

Análisis de Performance con Artillery

Iniciamos el servidor en una terminal sea en modo cluster como en modo fork y comparamos los resultados.

```
File Edit Selection View Go Run Terminal Help
result_cluster U X
K result_cluster
1 Running scenarios...
2 Phase started: unnamed (index: 0, duration: 1s) 16:23:30(-0300)
3
4 Phase completed: unnamed (index: 0, duration: 1s) 16:23:39(-0300)
5
6 All VUs finished, Total time: 15 seconds
7
8
9 Summary report @ 16:23:51(-0300)
10
11
12 http.codes.200: ..... 1000
13 http.request_rate: ..... 50/sec
14 http.requests: ..... 1000
15 http.response_time:
16 min: ..... 49
17 max: ..... 1000
18 median: ..... 85.6
19 p95: ..... 689.5
20 p99: ..... 884.5
21 http.responses: ..... 1000
22 vusers.completed: ..... 20
23 vusers.created: ..... 20
24 vusers.created_by_name.0: ..... 20
25 vusers.failed: ..... 0
26 vusers.session_length:
27 min: ..... 8358.1
28 max: ..... 10015.2
29 median: ..... 9607.1
30 p95: ..... 9999.2
31 p99: ..... 9999.2
32
```

```
File Edit Selection View Go Run Terminal Help
result_fork U X
K result_fork
4 Phase completed: unnamed (index: 0, duration: 1s) 16:22:11(-0300)
5
6 All VUs finished, Total time: 15 seconds
7
8
9 Summary report @ 16:22:23(-0300)
10
11
12 http.codes.200: ..... 1000
13 http.request_rate: ..... 45/sec
14 http.requests: ..... 1000
15 http.response_time:
16 min: ..... 49
17 max: ..... 982
18 median: ..... 87.4
19 p95: ..... 671.8
20 p99: ..... 727.9
21 http.responses: ..... 1000
22 vusers.completed: ..... 20
23 vusers.created: ..... 20
24 vusers.created_by_name.0: ..... 20
25 vusers.failed: ..... 0
26 vusers.session_length:
27 min: ..... 8354.9
28 max: ..... 9985.8
29 median: ..... 9416.8
30 p95: ..... 9801.2
31 p99: ..... 9801.2
32
```

Como podemos ver en el siguiente cuadro:

	Modo CLUSTER	Modo FORK
Total Time	15 sec	15 sec
Request Rate	50/sec	45/sec
Response_time median	85.6	87.4
Vusers.session.length median	9607.1	9416.8

```
C:\Users\flore\Desktop\node\entrega3>curl -X POST -d "username=Pepe@mail.com&password=perz" http://localhost:8080/login
Found. Redirecting to /api/productos
C:\Users\flore\Desktop\node\entrega3>artillery quick "http://localhost:8080/api/productos" > result_cluster.txt
```

```
curl -X POST -d "username=Pepe@mail.com&password=perz" http://localhost:8080/login
```

```
artillery quick --count=20 --num=50 "http://localhost:8080/api/productos" > result_fork.txt
```

Vemos que los resultados obtenidos son muy similares. Obteniendo una mayor cantidad de request al segundo en cluster, teniendo un mejor tiempo de respuesta medio.