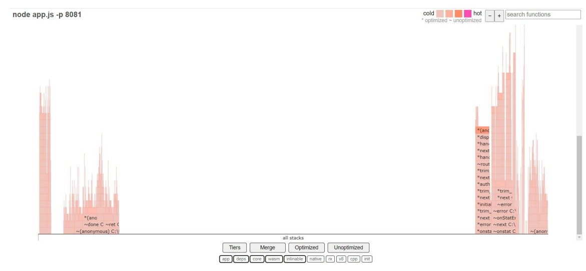
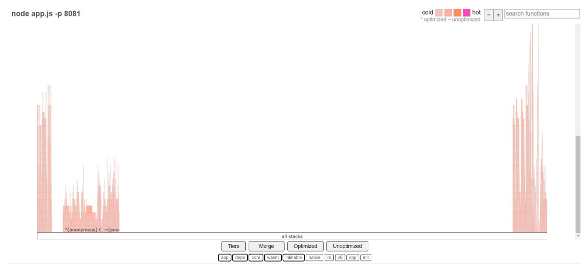
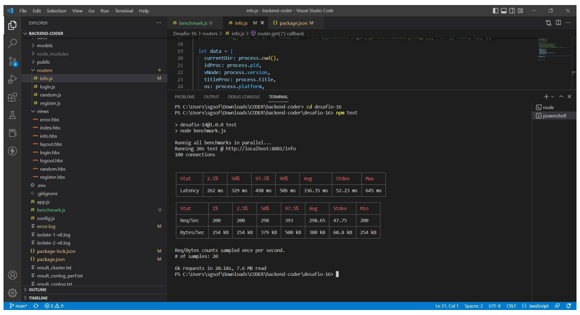
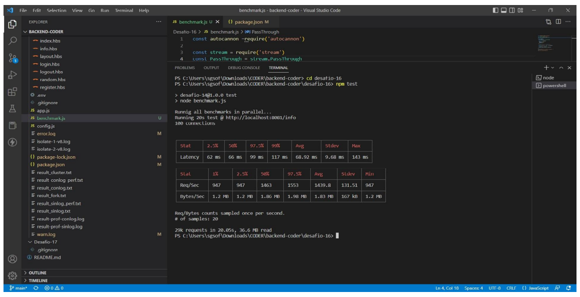
Probando /info

Con Log	Sin Log
Metrics for period to: 15:39:00(-0300) (width: 1.636s) response - median: 30.3	Metrics for period to: 15:38:00(-0300) (width: 0.508s) response - median: 25.8

`[Summary]:`
`   ticks  total  nonlib   name`
`     18    0.2%  100.0%  JavaScript`
`      0    0.0%    0.0%  C++`
`     19    0.2%  105.6%  GC`
`  11347   99.8%          Shared libraries`
	`[Summary]:`
	`   ticks  total  nonlib   name`
	`     17    0.0%  100.0%  JavaScript`
	`      0    0.0%    0.0%  C++`
	`     10    0.0%   58.8%  GC`
	`  69199  100.0%          Shared libraries`

Con Log:

```
let data = {
  currentDir: process.cwd(),
  idProc: process.pid,
  vNode: process.version,
  titleProc: process.title,
  os: process.platform,
```

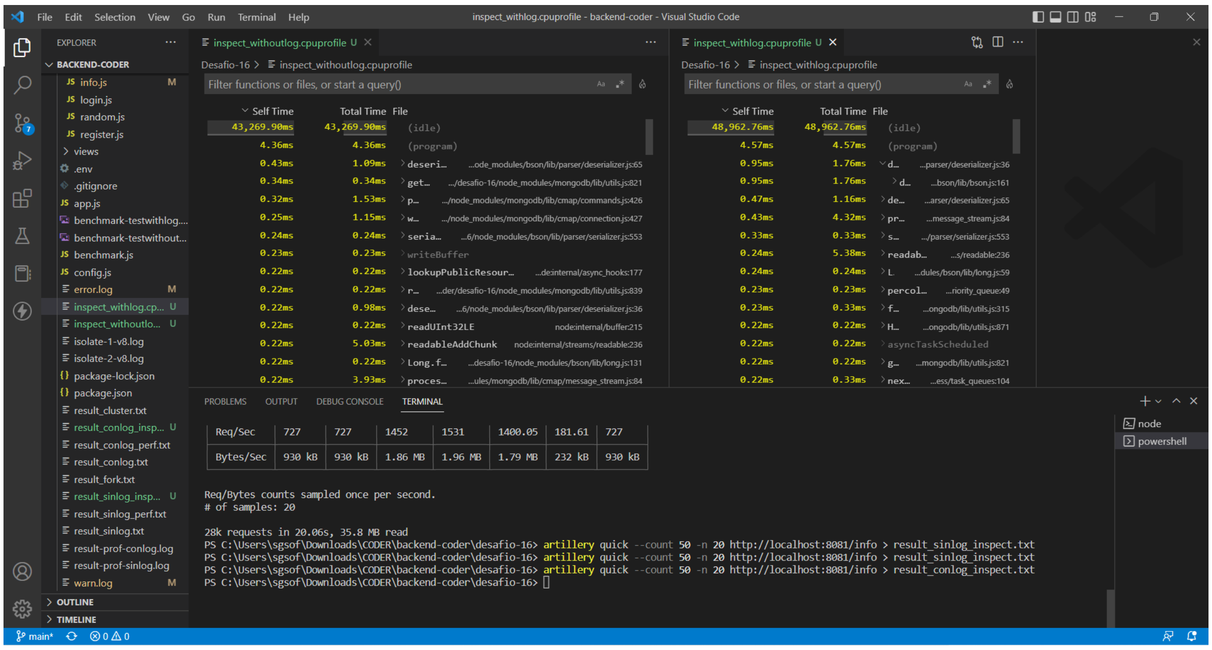
```
PS C:\Users\sgsof\Downloads\CODER\backend-coder> cd desafio-16
PS C:\Users\sgsof\Downloads\CODER\backend-coder\desafio-16> npm test

> desafio-14@1.0.0 test
> node benchmark.js

Runnig all benchmarks in parallel...
Running 20s test @ http://localhost:8081/info
100 connections

┌─────────┬────────┬────────┬────────┬────────┬───────────┬──────────┬────────┐
│ Stat    │ 2.5%   │ 50%    │ 97.5%  │ 99%    │ Avg       │ Stdev    │ Max    │
├─────────┼────────┼────────┼────────┼────────┼───────────┼──────────┼────────┤
│ Latency │ 262 ms │ 329 ms │ 498 ms │ 506 ms │ 336.35 ms │ 52.23 ms │ 645 ms │
└─────────┴────────┴────────┴────────┴────────┴───────────┴──────────┴────────┘
┌───────────┬────────┬────────┬────────┬────────┬────────┬─────────┬────────┐
│ Stat      │ 1%     │ 2.5%   │ 50%    │ 97.5%  │ Avg    │ Stdev   │ Min    │
├───────────┼────────┼────────┼────────┼────────┼────────┼─────────┼────────┤
│ Req/Sec   │ 200    │ 200    │ 298    │ 393    │ 298.65 │ 47.75   │ 200    │
├───────────┼────────┼────────┼────────┼────────┼────────┼─────────┼────────┤
│ Bytes/Sec │ 254 kB │ 254 kB │ 379 kB │ 500 kB │ 380 kB │ 60.8 kB │ 254 kB │
└───────────┴────────┴────────┴────────┴────────┴────────┴─────────┴────────┘

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

6k requests in 20.18s, 7.6 MB read
PS C:\Users\sgsof\Downloads\CODER\backend-coder\desafio-16>
```

Sin Log:

```
const autocannon =require('autocannon')

const stream = require('stream')
const PassThrough = stream.PassThrough
```

```
PS C:\Users\sgsof\Downloads\CODER\backend-coder> cd desafio-16
PS C:\Users\sgsof\Downloads\CODER\backend-coder\desafio-16> npm test

> desafio-14@1.0.0 test
> node benchmark.js

Runnig all benchmarks in parallel...
Running 20s test @ http://localhost:8081/info
100 connections

┌─────────┬───────┬───────┬───────┬────────┬──────────┬─────────┬────────┐
│ Stat    │ 2.5%  │ 50%   │ 97.5% │ 99%    │ Avg      │ Stdev   │ Max    │
├─────────┼───────┼───────┼───────┼────────┼──────────┼─────────┼────────┤
│ Latency │ 62 ms │ 66 ms │ 99 ms │ 117 ms │ 68.92 ms │ 9.68 ms │ 143 ms │
└─────────┴───────┴───────┴───────┴────────┴──────────┴─────────┴────────┘
┌───────────┬────────┬────────┬─────────┬─────────┬─────────┬────────┬────────┐
│ Stat      │ 1%     │ 2.5%   │ 50%     │ 97.5%   │ Avg     │ Stdev  │ Min    │
├───────────┼────────┼────────┼─────────┼─────────┼─────────┼────────┼────────┤
│ Req/Sec   │ 947    │ 947    │ 1463    │ 1553    │ 1439.8  │ 131.51 │ 947    │
├───────────┼────────┼────────┼─────────┼─────────┼─────────┼────────┼────────┤
│ Bytes/Sec │ 1.2 MB │ 1.2 MB │ 1.86 MB │ 1.98 MB │ 1.83 MB │ 167 kB │ 1.2 MB │
└───────────┴────────┴────────┴─────────┴─────────┴─────────┴────────┴────────┘

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

29k requests in 20.05s, 36.6 MB read
PS C:\Users\sgsof\Downloads\CODER\backend-coder\desafio-16>
```

Compare chrome://inspect


inspect_withlog.cpuprofile - backend-coder - Visual Studio Code
inspect_withoutlog.cpuprofile U
Desafio-16 > inspect_withoutlog.cpuprofile
inspect_withlog.cpuprofile U
Desafio-16 > inspect_withlog.cpuprofile
Filter functions or files, or start a query()

inspect_withoutlog.cpuprofile

Self Time	Total Time	File	
43,269.90ms	43,269.90ms	(idle)	
4.36ms	4.36ms	(program)	
0.43ms	1.09ms	deseri…	…ode_modules/bson/lib/parser/deserializer.js:65
0.34ms	0.34ms	get…	…/desafio-16/node_modules/mongodb/lib/utils.js:821
0.32ms	1.53ms	p…	…/node_modules/mongodb/lib/cmap/commands.js:426
0.25ms	1.15ms	w…	…/node_modules/mongodb/lib/cmap/connection.js:427
0.24ms	0.24ms	seria…	…6/node_modules/bson/lib/parser/serializer.js:553
0.23ms	0.23ms	writeBuffer	
0.22ms	0.22ms	lookupPublicResour…	…de:internal/async_hooks:177
0.22ms	0.22ms	r…	…der/desafio-16/node_modules/mongodb/lib/utils.js:839
0.22ms	0.98ms	dese…	…6/node_modules/bson/lib/parser/deserializer.js:36
0.22ms	0.22ms	readUInt32LE	node:internal/buffer:215
0.22ms	5.03ms	readableAddChunk	node:internal/streams/readable:236
0.22ms	0.22ms	Long.f…	…desafio-16/node_modules/bson/lib/long.js:131
0.22ms	3.93ms	proces…	…ules/mongodb/lib/cmap/message_stream.js:84

inspect_withlog.cpuprofile

Self Time	Total Time	File	
48,962.76ms	48,962.76ms	(idle)	
4.57ms	4.57ms	(program)	
0.95ms	1.76ms	d…	…parser/deserializer.js:36
0.95ms	1.76ms	d…	…bson/lib/bson.js:161
0.47ms	1.16ms	de…	…arser/deserializer.js:65
0.43ms	4.32ms	pr…	…message_stream.js:84
0.33ms	0.33ms	s…	…/parser/serializer.js:553
0.24ms	5.38ms	readab…	…s/readable:236
0.24ms	0.24ms	L.	…dules/bson/lib/long.js:59
0.23ms	0.23ms	percol…	…riority_queue:49
0.23ms	0.33ms	f…	…ongodb/lib/utils.js:315
0.22ms	0.22ms	H…	…ongodb/lib/utils.js:871
0.22ms	0.22ms	asyncTaskScheduled	
0.22ms	0.22ms	g…	…mongodb/lib/utils.js:821
0.22ms	0.33ms	nex…	…ess/task_queues:104

```
│ Req/Sec   │ 727    │ 727    │ 1452    │ 1531    │ 1400.05 │ 181.61 │ 727    │
│ Bytes/Sec │ 930 kB │ 930 kB │ 1.86 MB │ 1.96 MB │ 1.79 MB │ 232 kB │ 930 kB │

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

28k requests in 20.06s, 35.8 MB read
PS C:\Users\sgsof\Downloads\CODER\backend-coder\desafio-16> artillery quick --count 50 -n 20 http://localhost:8081/info > result_sinlog_inspect.txt
PS C:\Users\sgsof\Downloads\CODER\backend-coder\desafio-16> artillery quick --count 50 -n 20 http://localhost:8081/info > result_sinlog_inspect.txt
PS C:\Users\sgsof\Downloads\CODER\backend-coder\desafio-16> artillery quick --count 50 -n 20 http://localhost:8081/info > result_conlog_inspect.txt
PS C:\Users\sgsof\Downloads\CODER\backend-coder\desafio-16> 
```

El console.log es una actividad bloqueante por eso el comportamiento de nuestro servidor cuando agregamos o quitamos los logs.