

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS

**GABRIEL HENRIQUE MOTA RODRIGUES - 814411**

TESTE DESEMPENHO

BELO HORIZONTE - 2025

## Resumo Executivo

Os testes de desempenho foram realizados em dois cenários distintos da API:

### 1. Cenário CPU-bound – /crypto

A rota /crypto envolve operações intensivas de CPU.

**Capacidade máxima sustentada:  $\approx$  135 VUs**

Acima desse valor, a aplicação começou a apresentar falhas durante o teste de estresse, indicando que o processamento criptográfico saturou a CPU.

### 2. Cenário I/O-bound – /simple

A rota /simple representa operações leves, baseadas em I/O.

**Capacidade máxima sustentada:  $\approx$  10.000 VUs**

Durante o teste de carga constante nesse endpoint, os erros começaram a aparecer quando o sistema atingiu o platô de **10.000 usuários virtuais**, indicando que o gargalo ocorreu no limite de requisições simultâneas suportadas (I/O saturation).

## Análise de Estresse

### Limite de Estresse – /crypto (CPU)

O teste de estresse mostrou que a aplicação começou a falhar a partir de **135 VUs**, apresentando aumento abrupto no tempo de resposta, HTTP failures e timeouts

Isso demonstra que o endpoint CPU-bound atinge rapidamente o limite dos recursos de processamento.

### Limite de Estresse – /simple (I/O)

O primeiro ponto de falha ocorreu ao atingir o **platô de 10.000 VUs**, onde começaram a surgir falhas intermitentes, picos de latência acumulada e limitações de filas de I/O.

Isso indica que o gargalo desse endpoint está relacionado ao volume extremo de requisições simultâneas, e não ao processamento.

## Spike

```
HTTP
http_req_duration.....: avg=203.76ms min=99.48ms med=204.3ms max=312.58ms p(90)=283.8ms p(95)=293.63ms
  { expected_response:true }...: avg=203.76ms min=99.48ms med=204.3ms max=312.58ms p(90)=283.8ms p(95)=293.63ms
http_req_failed.....: 0.00% 0 out of 17800
http_reqs.....: 17800 160.137418/s

EXECUTION
iteration_duration.....: avg=1.2s min=1.09s med=1.2s max=1.31s p(90)=1.28s p(95)=1.29s
iterations.....: 17800 160.137418/s
vus.....: 2 min=1 max=300
vus_max.....: 300 min=300 max=300

NETWORK
data_received.....: 5.3 MB 47 kB/s
data_sent.....: 3.0 MB 27 kB/s
```

## Smoke

```
HTTP
http_req_duration.....: avg=155.98µs min=0s med=0s max=553.1µs p(90)=513.1µs p(95)=545.38µs
  { expected_response:true }...: avg=155.98µs min=0s med=0s max=553.1µs p(90)=513.1µs p(95)=545.38µs
http_req_failed.....: 0.00% 0 out of 30
http_reqs.....: 30 0.999257/s

EXECUTION
iteration_duration.....: avg=1s min=1s med=1s max=1s p(90)=1s p(95)=1s
iterations.....: 30 0.999257/s
vus.....: 1 min=1 max=1
vus_max.....: 1 min=1 max=1

NETWORK
data_received.....: 8.7 kB 289 B/s
data_sent.....: 2.3 kB 76 B/s
```

## Stress

```
HTTP
http_req_duration.....: avg=1.49s min=0s med=0s max=47.3s p(90)=5.02s p(95)=5.65s
  { expected_response:true }...: avg=11.67s min=51.2ms med=5.03s max=47.3s p(90)=37.25s p(95)=41.51s
http_req_failed.....: 88.60% 52337 out of 59068
http_reqs.....: 59068 156.123746/s

EXECUTION
iteration_duration.....: avg=2.49s min=1s med=1s max=48.3s p(90)=6.02s p(95)=6.65s
iterations.....: 59068 156.123746/s
vus.....: 8 min=2 max=999
vus_max.....: 1000 min=1000 max=1000

NETWORK
data_received.....: 2.1 MB 5.6 kB/s
data_sent.....: 1.4 MB 3.7 kB/s
```

## Load

```
HTTP
http_req_duration.....: avg=205.29ms min=100.6ms med=205.11ms max=311.61ms p(90)=286.49ms p(95)=294.06ms
  { expected_response:true }...: avg=205.29ms min=100.6ms med=205.11ms max=311.61ms p(90)=286.49ms p(95)=294.06ms
http_req_failed.....: 0.00% 0 out of 6878
http_reqs.....: 6878 32.666952/s

EXECUTION
iteration_duration.....: avg=1.2s min=1.1s med=1.2s max=1.31s p(90)=1.28s p(95)=1.29s
iterations.....: 6878 32.666952/s
vus.....: 1 min=1 max=50
vus_max.....: 50 min=50 max=50

NETWORK
data_received.....: 2.0 MB 9.7 kB/s
data_sent.....: 1.1 MB 5.4 kB/s
```