**Análisis de Capacidad**

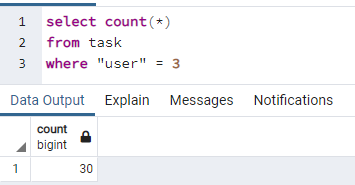
1. **Pruebas de estrés sobre el Modelo de Despliegue.**

**Escenario 1**. Se deberá definir un escenario donde se pueda probar cuál es la máxima cantidad de peticiones HTTP por minuto que soporta el API con 30 archivos publicados. Para hacer pruebas de estrés se sugiere utilizar la herramienta JMeter que debe instalarse en el ambiente de nube (para no sesgar los resultados de la prueba con la latencia que introduce la red, la máquina de pruebas debe ubicarse en el mismo segmento de red de la aplicación). El escenario y los resultados de las pruebas de estrés deberán ser documentados con gráficas que ilustran cómo se comporta el sistema a medida que el número de usuarios virtuales accediendo a la aplicación se incrementa hasta llegar al punto de degradar completamente el rendimiento del sistema.

**Restricciones del escenario.** En las pruebas de stress el tiempo de respuesta promedio de la página debe ser de máximo 1.500 ms, si este tiempo no se cumple, se concluye que el sistema NO soporta la cantidad de peticiones de la prueba. En caso de que durante una prueba se generen más de un 1% de errores en las peticiones de la prueba, se concluye que el sistema NO soporta la cantidad de peticiones.

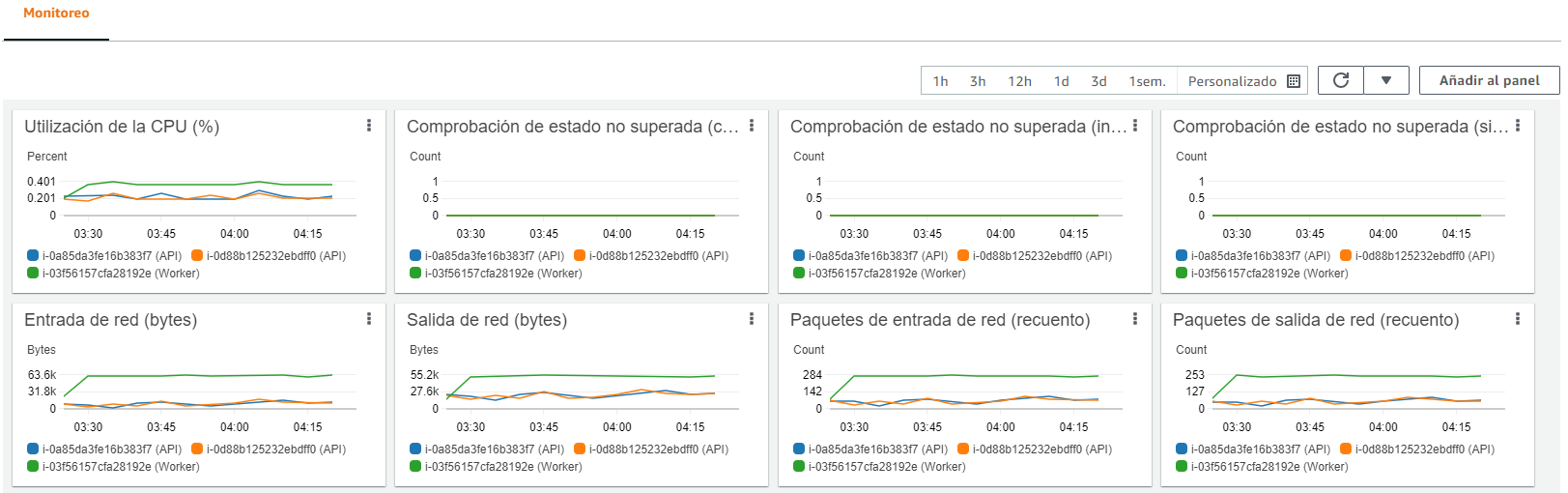
**Usuario con 30 archivos publicados**

En la imagen podemos ver que el usuario 3 cuenta con 30 tareas registradas

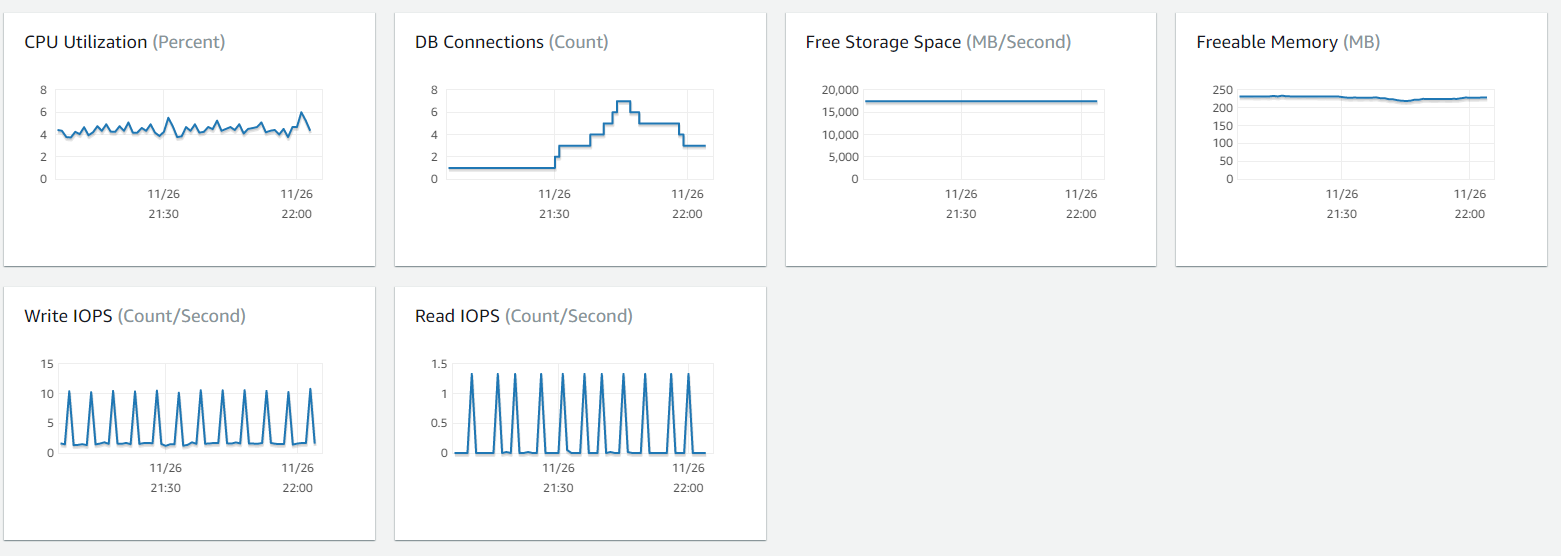


**Rendimiento inicial de los servidores**

Se muestra el rendimiento inicial de los 2 Servidores APIs y 1 servidor Worker



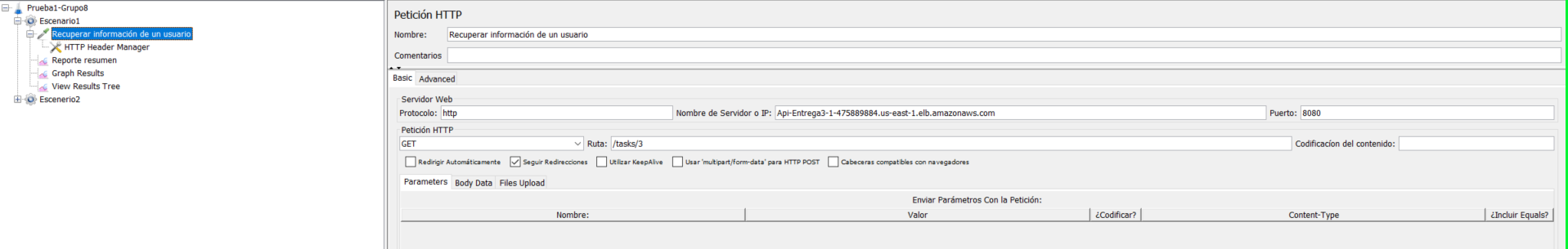
Se muestra el rendimiento inicial del Servidor de Base de Datos



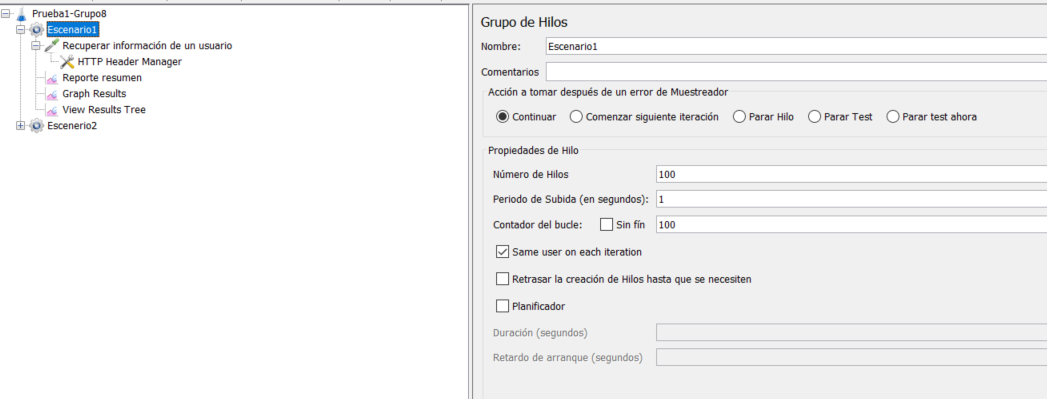
**Probamos el API “Recuperar tareas de un usuario” con 10000 muestras representadas en:**

* **100 hilos**
* **1 segundo de periodo de subida**
* **Contador del bucle en 100**

Se muestra la configuración del API en el JMeter

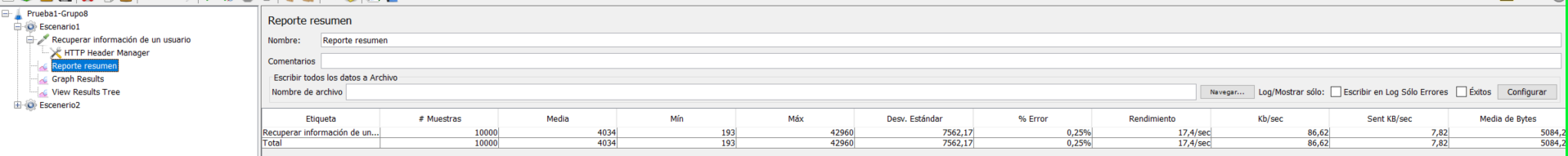


Se muestra la configuración de la prueba de estres a realizar

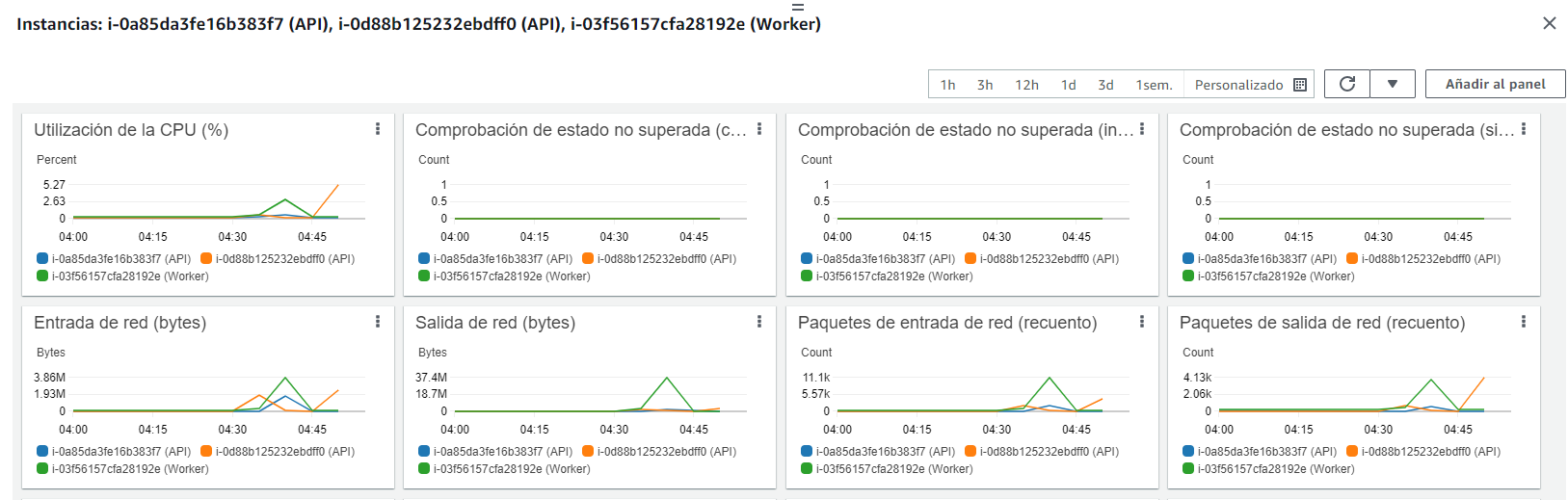


**RESULTADOS**

Se procesaron las 10000 peticiones arrojando una Media de 4034ms y un porcentaje de error de 0.25%

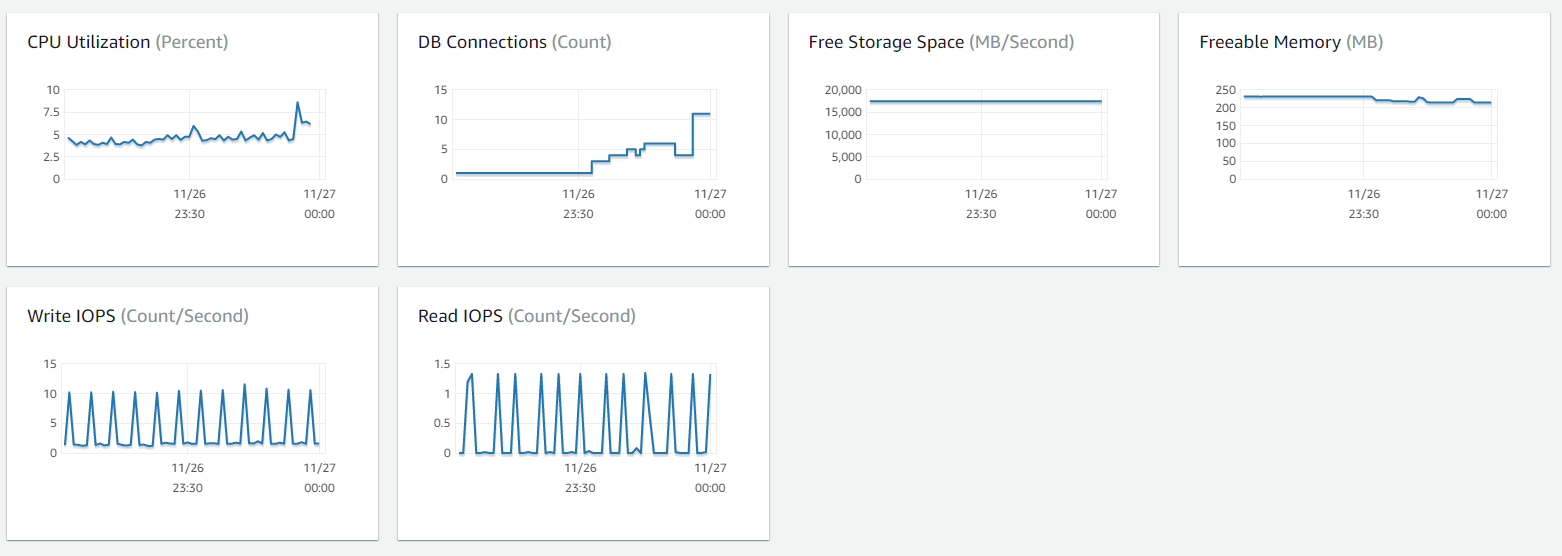


Se muestra el rendimiento de los servidores API y Worker en el momento de la prueba de estres, donde se puede observer un incremento en los distintos indicadores, por ejemplo en la Utilización del CPU el Servidor Worker alcanzó un pico de 2.63% (inicial 0.401%)



Se muestra el rendimiento del servidor de Base de Datos en el momento de la prueba de estres, donde se puede observer un incremento en los distintos indicadores, por ejemplo en la Utilización del CPU alcanzó un pico de 8.5% (inicial 6%)

También Podemos observar que el Write IOPS y Read IOPS el incremento es mínimo



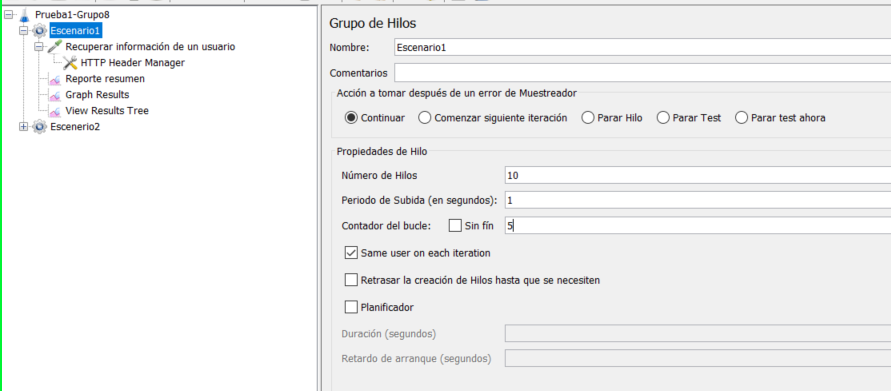
**Conclusión**

Para esta muestra no cumple lo esperado, pues supera los 1500ms y el 1% de error.

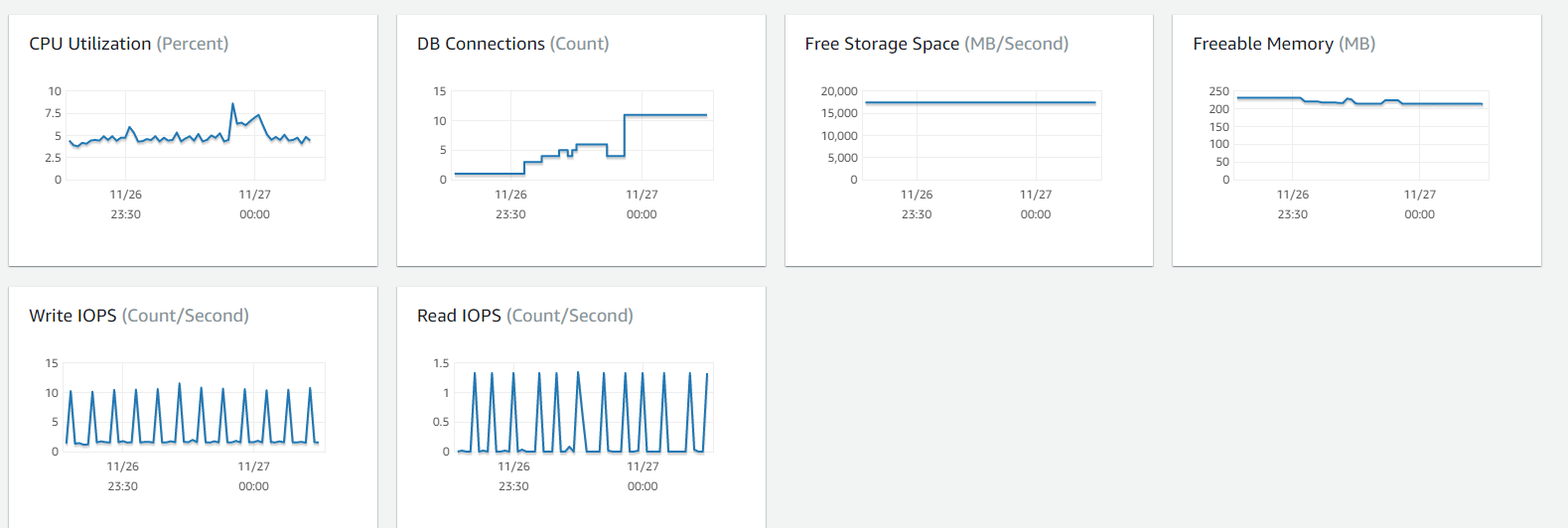
**Probamos el API “Recuperar tareas de un usuario” con 50 muestras representadas en:**

* **10 hilos**
* **1 segundo de periodo de subida**
* **Contador del bucle en 5**

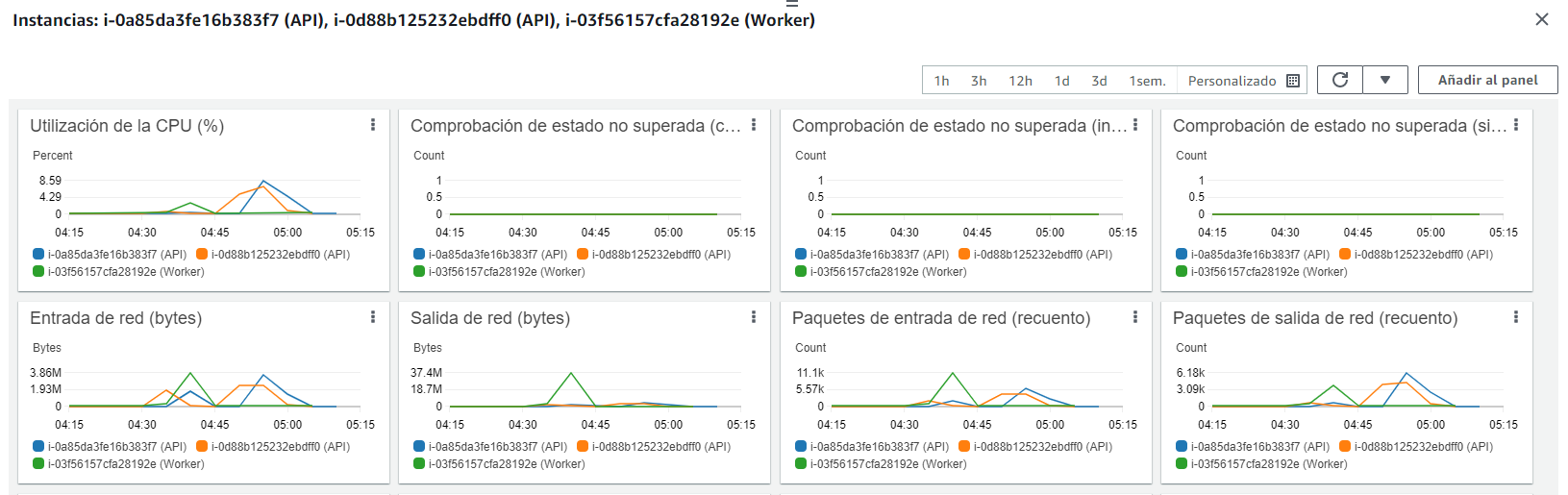
Se muestra la configuración de la prueba de estres a realizar



Rendimiento inicial del servidor de Base de Datos

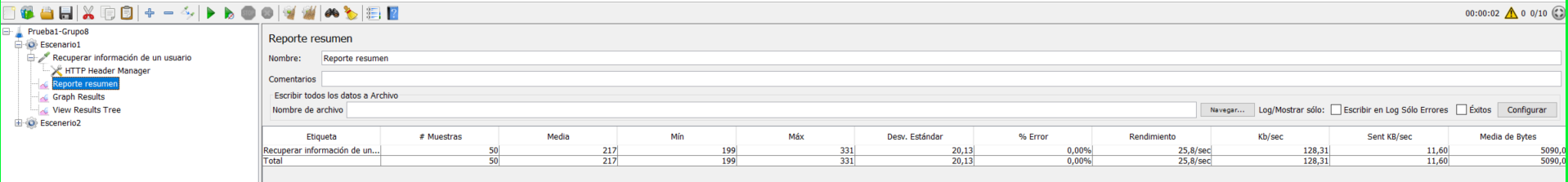


Rendimiento inicial de las instancias API y Worker

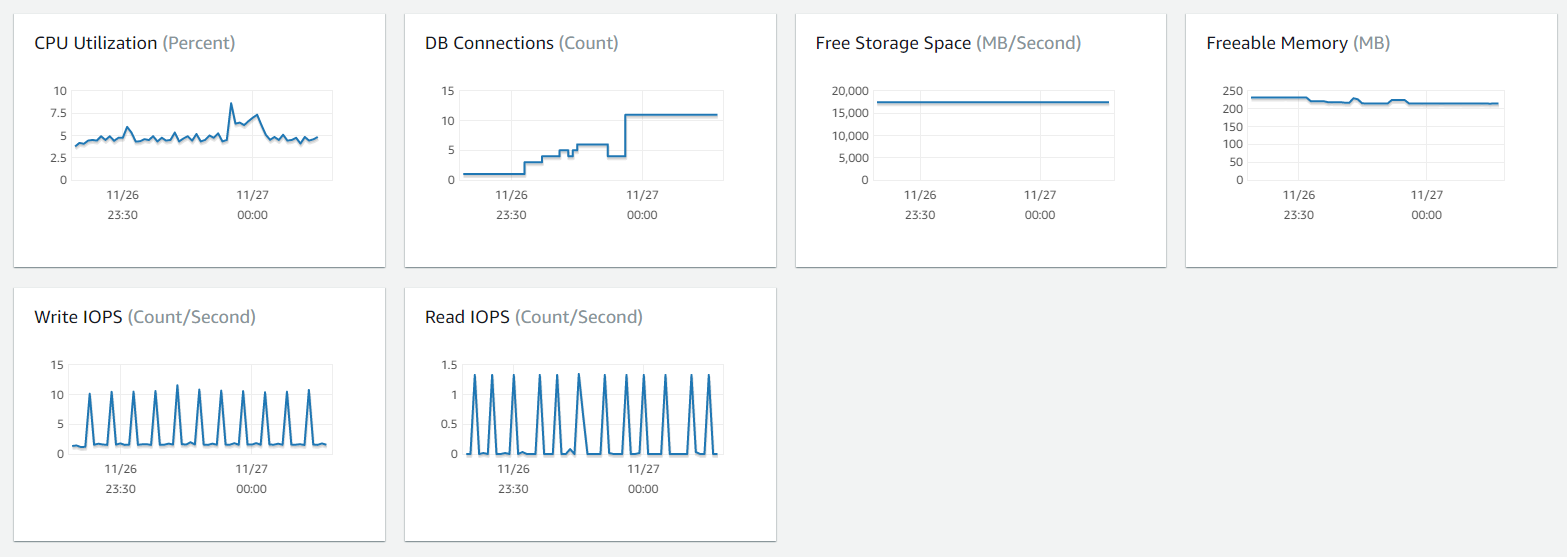


**Resultado**

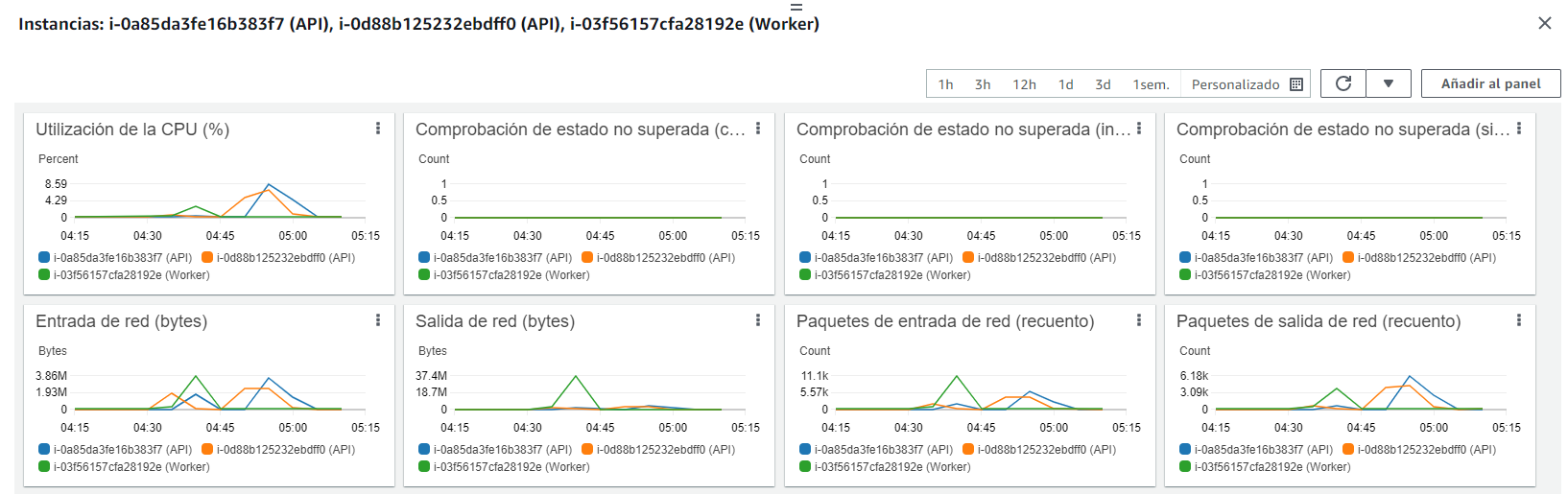
Se procesaron las 50 peticiones arrojando una Media de 217ms y un porcentaje de error de 0%



Se muestra el rendimiento del servidor de Base de Datos en el momento de la prueba de estres, donde se puede observar un leve incremento en los distintos indicadores



Se muestra el rendimiento de los servidores API y Worker en el momento de la prueba de estres, donde se puede observar un leve incremento en los distintos indicadores.



**Conclusión**

Para esta muestra cumple con creces lo esperado, pues se obtiene una media de 217ms y 0% de error.

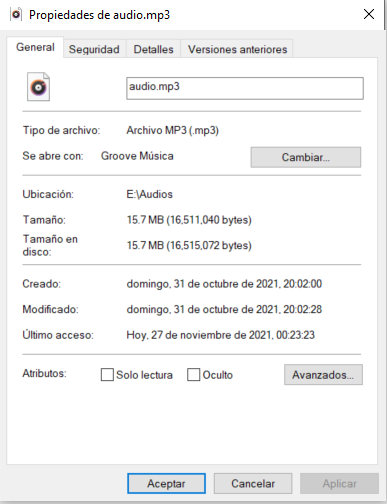
**Escenario 2.** Se deberá definir un escenario donde se pueda probar cuál es la máxima cantidad de archivos que pueden ser procesados por minuto en la aplicación. Para hacer pruebas de estrés se debe utilizar la herramienta JMeter que debe instalarse en el ambiente de nube (para no sesgar los resultados de la prueba con la latencia que introduce la red, la máquina de pruebas debe ubicarse en el mismo segmento de red de la aplicación). Las pruebas de estrés deberán realizarse desde otros equipos diferentes a los utilizados para ejecutar el servidor web y el servidor de base de datos. El escenario y los resultados de las pruebas de estrés deberán ser documentados con gráficas que ilustran cómo se comporta el sistema a medida que el número de usuarios convirtiendo archivos se incrementa, hasta llegar al punto en que el tiempo para iniciar el procesamiento de un archivo enviado por un usuario supere los 10 minutos (600 segundos).

Identifique los cuellos de botella que afectan el desempeño de la aplicación. Ejemplo cuántas transacciones soporta la instancia de base de datos seleccionada, este ejercicio lo debe realizar con cada uno de los componentes o servicios de nube que integra a la solución en la entrega vigente. Restricciones del escenario. El archivo enviado a convertir durante las pruebas debe ser de un tamaño superior a las 15 MiB.

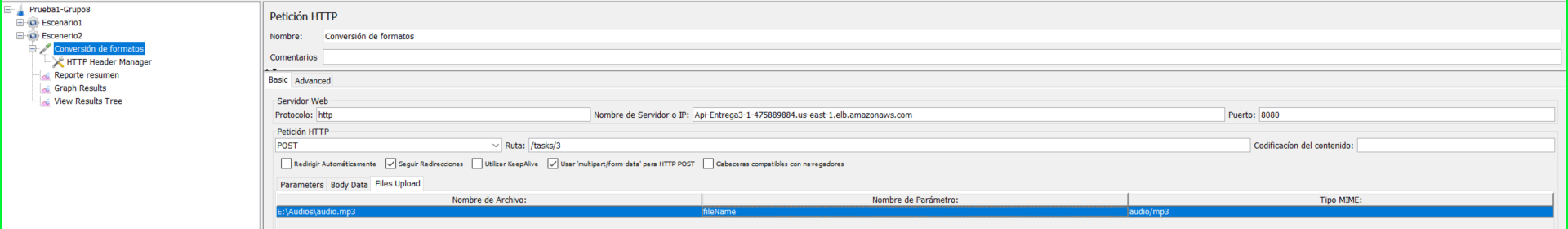
**Probamos el API “Conversión de formatos” con 200 muestras representadas en:**

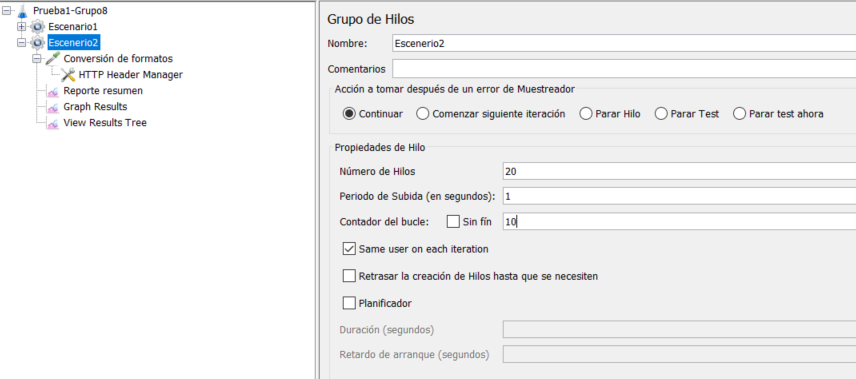
* **20 hilos**
* **1 segundo de periodo de subida**
* **Contador del bucle en 10**

Se muestra audio superior a 15Mb

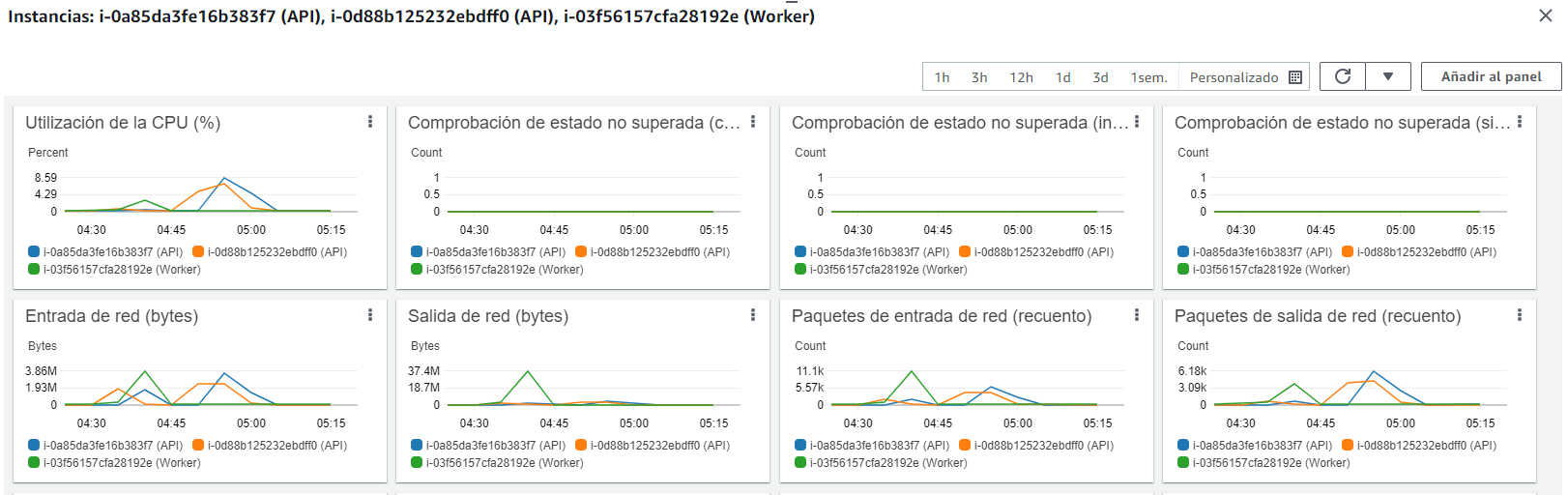


Se muestra la configuración del JMeter

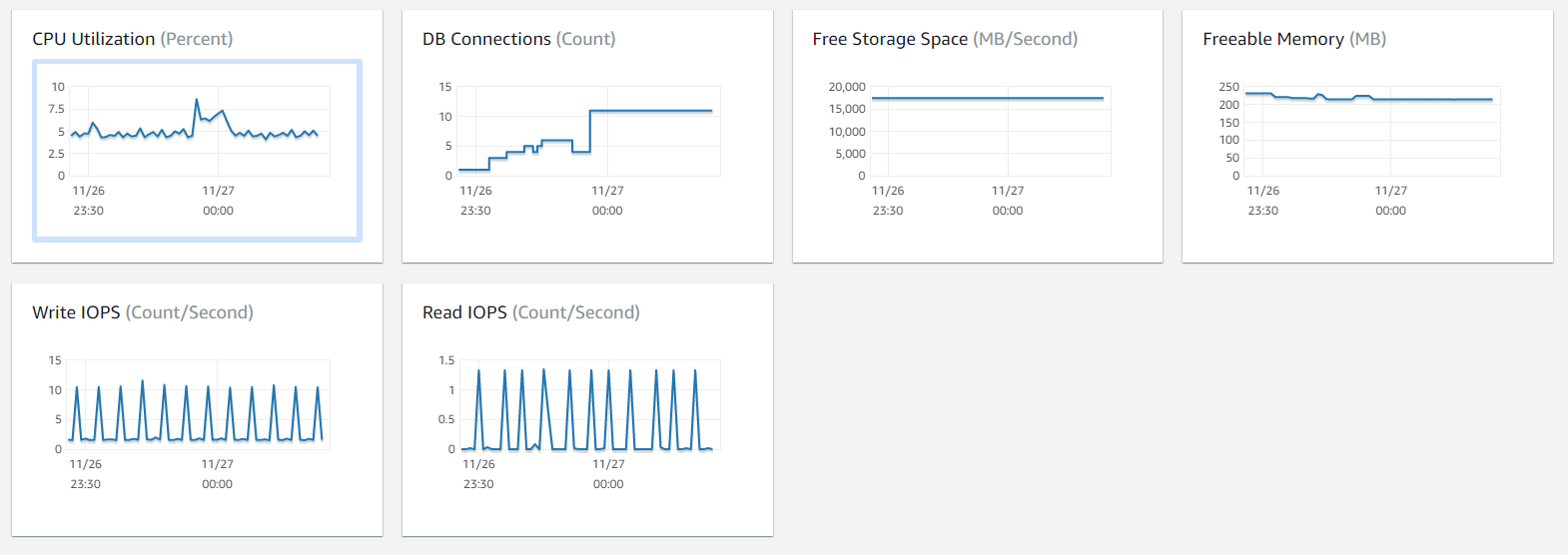




Rendimiento Inicial de los servidores

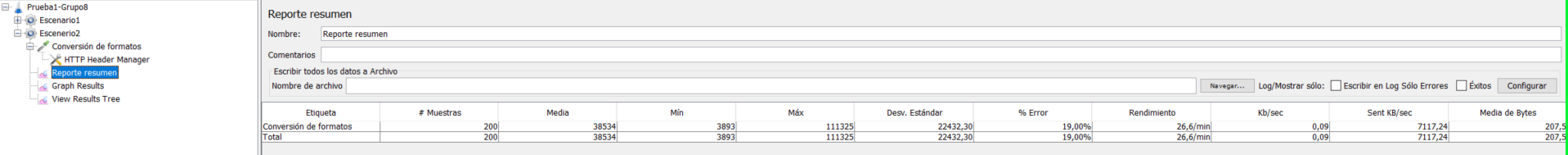


Rendimiento inicial de la Base de datos

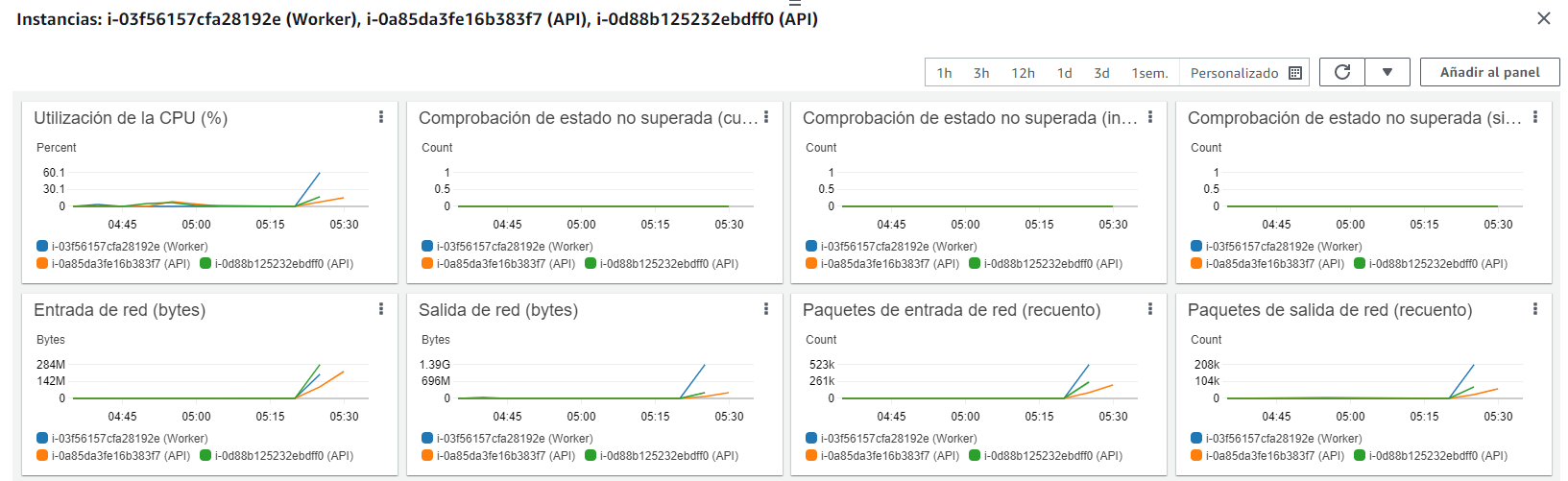


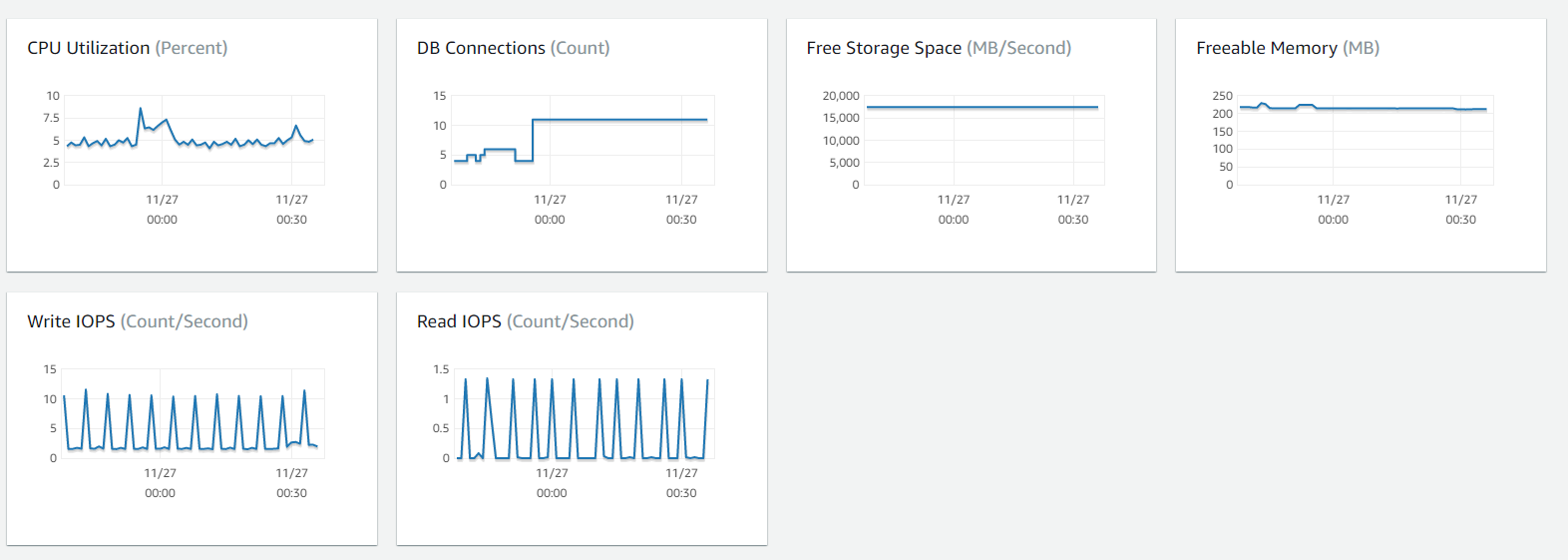
**Resultado de la prueba**

Se observa que la media es de 38534ms y tenemos un porcentaje de error de 19%



Renimiento de los servidores, se observa un incremento en los indicadores sin llegar al máximo



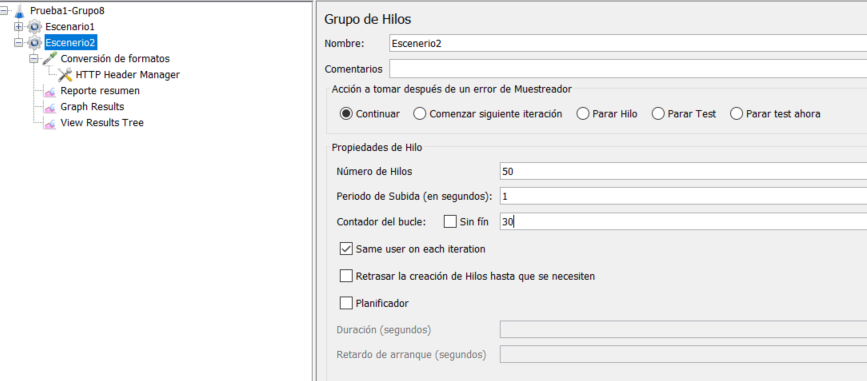


**Conclusión**: Tenemos un porcentaje de error elevado para el tipo de prueba realizado, por lo que se tiene que ajustar las configuraciones de las instancias, como política de escalamiento.

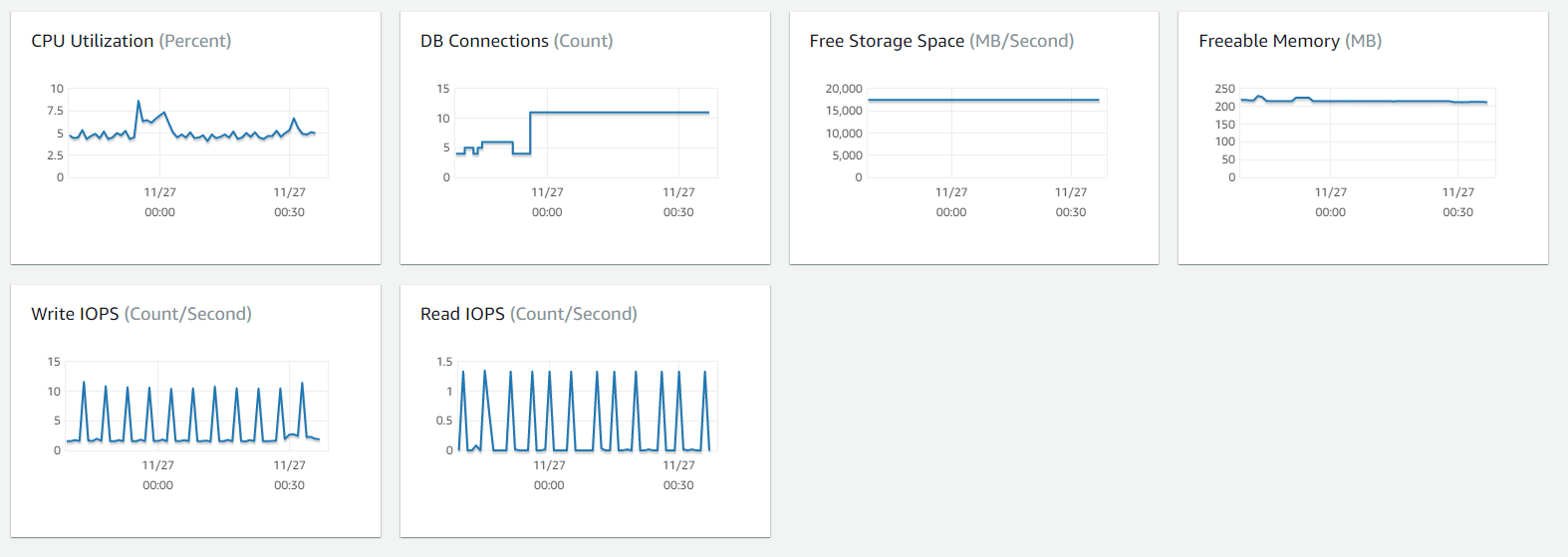
**Probamos el API “Conversión de formatos” con 1500 muestras representadas en:**

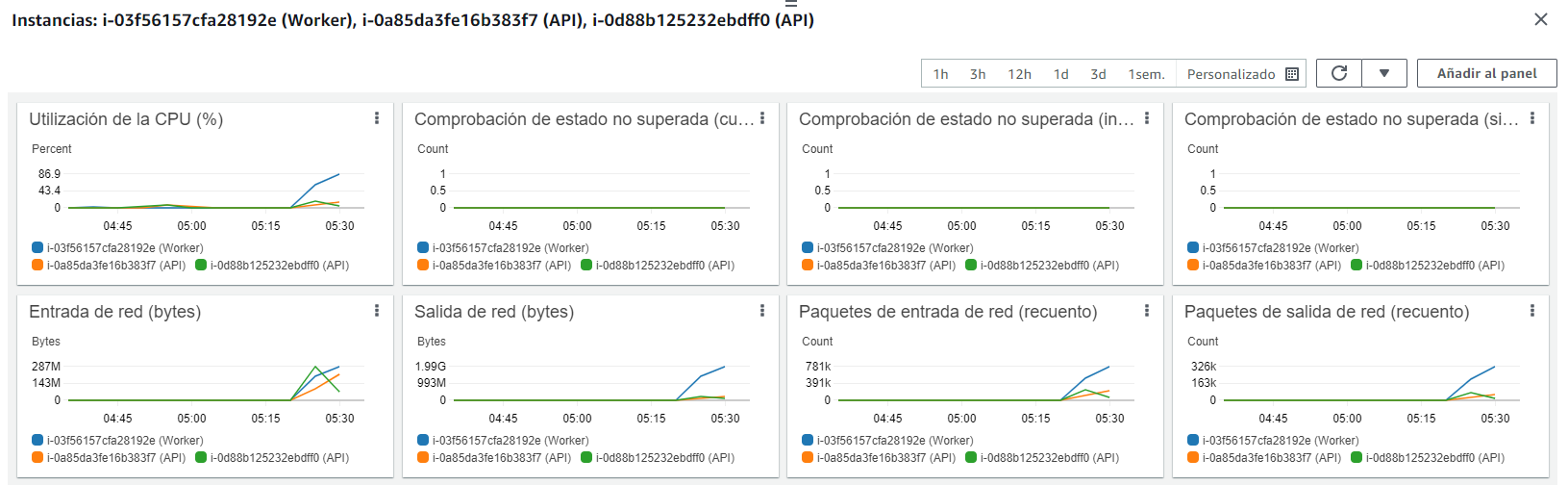
* **50 hilos**
* **1 segundo de periodo de subida**
* **Contador del bucle en 30**

Semuestra la configuración del JMeter

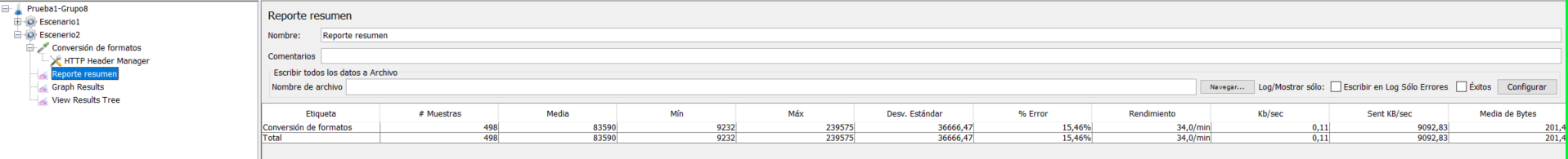


Se muestra el rendimiento inicial de los servidores

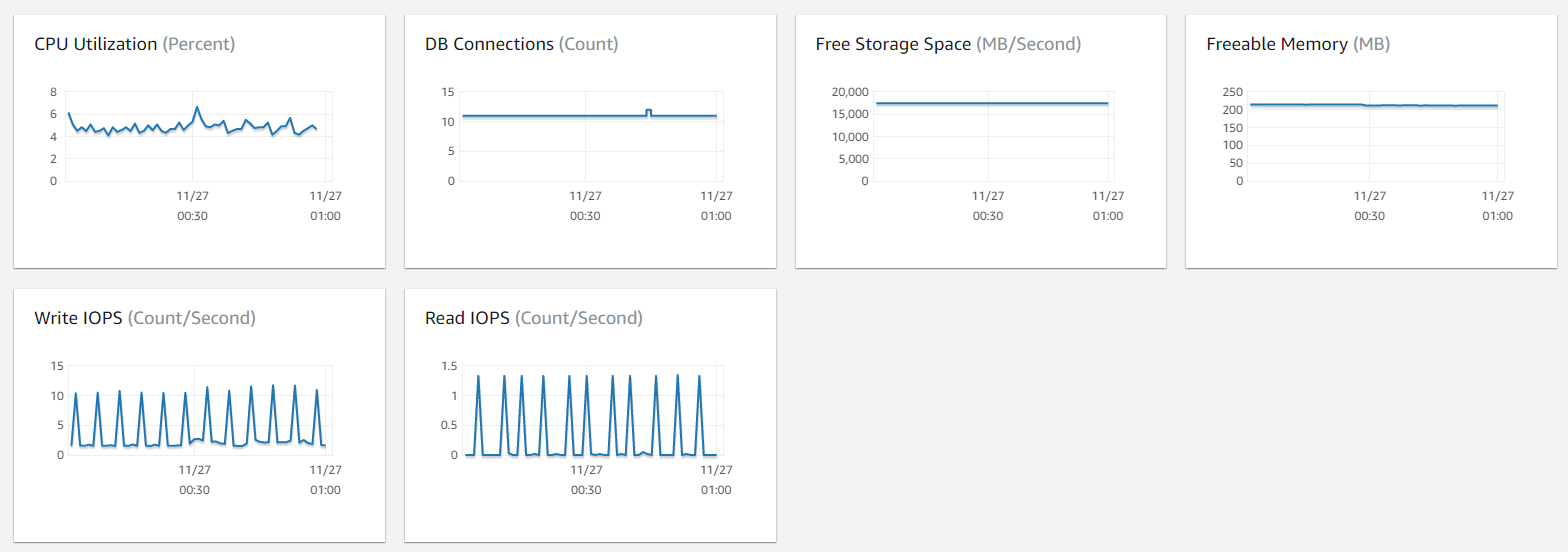


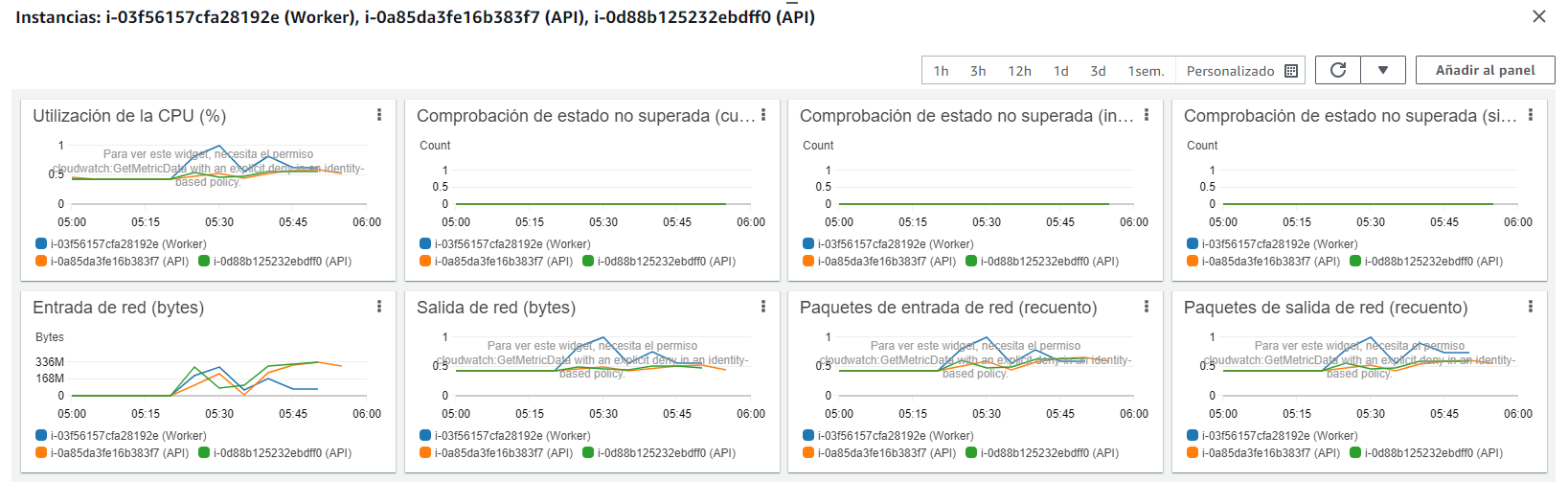


**Resultado de la prueba**

No se llegó a terminar la prueba por vencimiento del token, pero hasta los casi 500 procesos realizados vemos una media alta (83590ms) , igualmente un % de error elevado (15.46%)

Rendimiento de los servidores, se observa un incremento en los indicadores sin llegar al máximo





**Conclusión**: Tenemos un porcentaje de error elevado para el tipo de prueba realizado, por lo que se tiene que ajustar las configuraciones de las instancias, como política de escalamiento.