

**Федеральное государственное автономное образовательное
учреждение высшего образования
Национальный исследовательский
университет ИТМО**

Факультет программной инженерии и компьютерной техники

Направление подготовки:

Системное и Прикладное Программное Обеспечение

(09.03.04 Программная инженерия)

Дисциплина «Тестирование программного обеспечения»

Отчет

По лабораторной работе №4

Вариант:

<http://stload.se.ifmo.ru:8080/?token=496159540&user=-2105056041&config=1>
<http://stload.se.ifmo.ru:8080/?token=496159540&user=-2105056041&config=2>
<http://stload.se.ifmo.ru:8080/?token=496159540&user=-2105056041&config=3>

Студент:

**Карташев Владимир Сергеевич,
группа Р3315**

Практик:

Цопа Евгений Алексеевич

г. Санкт-Петербург, 2025 г.

Оглавление

Задание	3
Выполнение	4
Ссылка на код	4
Подготовка к выполнению	4
- - - Прокидываем порт	4
- - - Удаляем директории с результатами предыдущих тестов	4
Нагрузочное тестирование	4
- - - Конфигурация Thread Group config #1	5
- - - Конфигурация Thread Group config #2	6
- - - Конфигурация Thread Group config #3	7
- - - Конфигурация общих элементов	8
Результаты нагрузочного тестирования	10
- - - config #1	11
- - - config #2	12
- - - config #3	13
Стресс-тестирование	14
- - - Конфигурация Thread Group config #1	14
Результаты стресс-тестирования	15
- - - config #3	15
Вывод	16

Задание

С помощью программного пакета [Apache JMeter](#) провести нагрузочное и стресс-тестирование веб-приложения в соответствии с вариантом задания.

В ходе нагрузочного тестирования необходимо протестировать 3 конфигурации аппаратного обеспечения и выбрать среди них наиболее дешёвую, удовлетворяющую требованиям по максимальному времени отклика приложения при заданной нагрузке (в соответствии с вариантом).

В ходе стресс-тестирования необходимо определить, при какой нагрузке выбранная на предыдущем шаге конфигурация перестаёт удовлетворять требованиями по максимальному времени отклика. Для этого необходимо построить график зависимости времени отклика приложения от нагрузки.

Приложение для тестирования доступно только во внутренней сети кафедры.

Если запрос содержит некорректные параметры, сервер возвращает HTTP 403.

Если приложение не справляется с нагрузкой, сервер возвращает HTTP 503.

Параметры тестируемого веб-приложения:

- URL первой конфигурации (\$ 2100) -
<http://stload.se.ifmo.ru:8080?token=496159540&user=-2105056041&config=1>;
- URL второй конфигурации (\$ 3300) -
<http://stload.se.ifmo.ru:8080?token=496159540&user=-2105056041&config=2>;
- URL третьей конфигурации (\$ 3500) -
<http://stload.se.ifmo.ru:8080?token=496159540&user=-2105056041&config=3>;
- Максимальное количество параллельных пользователей - 5;
- Средняя нагрузка, формируемая одним пользователем - 40 запр. в мин.;
- Максимально допустимое время обработки запроса - 920 мс.

Выполнение

Ссылка на код

[КЛИК - КЛИК - КЛИК](#)

Подготовка к выполнению

--- Прокидываем порт

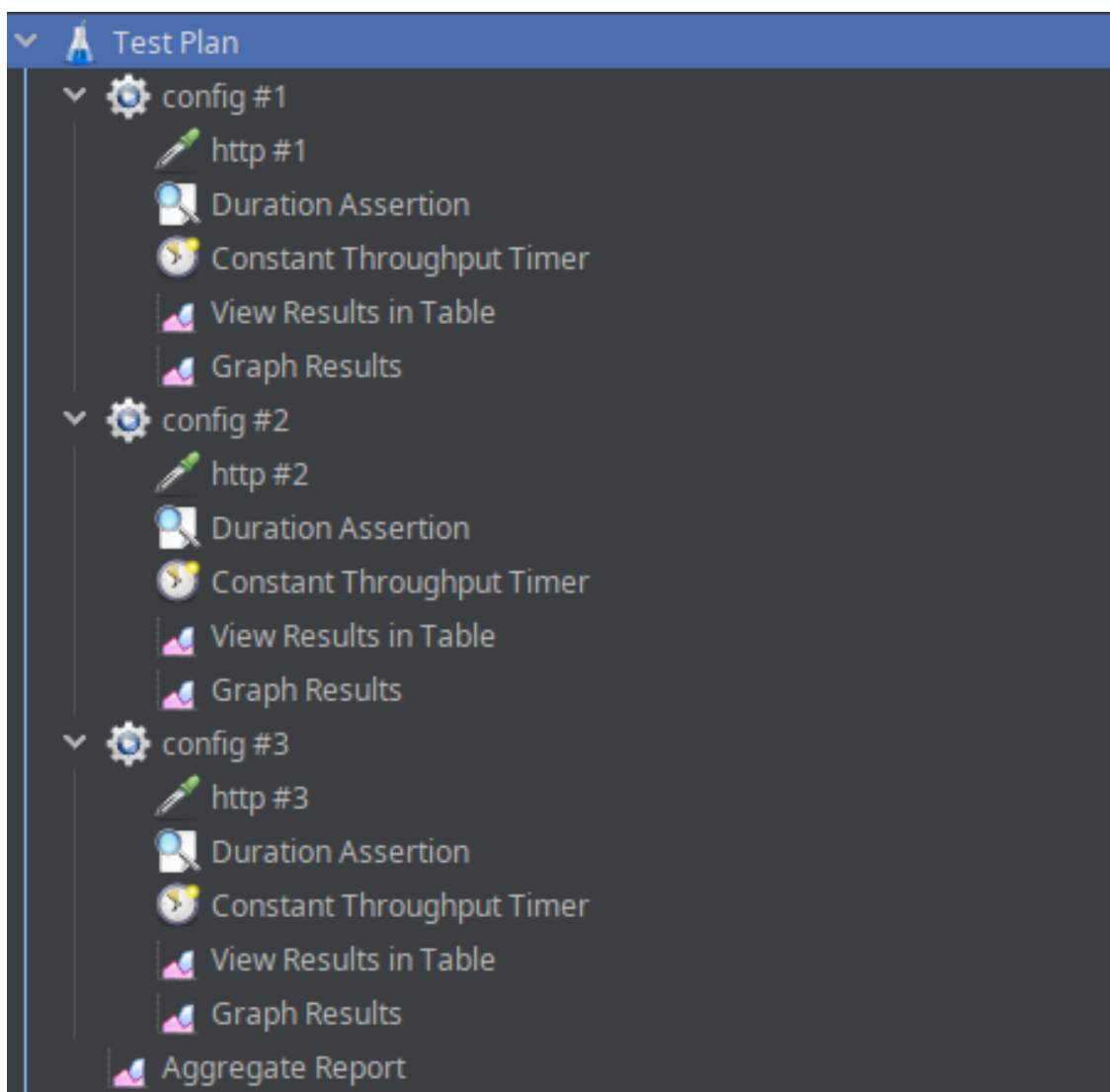
```
ssh -f -N -L 8080:stload.se.ifmo.ru:8080 s373440@se.ifmo.ru -p 2222
```

--- Удаляем директории с результатами предыдущих тестов

```
rm -fr load; rm -fr stress
```

Нагрузочное тестирование

древо конфигурации



--- Конфигурация Thread Group config #1

Thread Group config #1

Test Plan

config #1

http #1

Duration Assertion

Constant Throughput Timer

View Results in Table

Graph Results

config #2

config #3

Aggregate Report

Thread Group

Name: config #1

Comments:

Action to be taken after a Sampler error

Continue

Start Next Thread Loop

Stop Thread

Stop Test

Stop Test Now

Thread Properties

Number of Threads (users): 5

Ramp-up period (seconds): 60

Loop Count: ☐ Infinite 40

☒ Same user on each iteration

☐ Delay Thread creation until needed

☐ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

HTTP Request http #1

Test Plan

config #1

http #1

Duration Assertion

Constant Throughput Timer

View Results in Table

Graph Results

config #2

config #3

Aggregate Report

HTTP Request

Name: http #1

Comments:

Basic: Advanced

Web Server

Protocol (http): http

Server Name or IP: localhost

Port Number: 8080

HTTP Request

GET

Path: /

Content encoding:

☐ Redirect Automatically

☒ Follow Redirects

☒ Use Keepalive

☐ Use multipart/form-data

☐ Browser-compatible headers

Parameters

Body Data

Files Upload

Send Parameters With the Request

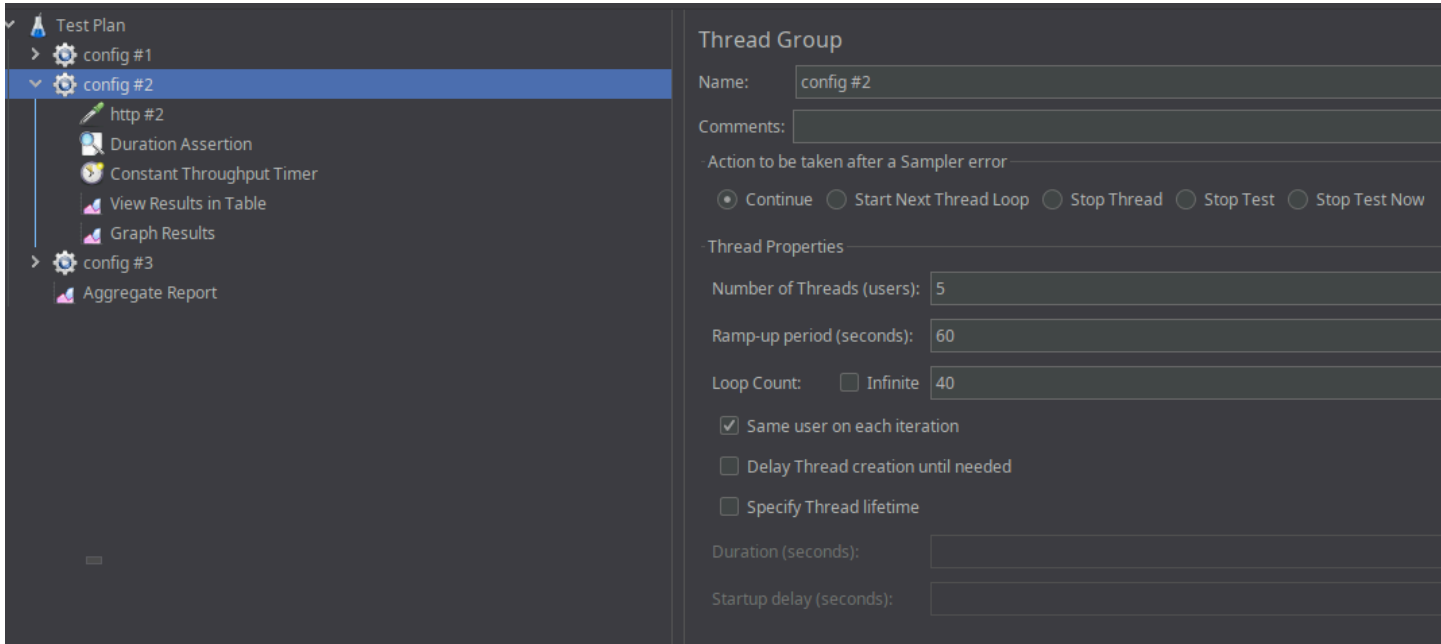
Name:	Value	URL Encode?	Content-Type	Include Equals?
token	496159540	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
user	-2105056041	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
config	1	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>

Таблица параметров:

Parameters	
token	496159540
user	-2105056041
config	1

--- Конфигурация Thread Group config #2

Thread Group config #2



HTTP Request http #2

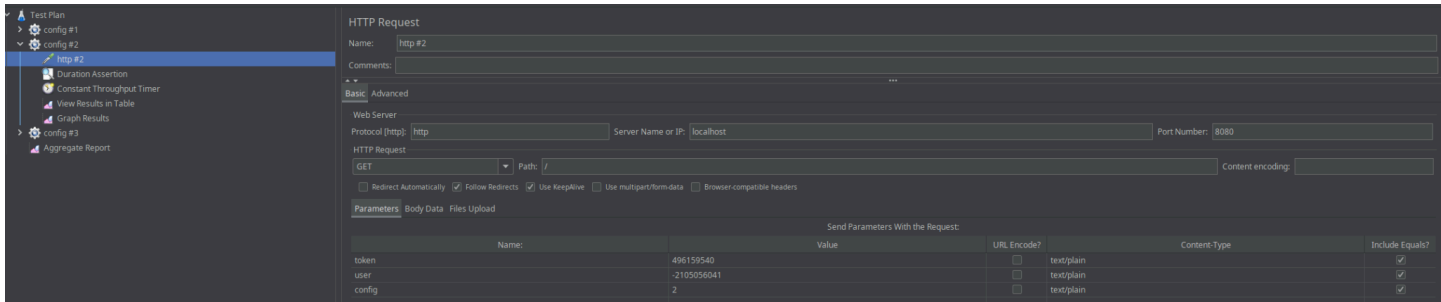


Таблица параметров:

Parameters	
token	496159540
user	-2105056041
config	2

- - - Конфигурация Thread Group config #3

Thread Group config #3

Test Plan

config #1

config #2

config #3

http #3

Duration Assertion

Constant Throughput Timer

View Results in Table

Graph Results

Aggregate Report

Thread Group

Name:

config #3

Comments:

Action to be taken after a Sampler error

☒ Continue

☐ Start Next Thread Loop

☐ Stop Thread

☐ Stop Test

☐ Stop Test Now

Thread Properties

Number of Threads (users):

5

Ramp-up period (seconds):

60

Loop Count:

☐ Infinite

40

☒ Same user on each iteration

☐ Delay Thread creation until needed

☐ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

HTTP Request http #3

Test Plan

config #1

config #2

config #3

http #3

Duration Assertion

Constant Throughput Timer

View Results in Table

Graph Results

Aggregate Report

HTTP Request

Name:

http #3

Comments:

Basic

Advanced

Web Server

Protocol (http):

http

Server Name or IP:

localhost

Port Number:

8080

HTTP Request

Method:

GET

Path:

/

Content encoding:

☐ Redirect Automatically

☒ Follow Redirects

☒ Use KeepAlive

☐ Use multipart/form-data

☐ Browser-compatible headers

Parameters

Body Data

Files Upload

Send Parameters With the Request:

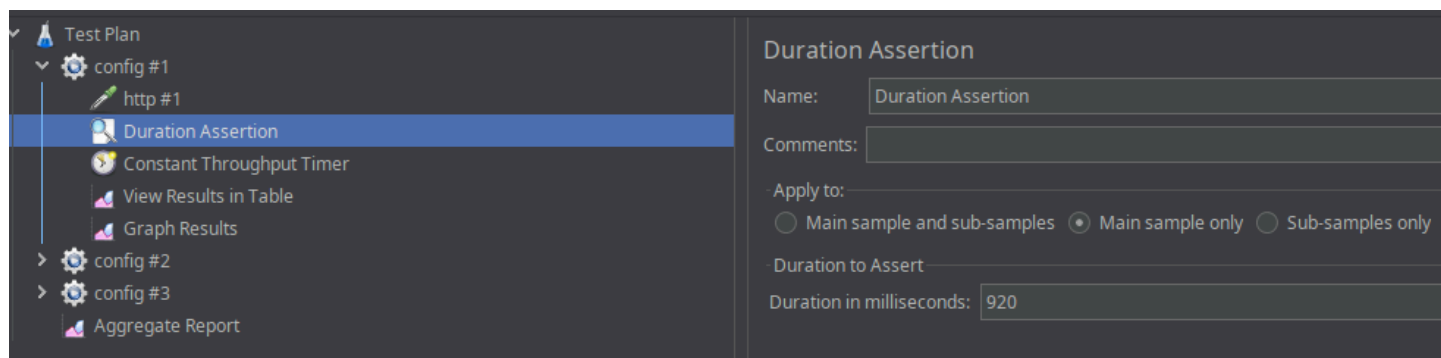
Name:	Value	URL Encode?	Content-Type	Include Equals?
token	496159540	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
user	-2105056041	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>
config	3	<input type="checkbox"/>	text/plain	<input checked="" type="checkbox"/>

Таблица параметров:

Parameters	
token	496159540
user	-2105056041
config	3

- - - Конфигурация общих элементов

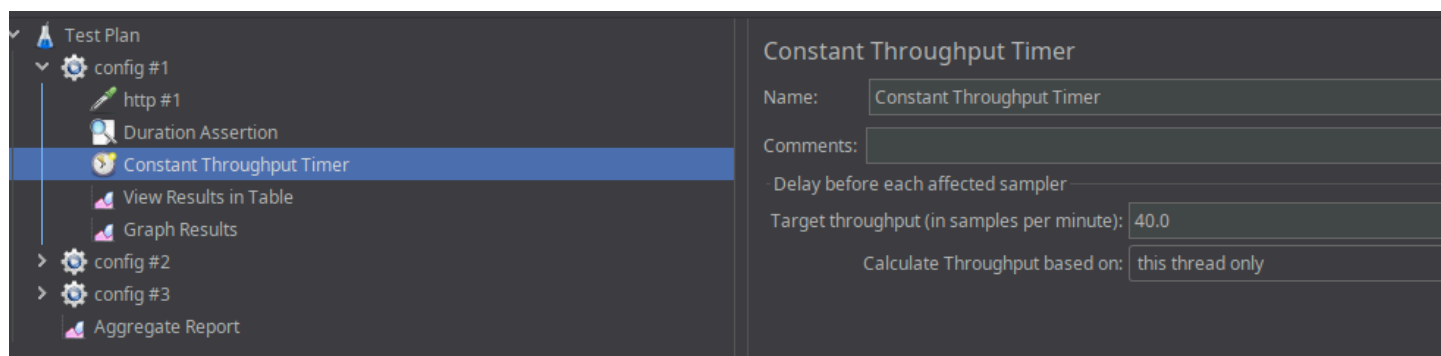
Duration Assertion



The screenshot shows the JMeter GUI with the 'Duration Assertion' element selected in the left-hand tree. The right-hand pane displays the configuration for this element. The 'Name' field is set to 'Duration Assertion'. The 'Comments' field is empty. The 'Apply to:' section has three radio buttons: 'Main sample and sub-samples' (unselected), 'Main sample only' (selected), and 'Sub-samples only' (unselected). The 'Duration to Assert' field is empty. The 'Duration in milliseconds' field is set to '920'.

Позволяет проверять, что время отклика (response time) запроса не превышает заданное значение (в миллисекундах).

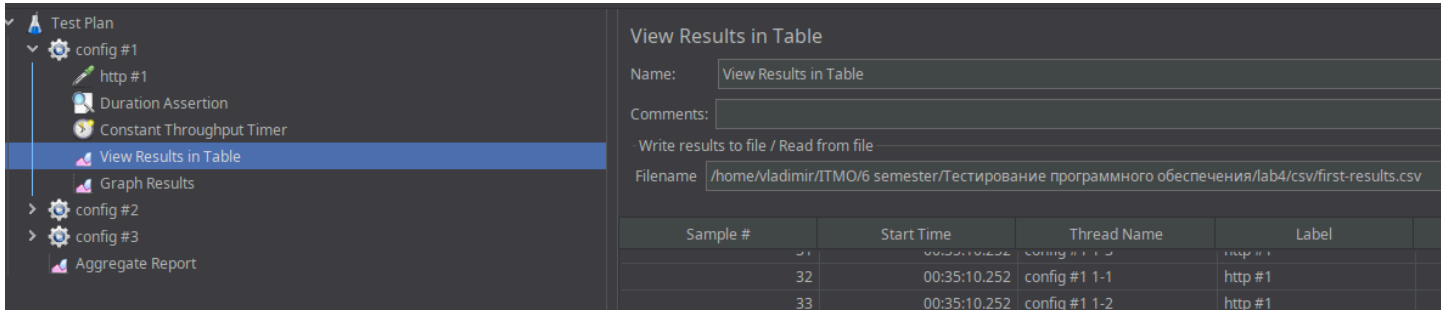
Constant Throughput Timer



The screenshot shows the JMeter GUI with the 'Constant Throughput Timer' element selected in the left-hand tree. The right-hand pane displays the configuration for this element. The 'Name' field is set to 'Constant Throughput Timer'. The 'Comments' field is empty. The 'Delay before each affected sampler' field is empty. The 'Target throughput (in samples per minute):' field is set to '40.0'. The 'Calculate Throughput based on:' dropdown menu is set to 'this thread only'.

Позволяет регулировать нагрузку, поддерживая постоянную пропускную способность (число запросов в минуту).

View Results in Table

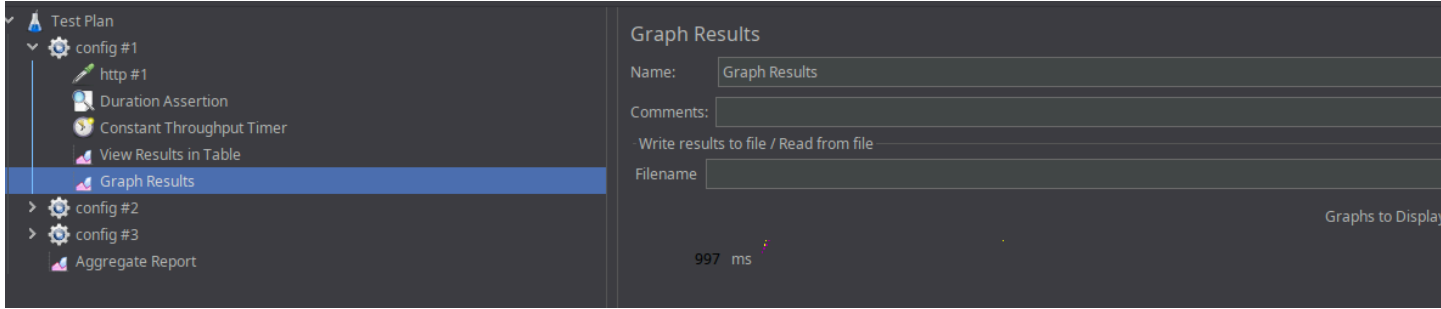


Позволяет представить запросы в табличном виде с определенными полями.

Таблица с описанием полей:

Поля	
Sample #	Номер запроса (порядковый).
Start Time	Время начала выполнения запроса (с точностью до миллисекунды).
Thread Name	Имя потока (вида Thread Group X-Y, где X – номер группы, Y – номер потока).
Label	Метка запроса (например, имя HTTP-запроса).
Sample Time (ms)	Время выполнения запроса (от отправки до получения ответа).
Status	После Constant Throughput Timer может показывать Success (если запрос выполнен) или Failed (если превышено время ожидания или assertion не пройден).
Bytes	Размер ответа (в байтах).
Sent Bytes	Размер отправленных данных (в байтах).
Latency	Время от отправки запроса до получения первого байта ответа.
Connect Time (ms)	Время установки TCP-соединения с сервером.

Graph Results



Позволяет представить результаты нагрузки графически, где ось X – время теста, а ось Y – значения (время отклика, количество запросов и т. д.).

Таблица значений на графике:

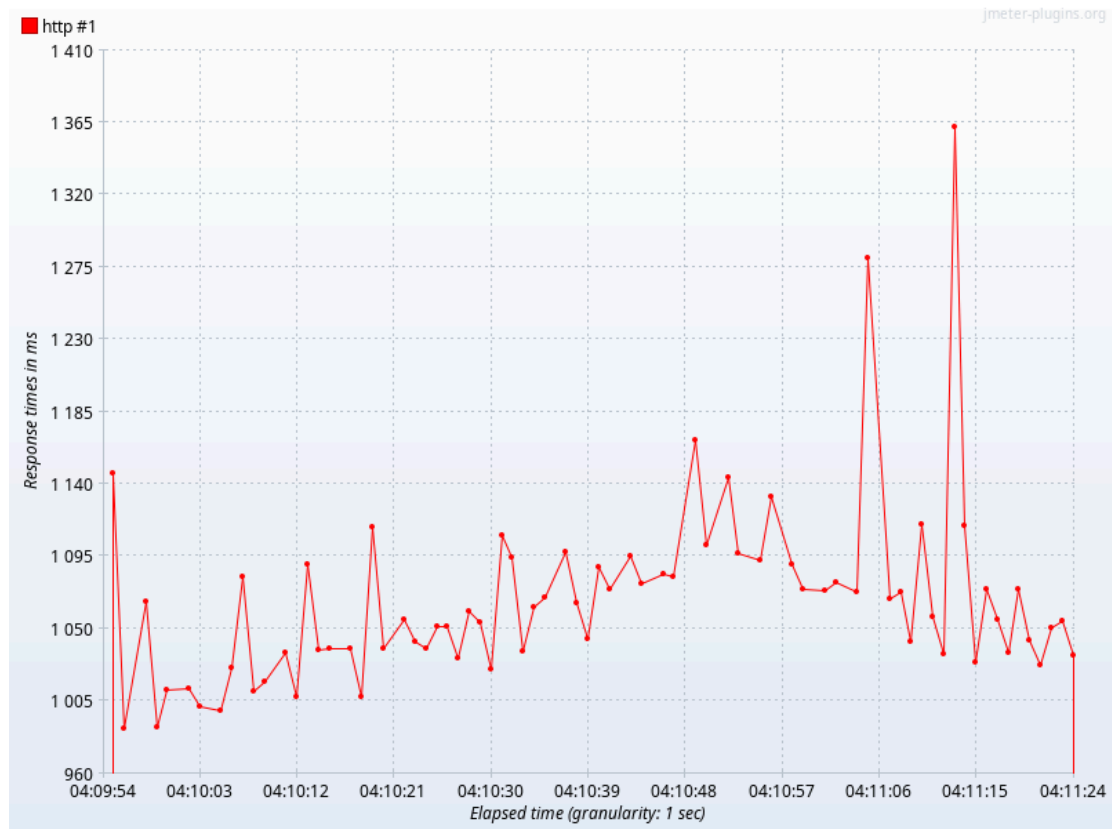
Значения	
Sample Time	среднее время отклика
Throughput	пропускная способность, запросов/сек
Deviation	отклонение времени ответа
Active Threads	число активных потоков

Результаты нагрузочного тестирования

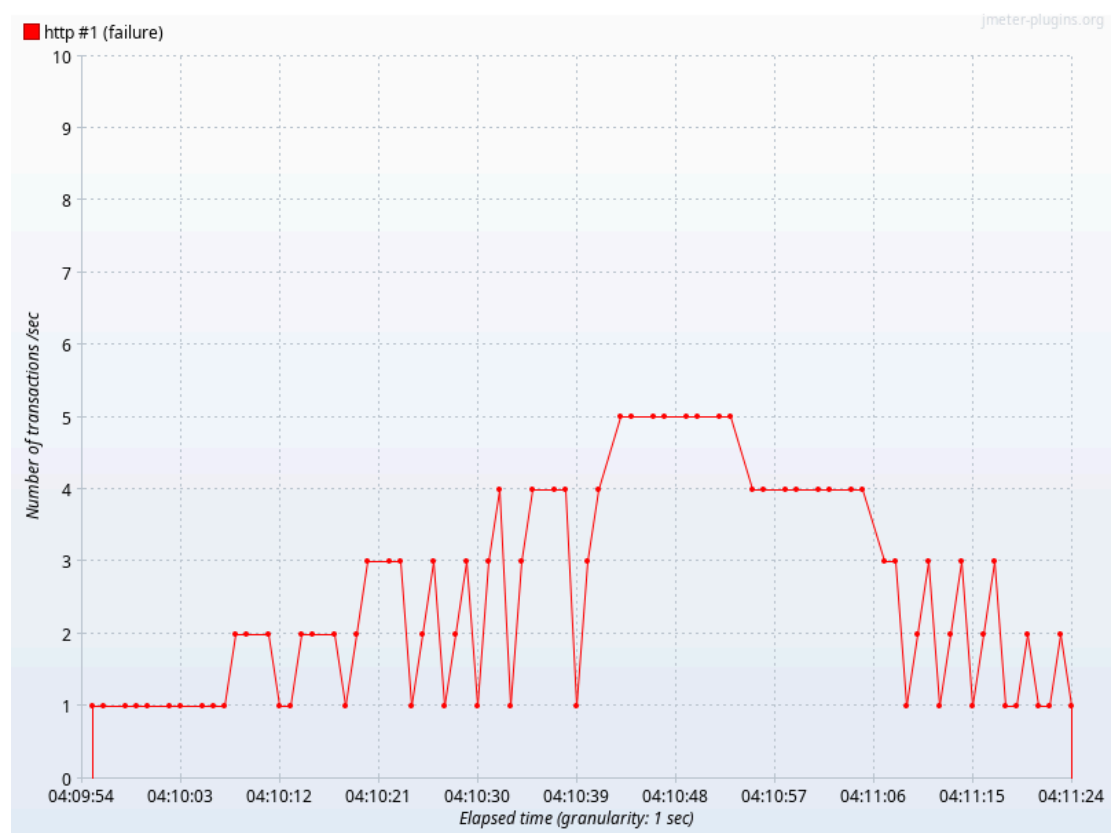
Label	# Samples	Average	Median	Min	Max	Error %	Throughput	Received KB/sec	Sent KB/sec
config #1	200	1077	1069	988	1363	100	1.9/sec	0.42	0.29
config #2	200	878	876	793	1006	14	1.9/sec	0.42	0.29
config #3	200	678	675	587	858	0	1.9/sec	0.42	0.29
TOTAL	600	878	876	587	1363	38	5.6/sec	1.26	0.87

--- config #1

Время ответа (мс) от времени запроса

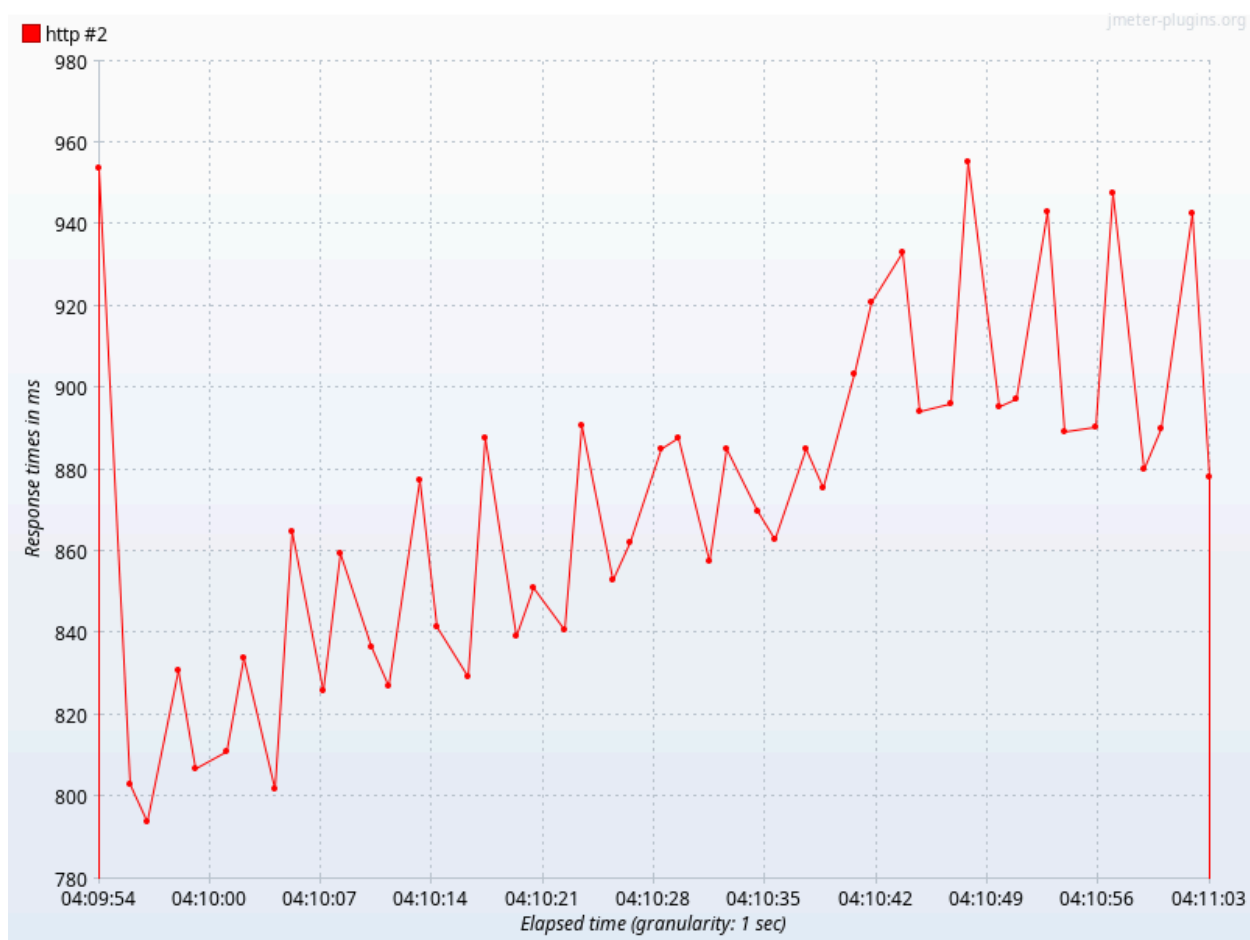


Запросы, соответствующие временному лимиту (920 мс)

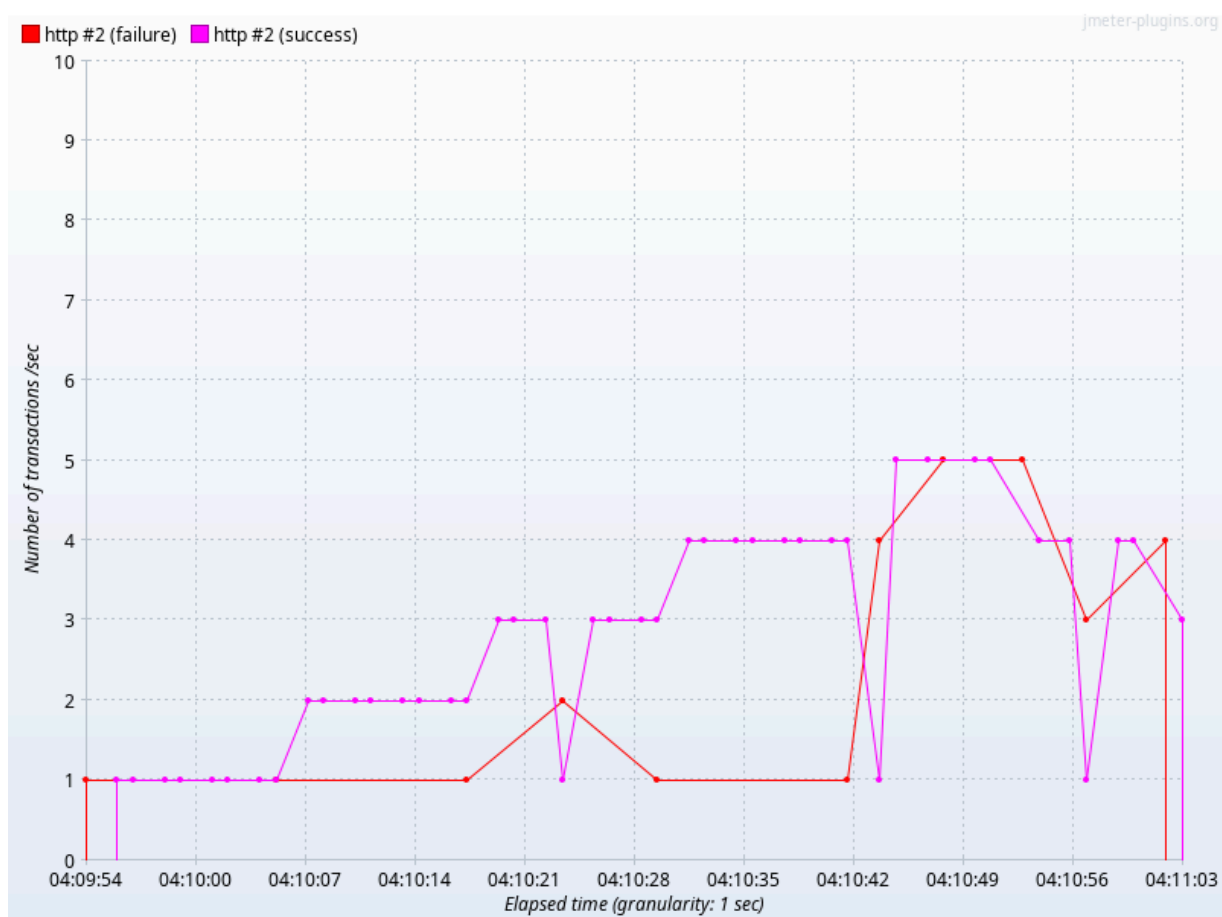


--- config #2

Время ответа (мс) от времени запроса

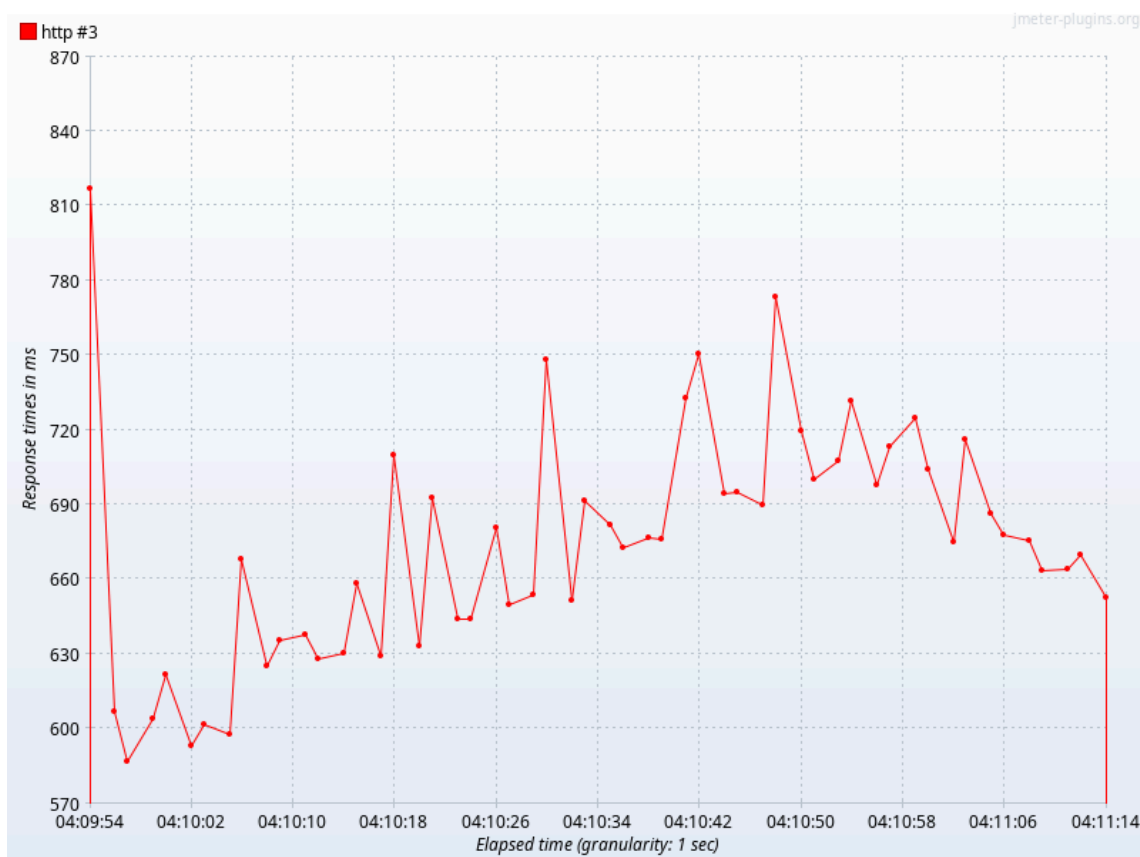


Запросы, соответствующие временному лимиту (920 мс)



--- config #3

Время ответа (мс) от времени запроса



Запросы, соответствующие временному лимиту (920 мс)



Стресс-тестирование

Для проведения стресс-тестирования была выбрана конфигурация №3, так как она имеет 100% соответствие требованиям логики и здравого смысла при ее стоимости в отличии от других конфигураций:

Расчет целесообразности конфигурации				
Конфигурация	Стоимость \$	Ошибка %	Успешность %	Цена ошибки \$/%
config #1	2100	100	0	2100
config #2	3300	14	86	33
config #3	3500	0	100	0

- * Цена ошибки рассчитывается через формулу $ЦО = C\$ / О\% * У\%$
- ** При успешности 0% считаем, что цена ошибки равна стоимости конфигурации, то есть конфигурация нецелесообразна

- - - Конфигурация Thread Group config #1

Thread Group config #3

Stress Testing Plan

config #3

http request

Constant Throughput Timer

View Results in Table

Graph Results

View Results Tree

jp@gc - Graphs Generator

Aggregate Report

Thread Group

Name: config #3

Comments:

Action to be taken after a Sampler error

☒ Continue

☐ Start Next Thread Loop

☐ Stop Thread

☐ Stop Test

☐ Stop Test Now

Thread Properties

Number of Threads (users): 500

Ramp-up period (seconds): 4

Loop Count: ☐ Infinite 1

☒ Same user on each iteration

☐ Delay Thread creation until needed

☐ Specify Thread lifetime

Duration (seconds):

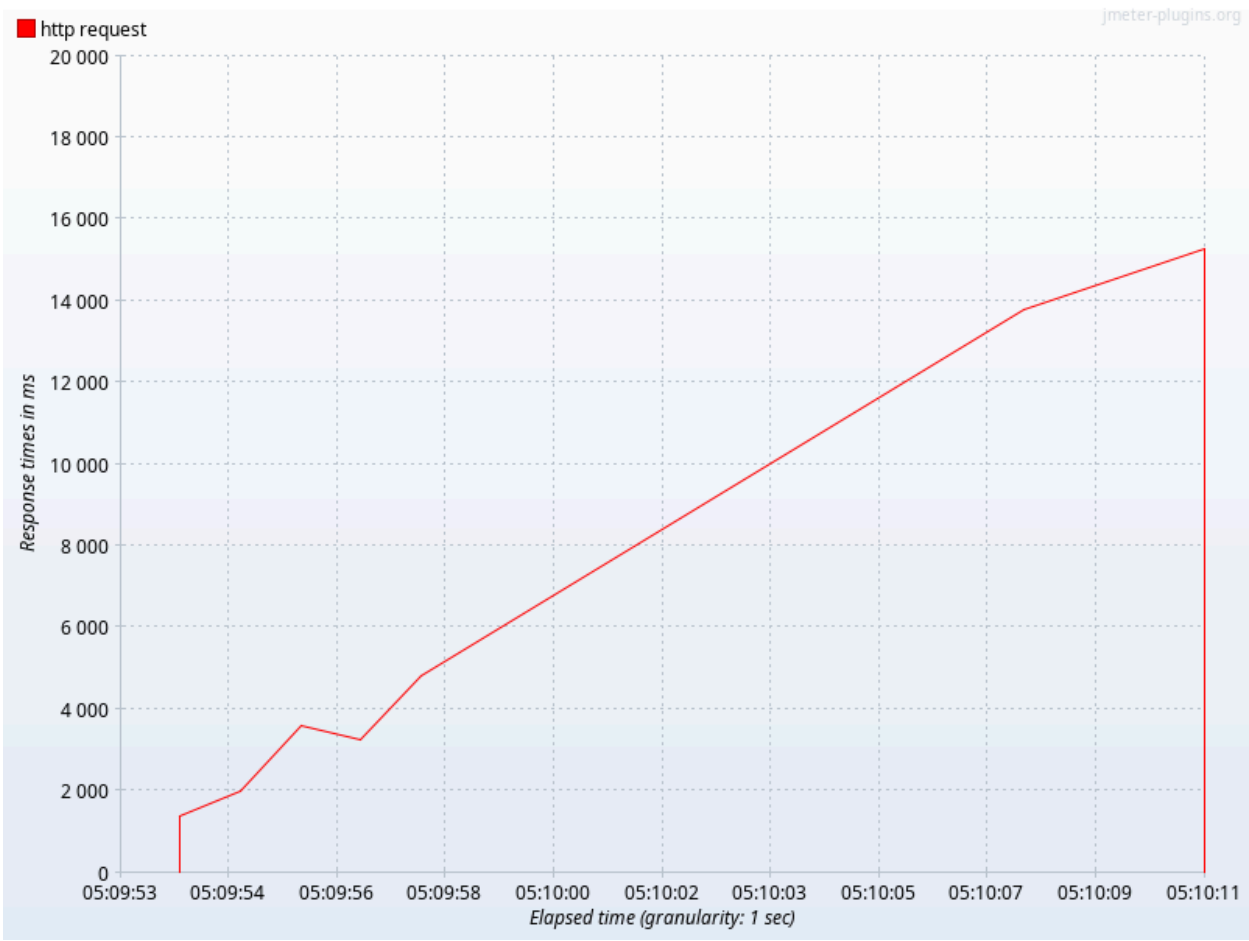
Startup delay (seconds):

Результаты стресс-тестирования

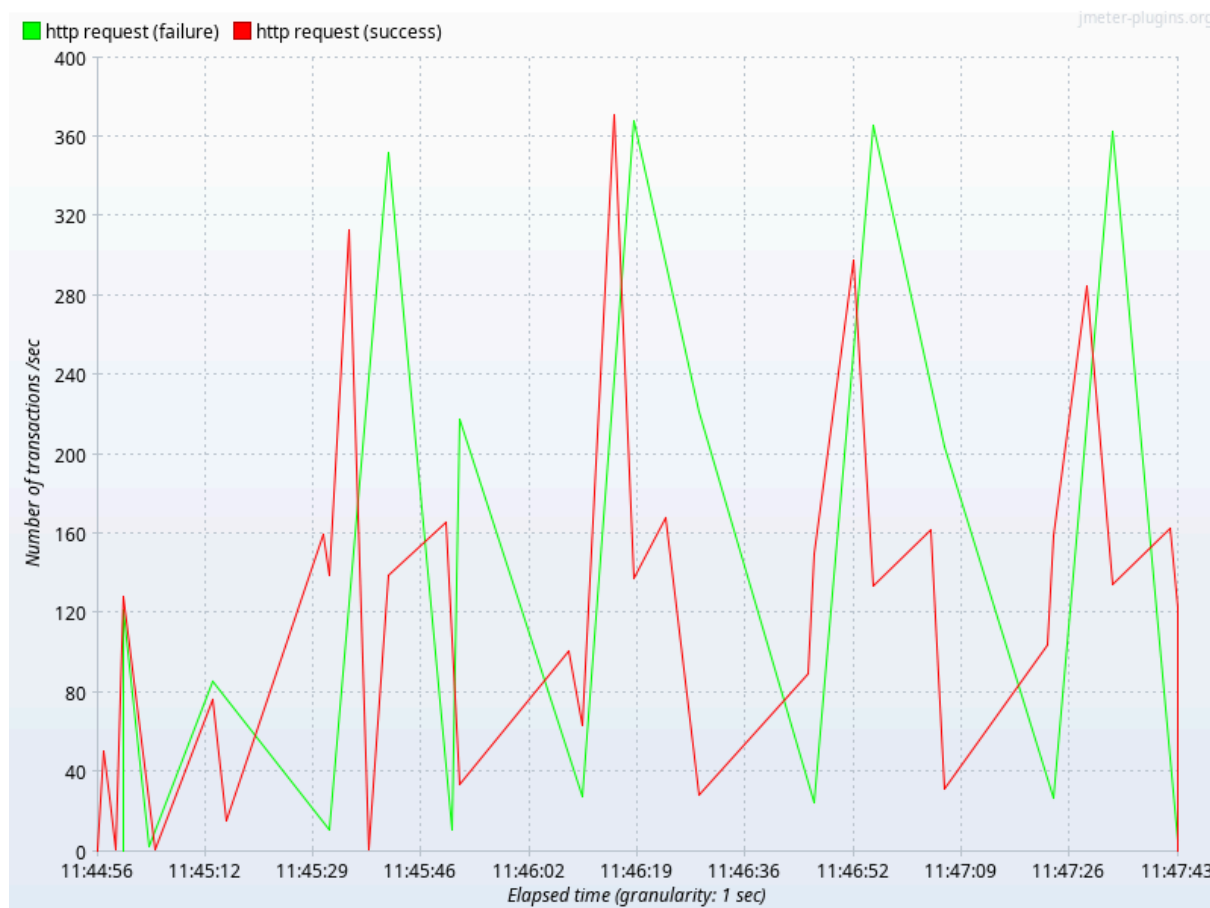
Label	# Samples	Average	Median	Min	Max	Error %	Throughput	Received KB/sec	Sent KB/sec
config #3	500	11749	14358	1366	16053	6.2	27.3/sec	6.16	4.24
TOTAL	500	11749	14358	1366	16053	6.2	27.3/sec	6.16	4.24

--- config #3

Время ответа (мс) от времени запроса



Решенные и нерешенные запросы пользователей



Вывод

В ходе лабораторной работы было проведено нагрузочное и стресс-тестирование веб-приложения с использованием Apache JMeter. Среди трёх конфигураций аппаратного обеспечения выбрана наиболее рациональная (конфигурация 3, \$3500), удовлетворяющая требованиям по времени отклика (≤ 920 мс) при заданной нагрузке (5 пользователей, 40 запросов в минуту).

Стресс-тестирование показало, что конфигурация №3 перестает справляться с нагрузкой при увеличении числа пользователей до 500, что подтверждается ростом времени отклика и появлением ошибок HTTP 503.

Полученные результаты демонстрируют важность тестирования производительности для оптимизации ресурсов и обеспечения стабильной работы приложения.