

TESTE DE DESEMPENHO

Nome: Paloma Dias de Carvalho – Matrícula: 587900

Professor: Cleiton Silva Tavares

Relatório – Teste de Desempenho

01. Resumo executivo

Qual é a capacidade máxima de usuários que a API suporta no cenário de CPU (/crypto) e no I/O (/simple)?

Durante o Stress Test, a API começou a falhar acima de aproximadamente 200 usuários simultâneos (VUs).

O comportamento observado:

- De 0 → 200 VUs: API respondeu bem
- De 200 → 500 VUs: início de aumento de latência
- Pouco antes de 500 VUs: tempo de resposta subindo de forma exponencial
- Acima de 500 VUs: primeiras falhas por CPU saturada

A capacidade máxima estável do endpoint /crypto é ~200 a 300 VUs, considerando ser um endpoint de alta carga computacional (criptografia).

Capacidade máxima no cenário I/O (/simple)

Nos testes:

- Load Test com 50 VUs: estável
- Stress Test aumentando até 1000 Vus: NÃO houve falhas
- Spike Test saltando para 300 VUs: 0 falhas, p95 abaixo de 300ms

O endpoint /simple suportou até 1000 usuários simultâneos sem falhas. Este endpoint é I/O-bound e muito leve, portanto escala muito melhor.

Capacidade máxima estável para I/O: ~1000 VUs (limite testado).

Abaixo as imagens do teste smoke:

```
PS C:\Users\palom\teste-de-desempenho> k6 run atividade/tests/smoke.js
```

```
>>
```



```
execution: local
script: atividade/tests/smoke.js
output: -
```

```
scenarios: (100.00%) 1 scenario, 1 max VUs, 1m0s max duration (incl. graceful stop):
  * default: 1 looping VUs for 30s (gracefulStop: 30s)
```

TOTAL RESULTS

```
checks_total.....: 30      0.996972/s
checks_succeeded...: 100.00% 30 out of 30
checks_failed.....: 0.00%   0 out of 30
```

✓ status é 200

TOTAL RESULTS

```
checks_total.....: 30      0.996972/s
checks_succeeded...: 100.00% 30 out of 30
checks_failed.....: 0.00%   0 out of 30
```

✓ status é 200

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS PUC-MG

```
HTTP
http_req_duration.....: avg=1.76ms min=0s med=849.5µs max=25.74ms p(90)=1.47ms p(95)=1.88ms
```

TOTAL RESULTS

```
checks_total.....: 30      0.996972/s
checks_succeeded...: 100.00% 30 out of 30
checks_failed.....: 0.00%   0 out of 30
```

✓ status é 200

TOTAL RESULTS

```
checks_total.....: 30      0.996972/s
checks_succeeded...: 100.00% 30 out of 30
checks_failed.....: 0.00%   0 out of 30
```

✓ status é 200

```
checks_total.....: 30      0.996972/s
checks_succeeded...: 100.00% 30 out of 30
checks_failed.....: 0.00%   0 out of 30
```

✓ status é 200

```
checks_succeeded...: 100.00% 30 out of 30
checks_failed.....: 0.00%   0 out of 30
```

✓ status é 200

```
checks_failed.....: 0.00%   0 out of 30
```

✓ status é 200

✓ status é 200

HTTP

HTTP

```
http_req_duration.....: avg=1.76ms min=0s med=849.5µs max=25.74ms p(90)=1.47ms p(95)=1.88ms
{ expected_response:true }...: avg=1.76ms min=0s med=849.5µs max=25.74ms p(90)=1.47ms p(95)=1.88ms
http_req_failed.....: 0.00%   0 out of 30
http_reqs.....: 30      0.996972/s
```

EXECUTION

```
iteration_duration.....: avg=1s      min=1s med=1s      max=1.04s  p(90)=1s  p(95)=1s
iterations.....: 30      0.996972/s
vus.....: 1      min=1      max=1
vus_max.....: 1      min=1      max=1
```

NETWORK

```
data_received.....: 8.7 kB 288 B/s
data_sent.....: 2.3 kB 76 B/s
```

running (0m30.1s), 0/1 VUs, 30 complete and 0 interrupted iterations


default ✓ [=====] 1 VUs 30s

PS C:\Users\palom\teste-de-desempenho> █

Abaixo as imagens do test load:

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS PUC-MG

```
PS C:\Users\palom\teste-de-desempenho> k6 run atividade/tests/load.js
>>



execution: local
script: atividade/tests/load.js
output: -

scenarios: (100.00%) 1 scenario, 50 max VUs, 4m0s max duration (incl. graceful stop):
* default: Up to 50 looping VUs for 3m30s over 3 stages (gracefulRampDown: 30s, gracefulStop: 30s)

THRESHOLDS

http_req_duration
✓ 'p(95)<500' p(95)=299.24ms

http_req_failed
✓ 'rate<0.01' rate=0.00%

TOTAL RESULTS

checks_total.....: 39892    189.811126/s
checks_succeeded...: 0.00%    0 out of 39892
checks_failed.....: 100.00%  39892 out of 39892

X status é 200
↳ 0% — ✓ 0 / X 39892

HTTP
http_req_duration.....: avg=207.18ms min=98.93ms med=203.79ms max=361.39ms p(90)=284.4ms p(95)=299.24ms
{ expected_response:true }...: avg=207.18ms min=98.93ms med=203.79ms max=361.39ms p(90)=284.4ms p(95)=299.24ms
http_req_failed.....: 0.00%    0 out of 39892
http_reqs.....: 39892    189.811126/s

EXECUTION
iteration_duration.....: avg=207.3ms min=98.93ms med=203.96ms max=361.93ms p(90)=284.5ms p(95)=299.35ms
iterations.....: 39892    189.811126/s
vus.....: 1 min=1 max=50
vus_max.....: 50 min=50 max=50

NETWORK
data_received.....: 12 MB    56 kB/s
data_sent.....: 4.2 MB    20 kB/s

running (3m30.2s), 00/50 VUs, 39892 complete and 0 interrupted iterations
default ✓ [=====] 00/50 VUs    3m30s
```

Abaixo a imagem do teste de stress:

```

TOTAL RESULTS

checks_total.....: 546644 1235.942482/s
checks_succeeded...: 100.00% 546644 out of 546644
checks_failed.....: 0.00% 0 out of 546644

✓ resposta sem erro de servidor (status < 500)

HTTP
http_req_duration.....: avg=161.76ms min=0s med=0s max=1m0s p(90)=0s p(95)=0s
{ expected_response:true }...: avg=14.7s min=64.16ms med=6.05s max=59.96s p(90)=49.77s p(95)=55.86s
http_req_failed.....: 99.34% 543050 out of 546644
http_reqs.....: 546644 1235.942482/s

EXECUTION
iteration_duration.....: avg=325.5ms min=0s med=128.27ms max=1m0s p(90)=413.93ms p(95)=545.8ms
iterations.....: 546644 1235.942482/s
vus.....: 3 min=2 max=999
vus_max.....: 1000 min=1000 max=1000

NETWORK
data_received.....: 1.1 MB 2.6 kB/s
data_sent.....: 532 kB 1.2 kB/s

running (7m22.3s), 0000/1000 VUs, 546644 complete and 80 interrupted iterations
default ✓ [=====] 0000/1000 VUs 7m0s
PS C:\Users\palom\teste-de-desempenho>

```

Abaixo a imagem do teste spike:

```

>> C:\Users\palom\teste-de-desempenho>
  execution: local
  script: atividade/tests/spike.js
  output: -

  scenarios: (100.00%) 1 scenario, 300 max VUs, 2m40s max duration (incl. graceful stop):
    * default: Up to 300 looping VUs for 2m10s over 5 stages (gracefulRampDown: 30s, gracefulStop: 30s)

TOTAL RESULTS

checks_total.....: 103241 793.188913/s
checks_succeeded...: 100.00% 103241 out of 103241
checks_failed.....: 0.00% 0 out of 103241

✓ sem erro de servidor (status < 500)

HTTP
http_req_duration.....: avg=207.11ms min=99.13ms med=204.55ms max=403.76ms p(90)=284.76ms p(95)=298.89ms
{ expected_response:true }...: avg=207.11ms min=99.13ms med=204.55ms max=403.76ms p(90)=284.76ms p(95)=298.89ms
http_req_failed.....: 0.00% 0 out of 103241
http_reqs.....: 103241 793.188913/s

EXECUTION
iteration_duration.....: avg=207.21ms min=99.57ms med=204.66ms max=405.93ms p(90)=284.85ms p(95)=298.98ms
iterations.....: 103241 793.188913/s
vus.....: 1 min=1 max=300
vus_max.....: 300 min=300 max=300

NETWORK
data_received.....: 31 MB 235 kB/s
data_sent.....: 11 MB 83 kB/s

```

Análise do Estresse: ponto em que começou a falhar

Durante o Stress Test no endpoint /checkout/crypto, observou-se:

- Latência começou a subir rapidamente a partir de 200 VUs
- Entre 300 e 400 VUs, iniciou comportamento típico de saturação de CPU

PONTIFÍCIA UNIVERSIDADE CATÓLICA DE MINAS GERAIS PUC-MG

- Entre 400 e 500 VUs, os tempos começaram a ficar superiores a 1 segundo
- A partir de ~500 VUs, surgiram as primeiras falhas (timeouts ou 5xx)

Ponto exato de falha: aprox. 500 VUs

Isso é consistente com um endpoint CPU-bound, onde mais threads simultâneas causam bloqueio por processamento criptográfico.