

# Desafio Clase 32

## Logs, Debug, Profiling

Guzzo Tomas

Verificar sobre la ruta /info con y sin compresión, la diferencia de cantidad de bytes devueltos en un caso y otro.

Sin gzip

Name	Status	Type	Initiator	Size	Time	Waterfall
info	200	document	Other	2.7 kB	86 ms	

Con gzip

info	200	document	Other	1.3 kB	90 ms	
------	-----	----------	-------	--------	-------	--

Test sobre la ruta /info, con y sin console.log

## Artillery

Con console.log

[Summary]:				
ticks	total	nonlib	name	
43	0.1%	97.7%	JavaScript	
0	0.0%	0.0%	C++	
27	0.1%	61.4%	GC	
29530	99.9%		Shared libraries	
1	0.0%		Unaccounted	

Sin console.log

[Summary]:				
ticks	total	nonlib	name	
58	0.6%	100.0%	JavaScript	
0	0.0%	0.0%	C++	
25	0.3%	43.1%	GC	
9039	99.4%		Shared libraries	

Test usando --inspect y con console.log

```
163 app.get('/info', compression(), (req, res) => {
164   2.3 ms const { argv, execPath, platform, version, pid, memoryUsage, cwd } = process;
165   0.3 ms const { rss } = memoryUsage();
166   5.5 ms console.log(argv,
167     execPath,
168     platform,
169     version,
170     pid,
171     rss,
172     CPUs,
173     cwd()
174   )
175
176   14.6 ms res.render("info", {
177     layout: "main",
178     argv,
179     execPath,
180     platform,
181     version,
182     pid,
183     rss,
184     CPUs,
185     0.1 ms currentDir: cwd(),
186   });
187 })
188
```

Test con Autocannon:

Sin console.log

50 connections

Stat	2.5%	50%	97.5%	99%	Avg	Stdev	Max
Latency	113 ms	357 ms	977 ms	993 ms	483.26 ms	309.13 ms	1354 ms

Stat	1%	2.5%	50%	97.5%	Avg	Stdev	Min
Req/Sec	49	49	101	150	100.75	15.98	49
Bytes/Sec	135 kB	135 kB	278 kB	412 kB	277 kB	43.9 kB	135 kB

Con console.log

50 connections

Stat	2.5%	50%	97.5%	99%	Avg	Stdev	Max
Latency	213 ms	905 ms	8723 ms	9343 ms	1560.35 ms	2041.65 ms	9561 ms

Req/Sec	0	0	22	92	31.5	30.18	3
Bytes/Sec	0 B	0 B	60.5 kB	253 kB	86.6 kB	82.9 kB	8.24 kB

# Test con 0x:

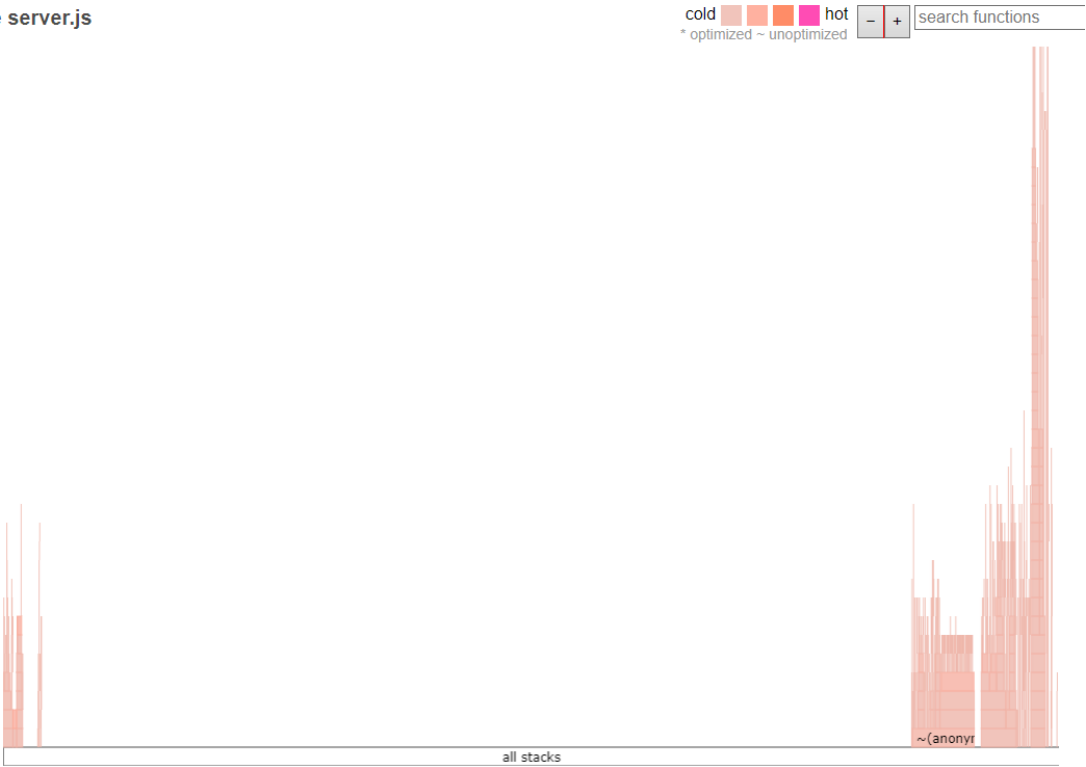
Con console.log

node server.js



Sin console.log

node server.js



## **Conclusión:**

La herramienta Gzip me parece muy útil ya que optimiza notablemente la ruta. El test con `--inspect` no fue de lo mejor para mí ya que tuve complicaciones a cuando lo quise probar sin `console.log`, me aparecía en la consola que se estaban ejecutando GET de las rutas `/json` y `/versión/json` cosa que la primera vez no me sucedió y se pudo ver bien pero no supe solucionarlo luego.