

Clase 32

AutoCannon

```
ca. Símbolo del sistema
> test
> node tarea-clase-32/src/utils/benchmark.js

Running 20s test @ http://localhost:8080/info
100 connections
```

Stat	2.5%	50%	97.5%	99%	Avg	Stdev	Max
Latency	30 ms	95 ms	151 ms	171 ms	93.26 ms	27.45 ms	262 ms

Stat	1%	2.5%	50%	97.5%	Avg	Stdev	Min
Req/Sec	722	722	1104	1165	1064.6	103.65	722
Bytes/Sec	470 kB	470 kB	719 kB	759 kB	693 kB	67.5 kB	470 kB

```
Req/Bytes counts sampled once per second.
# of samples: 20

21k requests in 20.05s, 13.9 MB read
C:\Users\Facu\Desktop\tareas BackEnd\Facundo Ibañez-clase 4\tarea-clase-32\src>
```

El devtools nunca dio ningún tipo de sección “run” por lo tanto no se pudo realizar la observación de cuanto tardo cada función en ejecutarse. El --inspect fue realizado en la ruta **node --inspect tarea-clase-32\src\server.js** y el testeo fue realizado tanto con artillery como con autocannon. Igualmente deje el archivo del resultado del artillery con el inspect en el profiler.txt

DevTools - Node.js - Opera			
DevTools is now available in Spanish! Always match Chrome's language Switch DevTools to Spanish Don't show again			
Connection	Console	Profiler	Sources Memory
Heavy (Bottom Up)			
Profiles	Self Time	Total Time	Function
CPU PROFILES	24028.2 ms	24028.2 ms	(idle)
Profile 1 Save	4422.7 ms 37.21 %	4422.7 ms 37.21 %	▸ writeUtf8String
	3233.2 ms 27.20 %	7837.5 ms 65.94 %	▸ consoleCall
	2067.3 ms 17.39 %	2067.3 ms 17.39 %	▸ getCPUs
	249.2 ms 2.10 %	249.2 ms 2.10 %	▸ writev
	67.0 ms 0.56 %	67.0 ms 0.56 %	(garbage collector)
	65.6 ms 0.55 %	69.2 ms 0.58 %	▸ asString
	59.9 ms 0.50 %	59.9 ms 0.50 %	▸ getColorDepth
	48.9 ms 0.41 %	48.9 ms 0.41 %	▸ stringify
	42.6 ms 0.36 %	11231.8 ms 94.50 %	▸ (anonymous)
	42.3 ms 0.36 %	66.1 ms 0.56 %	▸ nextTick
	41.8 ms 0.35 %	52.0 ms 0.44 %	▸ clearIncoming
	40.4 ms 0.34 %	60.3 ms 0.51 %	▸ writeHead

Autocannon y 0x:

Generamos el flamegraph y vemos que la lo que más tiempo lleva y debemos mejorar es el ruteo de info()

