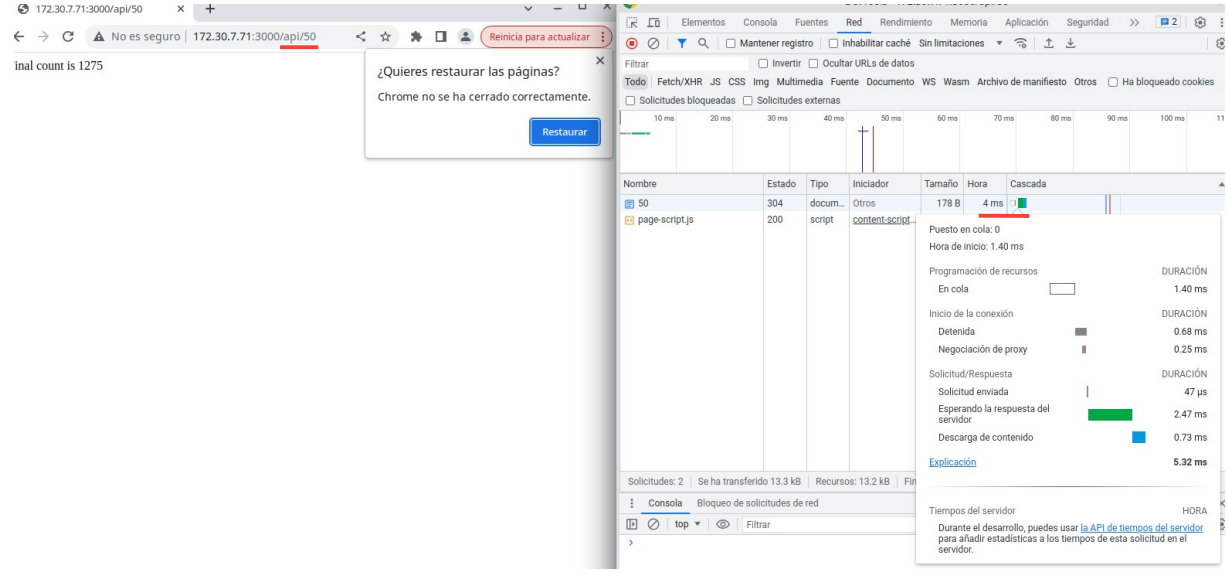
Práctica 3.3

Miguel Pozo Pérez

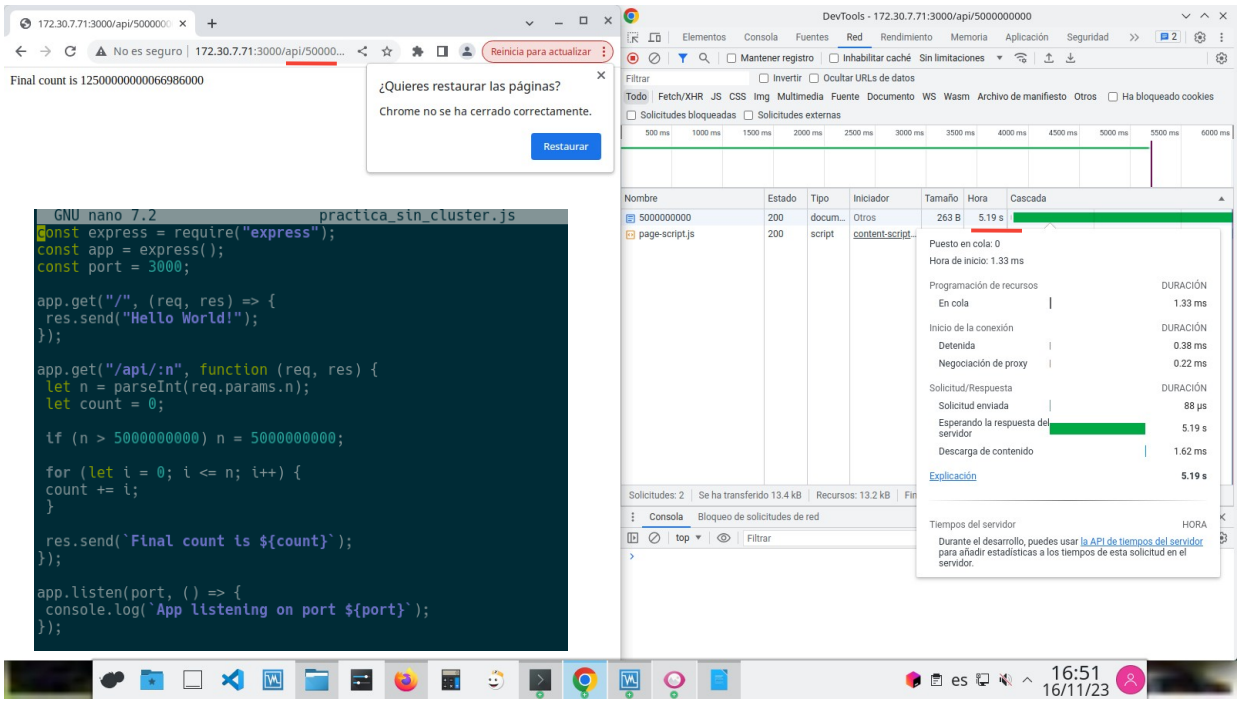
Índice

Pruebas sin clúster

Vamos a hacer dos comprobaciones para ver la ejecución y lo rápido que es, ahora vamos a poner un valor pequeño

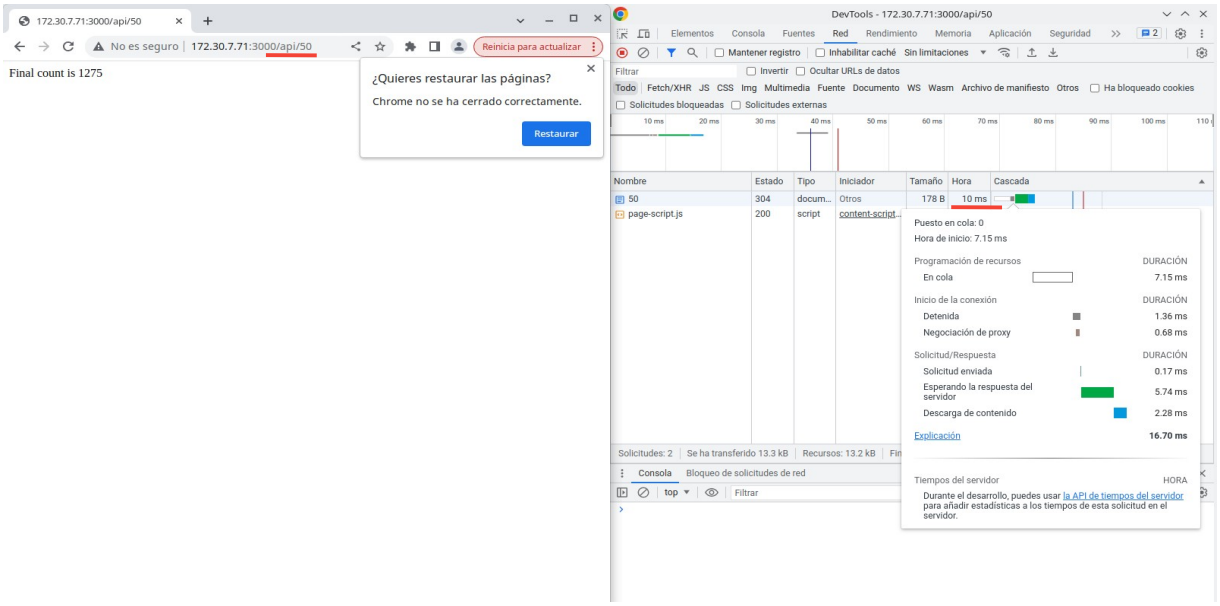


Ahora vamos a comprobar la velocidad con un valor más grande, como podemos comprobar tarda bastante más.

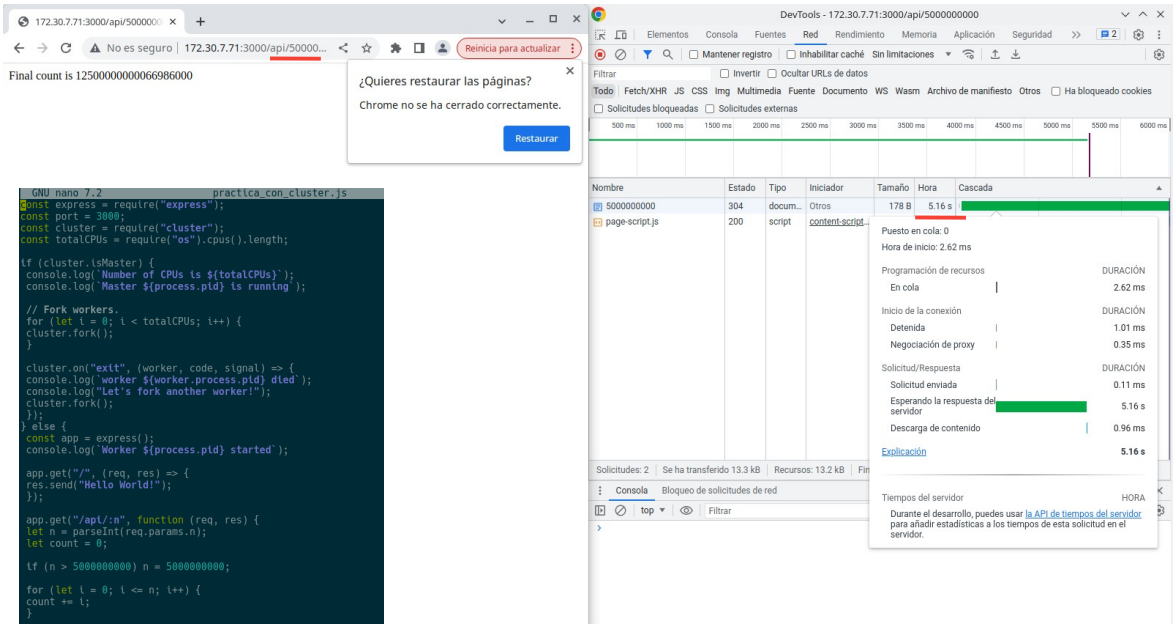


Pruebas con clúster

Ahora vamos a comprobar la practica anterior pero usaremos el módulo clúster, con un valor pequeño nos tarda poco.



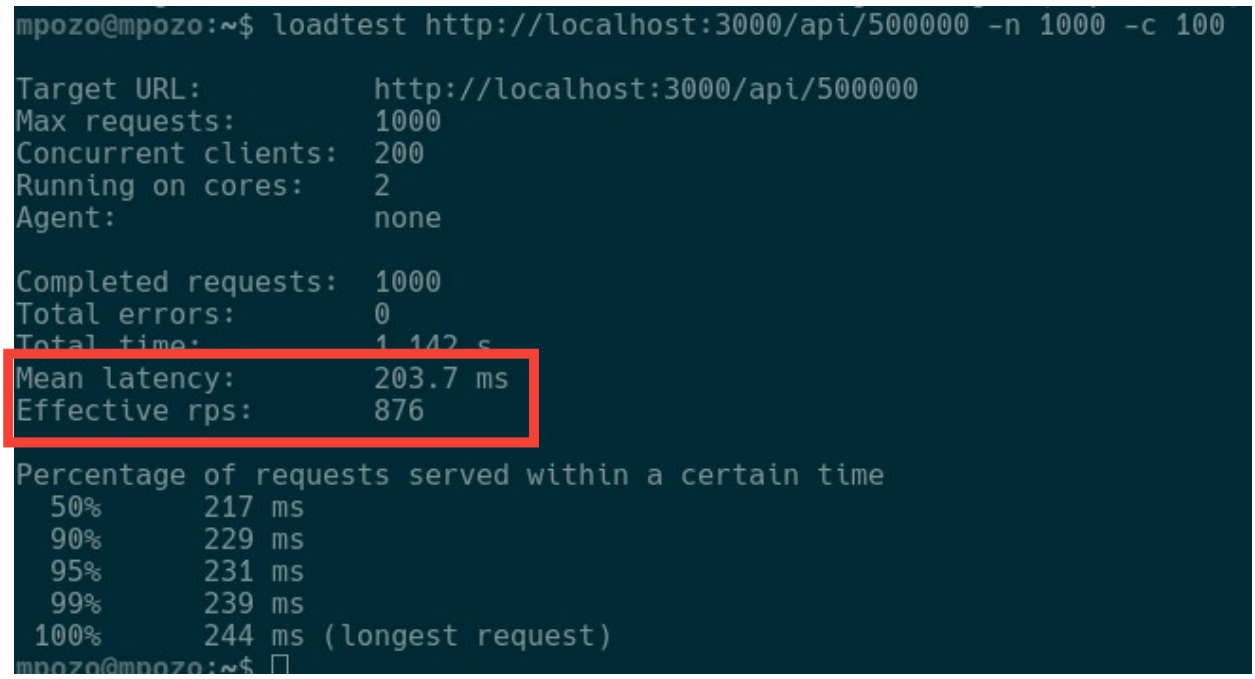
A continuación vamos a probar con un valor más grande, tendría que reducirse el tiempo porque cuando usamos el modulo clúster el proceso lo divide en los módulos de la CPU peor por algún motivo de la maquina virtual solo esta utilizando un núcleo y por eso nos esta tardando la solicitud.



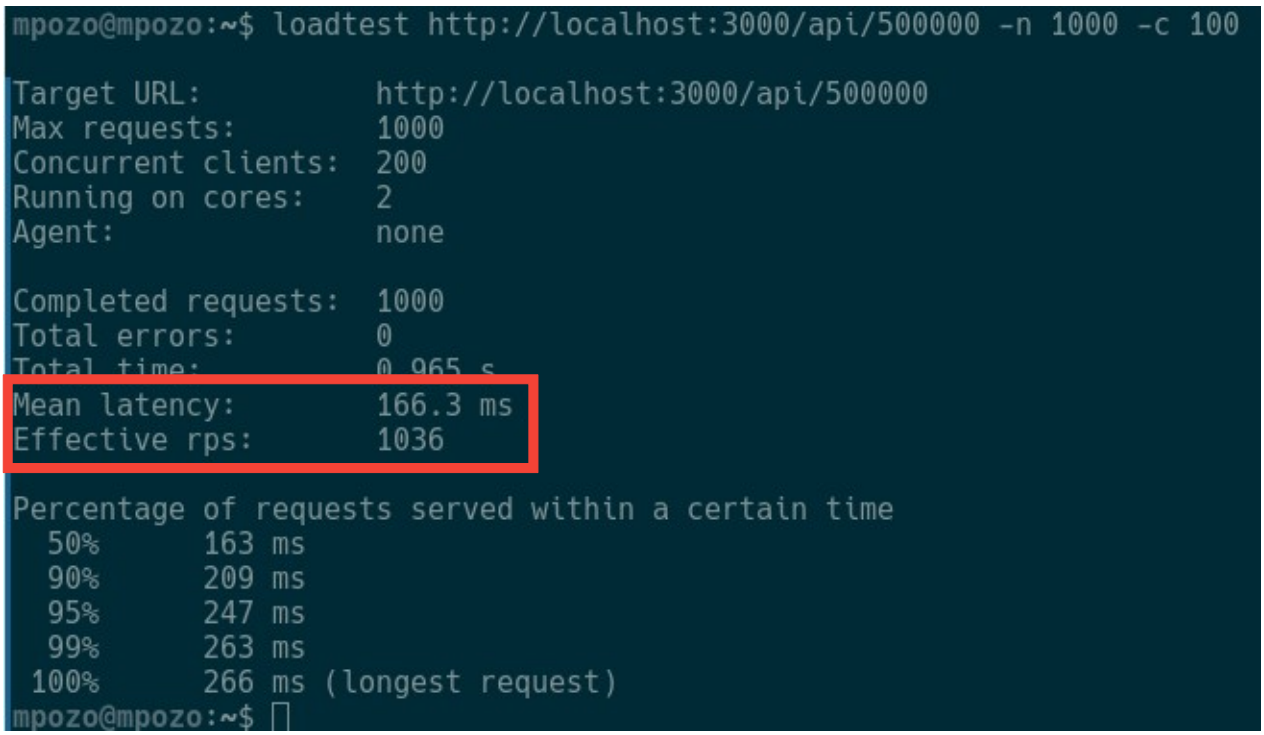
Pruebas con loadtest

Para la siguiente practica nos tendremos que instalar loadtest y ejecutar la aplicación sin el clúster y con el clúster, por algún motivo de la maquina virtual nos salen resultados parecidos.

Sin clúster



Con clúster



Pruebas con PM2

Para la última practica vamos a instalar PM2, a continuación ejecutaremos el js sin clusterizar.

```
mpozo@mpozo:~/Documentos/sinCluster$ pm2 start Node.js -i 0
[PM2] Applying action restartProcessId on app [Node](ids: [ 0, 1 ])
[PM2] [Node](0) ✓
[PM2] [Node](1) ✓
[PM2] Process successfully started

 id   name                 mode       ↺      status      cpu         memory

 0    Node                 fork       1      online      0%          39.0mb
 1    Node                 fork       30     online      0%          13.0mb

mpozo@mpozo:~/Documentos/sinCluster$
```

Podemos detener la aplicación con stop.

```
mpozo@mpozo:~/Documentos/sinCluster$ pm2 stop Node.js
[PM2] Applying action stopProcessId on app [Node.js](ids: [ 0, 1 ])
[PM2] [Node](0) ✓
[PM2] [Node.js](1) ✓

 id   name                 mode       ↺      status      cpu         memory

 0    Node                 fork       1      stopped     0%          0b
 1    Node                 fork       45     stopped     0%          0b

mpozo@mpozo:~/Documentos/sinCluster$
```

Ahora vamos a crear un archivo de configuración llamado Ecosystem para establecer configuraciones especificas y que nos clusterizar solo nuestros archivos.

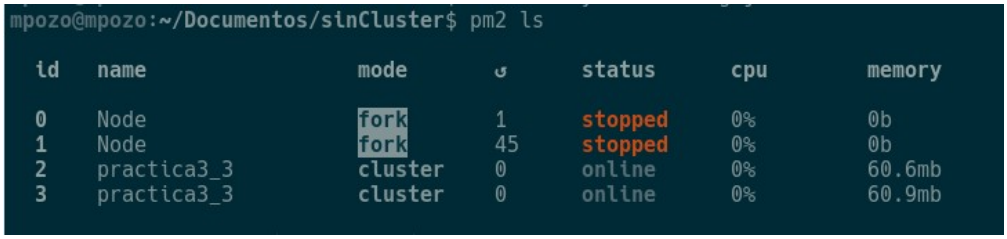
```
  GNU nano 7.2                    ecosystem.config.js
module.exports = {
 apps: [
 {
 name: "practica3_3",
 script: "Node.js",
 instances: 0,
 exec_mode: "cluster",
 },
 ],
};
```

Y podemos iniciar directamente ecosystem.

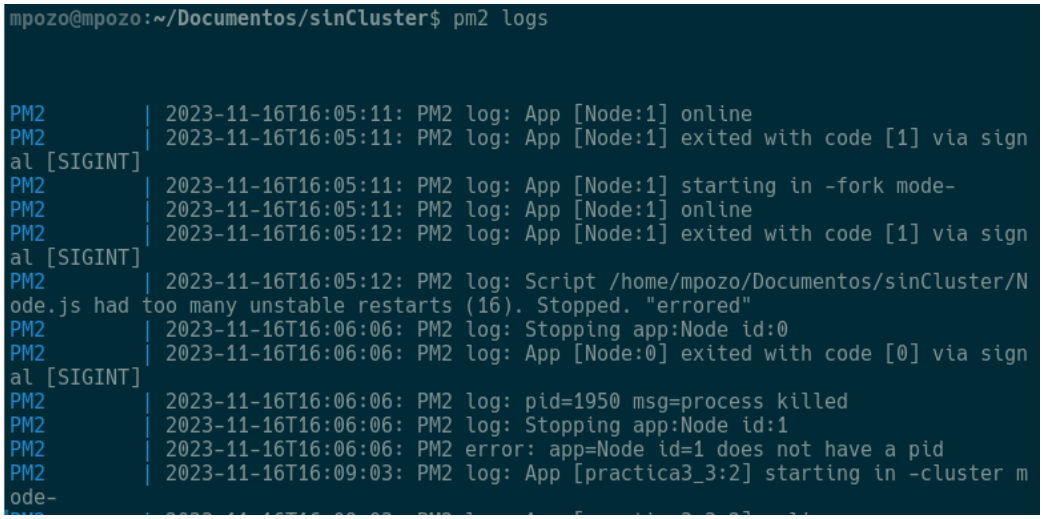
```
mpozo@mpozo:~/Documentos/sinCluster$ pm2 ls

 id   name                 mode       ↺      status      cpu         memory

 0    Node                 fork       1      stopped     0%          0b
 1    Node                 fork       45     stopped     0%          0b
 2    practica3_3          cluster    0      online      0%          60.6mb
 3    practica3_3          cluster    0      online      0%          60.9mb
```

Tarea

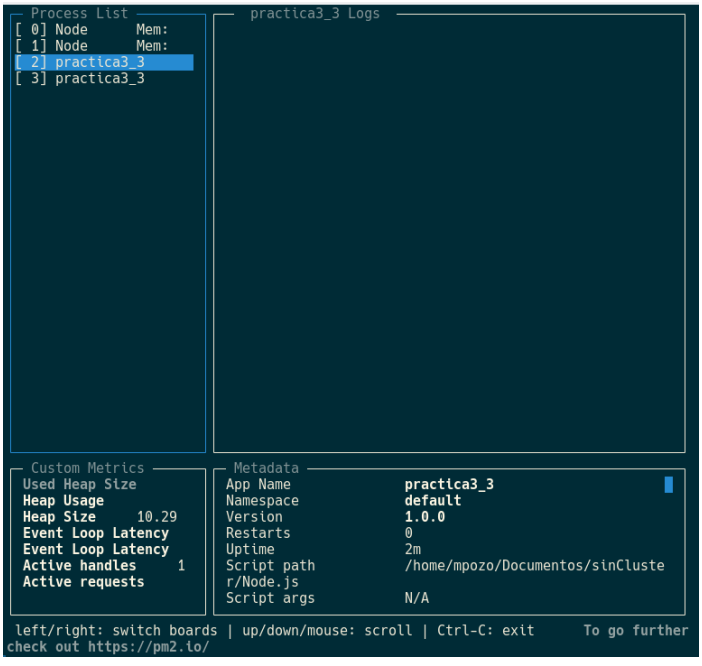
pm2 ls: Nos muestra una lista de todos los procesos de PM2 con detalles avanzados.



pm2 logs: Muestra los registros en tiempo real.



pm2 monit: Inicia un panel de control en tiempo real con información de cada aplicación.



Cuestiones

¿Sabrías decir por qué en algunos casos concretos, como este, la aplicación sin clusterizar tiene mejores resultados?

Estará utilizando un sistema mononúcleo donde sea ineficiente clusterizar ya que Node.js se ejecuta en un solo hilo.