

## Con console.log(info)

Artillery:

http.response\_time:

min: .....3  
max: .....1191  
median: .....572.6  
p95: .....1153.1  
p99: .....1176.4

Profiler:

[Summary]:

ticks	total	nonlib	name
22	0.5%	100.0%	JavaScript
0	0.0%	0.0%	C++
50	1.1%	227.3%	GC
4449	99.5%		Shared libraries

```
15 cantProcesadores: config.cantProcesadores
16 }
17
18 router.get('/info', (req, res) => {
19   console.log(info)
20   res.render(path.join(process.cwd(), 'src/views/info.ejs'), { info })
21   logger.info(`${req.method} ${req.path}`)
22 })
23
24 module.exports = router
```

Stat	2.5%	50%	97.5%	99%	Avg	Stdev	Max
Latency	613 ms	1077 ms	3221 ms	3846 ms	1266.64 ms	593.29 ms	4114 ms

Stat	1%	2.5%	50%	97.5%	Avg	Stdev	Min
Req/Sec	10	10	85	122	76.91	31.6	10
Bytes/Sec	19.4 kB	19.4 kB	165 kB	237 kB	149 kB	61.3 kB	19.4 kB

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

2k requests in 20.15s, 2.98 MB read

## Sin console.log(info)

Artillery:

http.response\_time:

min: .....2  
max: .....1108  
median: .....757.6  
p95: .....963.1  
p99: .....1085.9

Profiler:

[Summary]:

ticks	total	nonlib	name
25	0.4%	100.0%	JavaScript
0	0.0%	0.0%	C++
41	0.6%	164.0%	GC
6472	99.6%		Shared libraries

```
15 cantProcesadores: config.cantProcesadores
16 }
17
18 router.get('/info', (req, res) => {
19   res.render(path.join(process.cwd(), 'src/views/info.ejs'), { info })
20   logger.info(`${req.method} ${req.path}`)
21 })
22
23 module.exports = router
```

Stat	2.5%	50%	97.5%	99%	Avg	Stdev	Max
Latency	360 ms	1020 ms	2665 ms	3019 ms	1145.39 ms	443.34 ms	3290 ms

Stat	1%	2.5%	50%	97.5%	Avg	Stdev	Min
Req/Sec	10	10	94	105	84.4	25.24	10
Bytes/Sec	19.4 kB	19.4 kB	182 kB	204 kB	164 kB	48.9 kB	19.4 kB

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

2k requests in 20.11s, 3.27 MB read

