

Performance Test Report - Aug 2, 2025 (#1)

Open in Postman

Postman collection: FTP
Report exported on: Aug 2, 2025, 16:58:35 (GMT+5:30)

Test setup

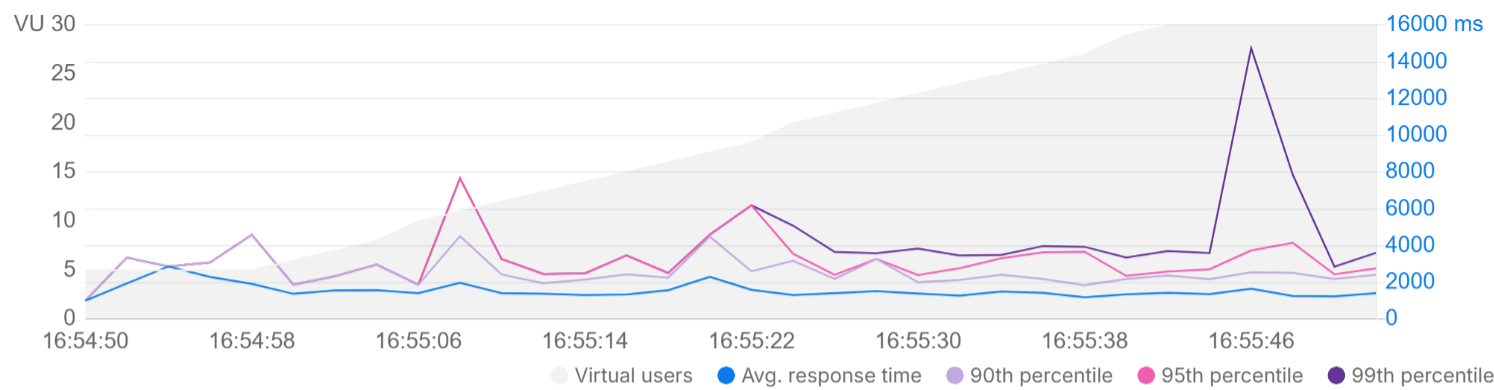
Virtual users	Start time	Load profile
30 VU	Aug 2, 16:54:46 (GMT+5:30)	Ramp up (45 seconds)
Duration	End time	Environment
1 minute	Aug 2, 16:55:53 (GMT+5:30)	-

1. Summary

Total requests sent	Throughput	Average response time	Error rate
547	8.12 requests/second	1,463 ms	1.46 %

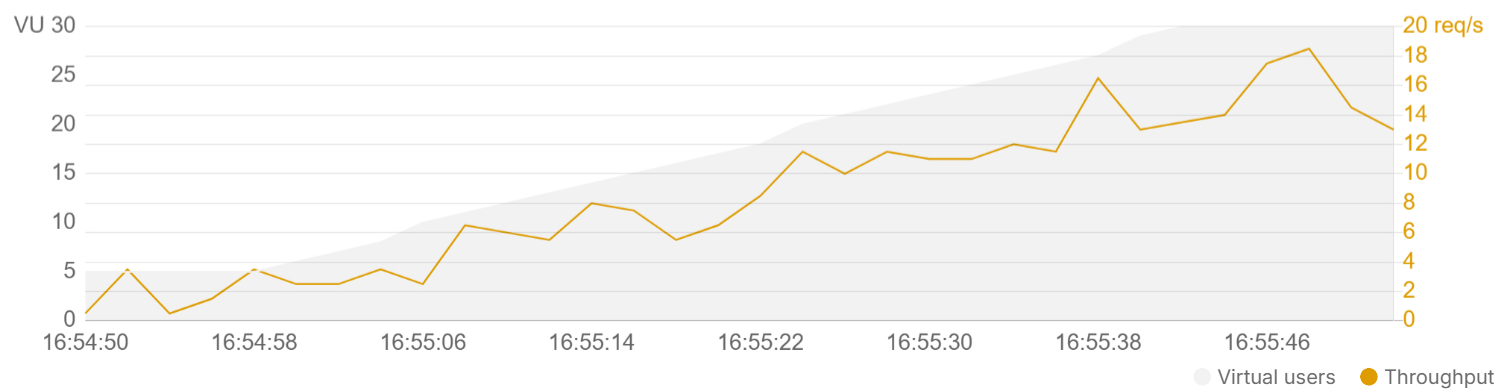
1.1 Response time

Response time trends during the test duration.



1.2 Throughput

Rate of requests sent per second during the test duration.



1.3 Requests with slowest response times

Top 5 slowest requests based on their average response times.

Request	Resp. time (Avg ms)	90th (ms)	95th (ms)	99th (ms)	Min (ms)	Max (ms)
GET getAccountDetails http://localhost:8080/accounts/29	1,474	2,458	3,504	4,530	465	14,759
POST createAccount http://localhost:8080/accounts	1,453	2,445	3,283	5,099	374	7,868

1.4 Requests with most errors

Top 5 requests with the most errors, along with the most frequently occurring errors for each request.

Request	Total error count	Error 1	Error 2	Other errors
POST createAccount http://localhost:8080/accounts	5	500 Server Error (5)	-	0
GET getAccountDetails http://localhost:8080/accounts/29	3	500 Server Error (3)	-	0

2. Metrics for each request

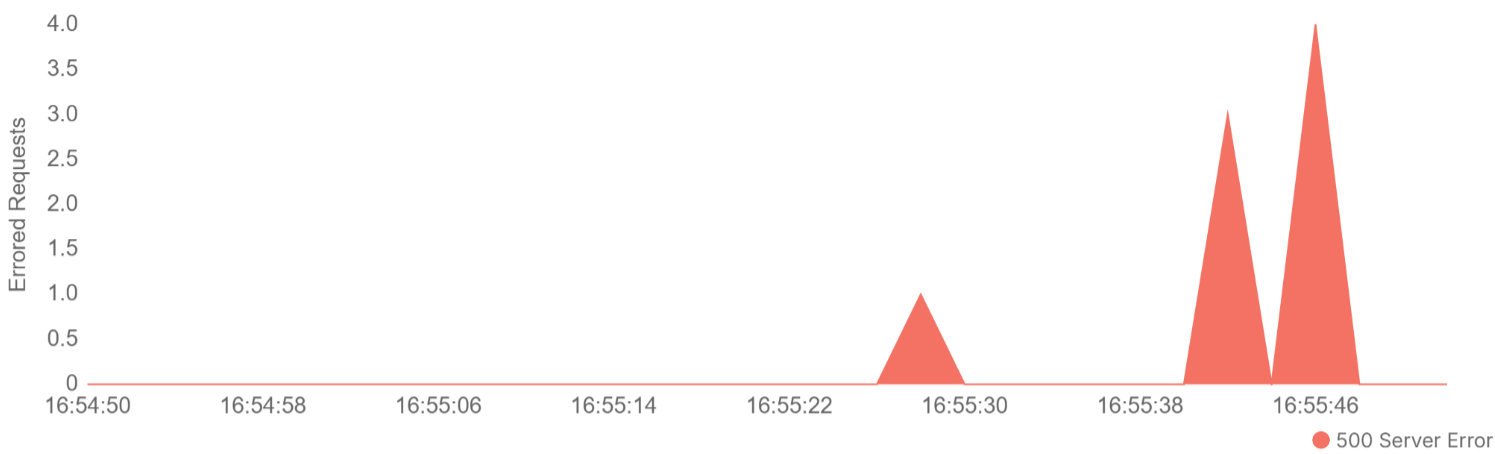
The requests are shown in the order they were sent by virtual users.

Request	Total requests	Requests/s	Min (ms)	Avg (ms)	90th (ms)	Max (ms)	Error %
POST createAccount http://localhost:8080/accounts	281	4.17	374	1,453	2,445	7,868	1.78
GET getAccountDetails http://localhost:8080/accounts/29	266	3.95	465	1,474	2,458	14,759	1.13

3. Errors

3.1 Error distribution over time

Top 5 error classes observed during the test duration.



3.2 Error distribution for requests

Errored requests grouped by error class, along with the error count for each class.

Error class	Total counts
500 Server Error	8
POST createAccount	5
GET getAccountDetails	3



Testing API performance on Postman

Postman enables you to simulate user traffic and observe how your API behaves under load. It also helps you identify any issues or bottlenecks that affect performance.

Learn more about [testing API performance](#).