

# Вводная информация

Параметры нагрузочных тестов взяты для примера для для среднего веб-сервиса.

## Характеристики проводимых тестов:

1. Базовый тест:
  - Цель: Установить базовую производительность системы при нормальных условиях.
  - Характеристики: • Небольшое или среднее количество пользователей • Типичные сценарии использования • Обычная нагрузка
  - Зачем нужен: Чтобы понять, как система работает в нормальных условиях, и иметь точку отсчета для сравнения с другими тестами.
2. Тест производительности:
  - Цель: Оценить производительность системы при ожидаемой пиковой нагрузке.
  - Характеристики: • Количество пользователей соответствует максимальной ожидаемой нагрузке • Разнообразные сценарии использования • Продолжительное время выполнения (обычно от 30 минут до нескольких часов)
  - Зачем нужен: Чтобы убедиться, что система может справиться с ожидаемой максимальной нагрузкой, и выявить потенциальные узкие места.
3. Стресс-тест:
  - Цель: Определить пределы возможностей системы и ее поведение при экстремальной нагрузке.
  - Характеристики: • Количество пользователей превышает ожидаемую максимальную нагрузку • Интенсивные сценарии использования • Может включать неожиданные паттерны нагрузки
  - Зачем нужен: Чтобы найти точку отказа системы, понять, как система ведет себя при перегрузке, и оценить способность восстановления после стресса.

## Ключевые различия:

- Базовый тест использует обычную нагрузку для установления стандарта производительности.
- Тест производительности проверяет систему при ожидаемой максимальной нагрузке.
- Стресс-тест намеренно перегружает систему, чтобы найти ее пределы.

## Определения параметров нагрузки:

### Action to be taken after a Sampler error:

Определяет, что делать, если во время выполнения теста возникает ошибка (продолжить, запустить следующий поток, остановить поток, остановить тест).

Number of Threads (users):

Определяет количество параллельных пользователей (потоков).  
Каждый поток симулирует одного пользователя.

Ramp-up period (in seconds):

Время, за которое JMeter создаст все потоки.  
Например, если у вас 100 потоков и время разгона 100 секунд, JMeter будет создавать 1 поток в секунду.

Loop Count:

Сколько раз каждый поток должен выполнить тестовый сценарий.  
"Infinite" означает бесконечное выполнение.

Same user on each iteration:

Если выбрано, JMeter будет использовать один и тот же пользовательский контекст для каждой итерации.

Delay Thread creation until needed:

Если выбрано, потоки будут создаваться только когда они действительно нужны.

Specify Thread lifetime:

*Duration (seconds)* - максимальное время жизни каждого потока.

*Startup delay (seconds)* - задержка перед стартом каждого потока.

**Пример настройки:**

Number of Threads: 100  
Ramp-up period: 60 секунд  
Loop Count: 10

Это означает, что JMeter создаст 100 виртуальных пользователей за 60 секунд, и каждый пользователь выполнит тестовый сценарий 10 раз.

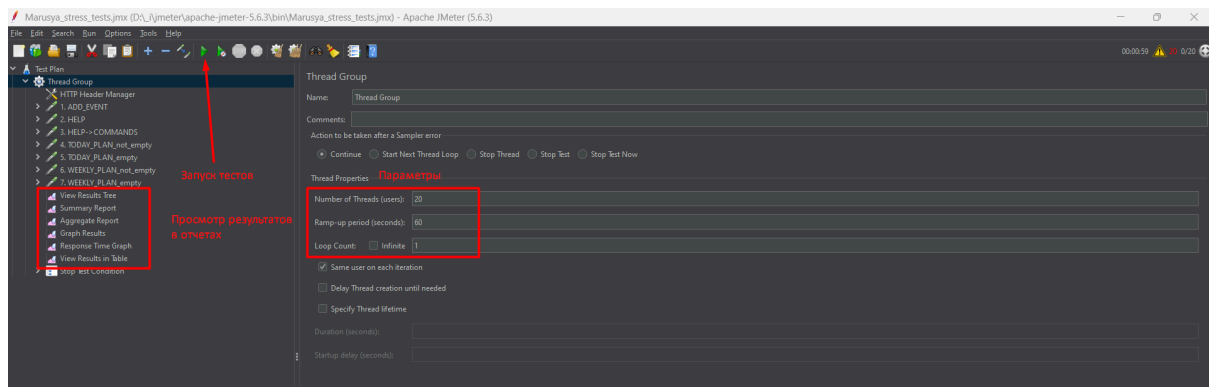
Правильная настройка этих параметров важна для создания реалистичной нагрузки на тестируемую систему. Выбор значений зависит от целей вашего тестирования и возможностей тестируемой системы.

**Важные колонки в Summary Report:**

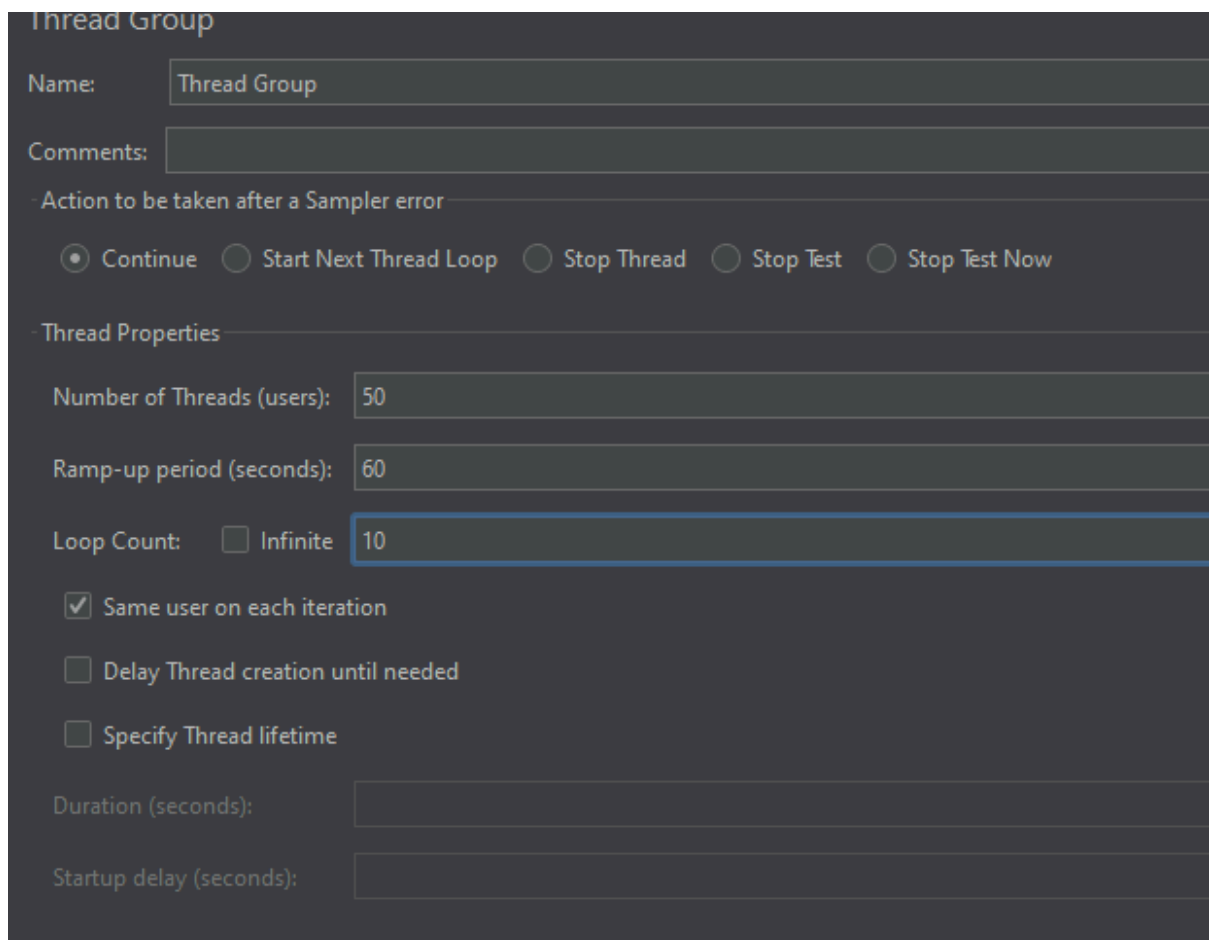
- Label: имя сэмплера
- Samples (отдельный запрос к серверу или выполнение определенной операции в тесте): количество сэмплов
- Average: среднее время отклика
- Error %: процент ошибок
- Throughput: пропускная способность (запросы в секунду)

## Запуск нагрузочного теста:

1. Перейти в раздел "Thread Group@"
2. Указать соответствующие параметры тестирования (указаны на соответствующих скринах ниже)
3. Нажать "Play"
4. Проанализировать результаты в последующих разделах



## 1. Базовый тест 50-60-10 Успешно



время выполнения 1:19

Summary Report

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
1. ADD_EVENT	560	399	265	2881	260.28	0.00%	2.3/min	0.03	0.05	856.0
2. HELP	560	281	263	1019	33.34	0.00%	2.3/min	0.03	0.05	826.0
3. HELP->COMMAN...	560	280	264	771	22.59	0.00%	2.3/min	0.03	0.05	722.0
4. TODAY_PLAN_net...	560	280	265	341	8.00	0.00%	2.3/min	0.03	0.08	819.0
5. TODAY_PLAN_em...	560	281	264	889	27.58	0.00%	2.3/min	0.03	0.08	818.0
6. WEEKLY_PLAN_no...	560	281	265	449	11.57	0.00%	2.3/min	0.05	0.10	1328.0
7. WEEKLY_PLAN_em...	560	280	264	741	21.78	0.00%	2.3/min	0.02	0.05	516.0
TOTAL	3920	297	263	2881	108.85	0.00%	16.0/min	0.22	0.47	840.7

2. Тест производительности 500-300-20 Успешно

Thread Group

Name:

Thread Group

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):

300

Loop Count:

☐ Infinite

20

☒ Same user on each iteration

☐ Delay Thread creation until needed

☐ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

время выполнения 5:41

Summary Report

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
1. ADD_EVENT	10560	316	256	2881	140.20	0.00%	41.1/min	0.57	0.89	856.0
2. HELP	10560	282	254	1019	21.34	0.00%	41.1/min	0.55	0.89	824.0
3. HELP->COMMAN...	10560	309	255	2470	125.02	0.00%	41.1/min	0.48	0.88	722.0
4. TODAY_PLAN_net...	10560	282	256	1009	19.87	0.00%	41.1/min	0.55	1.50	819.0
5. TODAY_PLAN_em...	10560	282	256	894	18.48	0.00%	41.1/min	0.55	1.50	818.0
6. WEEKLY_PLAN_no...	10560	282	256	809	18.82	0.00%	41.1/min	0.89	1.81	1328.0
7. WEEKLY_PLAN_em...	10560	281	254	988	19.40	0.00%	41.1/min	0.34	0.94	516.0
TOTAL	73920	291	254	2881	74.24	0.00%	4.8/sec	3.93	8.41	840.3

### 3. Стресс - тест 1000-120-1000 Успешно

Thread Group

Name:

Thread Group

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):

1000

Ramp-up period (seconds):

120

Loop Count:

☐ Infinite

1000

☒ Same user on each iteration

☐ Delay Thread creation until needed

☐ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

время выполнения 05:08:29

#### Summary Report

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
1. ADD_EVENT	378144	15256	256	38261989	535671.50	0.54%	4.3/sec	3.65	5.56	866.8
2. HELP	378000	15089	254	42398374	535236.78	0.53%	4.3/sec	3.52	5.54	836.8
3. HELP->COMMAN...	377871	15236	255	41781909	539166.03	0.53%	4.3/sec	3.08	5.53	733.5
4. TODAY_PLAN_not...	377741	14190	256	41785357	518010.76	0.53%	4.3/sec	3.49	9.41	829.8
5. TODAY_PLAN_em...	377610	14083	256	41781001	508839.63	0.54%	4.3/sec	3.48	9.40	829.1
6. WEEKLY_PLAN_no...	377471	13638	256	50353692	489017.01	0.54%	4.3/sec	5.61	11.30	1336.3
7. WEEKLY_PLAN_em...	377327	14612	254	50646413	527771.67	0.52%	4.3/sec	2.22	5.86	528.3
TOTAL	2644164	14586	254	50646413	522235.78	0.53%	30.1/sec	25.05	52.60	851.5

При стресс-тестировании показатель Error % = 0,54% в таблице Summary Report означает следующее:

- Уровень ошибок:
  - 0,54% всех запросов завершились с ошибкой.
  - 99,46% запросов были успешными.
- Интерпретация для стресс-теста:
  - Это относительно низкий процент ошибок, особенно для стресс-теста.
  - На каждые 1000 запросов приходится примерно 5-6 ошибок.
- Контекст стресс-тестирования:

- В стресс-тестах такой процент ошибок часто считается приемлемым или даже хорошим результатом.
- Стресс-тесты намеренно перегружают систему, поэтому наличие некоторого количества ошибок ожидаемо.

RESPONSE data (тело ответа сервера на запрос):

```
java.net.SocketException: Software caused connection abort: socket write error
    at java.net.SocketOutputStream.socketWrite0(Native Method)
    at java.net.SocketOutputStream.socketWrite(Unknown Source)
    at java.net.SocketOutputStream.write(Unknown Source)
    at sun.security.ssl.SSLSocketOutputRecord.flush(Unknown Source)
    at sun.security.ssl.HandshakeOutputStream.flush(Unknown Source)
    at sun.security.ssl.Finished$T13FinishedProducer.onProduceFinished(Unknown
Source)
    at sun.security.ssl.Finished$T13FinishedProducer.produce(Unknown Source)
    at sun.security.ssl.SSLHandshake.produce(Unknown Source)
    at sun.security.ssl.Finished$T13FinishedConsumer.onConsumeFinished(Unknown
Source)
    at sun.security.ssl.Finished$T13FinishedConsumer.consume(Unknown Source)
    at sun.security.ssl.SSLHandshake.consume(Unknown Source)
    at sun.security.ssl.HandshakeContext.dispatch(Unknown Source)
    at sun.security.ssl.HandshakeContext.dispatch(Unknown Source)
    at sun.security.ssl.TransportContext.dispatch(Unknown Source)
    at sun.security.ssl.SSLTransport.decode(Unknown Source)
    at sun.security.ssl.SSLSocketImpl.decode(Unknown Source)
    at sun.security.ssl.SSLSocketImpl.readHandshakeRecord(Unknown Source)
    at sun.security.ssl.SSLSocketImpl.startHandshake(Unknown Source)
    at sun.security.ssl.SSLSocketImpl.startHandshake(Unknown Source)
    at
org.apache.http.conn.ssl.SSLConnectionSocketFactory.createLayeredSocket(SSLConnec
tionSocketFactory.java:436)
    at
org.apache.http.conn.ssl.SSLConnectionSocketFactory.connectSocket(SSLConnectionSo
cketFactory.java:384)
    at
org.apache.jmeter.protocol.http.sampler.hc.LazyLayeredConnectionSocketFactory.connec
tSocket(LazyLayeredConnectionSocketFactory.java:87)
    at
org.apache.http.impl.conn.DefaultHttpClientConnectionOperator.connect(DefaultHttpClie
ntConnectionOperator.java:142)
    at
org.apache.jmeter.protocol.http.sampler.HTTPHC4Impl$JMeterDefaultHttpClientConnectio
nOperator.connect(HTTPHC4Impl.java:409)
    at
org.apache.http.impl.conn.PoolingHttpClientConnectionManager.connect(PoolingHttpClie
ntConnectionManager.java:376)
    at
org.apache.http.impl.execchain.MainClientExec.establishRoute(MainClientExec.java:393)
    at
org.apache.http.impl.execchain.MainClientExec.execute(MainClientExec.java:236)
```

```
        at org.apache.http.impl.execchain.ProtocolExec.execute(ProtocolExec.java:186)
        at org.apache.http.impl.execchain.RetryExec.execute(RetryExec.java:89)
        at org.apache.http.impl.execchain.RedirectExec.execute(RedirectExec.java:110)
        at
org.apache.http.impl.client.InternalHttpClient.doExecute(InternalHttpClient.java:185)
        at
org.apache.http.impl.client.CloseableHttpClient.execute(CloseableHttpClient.java:83)
        at
org.apache.jmeter.protocol.http.sampler.HTTPHC4Impl.executeRequest(HTTPHC4Impl.java:940)
        at
org.apache.jmeter.protocol.http.sampler.HTTPHC4Impl.sample(HTTPHC4Impl.java:651)
        at
org.apache.jmeter.protocol.http.sampler.HTTPSamplerProxy.sample(HTTPSamplerProxy.java:66)
        at
org.apache.jmeter.protocol.http.sampler.HTTPSamplerBase.sample(HTTPSamplerBase.java:1311)
        at
org.apache.jmeter.protocol.http.sampler.HTTPSamplerBase.sample(HTTPSamplerBase.java:1300)
            at org.apache.jmeter.threads.JMeterThread.doSampling(JMeterThread.java:651)
            at
org.apache.jmeter.threads.JMeterThread.executeSamplePackage(JMeterThread.java:570)
        )
        at
org.apache.jmeter.threads.JMeterThread.processSampler(JMeterThread.java:501)
        at org.apache.jmeter.threads.JMeterThread.run(JMeterThread.java:268)
        at java.lang.Thread.run(Unknown Source)
```

#### Sampler result:

Thread Name:Thread Group 1-308  
Sample Start:2024-07-17 02:46:37 GMT+03:00  
Load time:516512  
Connect Time:514684  
Latency:0  
Size in bytes:3486  
Sent bytes:0  
Headers size in bytes:0  
Body size in bytes:3486  
Sample Count:1  
Error Count:1  
Data type ("text"|"bin"|"" ):text  
Response code:Non HTTP response code: java.net.SocketException  
Response message:Non HTTP response message: Software caused connection abort:  
socket write error

HTTPSampleResult fields:

ContentType:  
DataEncoding: null

#### 4. Доп. тесты:

тест1: 20-60-1

Thread Group

Name: Thread Group

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): 20

Ramp-up period (seconds): 60

Loop Count: ☐ Infinite 1

☒ Same user on each iteration

☐ Delay Thread creation until needed

☐ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

время выполнения: 1 мин

#### Summary Report

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
1. ADD_EVENT	20	896	815	1416	123.38	0.00%	20.8/min	0.29	0.45	856.0
2. HELP	20	290	266	400	26.84	0.00%	21.2/min	0.28	0.46	826.0
3. HELP->COMMAN...	20	284	268	317	11.09	0.00%	21.2/min	0.25	0.46	722.0
4. TODAY_PLAN_not...	20	284	269	311	10.12	0.00%	21.2/min	0.28	0.78	819.0
5. TODAY_PLAN_em...	20	292	268	349	22.20	0.00%	21.2/min	0.28	0.78	818.0
6. WEEKLY_PLAN_no...	20	284	268	310	9.30	0.00%	21.2/min	0.46	0.93	1328.0
7. WEEKLY_PLAN_em...	20	283	266	325	11.74	0.00%	21.2/min	0.18	0.48	516.0
TOTAL	140	373	266	1416	218.84	0.00%	2.4/sec	1.93	4.13	840.7



тест2: 100-1-1

Thread Group

Name:

Thread Group

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):

100

Ramp-up period (seconds):

1

Loop Count:

☐ Infinite

1

☒ Same user on each iteration

☐ Delay Thread creation until needed

☐ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

время выполнения: 4 секунды

Summary Report

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
1. ADD_EVENT	120	1043	806	1516	206.89	0.00%	22.1/hour	0.01	0.01	856.0
2. HELP	120	280	258	400	13.81	0.00%	22.1/hour	0.00	0.01	826.0
3. HELP-> COMMAN...	120	276	257	317	7.82	0.00%	22.1/hour	0.00	0.01	722.0
4. TODAY_PLAN_not...	120	277	259	311	7.78	0.00%	22.1/hour	0.00	0.01	819.0
5. TODAY_PLAN_em...	120	282	259	349	15.70	0.00%	22.1/hour	0.00	0.01	818.0
6. WEEKLY_PLAN_no...	120	294	264	360	26.17	0.00%	22.1/hour	0.01	0.02	1328.0
7. WEEKLY_PLAN_em...	120	278	262	325	8.67	0.00%	22.1/hour	0.00	0.01	516.0
TOTAL	840	390	257	1516	278.07	0.00%	2.6/min	0.04	0.08	840.7

тест3: 500-1-1

Thread Group

Name:

Thread Group

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):

1

Loop Count:

☐ Infinite

1

☒ Same user on each iteration

☐ Delay Thread creation until needed

☐ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

\*\*\*

время выполнения: 7-13 секунд

Summary Report

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
1. ADD_EVENT	620	2392	806	4482	929.26	0.00%	1.9/min	0.03	0.04	856.0
2. HELP	620	389	256	1494	193.78	0.00%	1.9/min	0.03	0.04	826.0
3. HELP->COMMAN...	620	325	257	1180	83.98	0.00%	1.9/min	0.02	0.04	722.0
4. TODAY_PLAN_no...	620	313	299	762	55.77	0.00%	1.9/min	0.03	0.07	819.0
5. TODAY_PLAN_em...	620	339	299	1054	102.27	0.00%	1.9/min	0.03	0.07	818.0
6. WEEKLY_PLAN_no...	620	352	261	970	112.02	0.00%	1.9/min	0.04	0.08	1328.0
7. WEEKLY_PLAN_em...	620	332	253	792	80.66	0.00%	1.9/min	0.02	0.04	516.0
TOTAL	4340	635	253	4482	805.97	0.00%	13.2/min	0.18	0.39	840.7

тест4: 2000-1-1

Thread Group

Name:

Thread Group

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):

2000

Ramp-up period (seconds):

1

Loop Count:

☐ Infinite

1

☒ Same user on each iteration

☐ Delay Thread creation until needed

☐ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

время выполнения: 1 мин 53 секунды

Summary Report

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
1. ADD_EVENT	2620	7232	806	15631	3418.24	0.00%	7.9/min	0.11	0.17	856.0
2. HELP	2620	1519	256	4755	827.45	0.00%	7.9/min	0.11	0.17	826.0
3. HELP->COMMAN...	2620	1394	257	5154	739.89	0.00%	7.9/min	0.09	0.17	722.0
4. TODAY_PLAN_not...	2620	1388	259	3314	711.04	0.00%	7.9/min	0.11	0.29	819.0
5. TODAY_PLAN_em...	2620	1310	259	3393	691.33	0.00%	7.9/min	0.11	0.29	818.0
6. WEEKLY_PLAN_no...	2620	1205	261	3156	693.98	0.00%	7.9/min	0.17	0.35	1328.0
7. WEEKLY_PLAN_em...	2620	1015	253	3559	673.38	0.00%	7.9/min	0.07	0.18	516.0
TOTAL	18340	2152	253	15631	2538.23	0.00%	55.3/min	0.76	1.62	840.7

тест5: 3000-1-1 errors: 0.13% (очень незначительный % ошибок)

### Thread Group

Name:

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):

Ramp-up period (seconds):

Loop Count: ☐ Infinite

☒ Same user on each iteration  
☐ Delay Thread creation until needed  
☐ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

время выполнения: 30 секунд

## Summary Report

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
1. ADD_EVENT	3000	14953	1955	22018	2510.40	0.13%	135.7/sec	113.75	175.70	858.6
2. HELP	3000	1942	287	13182	1034.89	0.00%	135.8/sec	109.57	175.77	826.0
3. HELP->COMMAN...	3000	2026	276	4337	799.69	0.00%	188.3/sec	132.79	243.33	722.0
4. TODAY_PLAN_not...	3000	2246	279	5163	772.49	0.00%	175.7/sec	140.52	366.22	819.0
5. TODAY_PLAN_em...	3000	2163	292	6023	649.79	0.00%	161.0/sec	128.62	353.79	818.0
6. WEEKLY_PLAN_no...	3000	1949	286	5108	593.59	0.00%	160.0/sec	207.44	422.70	1328.0
7. WEEKLY_PLAN_em...	3000	1742	269	5128	709.77	0.00%	161.1/sec	81.17	220.53	516.0
TOTAL	21000	3860	269	22018	4684.50	0.02%	675.8/sec	555.07	1186.32	841.1

Sampler result

Request

Response data

Thread Name:Thread Group 1-908

Sample Start:2024-07-13 05:19:53 GMT+03:00

Load time:1871

Connect Time:1871

Latency:0

Size in bytes:2820

Sent bytes:0

Headers size in bytes:0

Body size in bytes:2820

Sample Count:1

Error Count:1

Data type ("text"|"bin"|"):text

Response code:Non HTTP response code: java.net.SocketException

Response message:Non HTTP response message: Connection reset

HTTPSampleResult fields:

ContentType:

DataEncoding: null

Sampler result	Request	Response data
Response Body	Response headers	
<div></div> <pre>java.net.SocketException: Connection reset     at java.net.SocketInputStream.read(Unknown Source)     at java.net.SocketInputStream.read(Unknown Source)     at sun.security.ssl.SSLSocketInputRecord.read(Unknown Source)     at sun.security.ssl.SSLSocketInputRecord.readHeader(Unknown Source)     at sun.security.ssl.SSLSocketInputRecord.decode(Unknown Source)     at sun.security.ssl.SSLTransport.decode(Unknown Source)     at sun.security.ssl.SSLSocketImpl.decode(Unknown Source)     at sun.security.ssl.SSLSocketImpl.readHandshakeRecord(Unknown Source)     at sun.security.ssl.SSLSocketImpl.startHandshake(Unknown Source)     at sun.security.ssl.SSLSocketImpl.startHandshake(Unknown Source)     at org.apache.http.conn.ssl.SSLConnectionSocketFactory.createLayeredSocket(SSLConnectionSocketFactory.java:436)     at org.apache.http.conn.ssl.SSLConnectionSocketFactory.connectSocket(SSLConnectionSocketFactory.java:384)     at org.apache.jmeter.protocol.http.sampler.HcLazyLayeredConnectionSocketFactory.connectSocket(LazyLayeredConnectionSocketFactory.java:87)     at org.apache.http.impl.conn.DefaultHttpClientConnectionOperator.connect(DefaultHttpClientConnectionOperator.java:142)     at org.apache.jmeter.protocol.http.sampler.HTTPHC4Impl\$JMeterDefaultHttpClientConnectionOperator.connect(HTTPHC4Impl.java:409)     at org.apache.http.impl.conn.PoolingHttpClientConnectionManager.connect(PoolingHttpClientConnectionManager.java:376)     at org.apache.http.impl.execchain.MainClientExec.establishRoute(MainClientExec.java:393)     at org.apache.http.impl.execchain.MainClientExec.execute(MainClientExec.java:236)     at org.apache.http.impl.execchain.ProtocolExec.execute(ProtocolExec.java:186)     at org.apache.http.impl.execchain.RetryExec.execute(RetryExec.java:89)     at org.apache.http.impl.execchain.RedirectExec.execute(RedirectExec.java:110)     at org.apache.http.impl.client.InternalHttpClient.doExecute(InternalHttpClient.java:185)     at org.apache.http.impl.client.CloseableHttpClient.execute(CloseableHttpClient.java:83)     at org.apache.jmeter.protocol.http.sampler.HTTPHC4Impl.executeRequest(HTTPHC4Impl.java:940)     at org.apache.jmeter.protocol.http.sampler.HTTPHC4Impl.sample(HTTPHC4Impl.java:651)     at org.apache.jmeter.protocol.http.sampler.HTTPSamplerProxy.sample(HTTPSamplerProxy.java:66)     at org.apache.jmeter.protocol.http.sampler.HTTPSamplerBase.sample(HTTPSamplerBase.java:1311)     at org.apache.jmeter.protocol.http.sampler.HTTPSamplerBase.sample(HTTPSamplerBase.java:1300)     at org.apache.jmeter.threads.JMeterThread.doSampling(JMeterThread.java:651)     at org.apache.jmeter.threads.JMeterThread.executeSamplePackage(JMeterThread.java:570)     at org.apache.jmeter.threads.JMeterThread.processSampler(JMeterThread.java:501)     at org.apache.jmeter.threads.JMeterThread.run(JMeterThread.java:268)     at java.lang.Thread.run(Unknown Source)</pre>		

тест6: 4000-1-1 errors: 1-5%% (незначительный % ошибок)

Thread Group

Name:

Thread Group

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):

4000

Ramp-up period (seconds):

1

Loop Count:

☐ Infinite

1

☒ Same user on each iteration

☐ Delay Thread creation until needed

☐ Specify Thread lifetime

Duration (seconds):

Startup delay (seconds):

время выполнения: 43 секунды

## Summary Report

Filename	<div><div>Browse...</div><div>Log/Display Only:</div><div><input type="checkbox"/> Errors</div><div><input type="checkbox"/> Successes</div><div>Configure</div></div>										
Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes	
1. ADD_EVENT	6620	14075	806	29138	6858.64	1.89%	19.9/min	0.29	0.42	893.1	
2. HELP	6620	2650	256	22474	2525.11	0.00%	19.9/min	0.27	0.43	826.0	
3. HELP->COMMAN...	6620	2595	257	5661	1427.26	0.00%	19.9/min	0.23	0.43	722.0	
4. TODAY_PLAN_not...	6620	2683	259	7025	1333.93	0.00%	19.9/min	0.27	0.73	819.0	
5. TODAY_PLAN_em...	6620	2613	259	5498	1302.18	0.00%	19.9/min	0.27	0.73	818.0	
6. WEEKLY_PLAN_no...	6620	2418	261	5435	1235.21	0.00%	19.9/min	0.43	0.88	1328.0	
7. WEEKLY_PLAN_em...	6620	1988	253	6076	1227.92	0.00%	19.9/min	0.17	0.45	516.0	
TOTAL	46340	4146	253	29138	5033.72	0.27%	2.3/sec	1.92	4.07	846.0	

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
1. ADD_EVENT	4000	18745	2120	26035	4123.36	5.30%	151.5/sec	142.00	186.00	960.1
2. HELP	4000	3158	279	21432	1976.36	0.00%	143.6/sec	115.81	185.77	826.0
3. HELP->COMMAN...	4000	3069	302	5544	891.34	0.00%	191.4/sec	134.92	247.24	722.0
4. TODAY_PLAN_not...	4000	3223	290	6814	836.56	0.00%	177.6/sec	142.01	390.31	819.0
5. TODAY_PLAN_em...	4000	3194	282	6528	765.08	0.00%	161.8/sec	129.28	355.59	818.0
6. WEEKLY_PLAN_no...	4000	3058	278	5885	719.01	0.00%	160.0/sec	207.52	422.86	1328.0
7. WEEKLY_PLAN_em...	4000	2568	280	5559	996.97	0.00%	165.2/sec	83.24	226.16	516.0
TOTAL	28000	5288	278	26035	5809.55	0.76%	663.9/sec	554.74	1159.16	855.6

Sampler resultRequestResponse data

Thread Name:Thread Group 1-908  
Sample Start:2024-07-13 05:19:53 GMT+03:00  
Load time:1871  
Connect Time:1871  
Latency:0  
Size in bytes:2820  
Sent bytes:0  
Headers size in bytes:0  
Body size in bytes:2820  
Sample Count:1  
Error Count:1  
Data type ("text"|"bin"|"):text  
Response code:Non HTTP response code: java.net.SocketException  
Response message:Non HTTP response message: Connection reset

HTTPSampleResult fields:  
ContentType:  
DataEncoding: null



Sampler resultRequestResponse data

Response BodyResponse headers

java.net.SocketException: Connection reset  
at java.net.SocketInputStream.read(Unknown Source)  
at java.net.SocketInputStream.read(Unknown Source)  
at sun.security.ssl.SSLSocketInputRecord.read(Unknown Source)  
at sun.security.ssl.SSLSocketInputRecord.readHeader(Unknown Source)  
at sun.security.ssl.SSLSocketInputRecord.decode(Unknown Source)  
at sun.security.ssl.SSLTransport.decode(Unknown Source)  
at sun.security.ssl.SSLSocketImpl.decode(Unknown Source)  
at sun.security.ssl.SSLSocketImpl.readHandshakeRecord(Unknown Source)  
at sun.security.ssl.SSLSocketImpl.startHandshake(Unknown Source)  
at sun.security.ssl.SSLSocketImpl.startHandshake(Unknown Source)  
at org.apache.http.conn.ssl.SSLConnectionSocketFactory.createLayeredSocket(SSLConnectionSocketFactory.java:436)  
at org.apache.http.conn.ssl.SSLConnectionSocketFactory.connectSocket(SSLConnectionSocketFactory.java:384)  
at org.apache.http.protocol.http.sampler.HcLazyLayeredConnectionSocketFactory.connectSocket(LazyLayeredConnectionSocketFactory.java:87)  
at org.apache.http.impl.conn.DefaultHttpClientConnectionOperator.connect(DefaultHttpClientConnectionOperator.java:142)  
at org.apache.jmeter.protocol.http.sampler.HTTPHC4Impl\$JMeterDefaultHttpClientConnectionOperator.connect(HTTPHC4Impl.java:409)  
at org.apache.http.impl.conn.PoolingHttpClientConnectionManager.connect(PoolingHttpClientConnectionManager.java:376)  
at org.apache.http.impl.execchain.MainClientExec.establishRoute(MainClientExec.java:393)  
at org.apache.http.impl.execchain.MainClientExec.execute(MainClientExec.java:236)  
at org.apache.http.impl.execchain.ProtocolExec.execute(ProtocolExec.java:186)  
at org.apache.http.impl.execchain.RetryExec.execute(RetryExec.java:89)  
at org.apache.http.impl.execchain.RedirectExec.execute(RedirectExec.java:110)  
at org.apache.http.impl.client.InternalHttpClient.doExecute(InternalHttpClient.java:185)  
at org.apache.http.impl.client.CloseableHttpClient.execute(CloseableHttpClient.java:83)  
at org.apache.jmeter.protocol.http.sampler.HTTPHC4Impl.executeRequest(HTTPHC4Impl.java:940)  
at org.apache.jmeter.protocol.http.sampler.HTTPHC4Impl.sample(HTTPHC4Impl.java:651)  
at org.apache.jmeter.protocol.http.sampler.HTTPSamplerProxy.sample(HTTPSamplerProxy.java:66)  
at org.apache.jmeter.protocol.http.sampler.HTTPSamplerBase.sample(HTTPSamplerBase.java:1311)  
at org.apache.jmeter.protocol.http.sampler.HTTPSamplerBase.sample(HTTPSamplerBase.java:1300)  
at org.apache.jmeter.threads.JMeterThread.doSampling(JMeterThread.java:651)  
at org.apache.jmeter.threads.JMeterThread.executeSamplePackage(JMeterThread.java:570)  
at org.apache.jmeter.threads.JMeterThread.processSampler(JMeterThread.java:501)  
at org.apache.jmeter.threads.JMeterThread.run(JMeterThread.java:268)  
at java.lang.Thread.run(Unknown Source)

Замечание:

После проведения тестов 5 и 6 стали появляться ошибки “java.net.SocketException: Connection reset” при параметрах теста4.

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
1. ADD_EVENT	4050	7920	1834	15946	2948.96	18.07%	73.3/sec	86.69	77.88	1211.0
2. HELP	4050	2671	300	11863	2270.52	0.00%	75.7/sec	61.07	97.97	826.0
3. HELP->COMMAN...	4050	1662	298	5210	612.11	0.00%	75.8/sec	53.43	97.90	722.0
4. TODAY_PLAN_not...	4050	1712	284	5365	571.14	0.00%	76.2/sec	60.92	167.43	819.0
5. TODAY_PLAN_em...	4050	1685	280	4352	517.57	0.00%	79.3/sec	63.35	174.25	818.0
6. WEEKLY_PLAN_no...	4050	1544	264	4647	531.55	0.00%	80.6/sec	104.52	212.97	1328.0
7. WEEKLY_PLAN_em...	4050	1282	259	4411	595.78	0.00%	80.9/sec	40.75	110.72	516.0
TOTAL	28350	2639	259	15946	2648.81	2.58%	464.7/sec	404.50	800.24	891.4

при повторном проведении теста4 - ошибок не обнаружено:

Label	# Samples	Average	Min	Max	Std. Dev.	Error %	Throughput	Received KB/sec	Sent KB/sec	Avg. Bytes
1. ADD_EVENT	2000	8994	950	13269	1971.16	0.00%	137.5/sec	114.91	178.28	856.0
2. HELP	2000	1613	282	4937	531.62	0.00%	131.1/sec	105.74	169.62	826.0
3. HELP->COMMAN...	2000	1660	287	4354	412.59	0.00%	125.5/sec	88.52	162.21	722.0
4. TODAY_PLAN_not...	2000	1608	282	3056	362.59	0.00%	134.2/sec	107.32	294.97	819.0
5. TODAY_PLAN_em...	2000	1551	284	3202	344.42	0.00%	148.1/sec	118.31	325.42	818.0
6. WEEKLY_PLAN_no...	2000	1493	275	3067	419.62	0.00%	156.2/sec	202.59	412.81	1328.0
7. WEEKLY_PLAN_em...	2000	1287	261	3202	548.29	0.00%	165.6/sec	83.44	226.72	516.0
TOTAL	14000	2601	261	13269	2747.39	0.00%	646.6/sec	530.86	1133.23	840.7

Параметры теста (Threads (users) = 2000, Ramp-up period = 1, Loop Count = 1), создают довольно интенсивную нагрузку, особенно с таким коротким периодом разгона:

1. Threads (users) = 2000:
- Это симулирует 2000 одновременных пользователей.

- Для "среднего" сервиса это может быть очень высокой нагрузкой, зависящей от специфики приложения и ожидаемого трафика.
- 2. Ramp-up period = 1 (секунда):
  - Все 2000 пользователей будут запущены в течение 1 секунды.
  - Это создает очень резкий скачок нагрузки, что не всегда реалистично для многих приложений.
- 3. Loop Count = 1:
  - Каждый виртуальный пользователь выполнит свой сценарий только один раз.
  - Тест будет относительно коротким, что может не отражать долгосрочное поведение системы под нагрузкой.

## Вывод:

Нагрузочное тестирование считаю успешно пройденным, так как при довольно интенсивной нагрузке процент ошибок был незначительным, а также при длительном стресс-тестировании: процент ошибок оказался крайне низким (0,5% для каждого запроса).