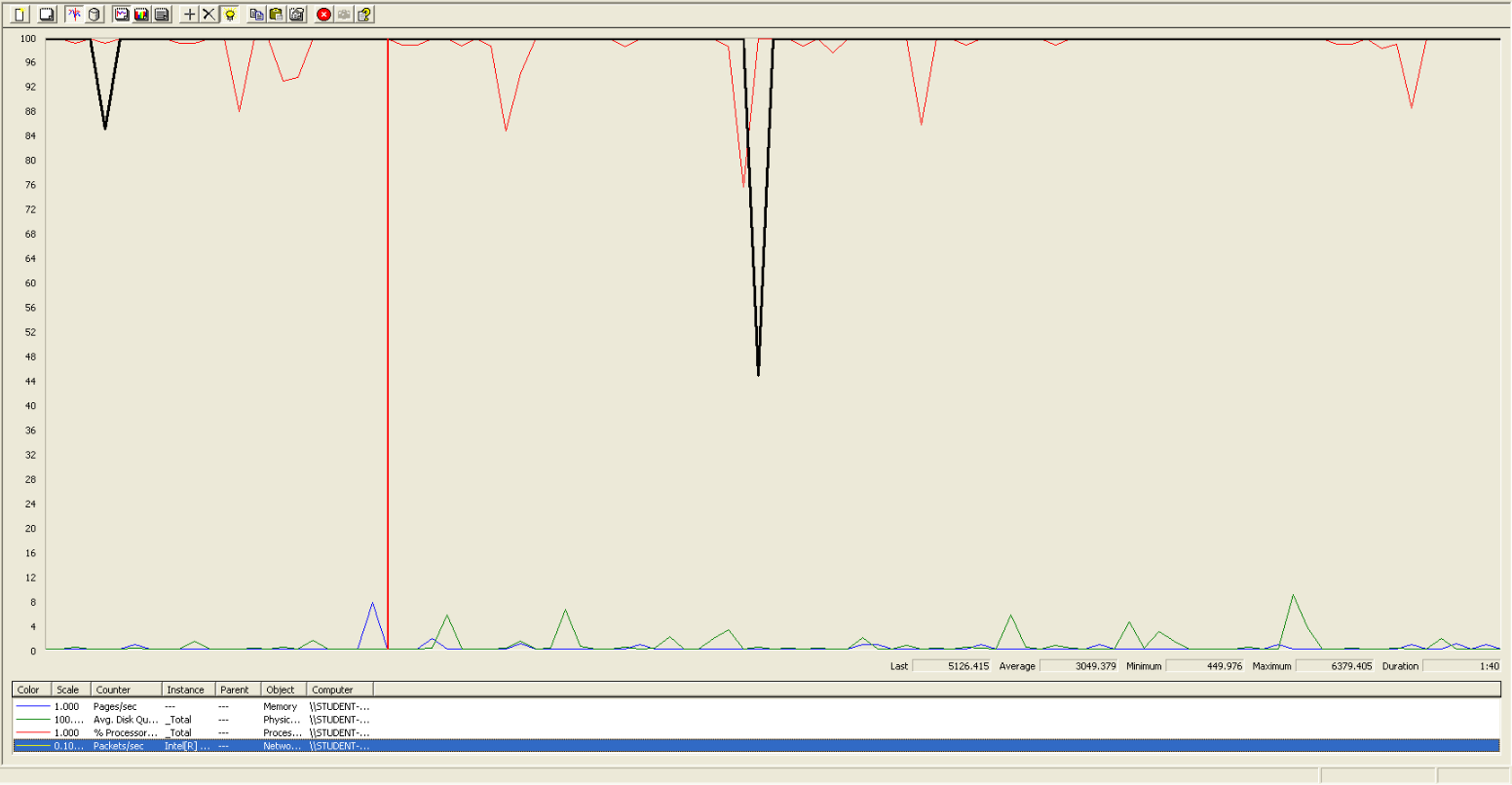
Como podemos observar fijándonos en 90% Line, las direcciones j\_spring\_security\_check, /actor/profile.do y /actor/agent/edit.do generan un tiempo de respuesta mayor a lo normal en este tipo de caso de uso, 456, 368 y 457 ms respectivamente.



En esta tabla, el rendimiento es de aproximadamente 21.800 ejecuciones por minuto (aumenta levemente al aumentar también en poca cantidad los usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Por lo que podemos afirmar que la aplicación soporta entre 180 y 200 usuarios en este case de uso. Ahora estudiaremos a que se deben los tiempos y errores:

Abrimos performance.exe y vemos la tarjeta de red tiene porcentajes muy altos, llegando al 100% en ocasiones, por lo que podemos estar ante un problema de la tarjeta de red.



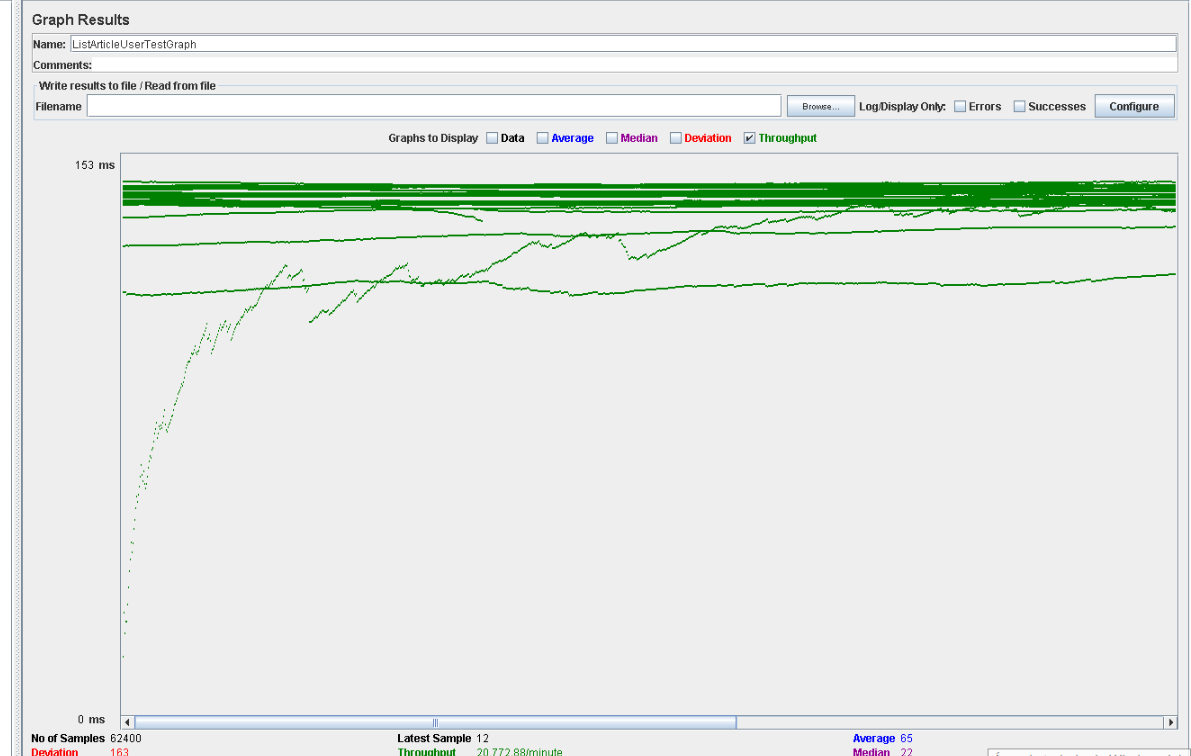
Performance con Packets/sec.

# Caso de uso: listar artículos de un usuario.

En este caso de uso, accedemos a la página, nos logueamos como un usuario, accedemos a la lista de artículos de un usuario y nos deslogueamos.

La configuración en esta ocasión es de 160 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores, aunque las esperas son algo considerables, lo que podemos ver en el Aggregate Report y en el Graph Results:



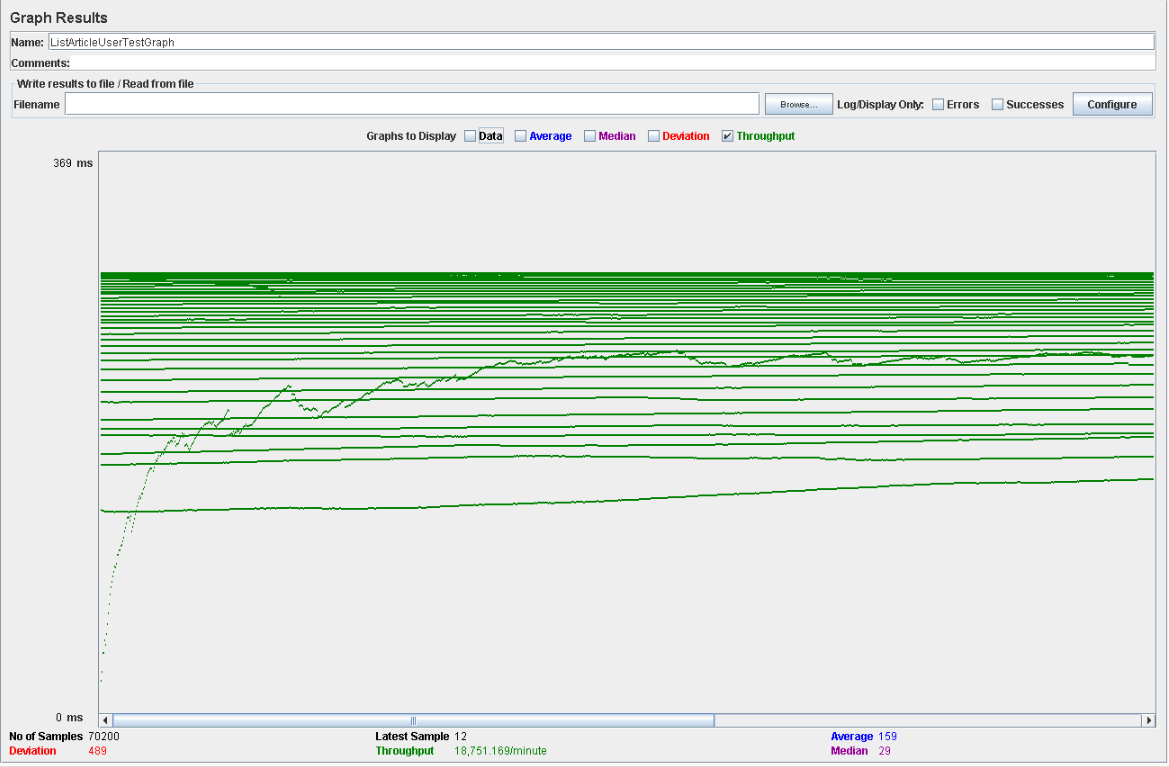


En esta tabla, el rendimiento es de aproximadamente 22.700 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Ahora probamos una configuración de 180 usuarios, 1 ramp-up y 30 acciones cada uno. Las gráficas correspondientes a este test son:



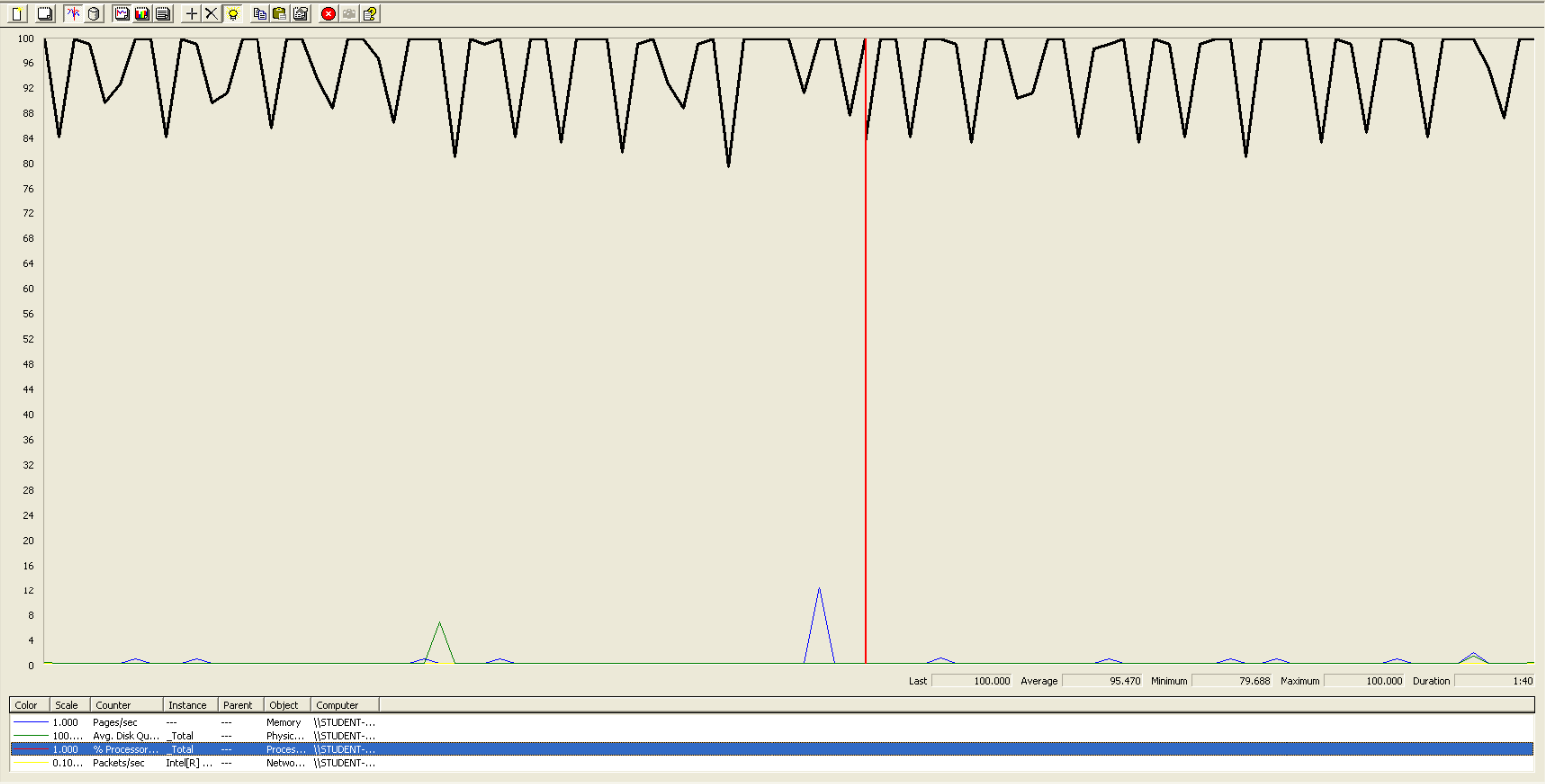
Como podemos observar fijándonos en 90% Line, las direcciones actor/user/list.do, /actor/user/display.do y /article/list.do generan un tiempo de respuesta mayor a lo establecido, 1350, 1312 y 1145 ms respectivamente.



Ya se producen muy leves errores y los tiempos de espera. En esta tabla, el rendimiento es de aproximadamente 18.700 ejecuciones por minuto (desciende al aumentar los usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Por lo que podemos afirmar que la aplicación soporta entre 180 y 200 usuarios en este case de uso. Ahora estudiaremos a que se deben los tiempos y errores:

Abrimos performance.exe y vemos el procesador tiene porcentajes muy altos, llegando al 100% en ocasiones, por lo que podemos estar ante un problema de la CPU.

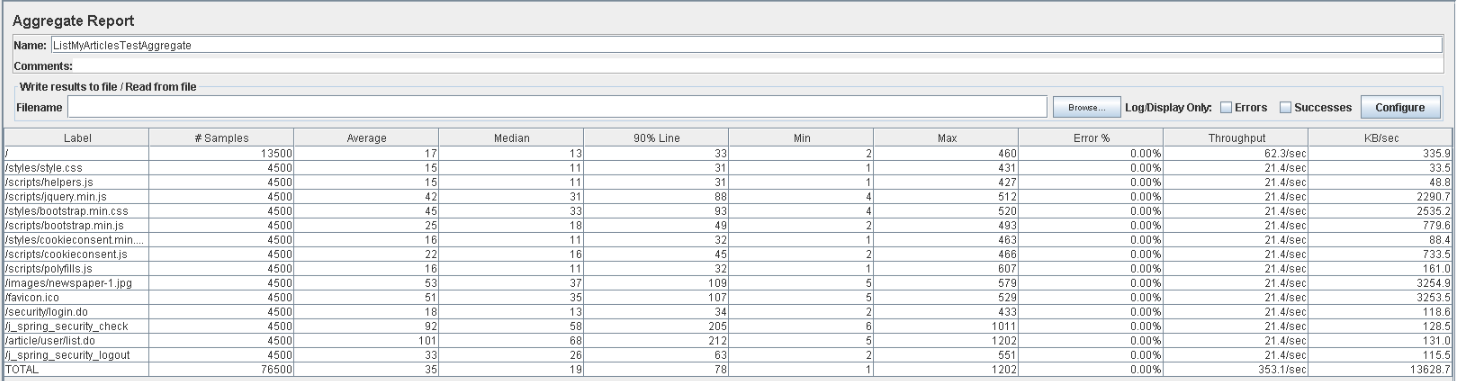


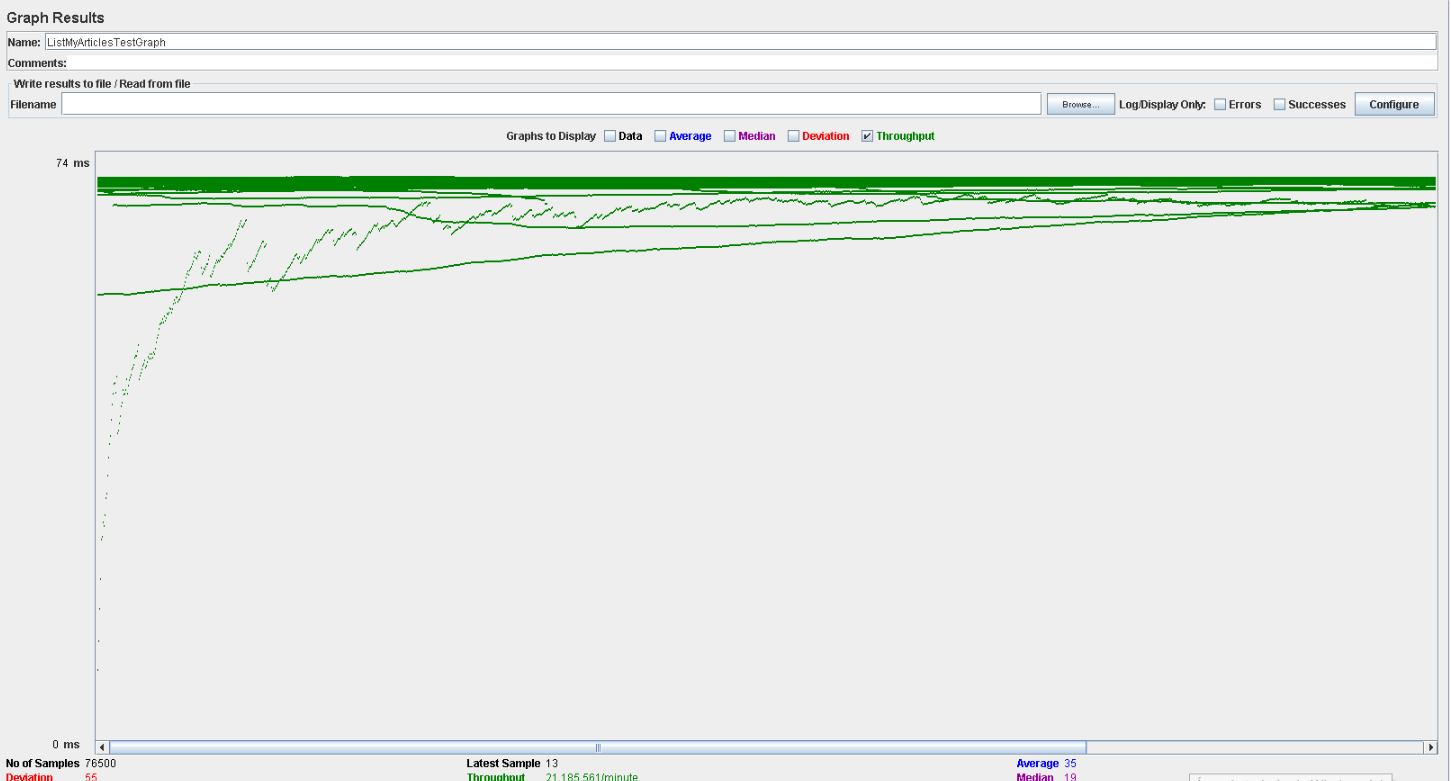
Performance con Processor Time.

# Caso de uso: listar mis artículos.

En este caso de uso, accedemos a la página, nos autentificamos como usuario, pulsamos Mis artículos y nos desautentificamos.

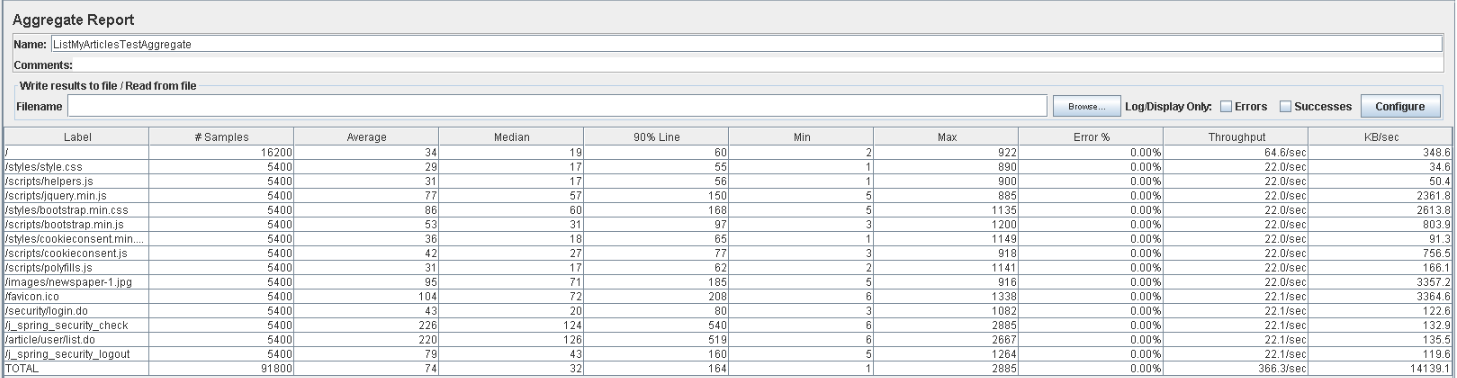
La configuración en esta ocasión es de 150 usuarios, 1 ramp-up y 30 acciones cada uno, produciéndose muy leves errores, pero unos tiempos de espera considerables, los que podemos ver en el Aggregate Report y en el Graph Results:



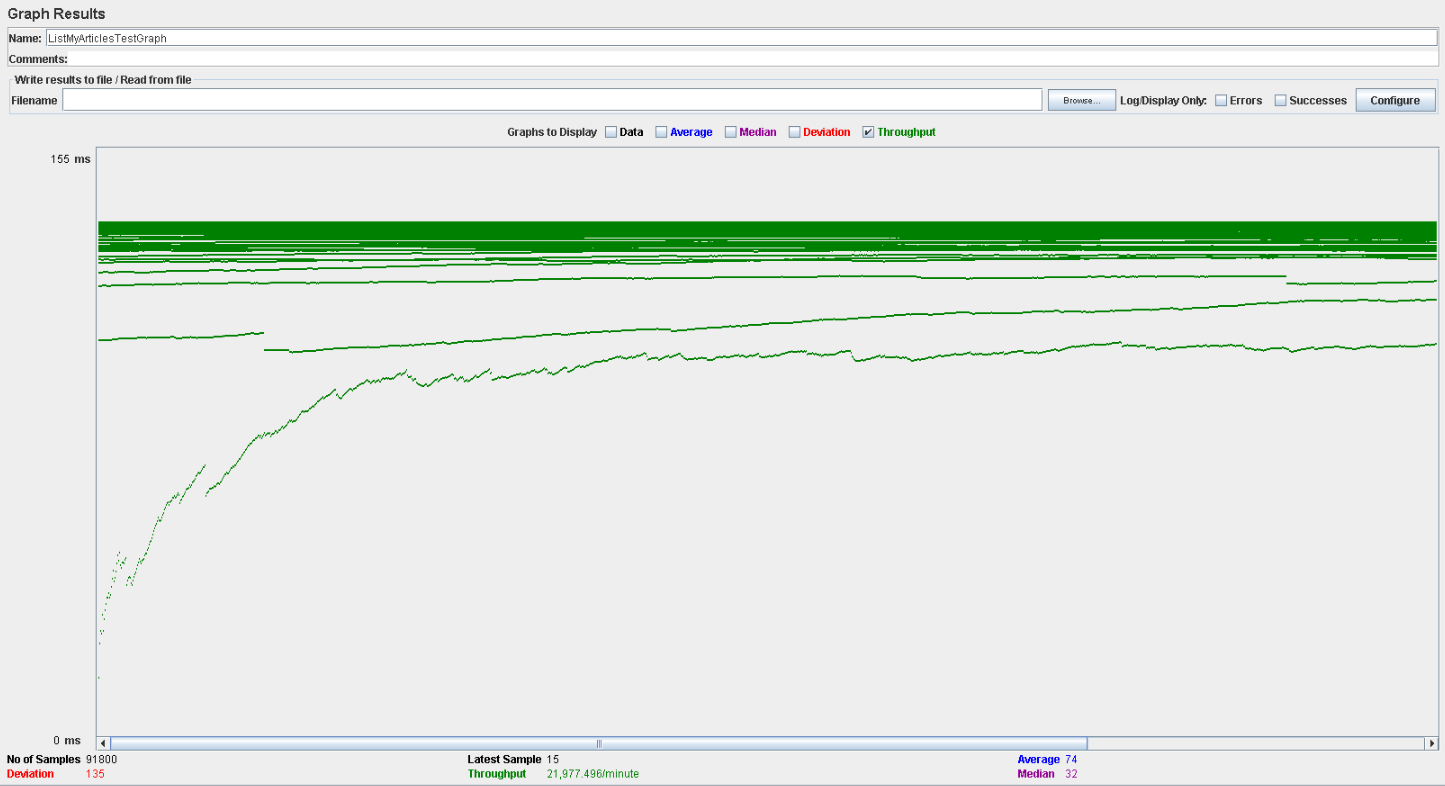


En esta tabla, el rendimiento es de aproximadamente 21.200 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Ahora probamos una configuración de 180 usuarios, 1 ramp-up y 30 acciones cada uno. Las gráficas correspondientes a este test son:



Como podemos observar fijándonos en 90% Line, las direcciones /j\_spring\_security\_check y /article/user/list.do generan un tiempo de respuesta (540 y 519 ms) mayor a lo esperado en un caso de uso de listado, además estos tiempos sumados a otros del tests, generan demasiada espera.



En esta tabla, el rendimiento es de aproximadamente 22.000 ejecuciones por minuto (aumenta brevemente al aumentar los usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Al aumentar el tiempo de espera, este caso de uso soporta entre 150 y 180 usuarios. Ahora estudiaremos a que se deben los tiempos y errores:

Abrimos performance.exe y vemos la CPU tienen porcentajes muy altos, llegando al 100% en ocasiones, por lo que podemos estar ante un problema de CPU, ya que se está utilizando un alto porcentaje de la capacidad del procesador.

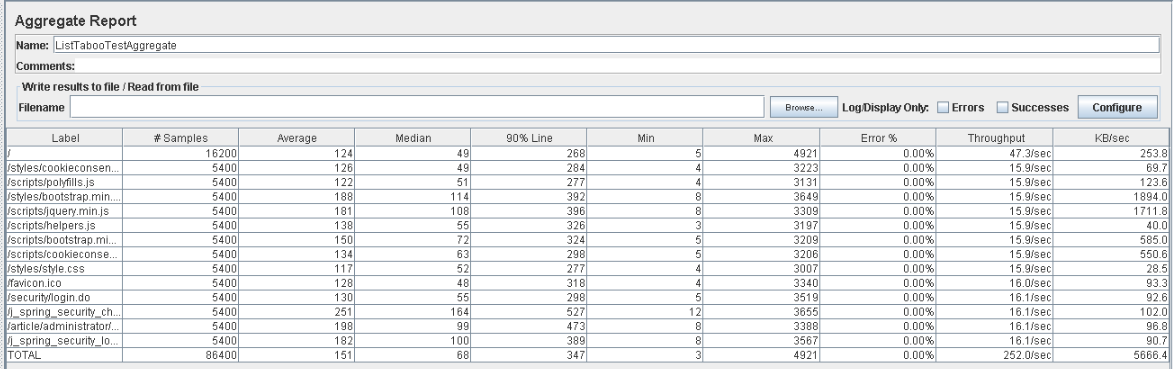


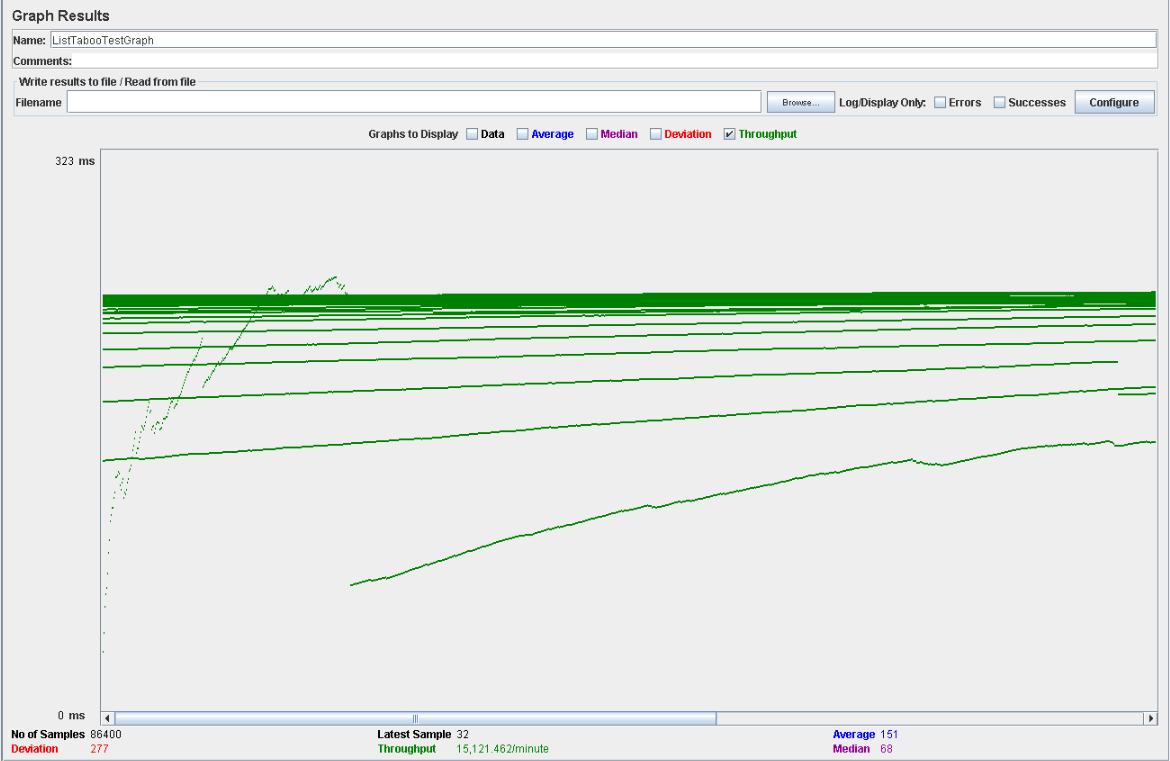
Performance con Processor Time.

# Caso de uso: listar artículos tabús.

En este caso de uso, accedemos a la página, nos logueamos como un administrador, accedemos a la lista de artículos tabús y nos deslogueamos.

La configuración en esta ocasión es de 180 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores, que podemos ver en el Aggregate Report y en el Graph Results:

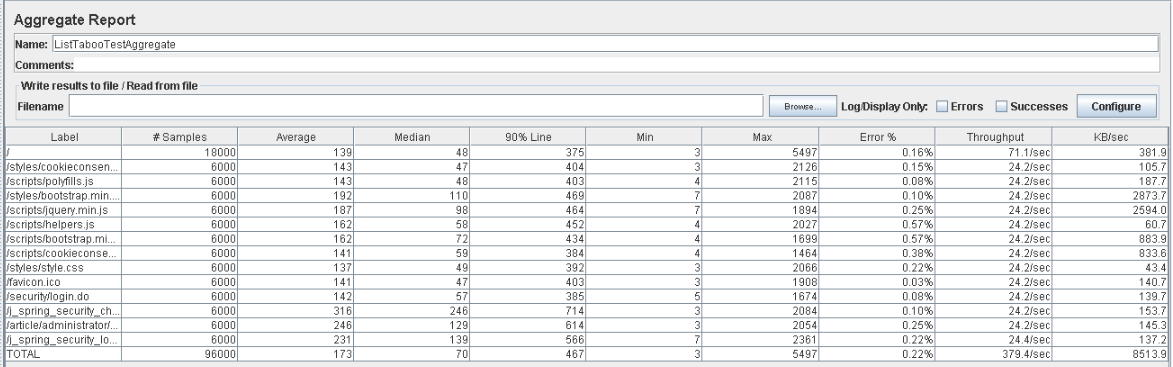


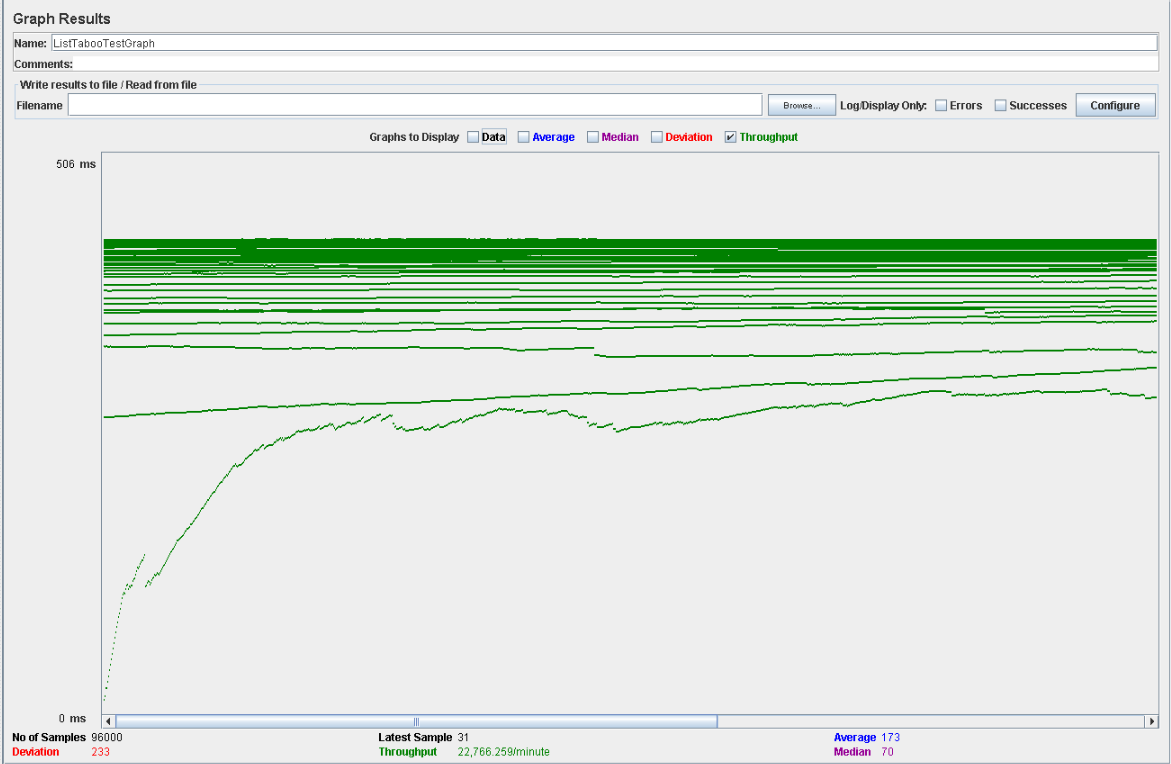


En esta tabla, el rendimiento es de aproximadamente 15.100 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

En un caso real, no sería normal encontrar un número tan alto de administradores en la aplicación. No obstante, hemos considerado suficiente ver un valor alto para así tener una estimación bastante buena tanto de los errores como de los tiempos de ejecución.

Ahora probamos una configuración de 200 usuarios, 1 ramp-up y 30 acciones cada uno, obteniéndose ahora errores, por lo que entre 180 y 200 son los usuarios permitidos, aunque en un caso de uso real no existirían tantos administradores al mismo tiempo. Las gráficas correspondientes a este test son:





En esta tabla, el rendimiento es de aproximadamente 22.700 ejecuciones por minuto (aumentando a pesar de los errores). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

En este caso, fijándonos en 90% Line, las direcciones /j\_spring\_security\_check y /article/administrator/listTaboo.do generan un tiempo de respuesta mayor al resto, aunque en todas las direcciones se produce unos tiempos de respuesta demasiado altos.

Ahora estudiaremos a que se deben los tiempos y errores:

Abrimos performance.exe y vemos la CPU tienen porcentajes muy altos, llegando al 100% en ocasiones, por lo que podemos estar ante un problema de CPU, ya que se está utilizando un alto porcentaje de la capacidad del procesador.

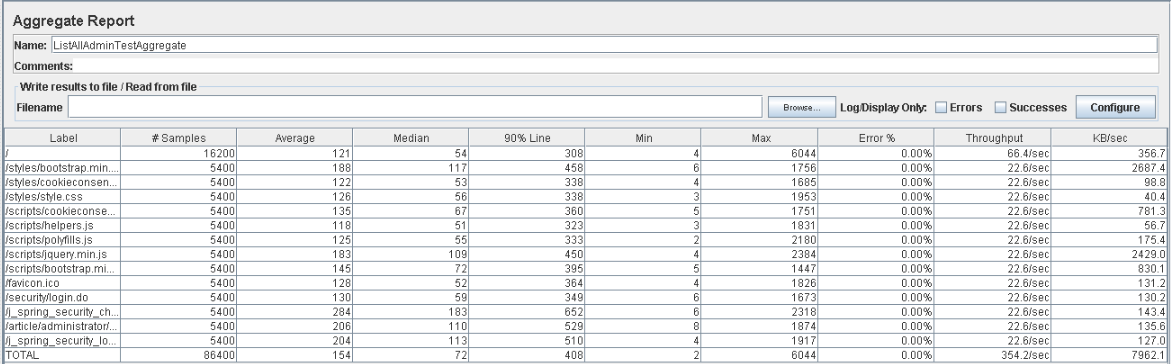


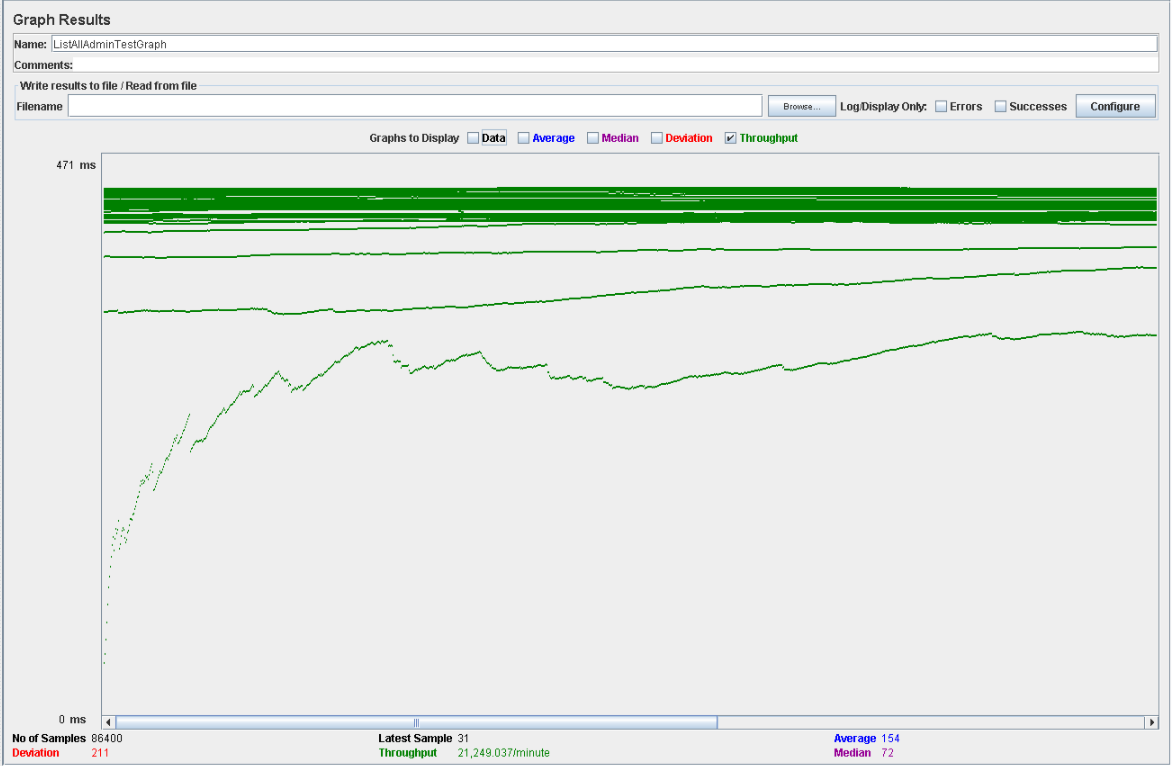
Performance con Processor Time.

# Caso de uso: listar todos los artículos administrador.

En este caso de uso, accedemos a la página, nos logueamos como un administrador, accedemos a la lista de artículos completa de un administrador y nos deslogueamos.

La configuración en esta ocasión es de 180 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores, pero esperas demasiado altas, que podemos ver en el Aggregate Report y en el Graph Results:

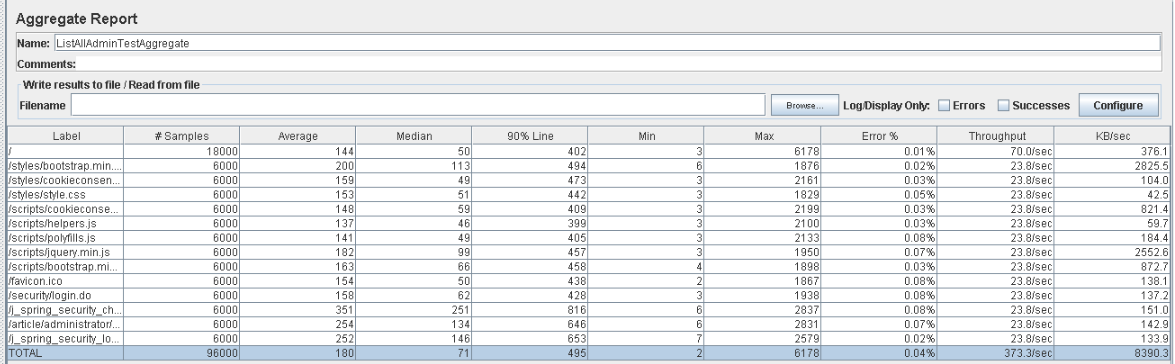


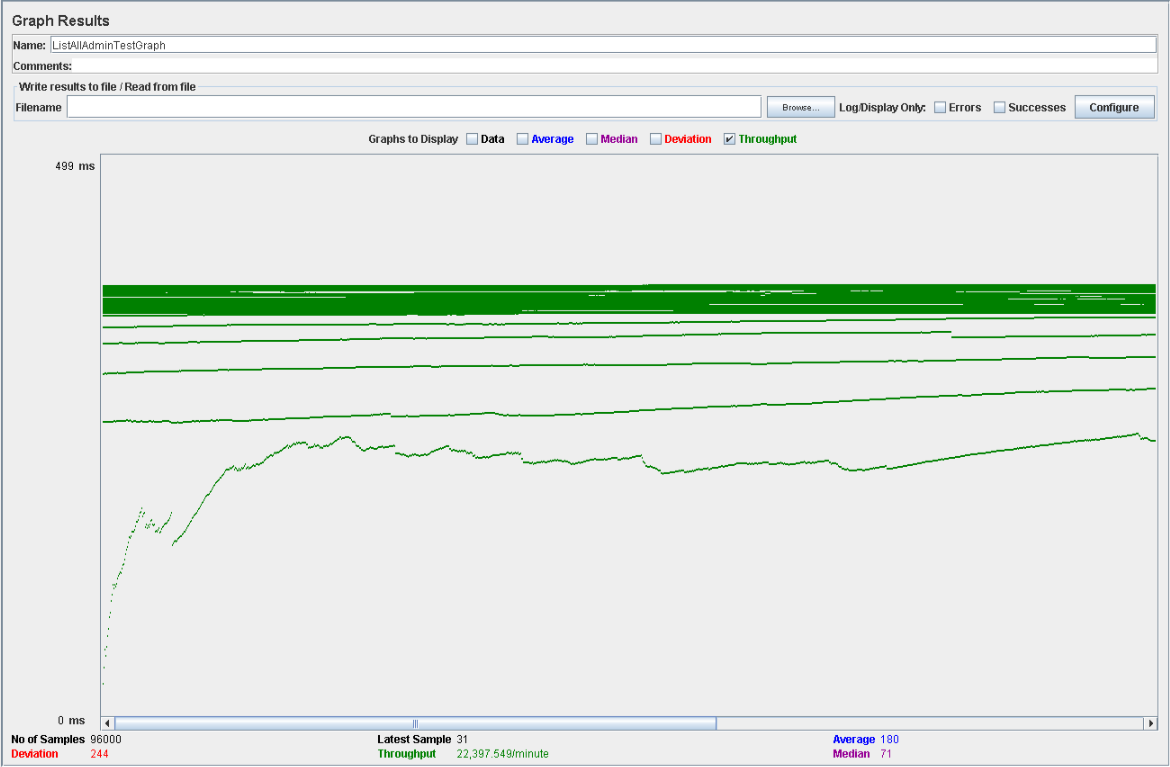


En esta tabla, el rendimiento es de aproximadamente 21.200 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

En un caso real, no sería normal encontrar un número tan alto de administradores en la aplicación. No obstante, hemos considerado suficiente ver un valor alto para así tener una estimación bastante buena tanto de los errores como de los tiempos de ejecución.

Ahora probamos una configuración de 200 usuarios, 1 ramp-up y 30 acciones cada uno, obteniéndose ahora errores, por lo que entre 180 y 200 son los usuarios permitidos, aunque en un caso de uso real no existirían tantos administradores al mismo tiempo. Las gráficas correspondientes a este test son:





En esta tabla, el rendimiento es de aproximadamente 22.400 ejecuciones por minuto (aumentando brevemente al aumentar los usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

En este caso, fijándonos en 90% Line, las direcciones /j\_spring\_security\_check, /article/administrator/list.do y j\_spring\_security\_logout generan un tiempo de respuesta mayor al resto, aunque en todas las direcciones se produce unos tiempos de respuesta demasiado altos.

Por lo que podemos afirmar que la aplicación soporta entre 180 y 200 usuarios en este case de uso. Ahora estudiaremos a que se deben los tiempos y errores:

Abrimos performance.exe y vemos el procesador tiene porcentajes muy altos, llegando al 100% en ocasiones, por lo que podemos estar ante un problema de la CPU.

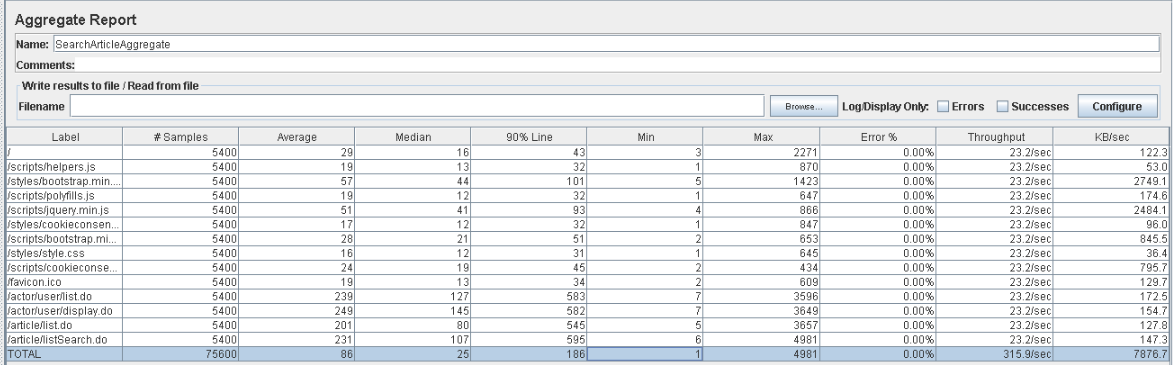


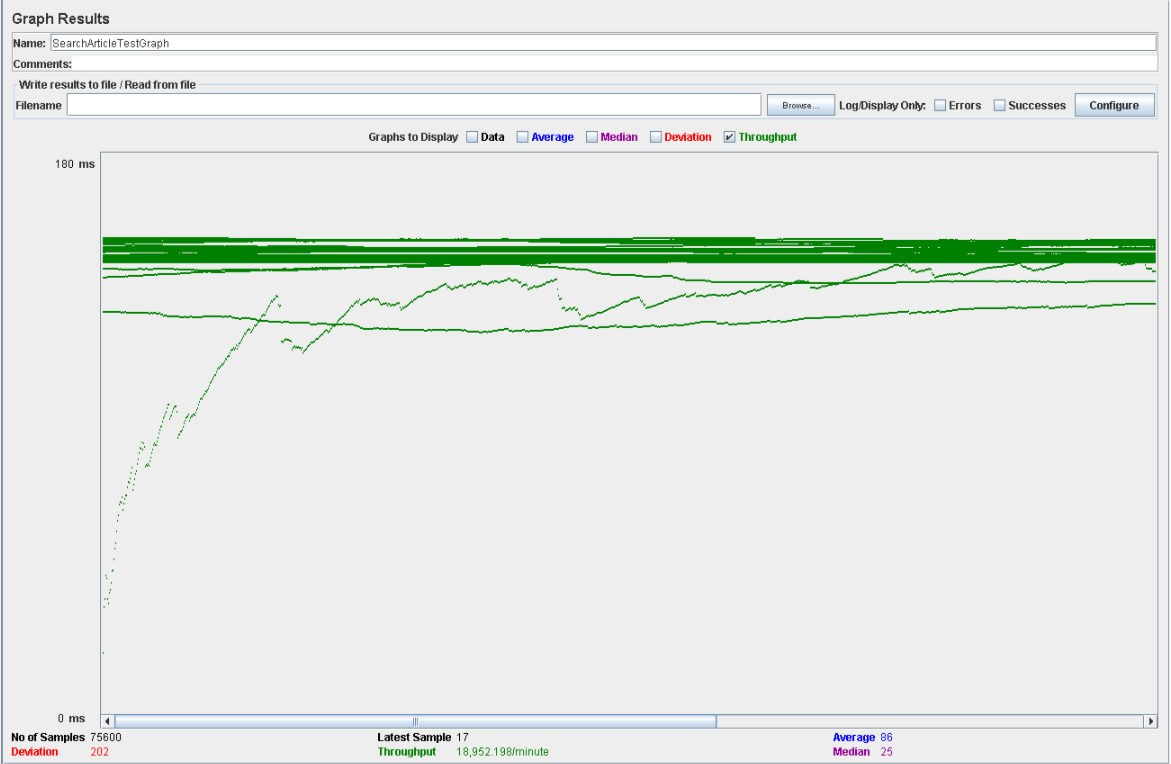
Performance con Processor Time.

# Caso de uso: buscar artículos.

En este caso de uso, accedemos a la aplicación, accedemos a la lista de usuarios, luego al display de uno de ellos, accedemos a Leer más artículos que nos llevará a la vista del listado de artículos de ese usuario, buscamos por un término clave y pasamos a la vista con los resultados. Este caso de uso se repite con todas las vistas de listado de artículos.

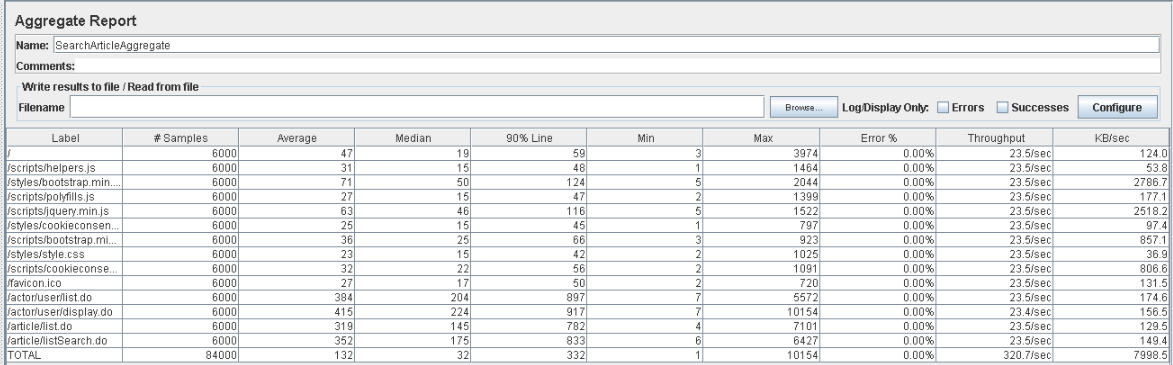
La configuración en esta ocasión es de 180 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores ni esperas demasiado altas, que podemos ver en el Aggregate Report y en el Graph Results:

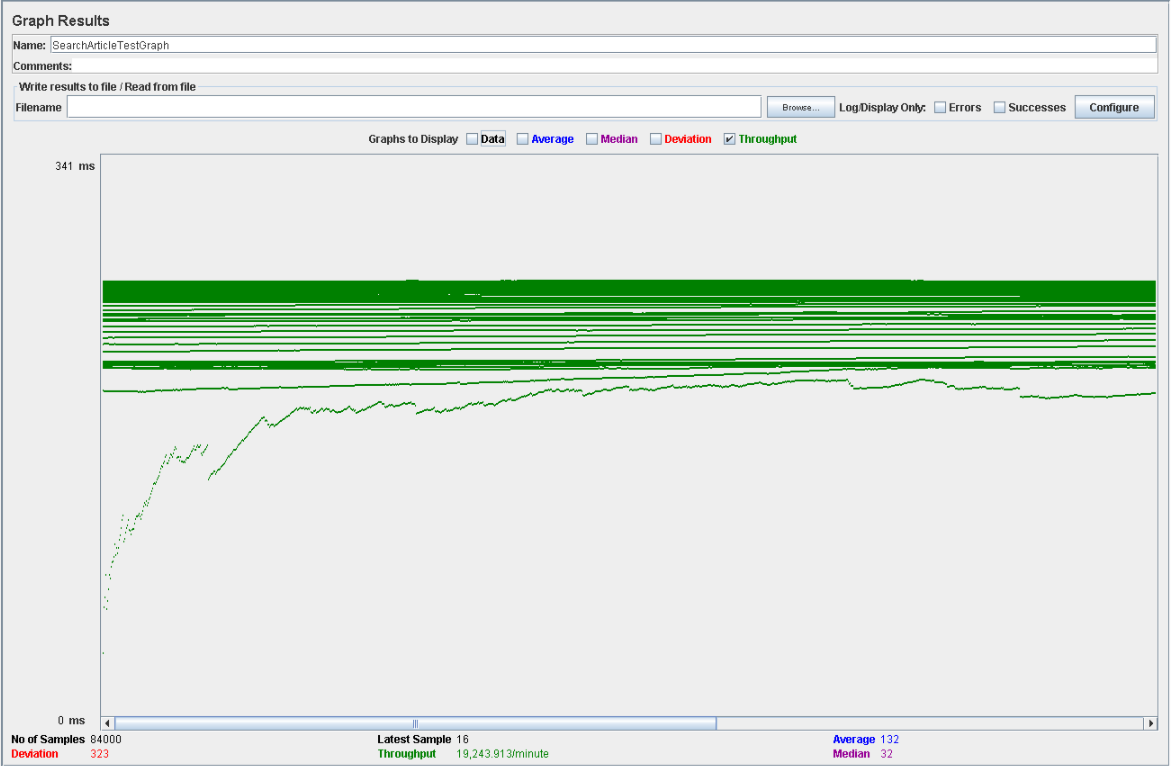




En esta tabla, el rendimiento es de aproximadamente 19.900 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Ahora probamos una configuración de 200 usuarios, 1 ramp-up y 30 acciones cada uno, obteniéndose ahora esperas demasiado altas, por lo que entre 180 y 200 son los usuarios permitidos. Las gráficas correspondientes a este test son:



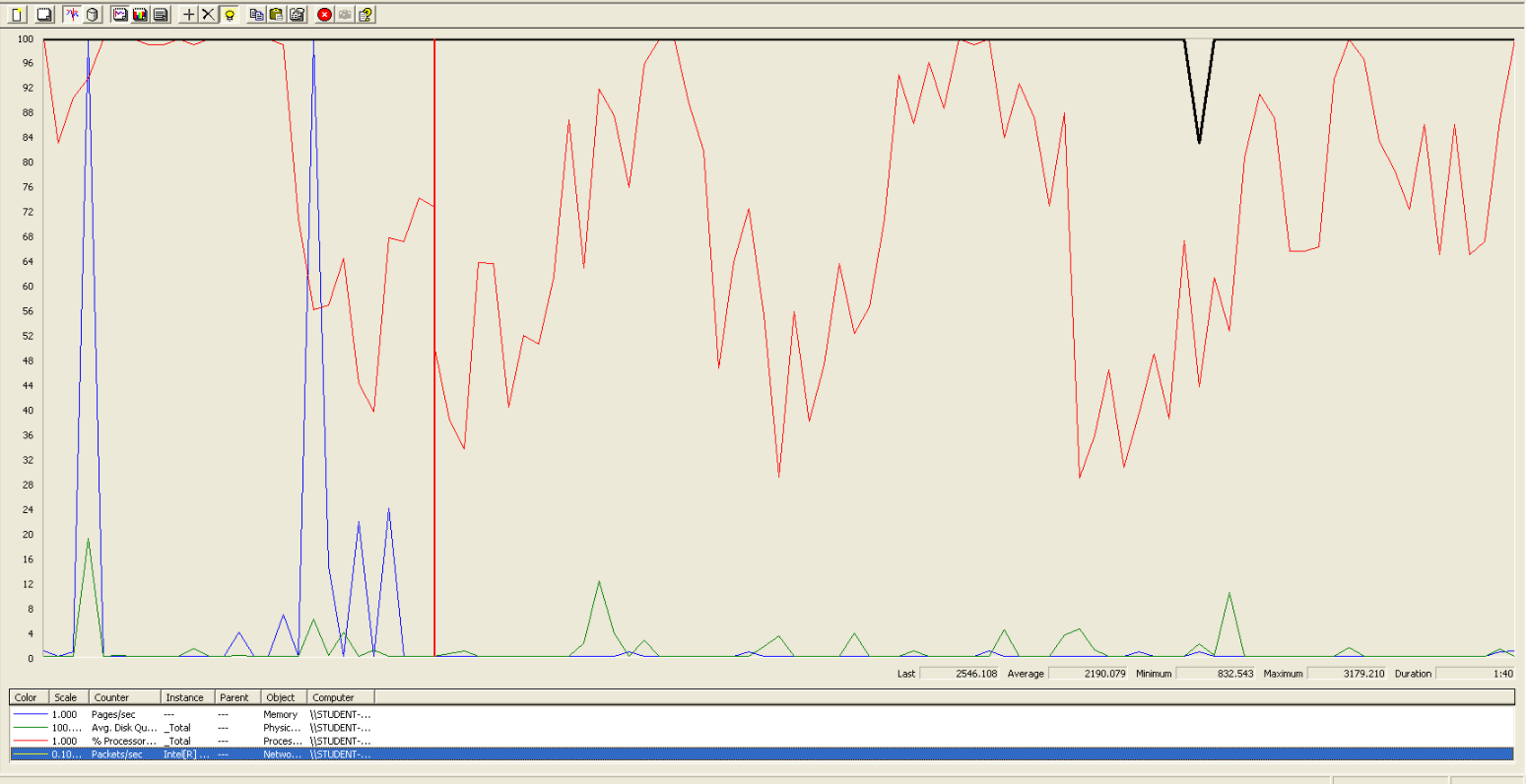


En esta tabla, el rendimiento es de aproximadamente 19.200 ejecuciones por minuto (descendiendo brevemente al aumentar los usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

En este caso, fijándonos en 90% Line, las direcciones /actor/user/list.do, /actor/user/display.do, /article/list.do y /article/listSearch.do generan unos tiempos de respuesta demasiados altos (897, 917, 782 y 833 ms respectivamente).

Ahora estudiaremos a que se deben los errores:

Abrimos performance.exe y vemos que la tarjeta de red tiene porcentajes muy altos, llegando al 100% en ocasiones, por lo que el problema se localizará en ella.



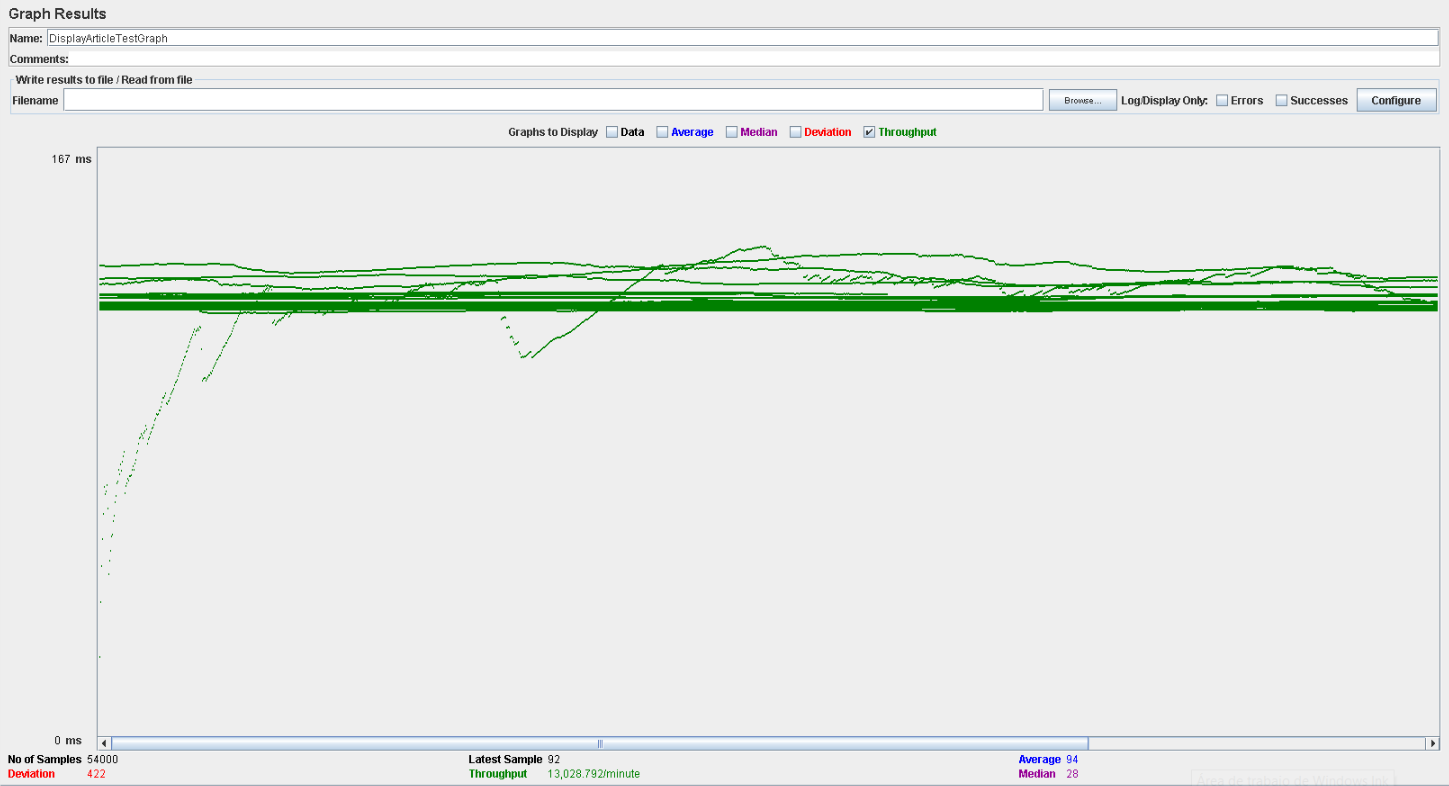
Performance con Packets/sec.

# Caso de uso: desplegar artículo.

En este caso de uso, accedemos a la aplicación, accedemos a la lista de periódicos, luego al display de uno de ellos y accedemos al display de un artículo.

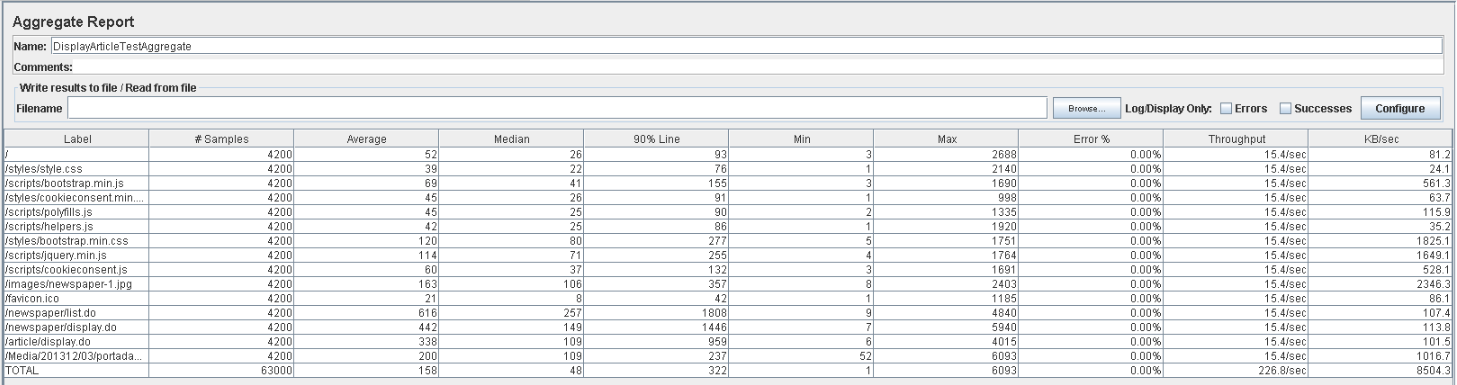
La configuración en esta ocasión es de 120 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores ni esperas demasiado altas, que podemos ver en el Aggregate Report y en el Graph Results:

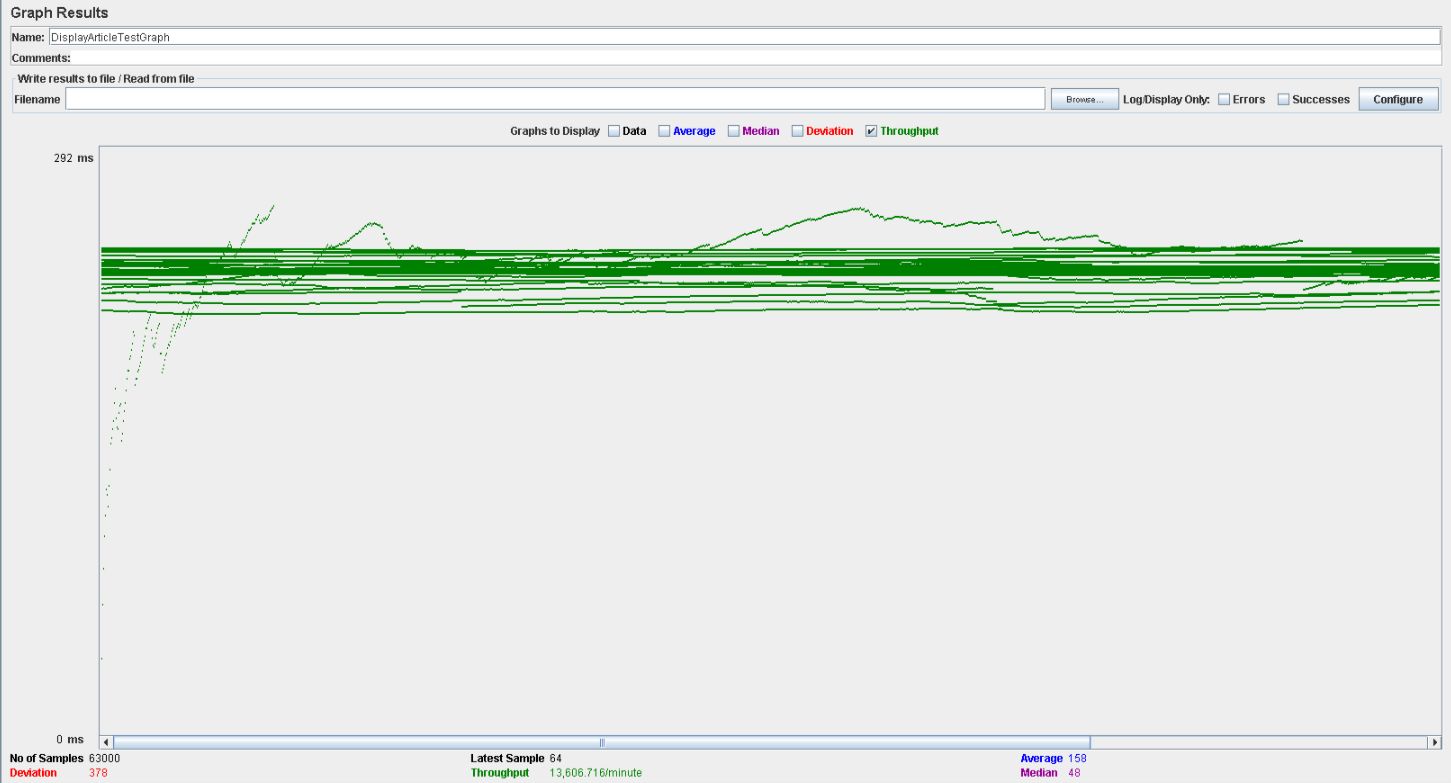




En esta tabla, el rendimiento es de aproximadamente 13.000 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Ahora probamos una configuración de 140 usuarios, 1 ramp-up y 30 acciones cada uno, obteniéndose ahora esperas demasiado altas además de leves errores, por lo que entre 160 y 180 son los usuarios permitidos. Las gráficas correspondientes a este test son:





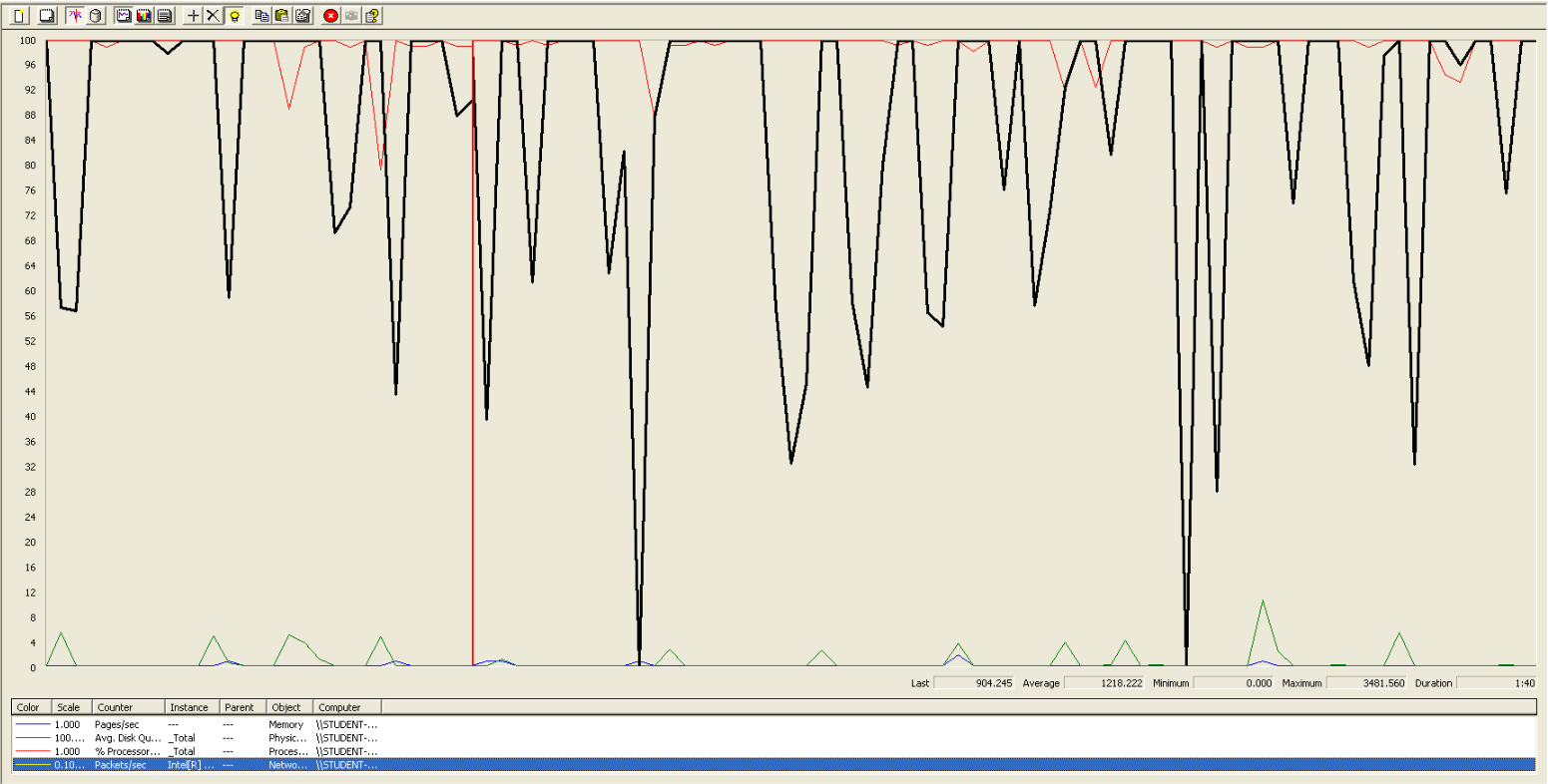
En esta tabla, el rendimiento es de aproximadamente 13.600 ejecuciones por minuto (aumenta brevemente al aumentar los usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

En este caso, fijándonos en 90% Line, las direcciones /newspaper/list.do, /newspaper/display.do y /article/display.do generan unos tiempos de respuesta demasiados altos: 1808, 1446 y 959 ms respectivamente.

Además, la dirección /Media/201312/03/portadas-periodicos--644x362.jpg es una imagen que lastra el rendimiento del sistema porque no está en nuestro servidor.

Ahora estudiaremos a que se deben los errores:

Abrimos performance.exe y vemos que la tarjeta de red tiene porcentajes muy altos, llegando al 100% en ocasiones, por lo que el problema se localizará en ella.



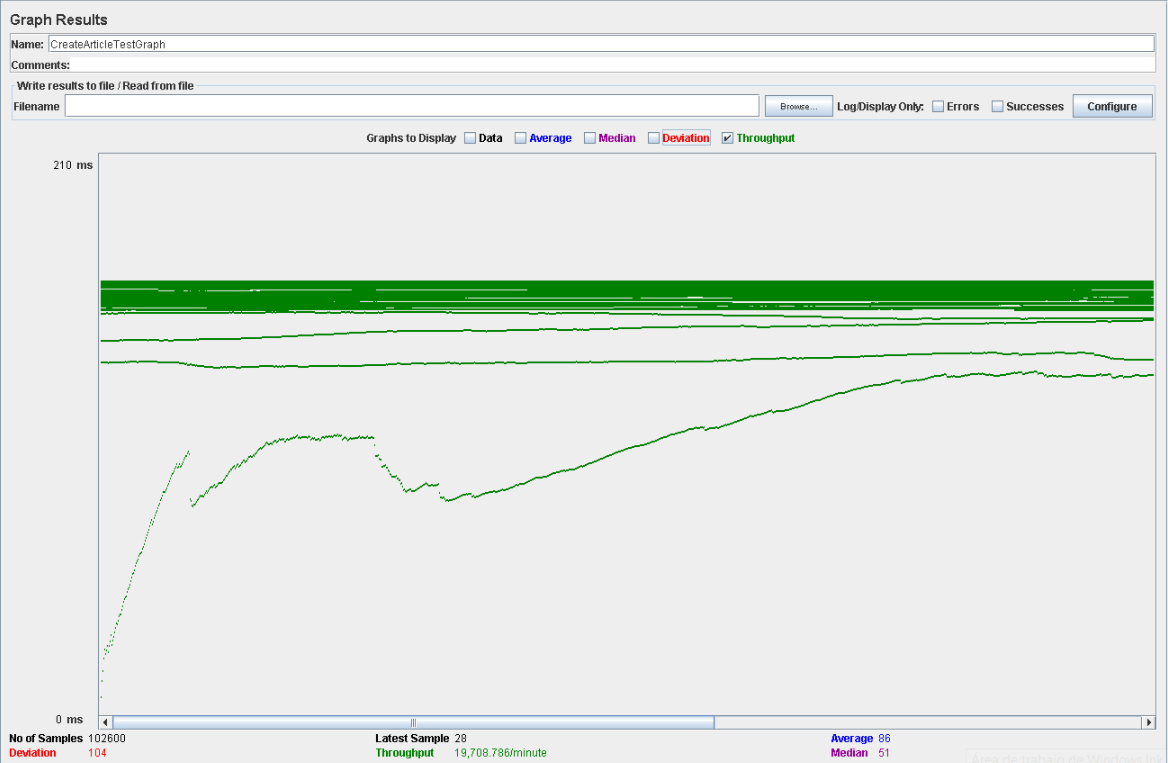
Performance con Packets/sec.

# Caso de uso: crear artículo.

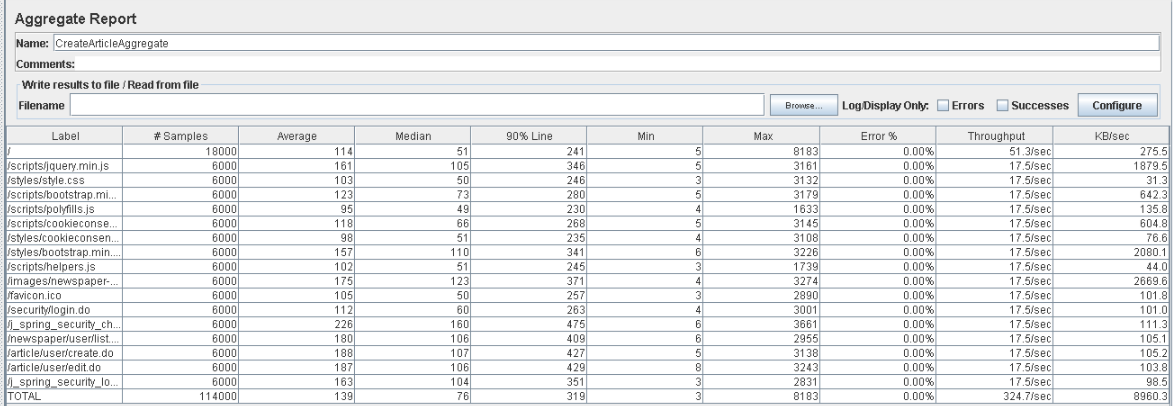
En este caso de uso, accedemos a la página, nos logueamos como un cliente, accedemos a la lista de periódicos, seleccionamos crear artículo, rellenamos el formulario, guardamos y nos deslogueamos.

La configuración en esta ocasión es de 180 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores ni tiempos de espera excesivamente altos, que podemos ver en el Aggregate Report y en el Graph Results.

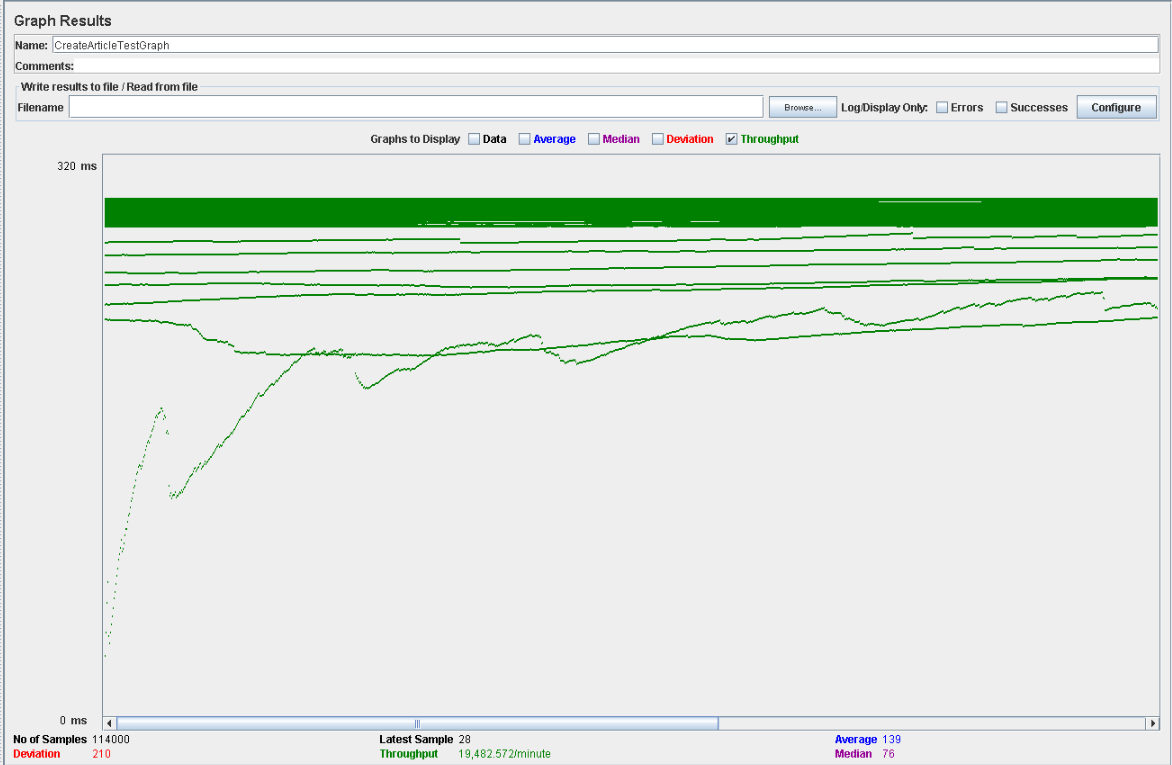


En esta tabla, el rendimiento es de aproximadamente 19.800 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Ahora, la configuración es de 200 usuarios, 1 ramp-up y 30 acciones cada uno, produciéndose esperas excesivas en algunos casos.



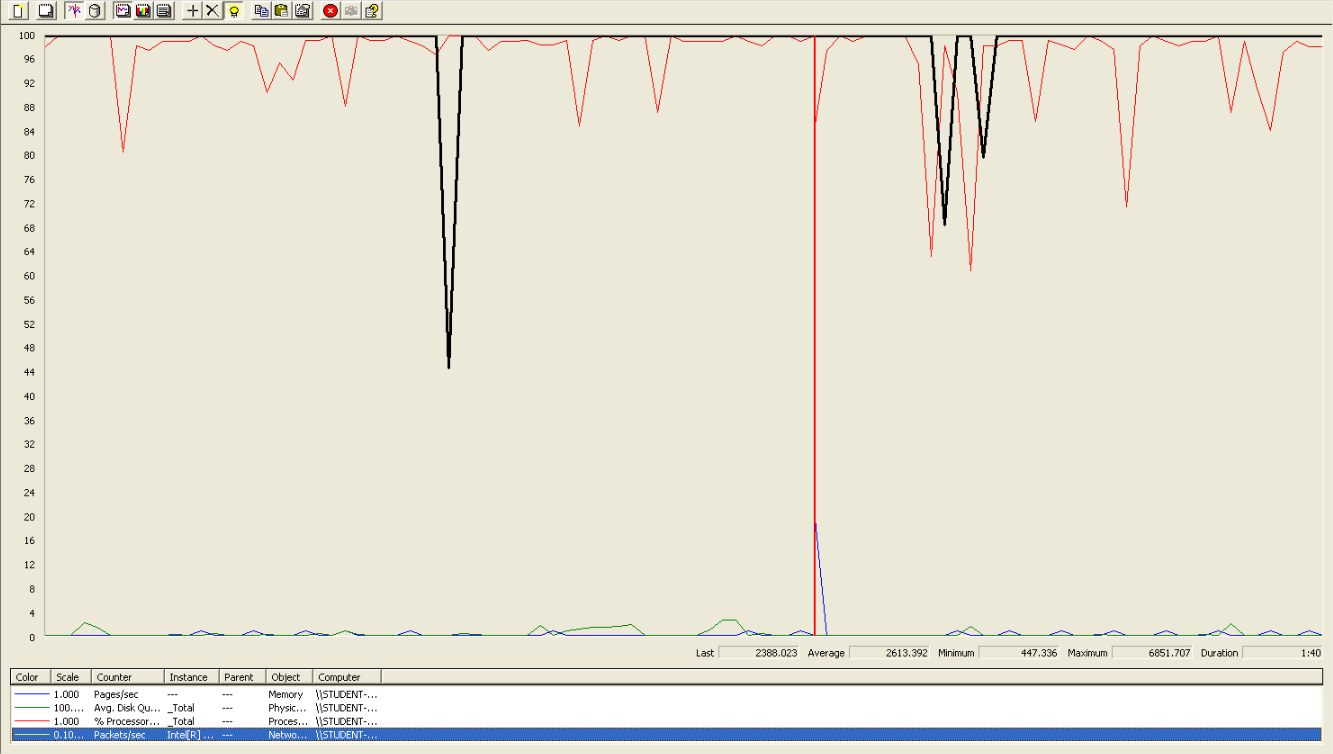
Como observamos fijándonos en 90% Line, las direcciones j\_spring\_security\_check, /newspaper/user/list.do, /article/user/create.do y /article/user/edit.do, generan un tiempo de respuesta mayor al resto, 475 ms, 409 ms, 427 ms y 429 ms respectivamente.



En esta tabla, el rendimiento es de aproximadamente 19.400 ejecuciones por minuto (desciende al aumentar el número de usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Como con 200 usuarios las esperas son excesivas, este caso de uso soporta entre 180 y 200 usuarios.

Ahora estudiaremos a que se deben los tiempos y errores:

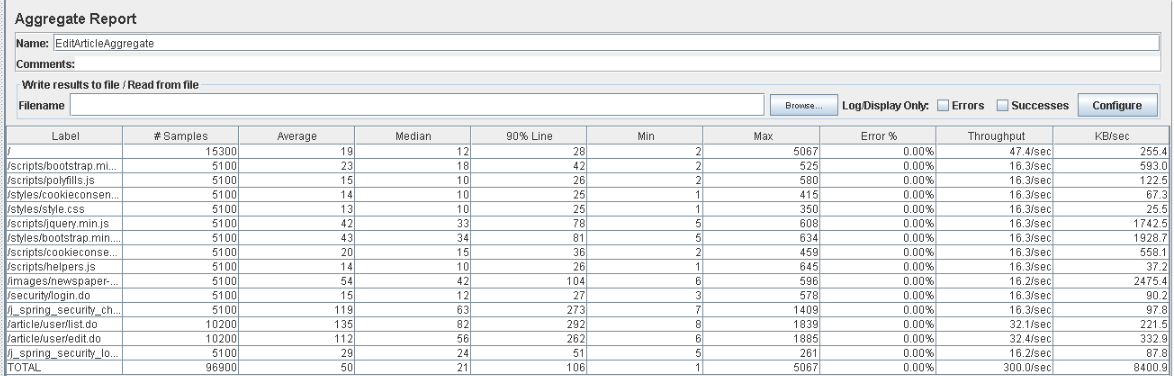
Abrimos performance.exe y vemos la tarjeta de red tienen porcentajes muy altos, llegando al 100% en ocasiones, por lo que podemos estar ante un problema de la tarjeta de red. 

Performance con Packets/sec.

# Caso de uso: editar un artículo.

En este caso de uso, accedemos a la página, nos logueamos como un usuario, accedemos a nuestros artículos, accedemos a la vista para editar, editamos algún parámetro, guardamos y nos deslogueamos.

Hemos establecido el número de hilos máximo acorde con las prestaciones de nuestros ordenadores a 170 usuarios, 1 ramp-up y 30 loop count, para este caso de uso. Con esto, obtenemos las gráficas Aggregate Report y Graph Results:

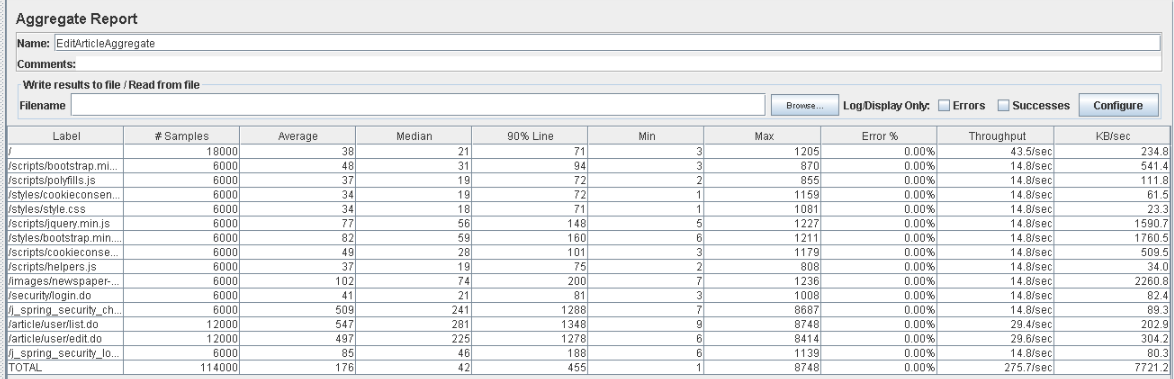


Como podemos observar fijándonos en 90% Line, las direcciones /article/list.do y /article/user/edit.do generan un tiempo de respuesta mayor al resto pero sin sobrepasar los límites, 292 ms y 262 ms respectivamente.

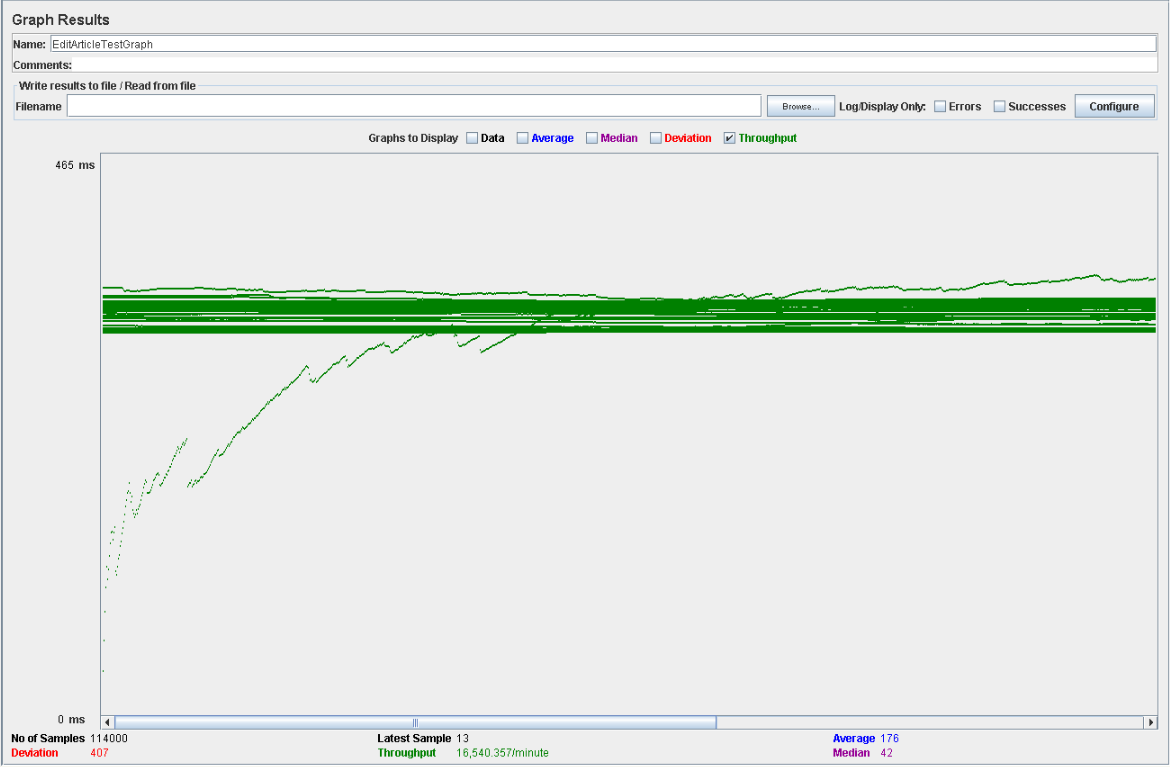


En esta tabla, el rendimiento es de aproximadamente 18.000 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Ahora, la configuración es de 200 usuarios, 1 ramp-up y 30 acciones cada uno, obteniéndose esperas demasiado altas.



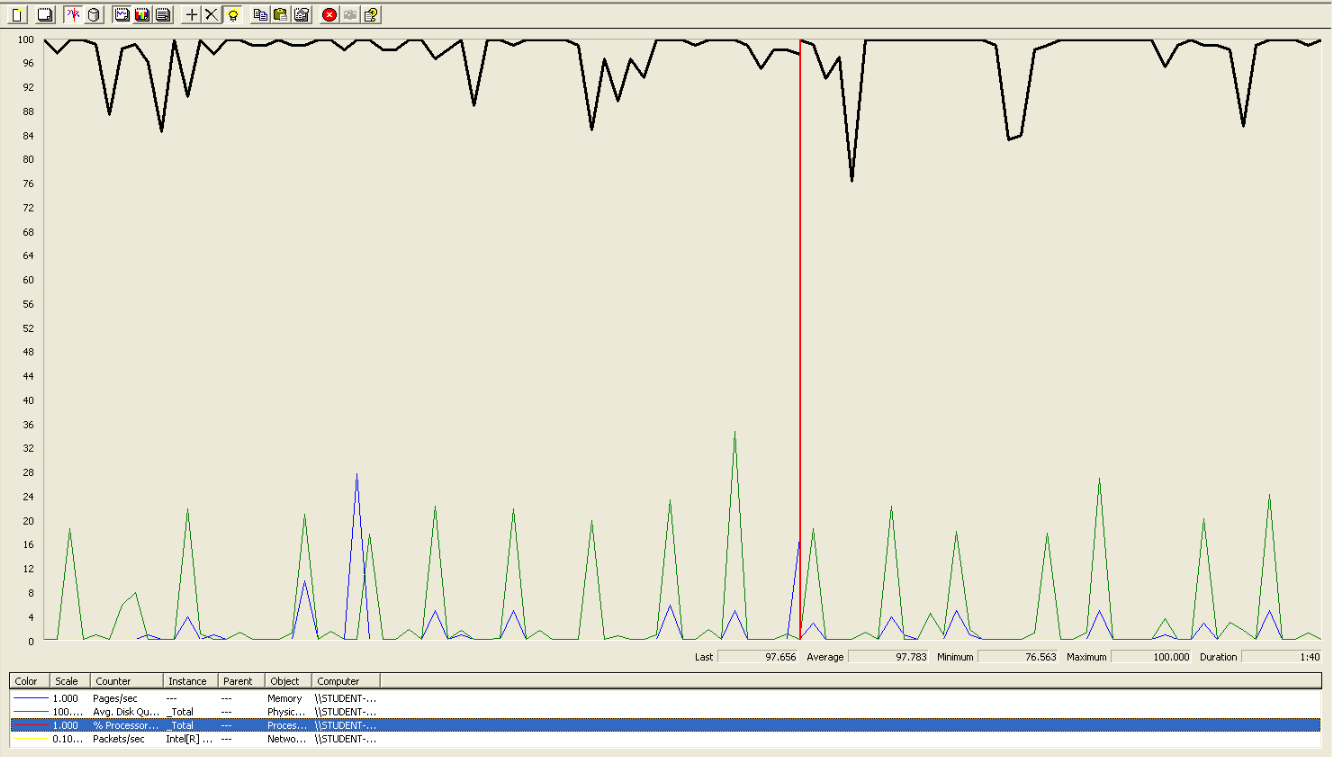
Como observamos fijándonos en 90% Line, las direcciones j\_spring\_security\_check, /article/user/list.do y /article/user/edit.do, generan un tiempo de respuesta mayor al resto, 1288 ms, 1348 ms y 1278 ms respectivamente.



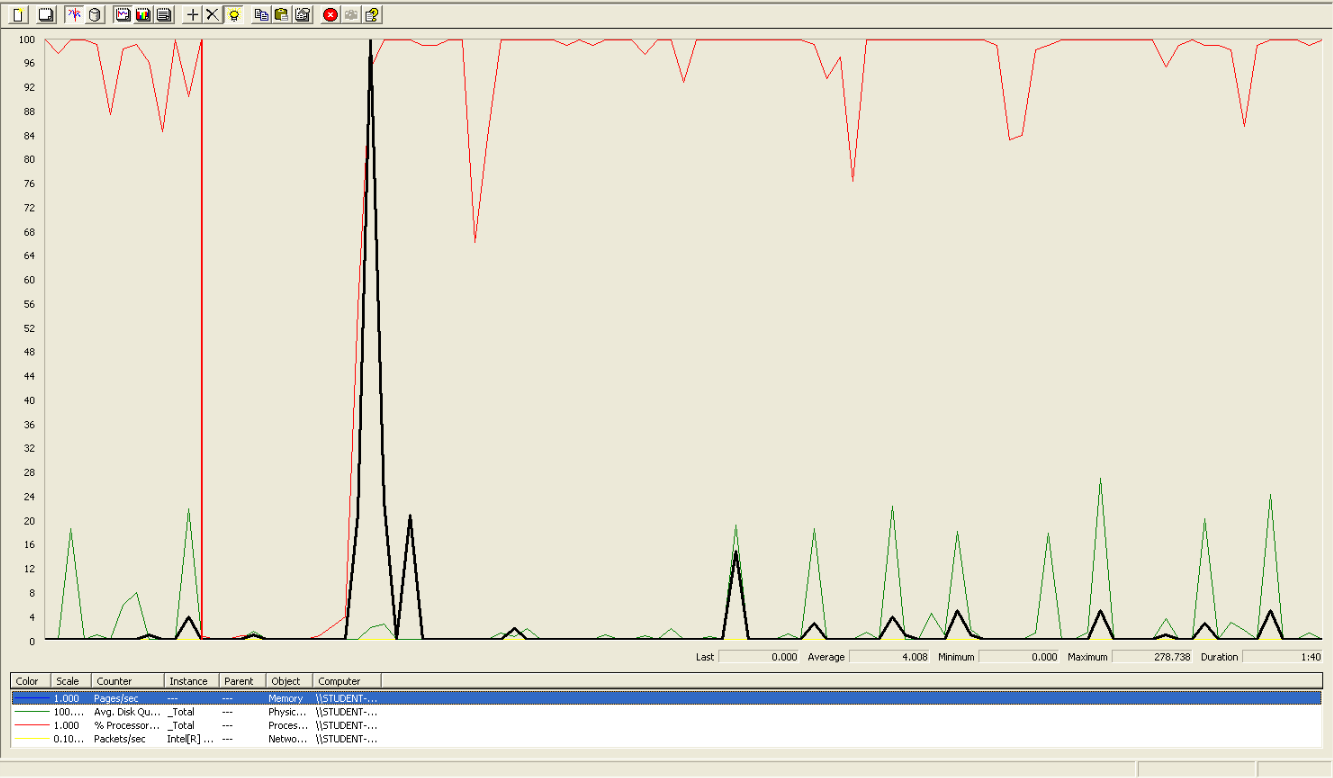
En esta tabla, el rendimiento es de aproximadamente 16.500 ejecuciones por minuto (desciende al aumentar los usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Como con 200 usuarios las esperas son excesivas, este caso de uso soporta entre 170 y 200 usuarios.

Para encontrar origen a estos resultados, revisamos performance.exe, obteniendo lo siguiente:



Observamos que podemos estar ante un problema de CPU, ya que se está utilizando un alto porcentaje de la capacidad del procesador.



No hay fallos de memoria ya que no destaca altos porcentajes en este caso.



La tarjeta de red no supone un problema.

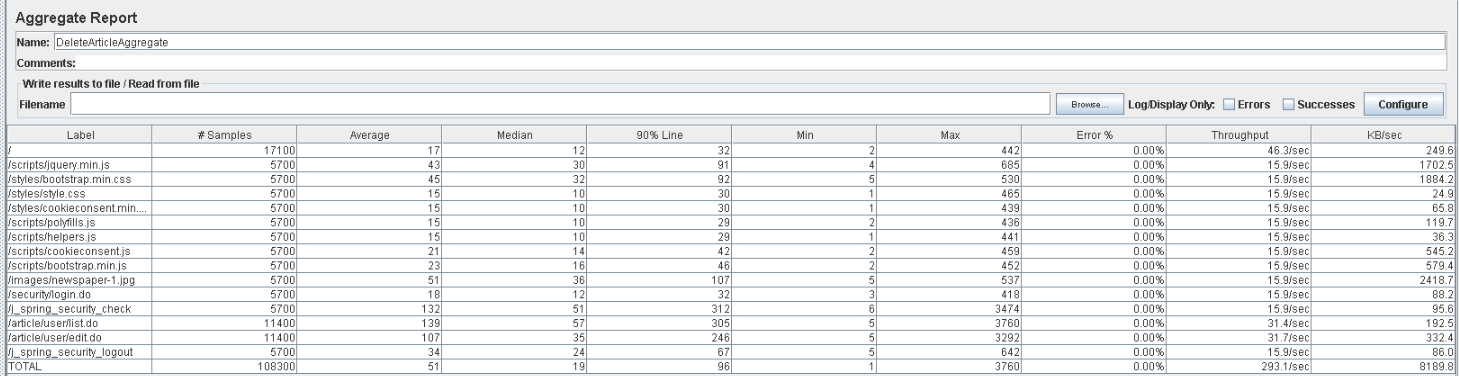


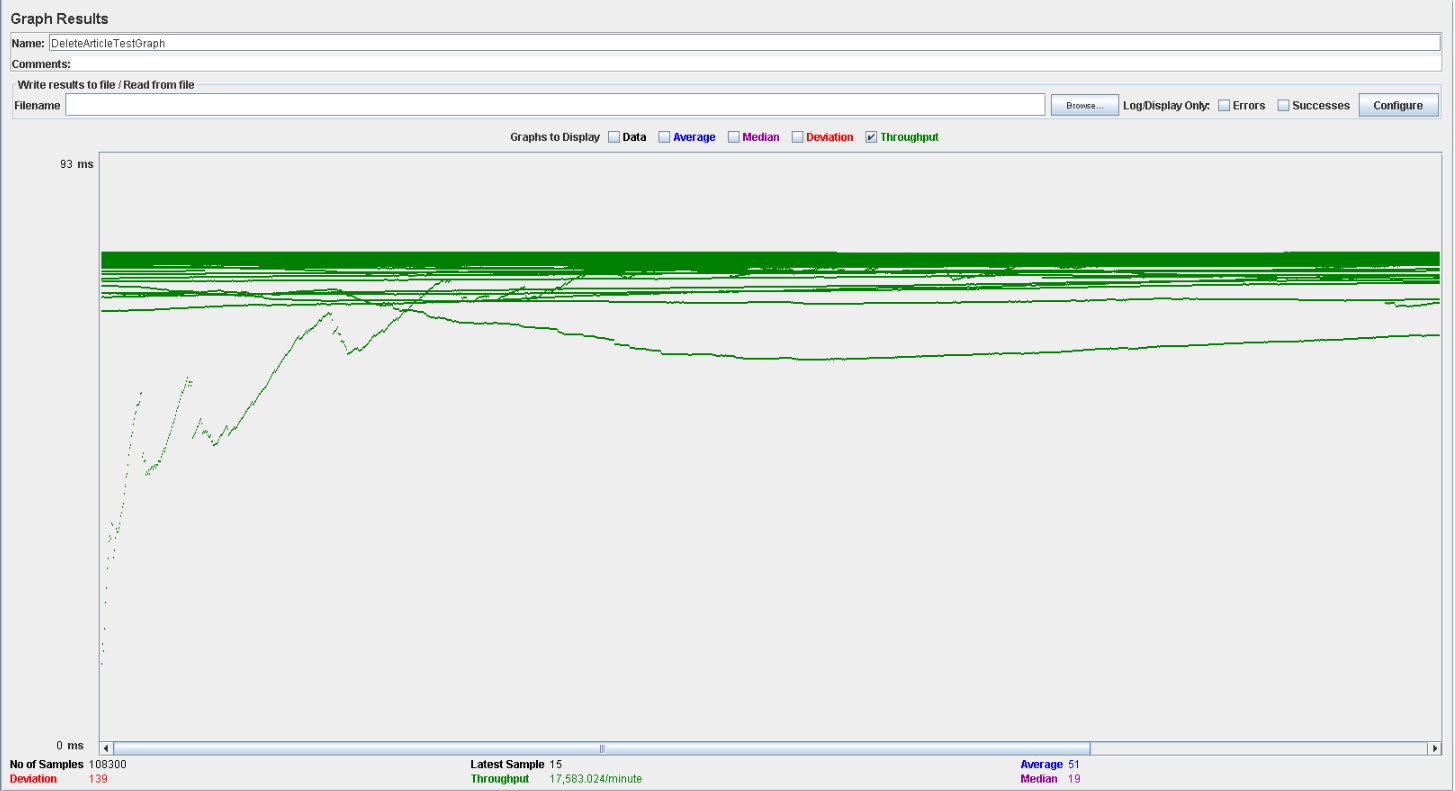
El disco no es el problema, puesto que los valores rondan el 15-20% de media.

# Caso de uso: borrar un artículo.

En este caso de uso, accedemos a la página, nos logueamos como un usuario, accedemos a nuestros artículos, accedemos a la vista para editar, borramos el artículo y nos deslogueamos.

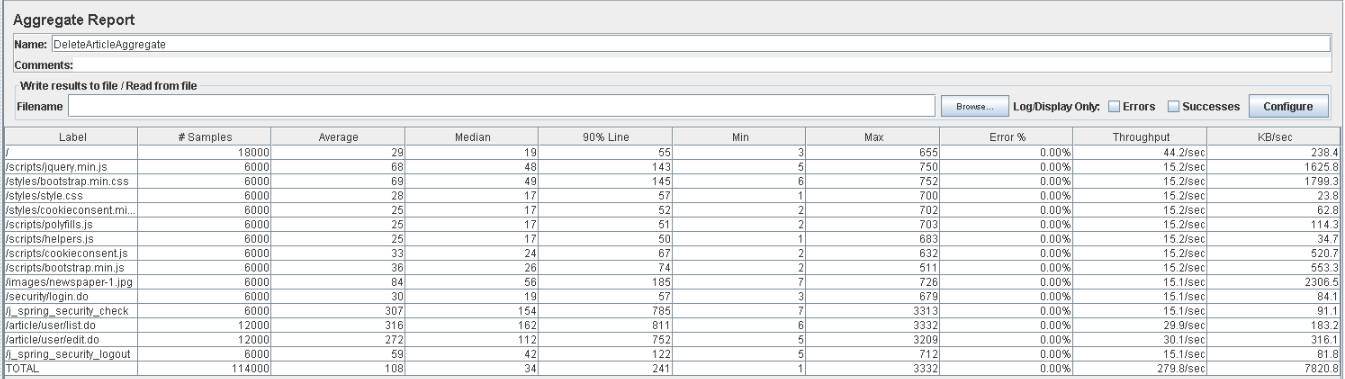
La configuración en esta ocasión es de 190 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores ni tiempos de respuestas muy altos. Obtenemos las gráficas:



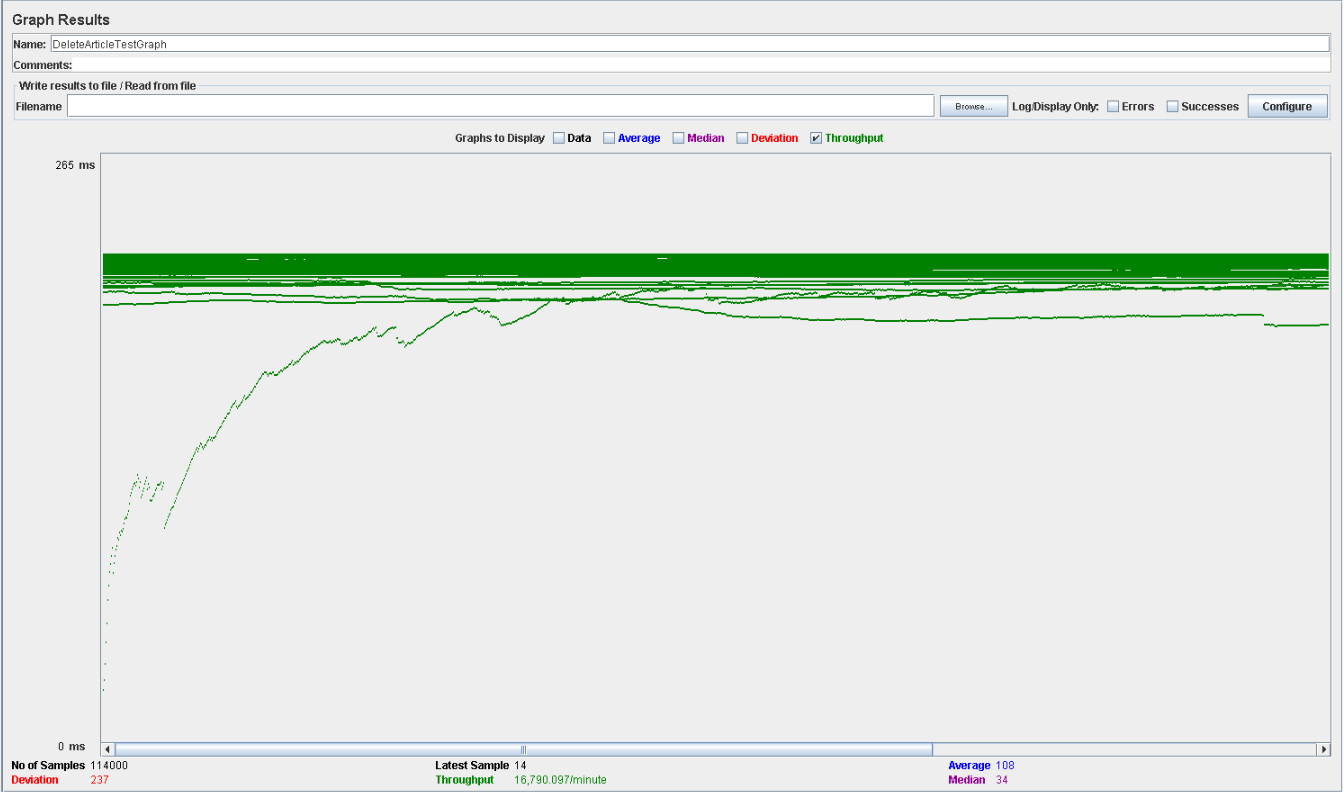


En esta tabla, el rendimiento es de aproximadamente 17.600 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Ahora, la configuración es de 200 usuarios, 1 ramp-up y 30 acciones cada uno, produciéndose pequeños porcentajes de errores. El número de usuarios permitidos está entre 180 y 200.



Como observamos fijándonos en 90% Line, las direcciones /article/user/list.do y /article/user/edit.do generan un tiempo de respuesta mayor al resto, 2170 ms y 2067 ms respectivamente.



En esta tabla, el rendimiento es de aproximadamente 16.800 ejecuciones por minuto (desciende en gran medida al aumentar los usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Como con 200 usuarios las esperas son excesivas, este caso de uso soporta entre 190 y 200 usuarios.

Para encontrar origen a estos resultados, revisamos performance.exe, obteniendo lo siguiente:

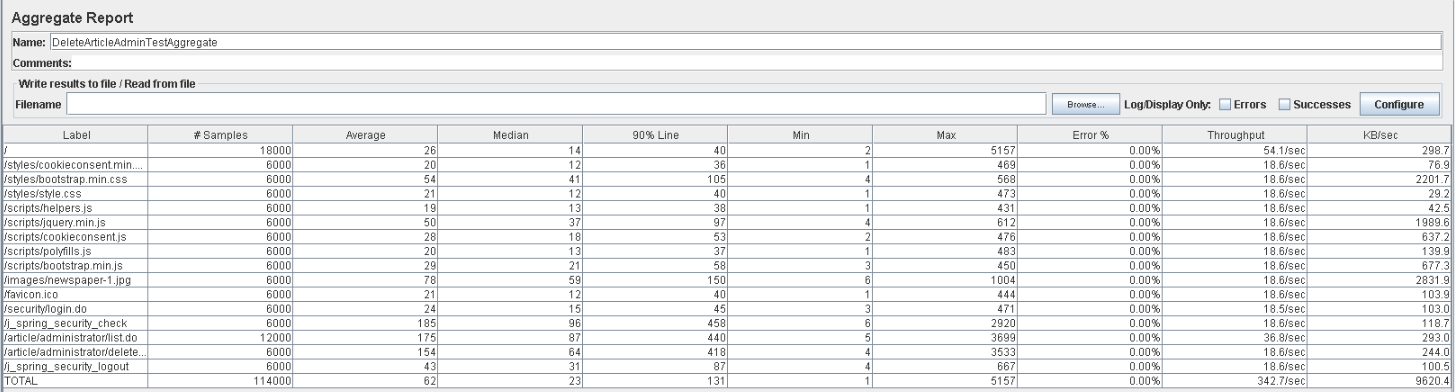


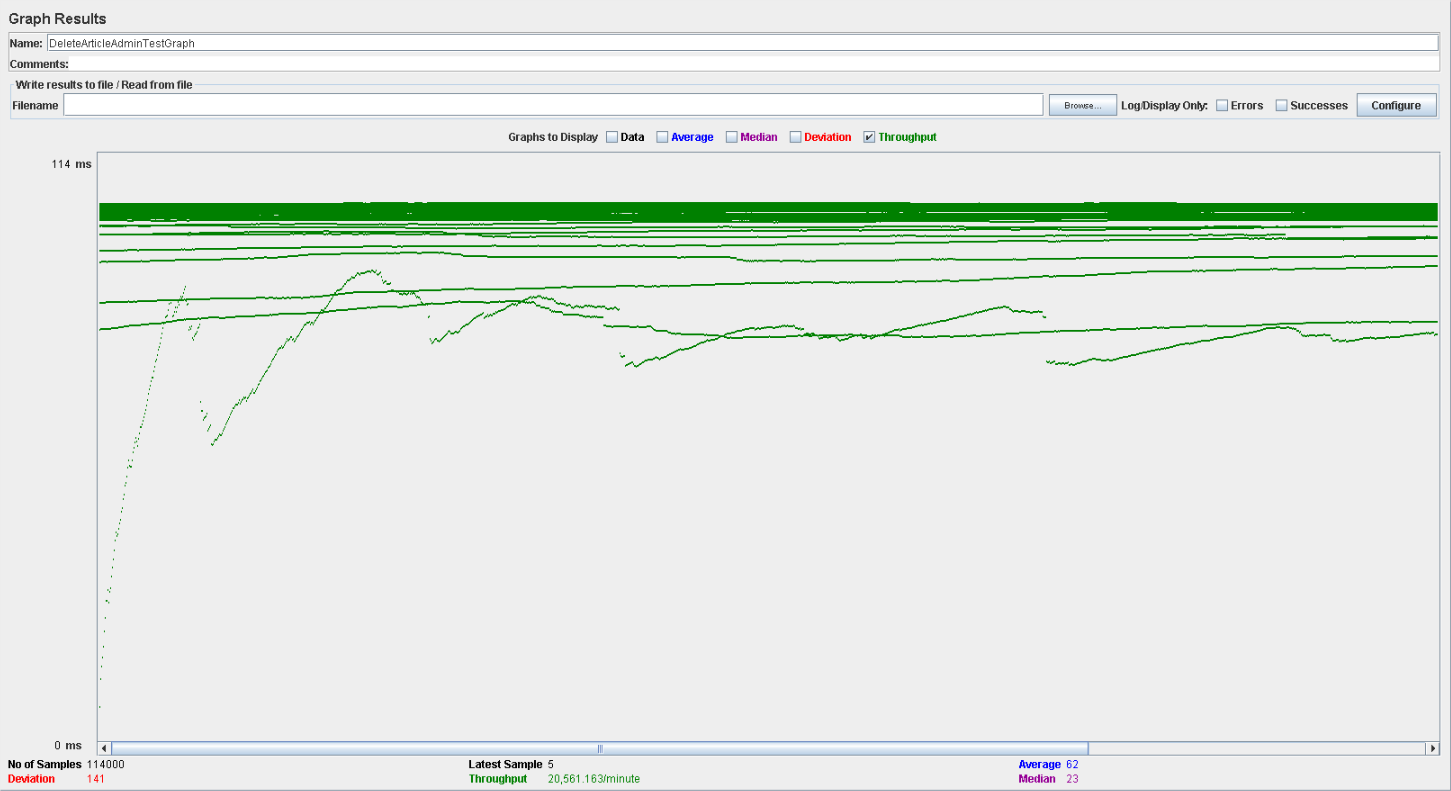
Observamos que podemos estar ante un problema de CPU, ya que se está utilizando un alto porcentaje de la capacidad del procesador la mayoría del tiempo.

# Caso de uso: borrar artículos administrador.

En este caso de uso, accedemos a la página, nos logueamos como un administrador, accedemos a la lista de artículos, borramos el artículo que deseemos y nos deslogueamos.

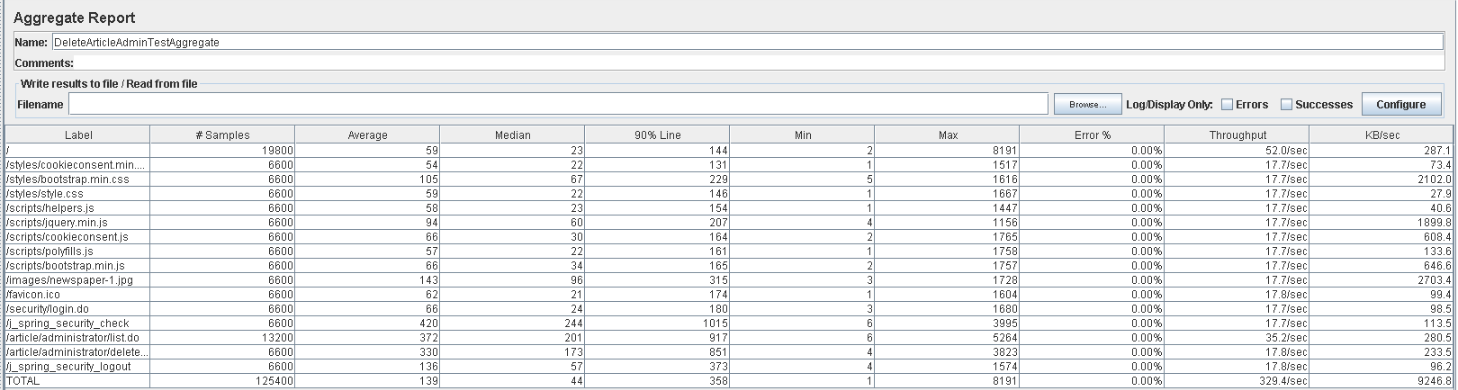
La configuración en esta ocasión es de 200 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores ni tiempos de respuestas demasiado altos. Obtenemos las gráficas:



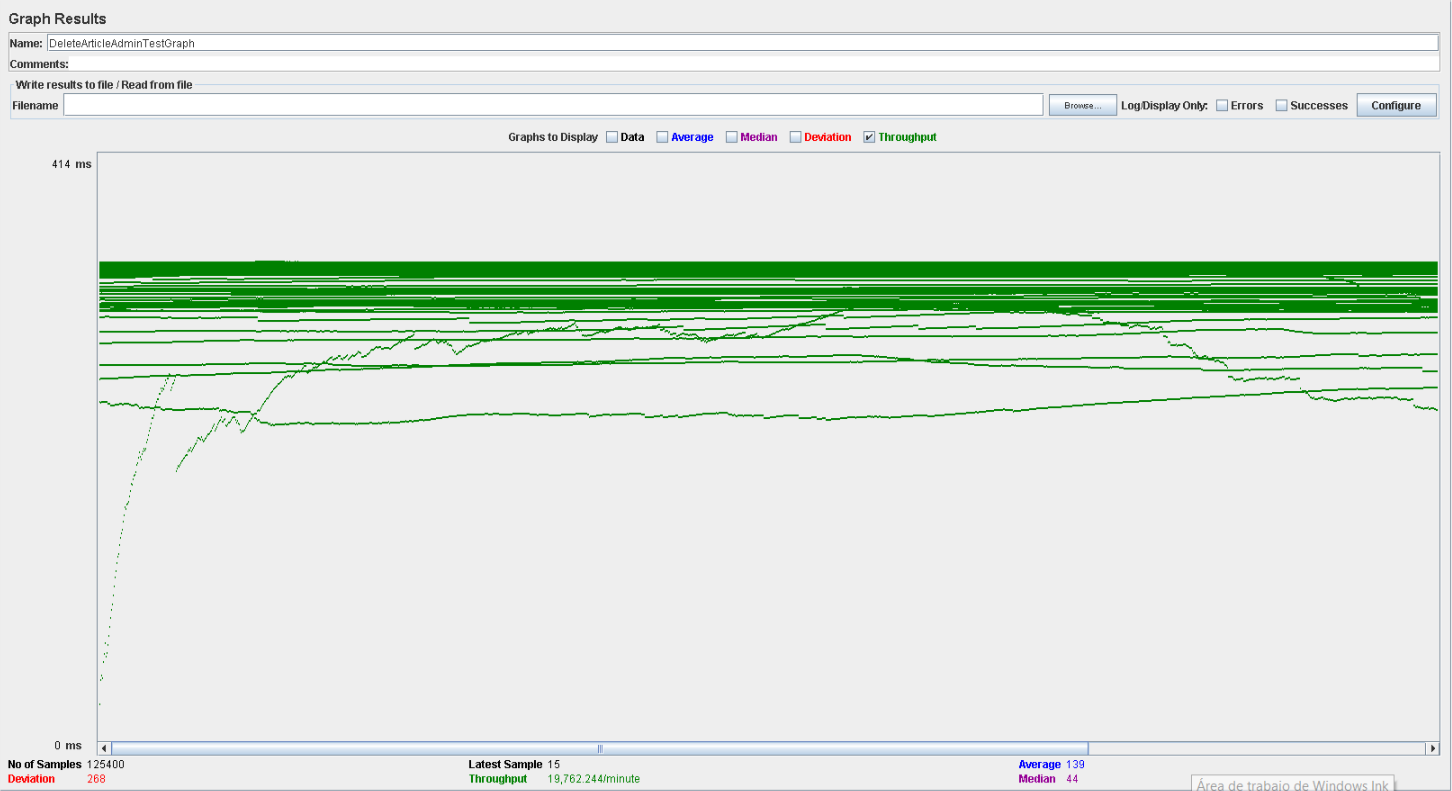


En esta tabla, el rendimiento es de aproximadamente 20.500 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

La configuración en esta ocasión es de 220 usuarios, 1 ramp-up y 30 acciones cada uno, produciéndose leves errores, que podemos ver en el Aggregate Report y en el Graph Results:



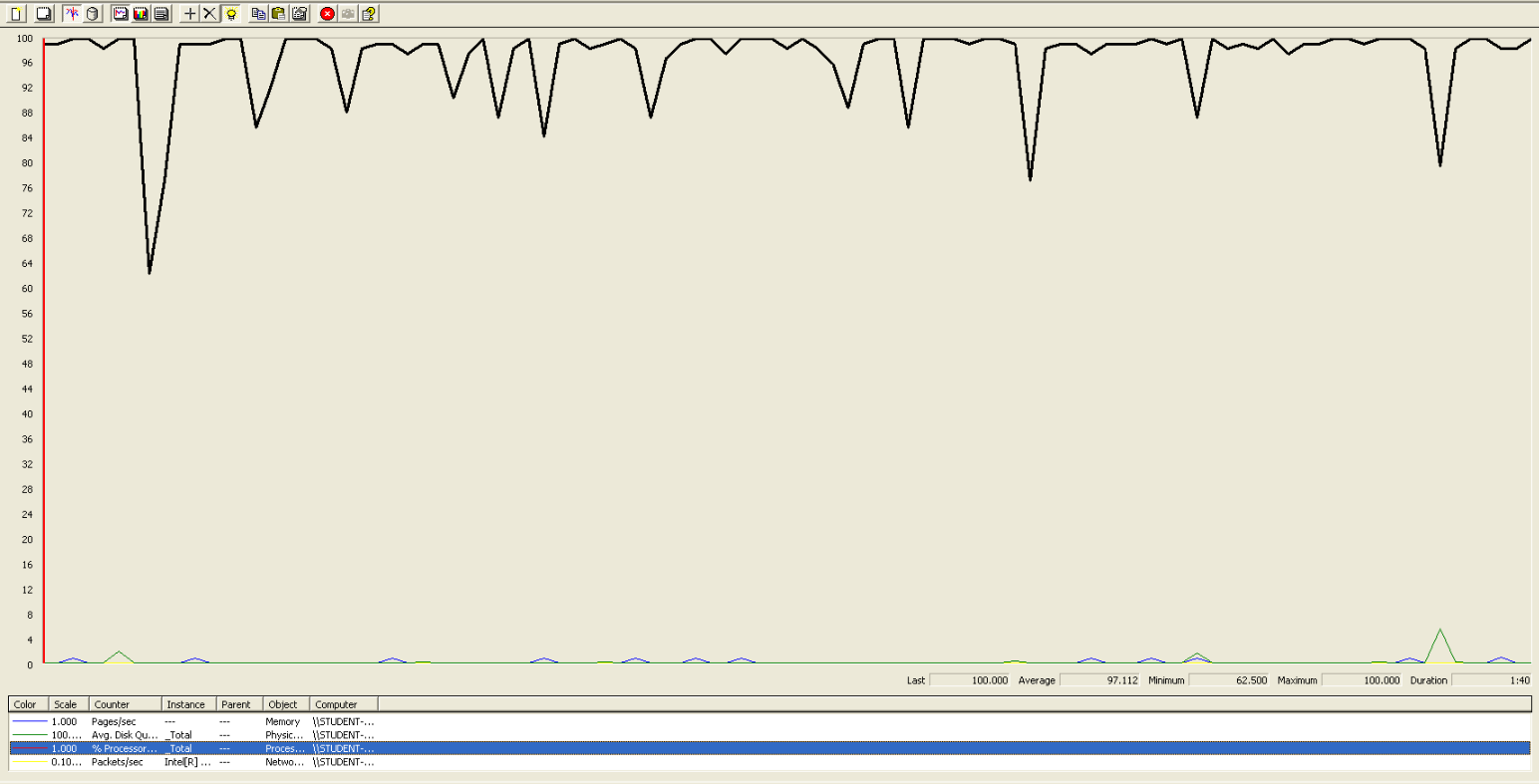
Como podemos observar fijándonos en 90% Line, las direcciones /j\_spring\_security\_check, /article/administrator/list.do y /article/administrator/deleteLis.do generan un tiempo de respuesta mayor a lo esperado, 1015, 917 y 851 ms respectivamente.



En esta tabla, el rendimiento es de aproximadamente 19.700 ejecuciones por minuto (desciende al aumentar los usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Ahora estudiaremos a que se deben los tiempos y los errores:

Abrimos performance.exe y vemos que la CPU tienen porcentajes muy altos, llegando al 100% en la mayoría de las ocasiones, por lo que el problema se localizará en la CPU.



Performance con Processor Time.

Por lo que, en este caso de uso, los usuarios permitidos serán entre 200 y 220.

# Caso de uso: listar suscripciones a periódicos de un cliente.

En este caso de uso, accedemos a la página, nos logueamos como un administrador, accedemos a la lista de artículos, borramos el artículo que deseemos y nos deslogueamos.

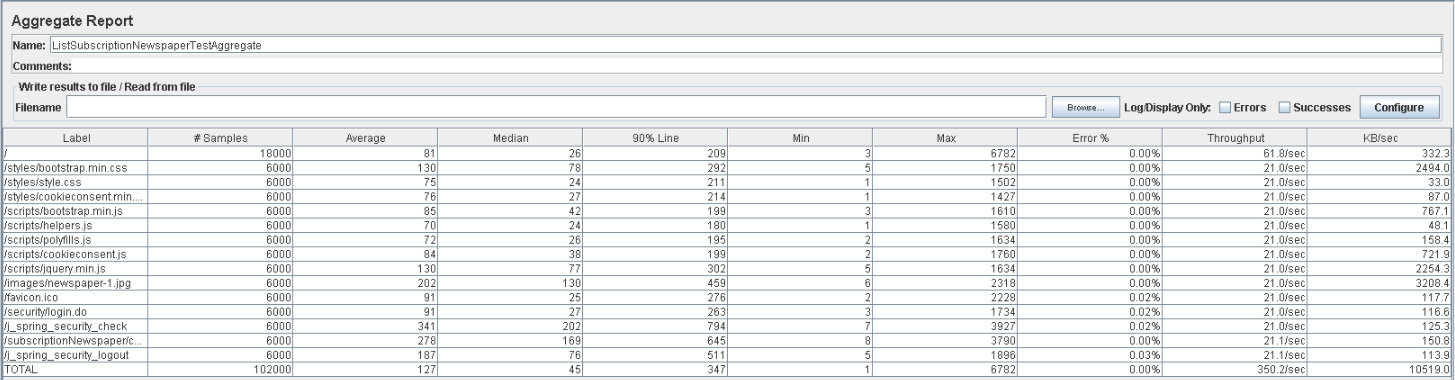
La configuración en esta ocasión es de 180 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores ni tiempos de respuestas demasiado altos. Obtenemos las gráficas:



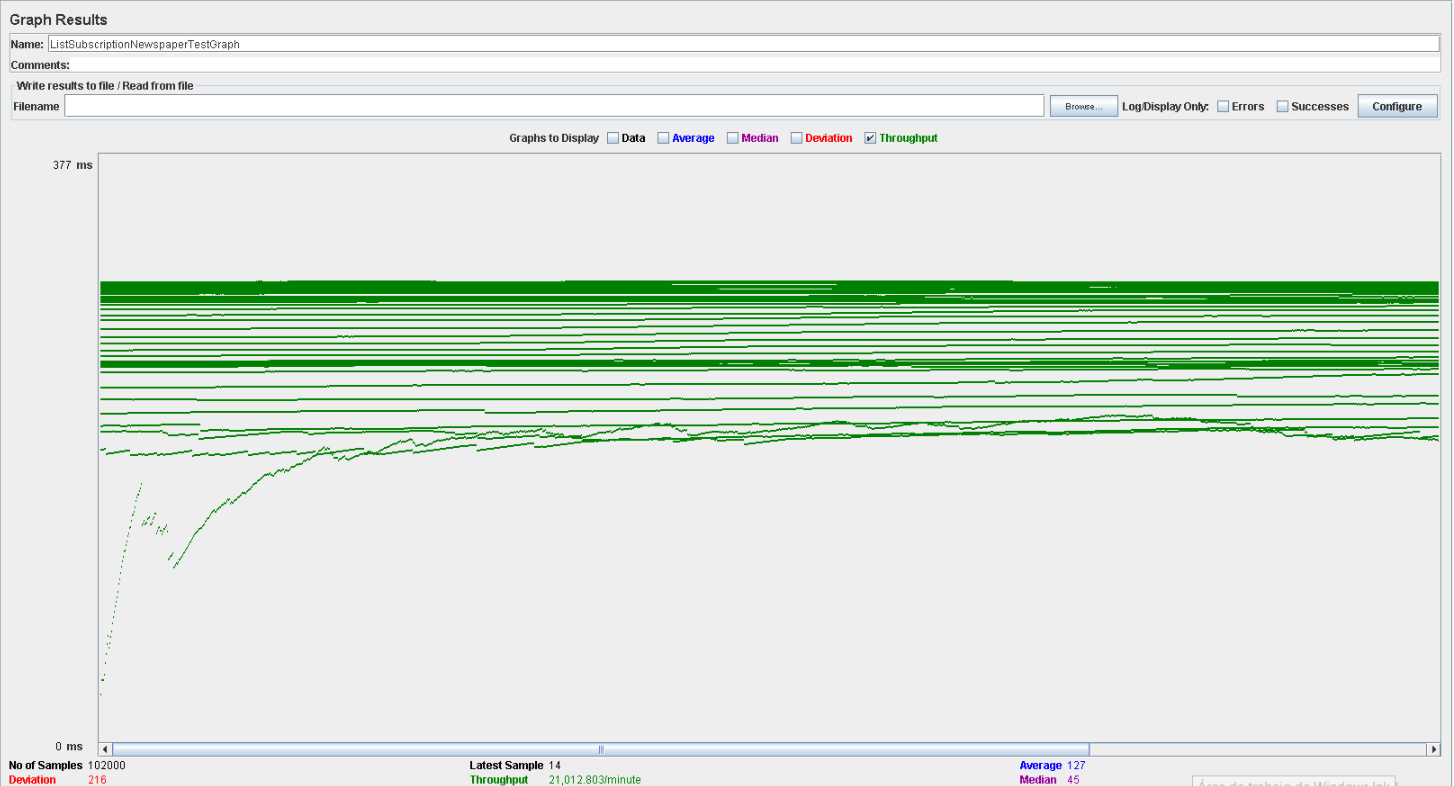


En esta tabla, el rendimiento es de aproximadamente 21.500 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

La configuración en esta ocasión es de 200 usuarios, 1 ramp-up y 30 acciones cada uno, produciéndose leves errores, que podemos ver en el Aggregate Report y en el Graph Results:



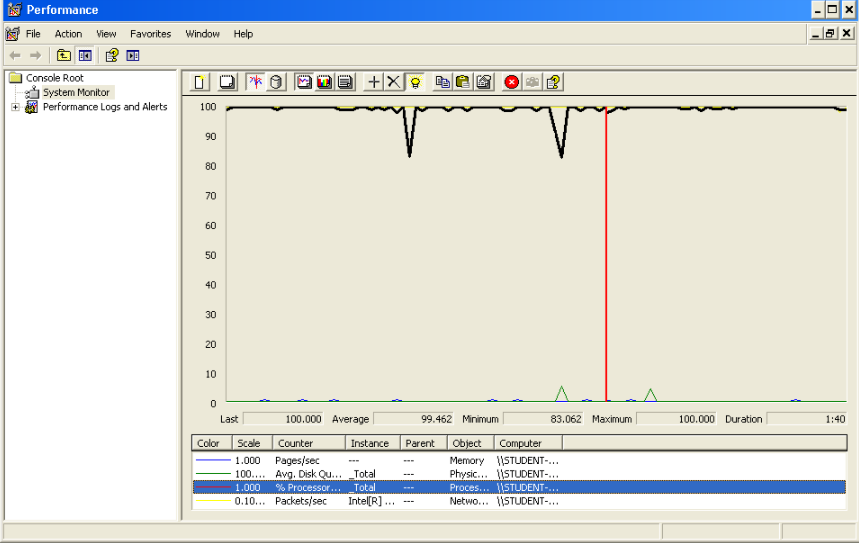
Como podemos observar fijándonos en 90% Line, las direcciones /j\_spring\_security\_check y /subscriptionNewspaper/customer/list.do generan un tiempo de respuesta mayor a lo esperado para un caso de uso de listar, 794 y 645 ms respectivamente.



En esta tabla, el rendimiento es de aproximadamente 21.000 ejecuciones por minuto (disminuye al aumentar los usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Ahora estudiaremos a que se deben los tiempos y los errores:

Abrimos performance.exe y vemos que la CPU tiene porcentajes muy altos, llegando al 100% en ocasiones, por lo que el problema se encontrará en la CPU.



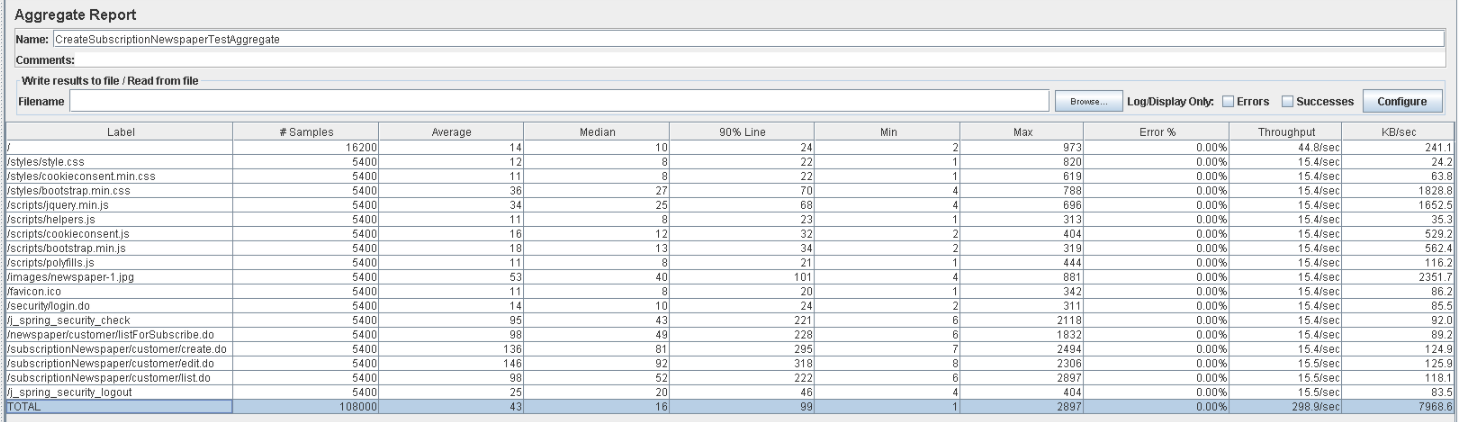
Performance con Processor Time.

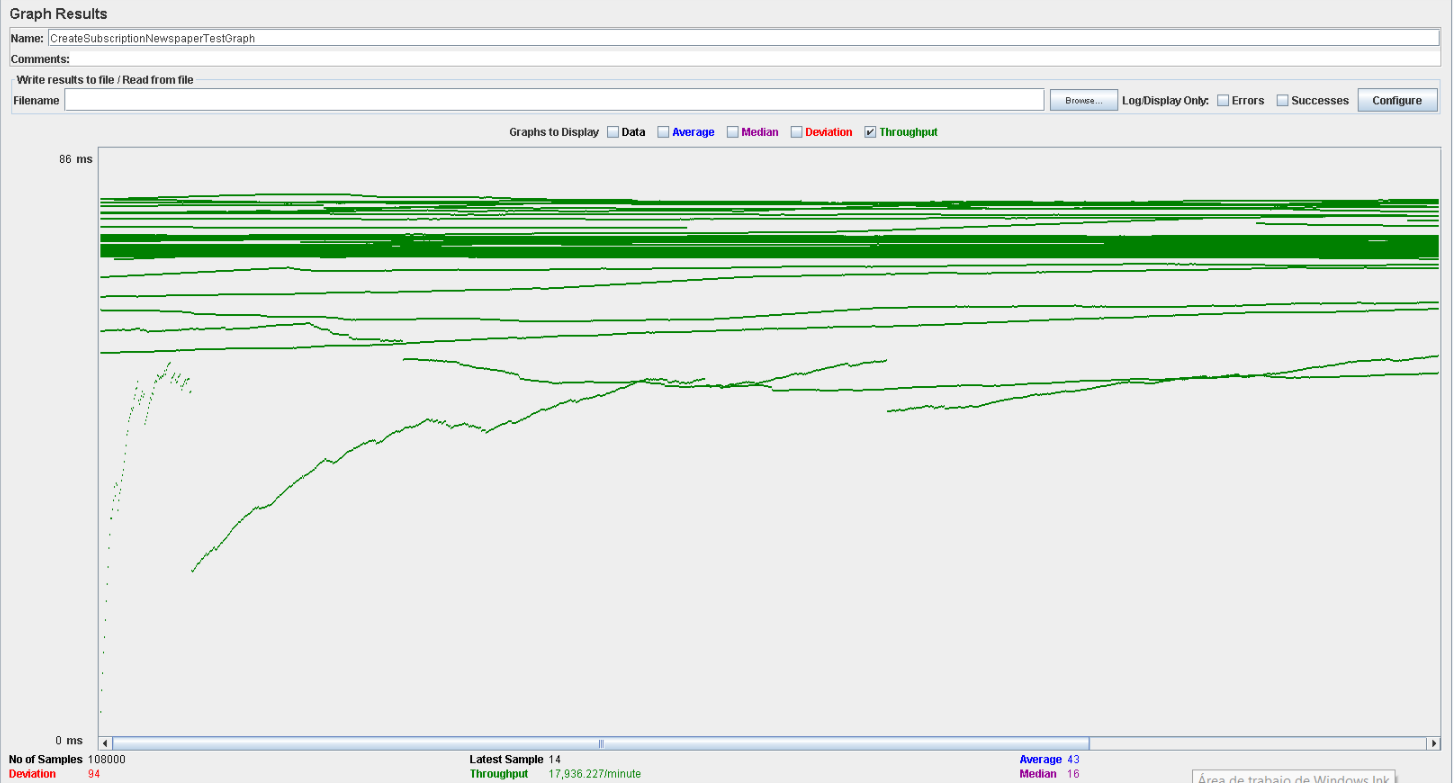
Por lo que, en este caso de uso, los usuarios permitidos serán entre 180 y 200.

# Caso de uso: suscribirse a un periódico.

En este caso de uso, accedemos a la página, nos logueamos como cliente, accedemos al listado de newspapers para suscribirnos, a continuación, al formulario para suscribirse y finalmente accedemos nos deslogueamos.

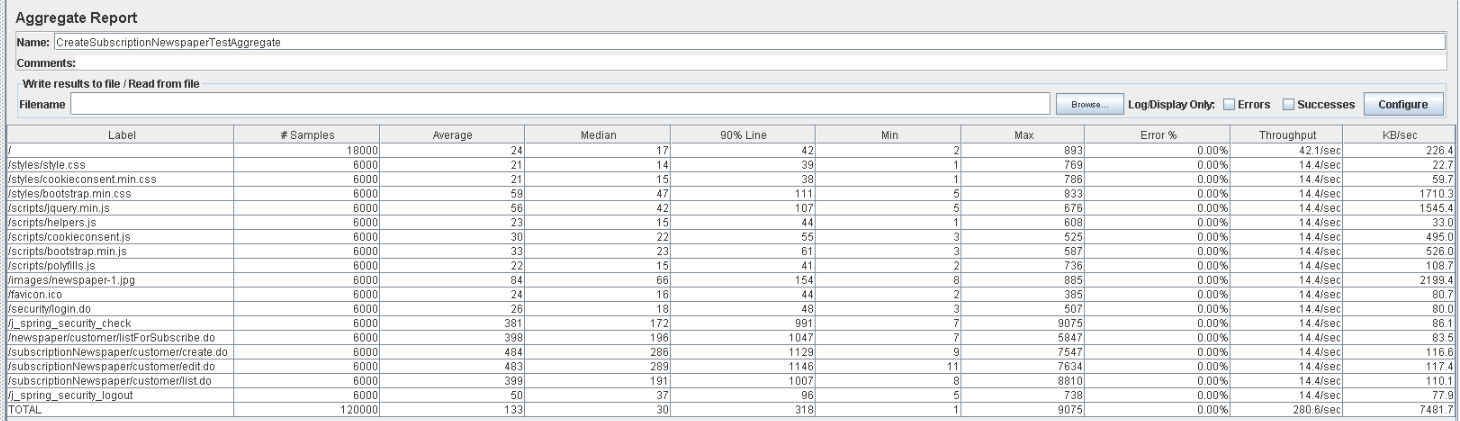
Probamos una configuración 180 usuarios, 1 ramp-up y 30 acciones cada uno, no obteniendo errores y las esperas no son excesivamente altas:





En esta tabla, el rendimiento es de aproximadamente 17.900 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Volvemos a probar con una configuración 200 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores, pero esperas demasiado altas, que podemos ver en el Aggregate Report y en el Graph Results:



Como podemos observar fijándonos en 90% Line, las direcciones /newspaper/customer/listForSubscribe.do, /subscriptionNewspaper/customer/create.do, /subscriptionNewspaper/customer/edit.do y /subscriptionNewspaper/customer/list.do generan un tiempo de respuesta mucho mayor a lo esperado (1000 ms), 1047, 1129, 1146 y 1007 ms respectivamente.

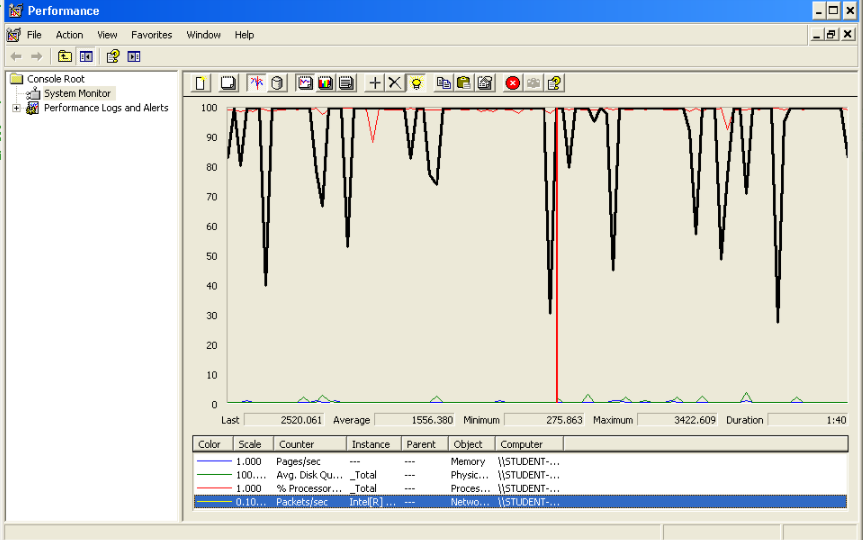


En esta tabla, el rendimiento es de aproximadamente 16.800 ejecuciones por minuto (desciende al aumentar los usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché. Ahora estudiaremos a que se deben los tiempos de espera:

Abrimos performance.exe y vemos que tanto la tarjeta de red como la CPU tienen porcentajes muy altos, llegando al 100% en ocasiones, por lo que el problema se localizará al menos en una de las dos.



Performance con Processor Time. Casi en todo momento la CPU se encuentra al 100%.



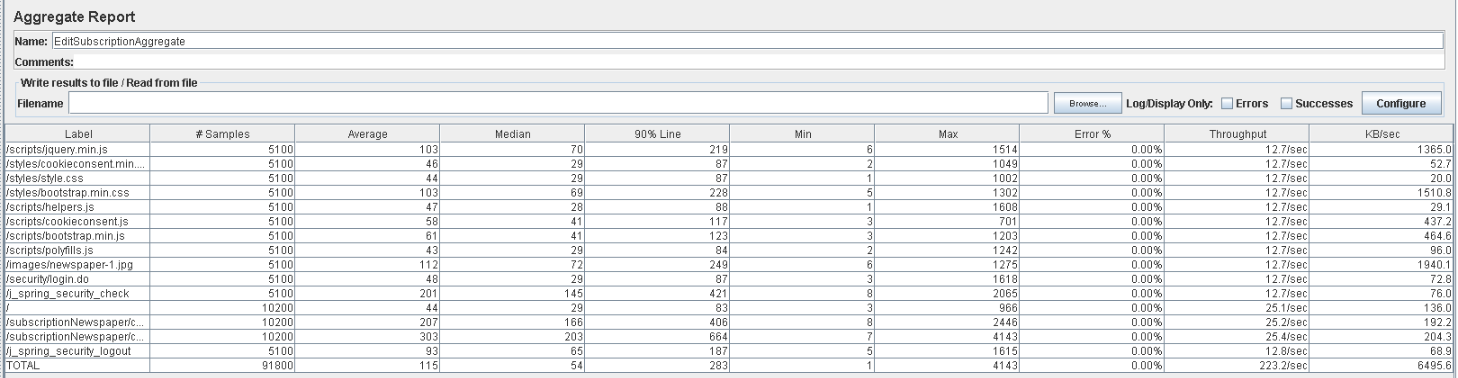
Performance con Packets/sec. Con alto porcentaje de la tarjeta de red.

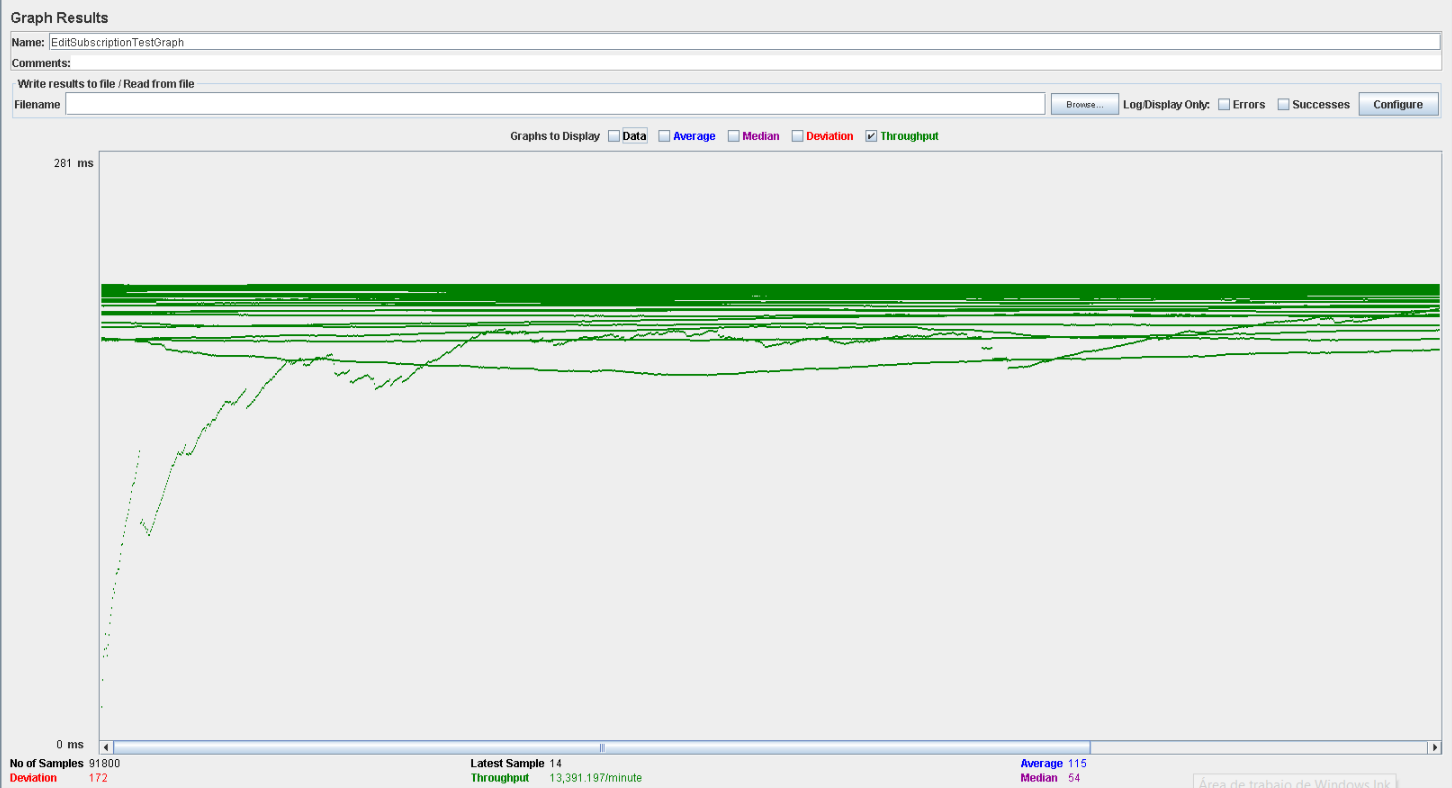
Por lo que, en este caso de uso, los usuarios permitidos serán entre 180 y 200.

# Caso de uso: editar suscripción.

En este caso de uso, accedemos a la página, nos logueamos como un cliente, accedemos a nuestras suscripciones, accedemos a la vista para editar, editamos algún parámetro, guardamos y nos deslogueamos.

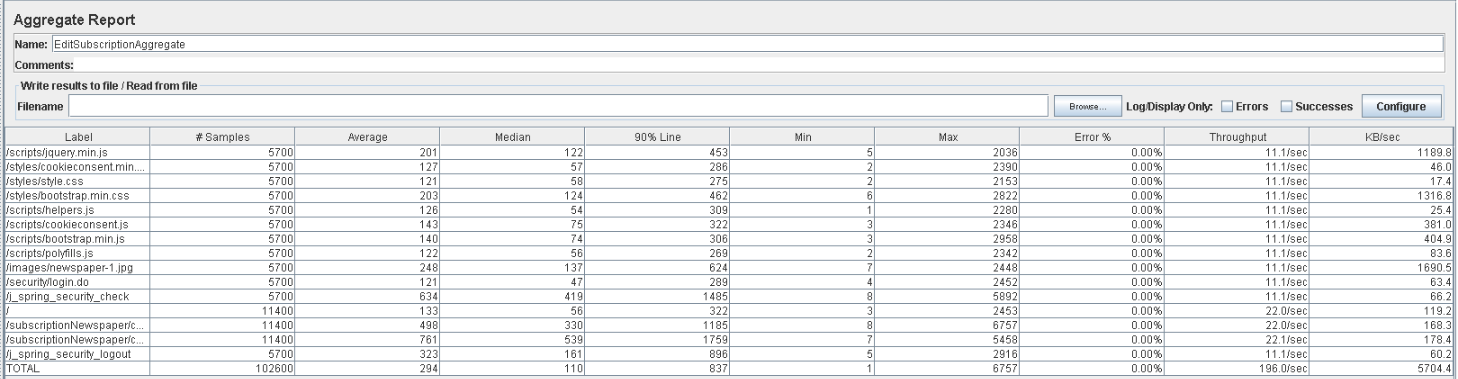
Probamos una configuración 170 usuarios, 1 ramp-up y 30 acciones cada uno, no obteniendo errores y las esperas no son excesivamente altas:



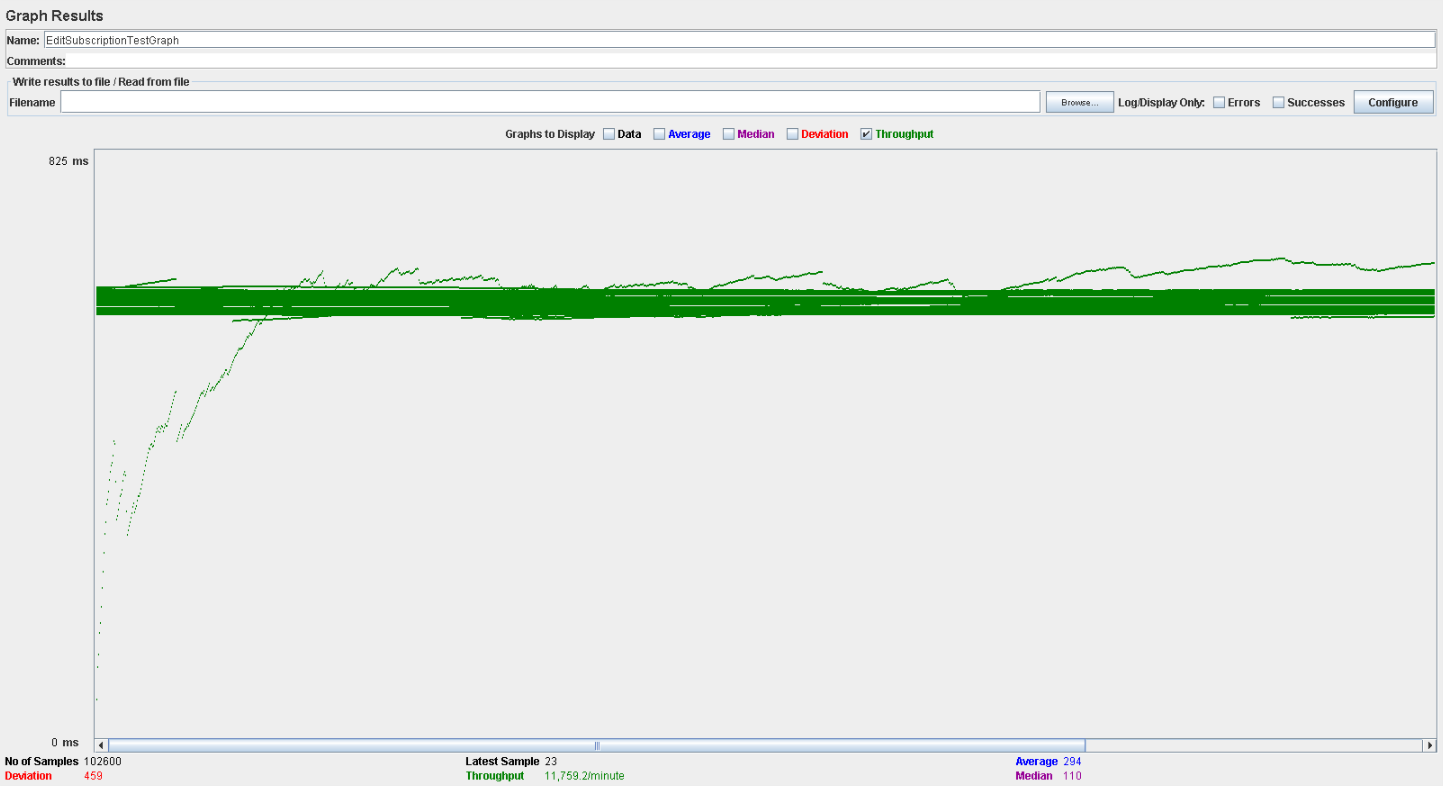


En esta tabla, el rendimiento es de aproximadamente 13.400 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Volvemos a probar con una configuración 190 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores, pero esperas demasiado altas, que podemos ver en el Aggregate Report y en el Graph Results:



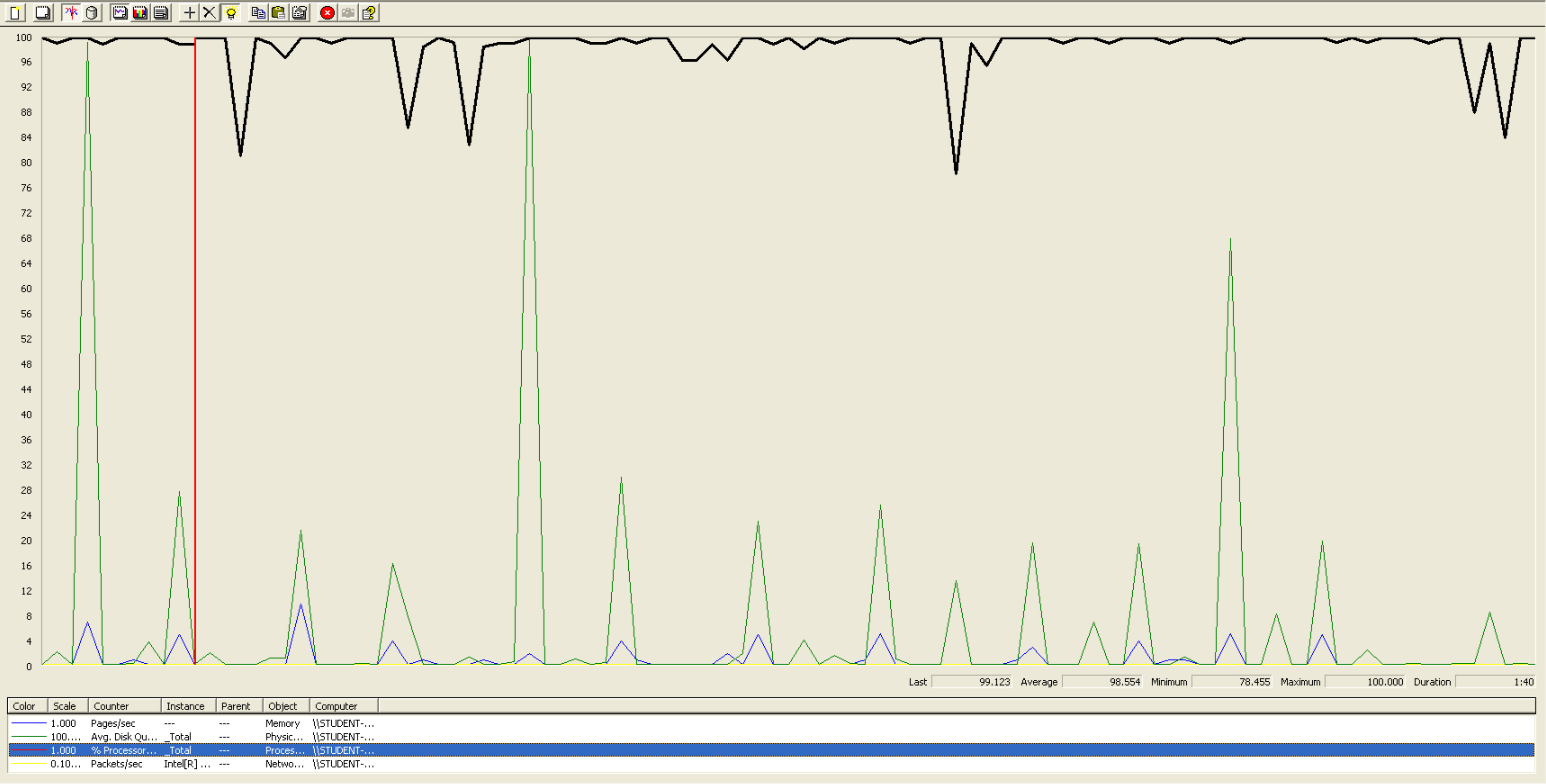
Como podemos observar fijándonos en 90% Line, las direcciones /j\_spring\_security\_check, /subscriptionNewspaper/customer/list.do y /subscriptionNewspaper/customer/edit.do generan un tiempo de respuesta mayor al resto, 1485 ms, 1185 ms y 1759 ms respectivamente.



En esta tabla, el rendimiento es de aproximadamente 11.700 ejecuciones por minuto (desciende al aumentar los usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Ahora estudiaremos a que se deben los tiempos de espera:

Abrimos performance.exe y vemos que el procesador tiene porcentajes muy altos, llegando al 100% casi siempre, por lo que el problema se localizará probablemente en la CPU.



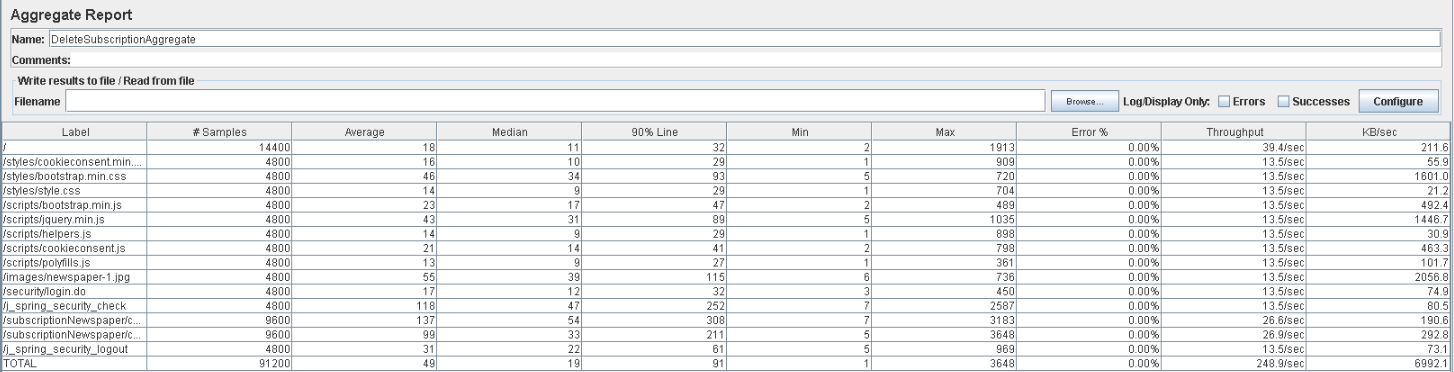
Performance con Processor Time. Casi en todo momento la CPU se encuentra al 100%.

Por lo que, en este caso de uso, los usuarios permitidos serán entre 170 y 190.

# Caso de uso: borrar una suscripción.

En este caso de uso, accedemos a la página, nos logueamos como un usuario, accedemos a nuestros artículos, accedemos a la vista para editar, borramos el artículo y nos deslogueamos.

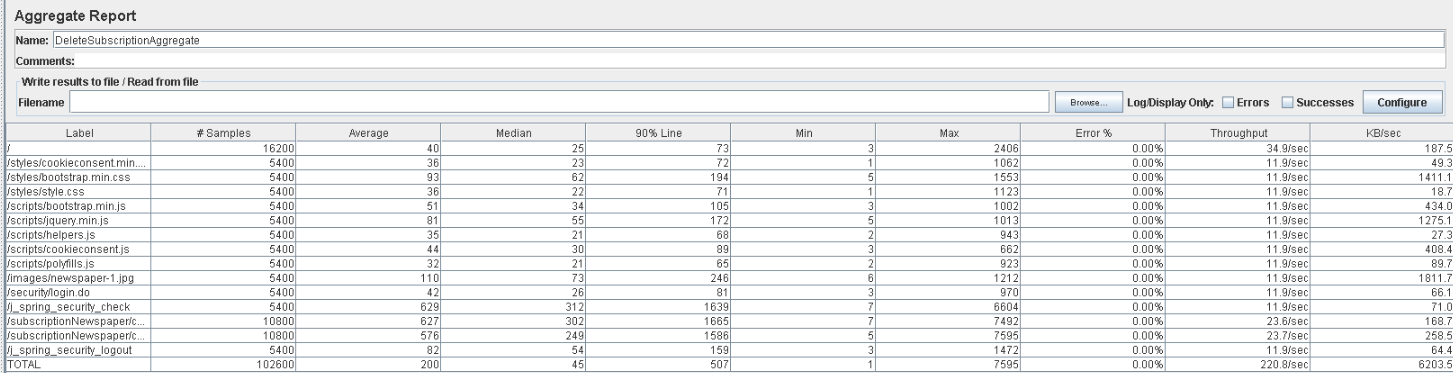
Probamos una configuración 160 usuarios, 1 ramp-up y 30 acciones cada uno, no obteniendo errores y las esperas no son excesivamente altas:





En esta tabla, el rendimiento es de aproximadamente 14.900 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Volvemos a probar con una configuración 180 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores, pero esperas demasiado altas, que podemos ver en el Aggregate Report y en el Graph Results:



Como podemos observar fijándonos en 90% Line, las direcciones /j\_spring\_security\_check, /subscriptionNewspaper/customer/list.do y /subscriptionNewspaper/customer/edit.do generan un tiempo de respuesta mayor lo permitido (1000 ms), 1639 ms, 1665 ms y 1586 ms respectivamente.



En esta tabla, el rendimiento es de aproximadamente 13.200 ejecuciones por minuto (desciende al aumentar los usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Ahora estudiaremos a que se deben los tiempos de espera:

Abrimos performance.exe y vemos que el procesador tiene porcentajes muy altos, llegando al 100% en varias ocasiones, por lo que el problema se localizará probablemente en la CPU.



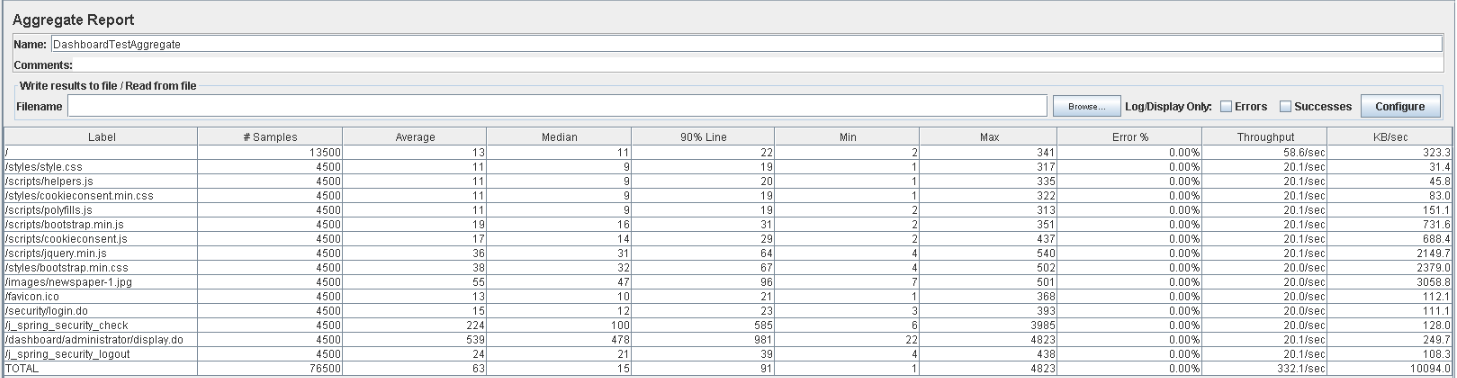
Performance con Processor Time. Casi en todo momento la CPU se encuentra al 100%.

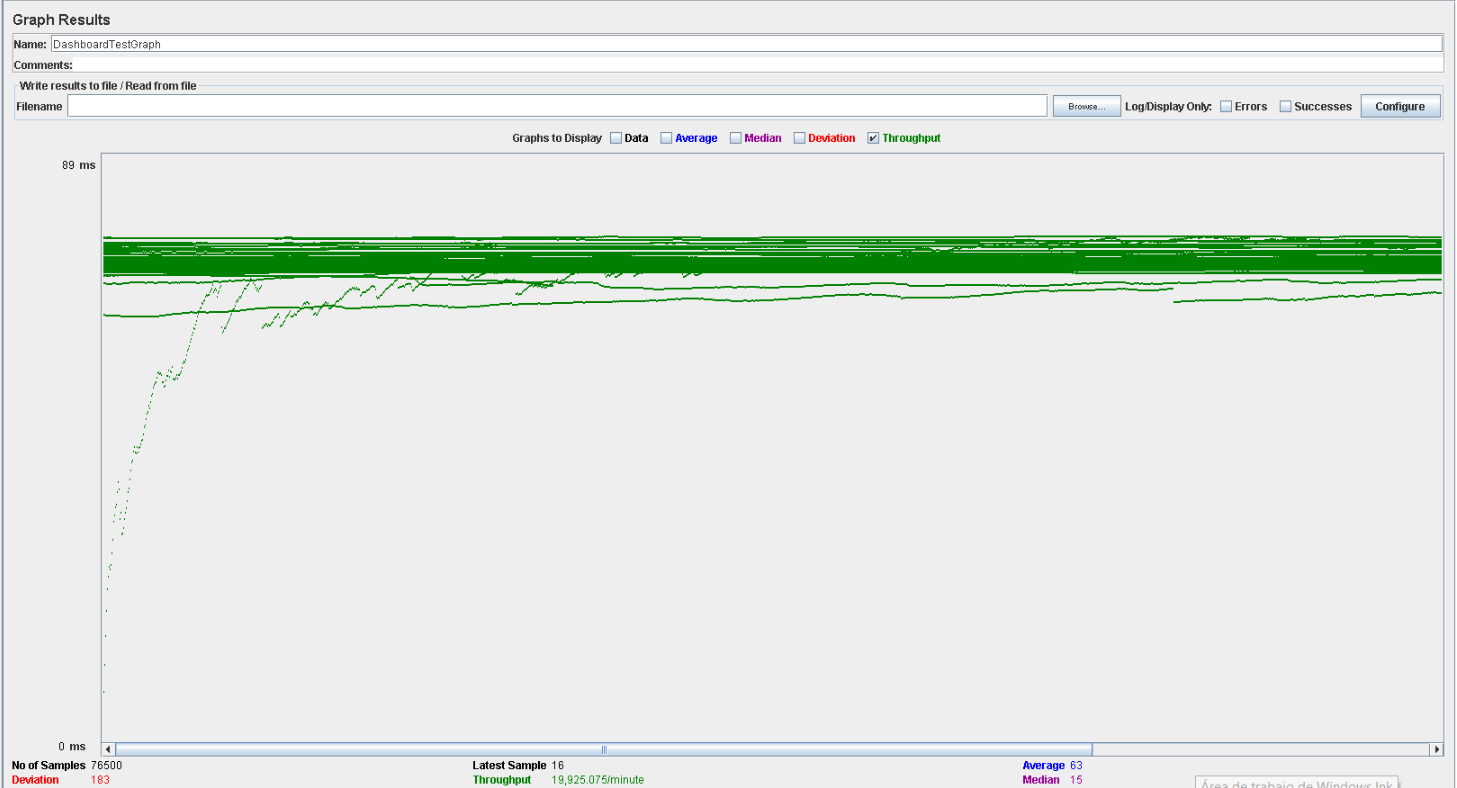
Por lo que, en este caso de uso, los usuarios permitidos serán entre 170 y 190.

# Caso de uso: tablón de mandos del administrador.

Nos autenticamos como administrador, vemos el dashboard, navegamos por él, y nos des autenticamos.

Probamos una configuración 150 usuarios, 1 ramp-up y 30 acciones cada uno, no obteniendo errores y aunque las esperas no son excesivamente altas, aunque están en el límite de lo permitido:

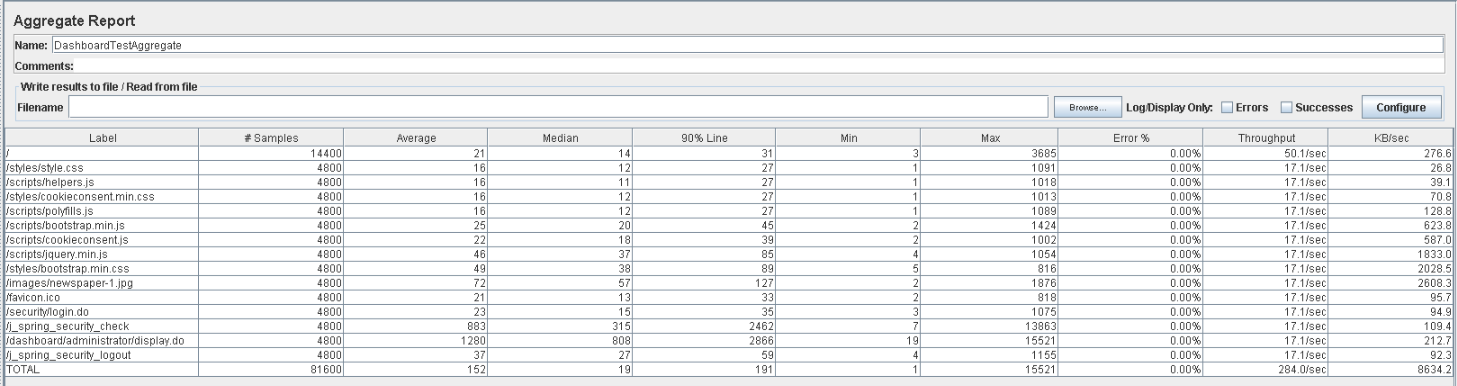




En esta tabla, el rendimiento es de aproximadamente 19.900 ejecuciones por minuto. El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

En un caso real, no sería normal encontrar un número tan alto de administradores en la aplicación. No obstante, hemos considerado suficiente ver un valor alto para así tener una estimación bastante buena tanto de los errores como de los tiempos de ejecución.

Volvemos a probar con una configuración 160 usuarios, 1 ramp-up y 30 acciones cada uno, no produciéndose errores, pero con esperas ya demasiado altas, que podemos ver en el Aggregate Report y en el Graph Results:



Como podemos observar fijándonos en 90% Line, la dirección /dashboard/administrator/display.do genera un tiempo de respuesta mucho mayor a lo esperado (1000 ms), un total de 2866 ms.



En esta tabla, el rendimiento es de aproximadamente 17.000 ejecuciones por minuto (desciende al aumentar el número de usuarios). El rendimiento inicial es pequeño porque los hilos están comenzando y aumenta a medida que aumenta la carga de trabajo y el sistema inicia el almacenamiento en caché.

Por lo que, en este caso de uso, los usuarios permitidos serán entre 150 y 160. Ahora estudiaremos a que se deben los tiempos de espera. Abrimos performance.exe y vemos que la CPU tiene porcentajes muy altos, llegando al 100% en ocasiones, por lo que el problema se localizará en el procesador.



Performance con Processor Time. Casi en todo momento la CPU se encuentra al 100%.