

Consigna

Vamos a trabajar con la ruta `"/info"` en modo `"fork"`, agregando o extrayendo un `console.log` de la información colectada antes de devolverla al cliente. Además, desactivaremos el `child_process` de la ruta `"/randoms"`.

Para ambas condiciones (Con o sin `console.log`) en la ruta `"/info"` OBTENER:

1) El perfilamiento del servidor, realizando el test con `–prof` de `node.js`. Analizar los resultados obtenidos luego de procesarlos con `–prof-process`.

Utilizaremos como test de carga Artillery en línea de comandos, emulando 50 conexiones concurrentes con 20 request por cada una. Extraer un reporte con los resultados en archivo de texto.

El informe de artillery tanto con console log como sin él, resultaron en que en promedio el servidor corriendo con console.log tardó alrededor de 399.5 de media.

Dejo los informes detallados en la carpeta `profiling/artillery`

Luego utilizaremos Autocannon en línea de comandos, emulando 100 conexiones concurrentes realizadas en un tiempo de 20 segundos. Extraer un reporte con los resultados(puede ser un print de pantalla de la consola).

2) El perfilamiento del servidor con el modo inspector de `node.js` `–inspect`. Revisar el tiempo de los procesos menos performantes sobre el archivo fuente de inspección.

En la carpeta `"/profiling/autocannon"` dejo las imágenes de la respuesta con `console.log` y sin `console.log`.

3) El diagrama de flama con `Ox`, emulando la carga con autocannon con los mismos parámetros anteriores.

En la carpeta `"/profiling/Ox"` dejo los diagramas de flama