

Федеральное государственное автономное образовательное учреждение высшего
образования

«Национальный исследовательский университет ИТМО»

Факультет программной инженерии и компьютерной техники

Направление подготовки 09.03.01 «Информатика и вычислительная техника» —

Компьютерные системы и технологии

Отчёт

к лабораторной работе №4 по дисциплине

«Тестирование программного обеспечения»

Выполнили:

студент 3 курса

Краков Кирилл Константинович

Группа: Р3331

Принял:

Гаврилов Антон Валерьевич

Отчёт принят «__»_____ 2025 г.

Оценка: _____

г. Санкт-Петербург, 2025

Оглавление

Описание задания.....	3
Описание конфигурации JMeter для нагрузочного тестирования.....	3
Результаты нагрузочного тестирования и графики пропускной способности приложения	4
Анализ результатов нагрузочного тестирования и выводы по выбранной конфигурации аппаратного обеспечения	5
Описание конфигурации JMeter для стресс-тестирования.....	5
График изменения времени отклика от нагрузки для выбранной конфигурации, полученный в ходе стресс-тестирования системы.	6
Выводы по работе.....	8

Описание задания

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 требованиям 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 (\$ 5200)
- <http://stload.se.ifmo.ru:8080?token=495384109&user=-2105081711&config=1;>
- URL второй конфигурации (\$ 9400)
- <http://stload.se.ifmo.ru:8080?token=495384109&user=-2105081711&config=2;>
- URL третьей конфигурации (\$ 11300)
- <http://stload.se.ifmo.ru:8080?token=495384109&user=-2105081711&config=3;>
- Максимальное количество параллельных пользователей - 12;
- Средняя нагрузка, формируемая одним пользователем - 20 запр. в мин.;
- Максимально допустимое время обработки запроса - 760 мс.

Описание конфигурации JMeter для нагрузочного тестирования

Для проведения нагрузочного тестирования в Apache JMeter была создана тестовая конфигурация, учитывающая все требования задания:

1. Thread Groups (Группы потоков)

- Для каждой аппаратной конфигурации создана отдельная группа потоков: «Config 1 Group», «Config 2 Group», «Config 3 Group».
- *Number of Threads (users)* – Количество потоков (пользователей): 12 в каждой группе
- *Ramp-up period* – Период нарастания нагрузки: 60 секунд для плавного входа в режим полной нагрузки
- *Loop Count* – Счётчик циклов: бесконечность (ограничиваю длительность теста в поле *Duration (seconds)* – продолжительность в секундах: 90)

2. HTTP Request Samplers (Сэмплеры HTTP-запросов)

- Каждый сэмплер («Config 1», «Config 2», «Config 3») настроен на свою конфигурацию
- Метод: GET

- Протокол: HTTP
- IP: localhost
- Путь: /
- Порт: 32623
- Параметры запроса:
 - token: 495384109
 - user: -2105081711
 - config: 1, 2 или 3 (в зависимости от тестируемой конфигурации)
- Настройки соединения:
 - *Follow Redirects*: включено (Следовать за редиректами)
 - *Use KeepAlive*: включено (Не разрывать соединение после запроса)

3. Assertions (Проверки)

- *Duration Assertion* (Проверка длительности): 760 мс
- Применяется к: *Main sample only*
- Данная проверка соответствует требованию по максимально допустимому времени обработки запроса

4. Timers (Таймеры)

- *Constant Throughput Timer* – Таймер постоянной пропускной способности: 20.0 запросов в минуту
- Расчет пропускной способности: на основе данного потока (*this thread only*)
- Обеспечивает заданную нагрузку в 20 запросов в минуту на одного пользователя

5. Listeners (Слушатели)

- *View Results in Table* – Просмотр результатов в таблице и их запись в .csv формате (load_total.csv)
- *Graph Results* – Графические результаты
- *Aggregate Report* – Сводный отчет

Данная конфигурация JMeter обеспечивает эмуляцию 12 параллельных пользователей, каждый из которых генерирует 20 запросов в минуту к каждой из трех аппаратных конфигураций, с контролем времени отклика не более 760 мс.

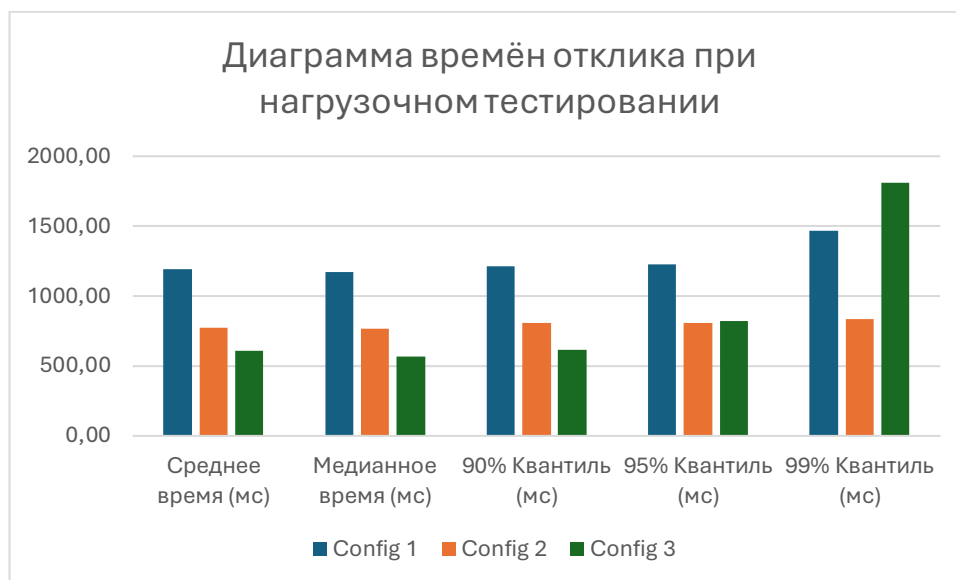
Результаты нагрузочного тестирования и графики пропускной способности приложения

Нагрузочное тестирование осуществлялось поочередно и отдельно для каждой группы потоков, описывающих разные конфигурации (это исключило доп. задержки на конфликт между группами). Но при этом, так как я осуществлял его из дома, для корректного соединения к внутренней сети кафедры использовал SSH-туннель (`ssh -f -N -L 32623:stload.se.ifmo.ru:8080 s368373@helios.cs.ifmo.ru -p 2222`), то это могло дать довольно существенные задержки (по оценке, проведённой ранее другими студентами, до 100-150 мс). Так или иначе, в ходе нагрузочного тестирования были получены следующие результаты для трех аппаратных конфигураций:

Конфигурация	Цена (\$)	Количество запросов в	Среднее время (мс)	Медианное время (мс)	90% Квантиль (мс)	95% Квантиль (мс)	99% Квантиль (мс)	Минимальное время (мс)	Минимальное время (мс)	Ошибки (%)	Пропускная способность
--------------	-----------	-----------------------	--------------------	----------------------	-------------------	-------------------	-------------------	------------------------	------------------------	------------	------------------------

											(запрос/сек)
Config 1	5200	260	1189	1169	1214	1227	1470	511	2454	100.00	1.98892
Config 2	9400	257	771	764	805	807	837	725	1164	61.089	2.83018
Config 3	11300	256	612	570	618	822	1808	525	1898	5.469	2.78983
TOTAL		773	859	767	1201	1213	1471	511	2454	55.757	1.58676

Получилась следующая диаграмма с указанием разных статистик для времени отклика у разных конфигураций приложения:



Анализ результатов нагрузочного тестирования и выводы по выбранной конфигурации аппаратного обеспечения

Результаты нагрузочного тестирования показывают, что наименьшее время нагрузочного отклика имеет 3 конфигурация, которая, что логично, является при этом самой дорогой. Первая конфигурация при этом является наименее производительной, имея значительно превышающее максимально возможное время отклика (медианное 1169 при максимальном 760) имеет, соответственно 100% ошибок из-за превышения этого времени, а также наименьшую пропускную способность из всех. Наконец, вторая конфигурация, несмотря на 61% ошибок в большей части случаев лишь незначительно превышает максимально допустимое время отклика (медианное 764 против максимального 760), а также имеет наибольшую пропускную способность и меньшую, чем у конфигурации 3 цену (9400 вместо 11300). Учитывая дополнительные задержки, получаемые из-за удалённого подключения через SSH-туннель, можно уверенно сказать, что при прямом подключении к внутренней сети кафедры именно вторая конфигурация при наименьшей цене дала бы удовлетворяющее условию время отклика (конфигурация 1 всё еще имела бы слишком высокое время отклика, так как задержки из-за туннеля не превышают 150-200 мс).

Итак, для дальнейшего стресс-тестирования я выбираю конфигурацию №2 (\$9400), но при этом увеличу порог допустимого времени отклика до 3000 мс вместо изначальных 760 мс, чтобы учесть задержки из-за удалённого подключения через SSH-туннель и сделать стресс-тестирование более репрезентативным на большой выборке пользователей

Описание конфигурации JMeter для стресс-тестирования

Для проведения стресс-тестирования была выбрана конфигурация #2 как наиболее подходящая по результатам нагрузочного тестирования. Стресс-тестирование проводилось

с целью определения точки отказа системы, когда время отклика превышает допустимый порог в 3000 мс.

1. Thread Groups (Группы потоков)

- Имя: Stress Test Config 2 Group
- Количество потоков (пользователей): от 12 до 107 с шагом увеличения 5 пользователей
- *Ramp-up period* – Период нарастания нагрузки: 10 секунд
- *Loop Count* – Счётчик циклов: бесконечность (ограничиваю длительность теста в поле *Duration (seconds)* – продолжительность в секундах: 60)

2. HTTP Request Samplers (Сэмплеры HTTP-запросов)

- Метод: GET
- Протокол: HTTP
- IP: localhost
- Путь: /
- Порт: 32623
- Параметры запроса:
 - token: 495384109
 - user: -2105081711
 - config: 2 (выбранная конфигурация)

3. Assertions (Проверки)

- *Duration Assertion* (Проверка длительности): 3000 мс
- Применяется к: *Main sample only*

4. Timers (Таймеры)

- *Constant Throughput Timer* – Таймер постоянной пропускной способности: 20.0 запросов в минуту
- Расчет пропускной способности: на основе данного потока (*this thread only*)

5. Listeners (Слушатели)

- *View Results in Table* – Просмотр результатов в таблице и их запись в .csv формате (stress12.csv, stress17.csv, ..., stress112.csv)
- *Graph Results* – Графические результаты
- *Aggregate Report* – Сводный отчет

Стресс-тестирование проводилось в 20 итераций с последовательным увеличением нагрузки для определения точки отказа системы. Это соответствует принципам тестирования емкости (capacity testing), который является подтипом стресс-тестирования и направлен на определение максимальной нагрузки, которую система может выдержать, оставаясь в пределах допустимых параметров производительности.

График изменения времени отклика от нагрузки для выбранной конфигурации, полученный в ходе стресс-тестирования системы.

По результатам стресс тестирования был получен следующая таблица и график зависимости времени отклика от нагрузки для конфигурации №2:

Кон фигу раци я	Количе ство запрос ов	Средн ее время (мс)	Медиан ное время (мс)	90% Квант иль (мс)	95% Квант иль (мс)	99% Квант иль (мс)	Минима льное время (мс)	Минима льное время (мс)	Ош ибк и (%)	Пропускная способность (запрос/мин)	Полу чено КБ/се к	Отпра влено КБ/се к
12	228	869	797	1069	1291	1572	746	2069	0,0 00	3,749	0,850	0,590
17	323	854	805	1018	1136	1360	726	1550	0,0 00	5,313	43831	0,840
22	415	881	818	1090	1257	1610	733	1796	0,0 00	6,804	19360	45839
27	511	896	834	1067	1218	1759	737	2500	0,0 00	8,396	32509	11689
32	603	919	858	1094	1255	1643	737	2440	0,0 00	9,918	45323	20455
37	698	918	880	1077	1180	1482	728	1698	0,0 00	11,474	21582	29221
42	792	972	909	1148	1348	2024	745	2630	0,0 00	12,982	34001	45749
47	886	996	938	1185	1424	1779	740	2061	0,0 00	14,550	46813	47150
52	979	1024	975	1210	1491	1895	753	2136	0,0 00	15,960	21976	18660
57	1074	1062	1007	1281	1467	2070	744	2683	0,0 00	17,622	35855	28157
62	1168	1087	1058	1265	1417	1962	738	2367	0,0 00	19,123	11414	45660
67	1262	1114	1097	1298	1438	1775	761	2075	0,0 00	20,672	24198	45717
72	1355	1182	1158	1373	1562	2015	750	2409	0,0 00	22,112	36251	17593
77	1450	1249	1235	1466	1599	1857	760	2022	0,0 00	23,646	12175	26359
82	1529	2186	2332	3107	3301	3671	747	3872	13, 669	24,491	19115	31107
87	1539	2695	3082	3369	4178	4269	746	4479	63, 093	24,633	20576	31837
92	1574	2878	3354	3645	3698	4283	739	4788	69, 822	25,108	24228	34759
97	1571	3104	3515	4038	4311	4728	749	5107	64, 799	24,839	22037	33298
102	1573	3262	3827	4242	4362	4612	738	5947	67, 896	25,061	23863	34394
107	1094	4912	4068	4815	21077	22319	744	22321	65, 082	18,190	46600	21217



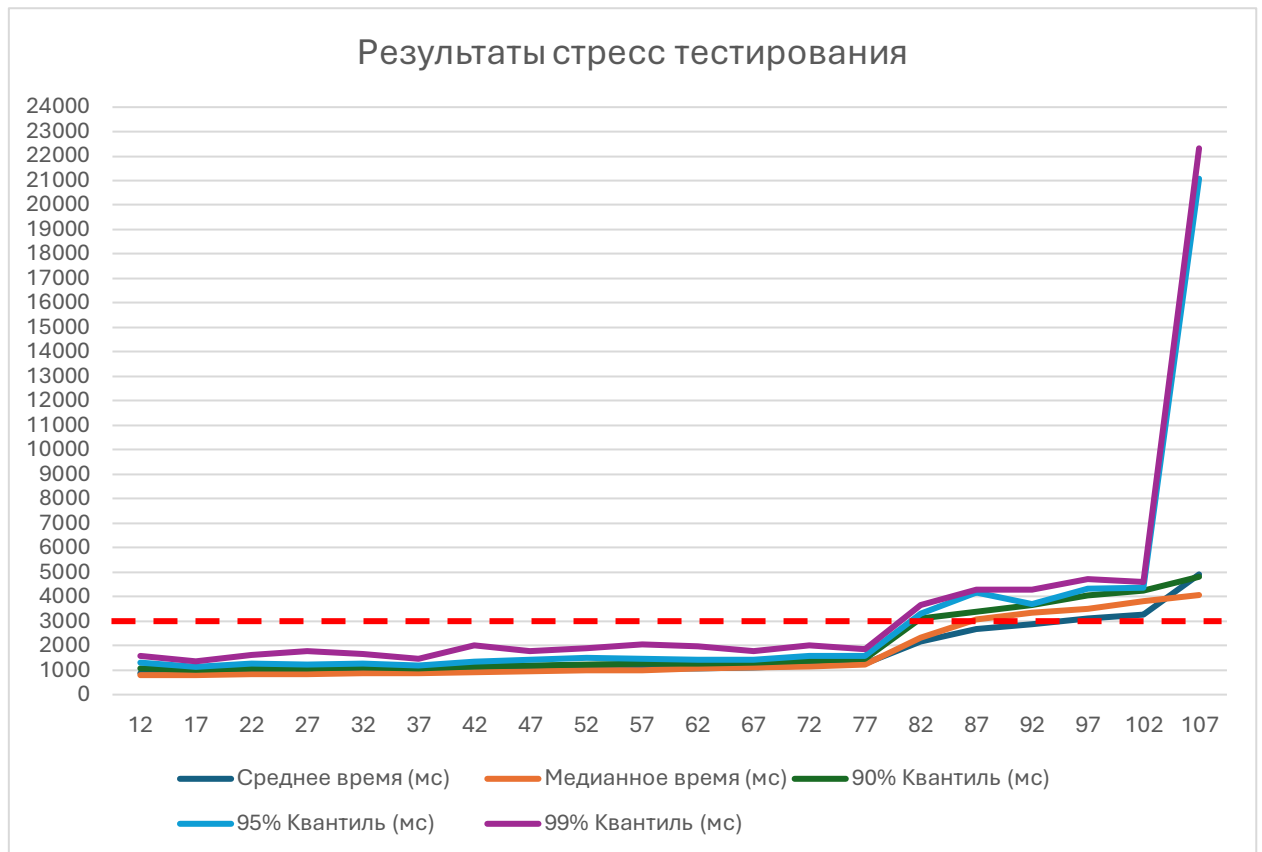


График демонстрирует важную особенность производительности системы: до определенной точки нагрузки время отклика растет умеренно, но затем наблюдается резкий скачок. Такое поведение характерно для многих систем и соответствует теоретическим принципам стресс-тестирования, где идентификация точки излома (knee point) является одной из ключевых задач.

Выводы по работе

На основе проведенных нагрузочного и стресс-тестирования можно сделать следующие выводы:

Оптимальная конфигурация аппаратного обеспечения:

- Конфигурация #2 стоимостью \$9400 показала наилучшие результаты по соотношению цена/производительность (с учетом задержек из-за удалённого подключения)

Предельная нагрузка:

- Система способна безошибочно обрабатывать приблизительно до 75-80 одновременных пользователей (при 20 запросах в минуту на каждого) с временем отклика до 3000 мс
- При превышении данной нагрузки наблюдается экспоненциальный рост времени отклика (который, однако, затем останавливается), что является классическим признаком достижения предела производительности

Точка излома производительности:

- Наблюдается при нагрузке около 77 пользователей

После этой точки даже небольшое увеличение нагрузки приводит к значительному росту времени отклика, который затем, однако также стабилизируется (по крайней мере на короткой дистанции)

Проведенное тестирование соответствует методологии CARAT и охватывает следующие аспекты:

- Capacity (Емкость): определены пределы системы по количеству одновременных пользователей - 100 исходя из 99-перцентиля и 125 исходя из 95 перцентиля.
- Availability (Доступность): измерен процент успешных запросов при различных уровнях нагрузки - если мы хотим максимальную доступность, то стоит ограничиться 70-ю пользователями.
- Reliability (Надежность): тестирование проводилось с постепенным увеличением нагрузки для оценки стабильности работы
- Accuracy (Точность): использовались различные метрики (среднее, медиана, перцентиля) для точной оценки производительности
- Throughput (Пропускная способность): проанализирована зависимость пропускной способности от количества пользователей

В целом, исследование демонстрирует важность комплексного подхода к тестированию производительности, сочетающего нагрузочное и стресс-тестирование для получения полной картины о возможностях системы и ее пределах. Мне очень понравилось работать с Apache JMeter в виду его простоты, удобства и интуитивности - описал тестовый план с группами потоков, создал сэмплеров с http запросами, добавил проверок, таймеров, слушателей и получил красивый и наглядный отчёт и производительности. Правда не понравились графики, их отдельно строил в Excel, можно написать программу на Python для ещё более информативных, но решил, что это излишне.