

# 1 Vegeta

- Como usar:

```
echo "GET http://localhost:8080/threads/virtual" | vegeta attack -duration=1s -rate=19000 | vegeta report
```

- Limite aproximado: 19000 requisições por segundo

# 2 JMeter

- Criar arquivo .jmx
- Limite: número de threads depende da capacidade do PC
- Em minha máquina: aproximadamente 1500 a 2300 requisições por segundo

# 3 Gatling

- Não consegui instalar

# 4 k6

- Criar arquivo .js
- Limite aproximado: 5000 requisições por segundo

# 5 ApacheBench (ab)

```
ab [opções] [URL]
```

Exemplo:

```
ab -n 100 -c 10 http://localhost:8080/threads/virtual
```

Onde:

- -n 100 → número total de requisições
- -c 10 → número de requisições concorrentes

# 6 Principais opções do ab

## 6.1 Requisições e concorrência

```
-n <numero> # Total de requisições  
-c <numero> # Conexões simultâneas
```

## 6.2 Cabeçalhos HTTP

```
-H "Header: valor" # Adiciona cabeçalho customizado  
-A usuario:senha # Autenticação básica HTTP
```

## 6.3 Tipo de requisição

```
-p arquivo # Envia dados POST a partir de arquivo  
-T tipo # Tipo de conteúdo (ex: application/json)  
-u arquivo # Envia dados PUT a partir de arquivo
```

## 6.4 Cookies

```
-C "nome=valor" # Envia cookie na requisição
```

## 6.5 TLS / HTTPS

```
-s segundos      # Timeout para cada requisição  
-Z cipherlist    # Cipher SSL customizado  
-X host:porta    # Proxy HTTP  
-v nivel         # Verbosidade (0 a 4)
```

## 6.6 Logs e relatórios

```
-e arquivo.csv   # Salva resultados detalhados em CSV  
-g arquivo.dat   # Salva dados para plot no gnuplot  
-r              # Não aborta em erro HTTP
```

## 6.7 Testes avançados

```
-k              # Habilita Keep-Alive  
-v nivel       # Mostra detalhes de cabeçalhos e respostas  
-q            # Suprime saída de progresso
```

# 7 Exemplos completos

## 7.1 Teste GET simples

```
ab -n 500 -c 50 http://localhost:8080/threads/virtual
```

## 7.2 Teste POST com JSON

```
ab -n 200 -c 20 -p dados.json -T application/json http://localhost:8080/api
```

## 7.3 Teste com Keep-Alive e headers personalizados

```
ab -n 1000 -c 100 -k -H "Authorization: Bearer TOKEN" http://localhost:8080/  
threads/virtual
```

## 7.4 Salvar resultados em CSV

```
ab -n 1000 -c 100 -e resultados.csv http://localhost:8080/threads/virtual
```

## 7.5 Salvar headers e status de cada requisição

```
ab -n 1000 -c 100 -v 2 http://localhost:8080/threads/virtual/ > log.txt
```

## 7.6 Definir timeout e limite de duração do teste

-s define o timeout em segundos para cada requisição -t define o tempo máximo do teste em segundos

```
ab -c 100 -t 10 -s 5 http://localhost:8080/threads/virtual/
```

## Limite de Simultaneidade de Portas no macOS

Verificar o intervalo de portas efêmeras utilizadas pelo sistema operacional para conexões TCP de saídas:

```
sysctl net.inet.ip.portrange.first  
sysctl net.inet.ip.portrange.last
```

- **first:** porta inicial do intervalo de portas efêmeras.
- **last:** porta final do intervalo de portas efêmeras.

```
net.inet.ip.portrange.first: 49152  
net.inet.ip.portrange.last: 65535
```

### Cálculo da Capacidade Máxima Teórica

O número total de portas disponíveis é dado por:

$$65535 - 49152 + 1 = 16384$$

Há aproximadamente **16384 portas disponíveis para conexões TCP simultâneas de saída.**