

**בחירות**

**מערכת RFID**

בדיקת עומסים UploadScannedTags

תוכן העניינים

[דגשים 3](#_Toc67004129)

[מטרת הבדיקה 3](#_Toc67004130)

[נושאים שנבדקו 3](#_Toc67004131)

[נושאים שלא נכללים בבדיקה 3](#_Toc67004132)

[ארכיטקטורת המערכת הבודקת 3](#_Toc67004133)

[רשת 3](#_Toc67004134)

[תצורה 3](#_Toc67004135)

[תסריט הבדיקה 4](#_Toc67004136)

[תיאור (מפתח המערכת) 4](#_Toc67004137)

[אופן ביצוע 4](#_Toc67004138)

[אופן ההעמסה 4](#_Toc67004139)

[תוצאות 6](#_Toc67004140)

[הרצה בתאריך 18/03/2021 7](#_Toc67004141)

[שעה 14:39:46 7](#_Toc67004142)

[שעה 15:16:28 13](#_Toc67004143)

[מסקנות ופעולות להמשך 21](#_Toc67004144)

[מסקנות 18/03/2021 21](#_Toc67004145)

[נספחים 22](#_Toc67004146)

[נספח א' – תצורת אפליקציה 22](#_Toc67004147)

[נספח ב' – מילון מונחים 23](#_Toc67004148)

[Summary Report 23](#_Toc67004149)

[Aggregate Report 24](#_Toc67004150)

[non-gui-test 25](#_Toc67004151)

# דגשים

* בזמן אמת, כל סריקה שמבוצעת מעדכנת את הבסיס נתונים.
* הבדיקה מורצת מול בסיס נתונים סביבת טסט על מנת לא להכניס מידע לא נדרש לבסיס נתונים ייצור.
* בפועל, כמות המשתמשים/רכיבים המרבית המריצים עדכון בו זמנית הנו 200.
* רוחב פס התקשורת המתוכנן הנו 150mb.

# מטרת הבדיקה

מטרת הבדיקה – סימולציה של התנהגות משאבי המערכת בזמן הרצת הבקשה ע"י כמות משתמשים מרבית.

# 

# נושאים שנבדקו

1. זמני תגובה של מתן התשובה.
2. התנהגות משאבי שרתים בזמן ההעמסהCPU, Memory-RAM, Swap, Disks I/O, Network I/O, TCP

# נושאים שלא נכללים בבדיקה

1. בדיקת ביצועים מצד משתמש קצה (Client).
2. תוכן התשובה הנשלחת בעקבות הבקשה.

# ארכיטקטורת המערכת הבודקת

## רשת

רוחב פס התקשורת מהמערכת הבודקת למערכת הנבדקת הנו 150mb כפי שיהיה ב Real Time.

## תצורה

המערכת הבודקת הנה אפליקציית Jmeter המותקנת בתצורת Master-Slave כאשר שרת אחד הנו שרת ניהול המערכת (master) המהווה ניהול וריכוז של כלל הפעולות המבוצעות במערכת + 11 שרתים המריצים את הבקשות מול המערכת הנבדקת.

תצורה זו מאפשרת העמסה מבוזרת על המערכת הנבדקת.

למידע נוסף בנושא ראה

* אפליקציה Jmeter - <https://jmeter.apache.org/>
* העמסה מבוזרת באמצעות Jmeter –

<https://jmeter.apache.org/usermanual/jmeter_distributed_testing_step_by_step.html>

למידע נוסף בנושא תצורת האפליקציה ראה נספח א' – תצורת אפליקציה.

# תסריט הבדיקה

#### תיאור (מפתח המערכת)

* API שמתבצע לאחר סיום של כל סריקה,
* למעשה הוא מרכז בטבלה את כל הסריקות שבוצעו במערכת לטובת הצגת הנתונים ב Dashboard.

(המידע עולה בקבצים גדולים וקורה הרבה פעמים (כל סריקה מעלים מידע) כאשר כל סיום סריקה

יהיה 350 רשומות בגודל.)

* **API זה הוא חשוב לבדיקה מכיוון שזהו ה- API שיתבצע הכי הרבה פעמיים במערכת, בכל סיום סריקה תתבצע קריאה אליו**

#### אופן ביצוע

תסריט הבדיקה הנה פקודת API המתבצעת מול שרת LB (Load Balancer) בכתובת 10.50.1.10.

הבדיקה מורצת מול בסיס נתונים סביבת טסט ולכן

LB\_URL= 10.50.1.10/WSElections\_test/api/

${LB\_URL}/UploadItems/UploadScannedTags

פקודת ה- API הנה שאילתת SQL לשם הכנסת נתונים לבסיס הנתונים

1. Exec ClearScannedItems @ContainerRFID, @OperationType
2. Exec UpdateScannedItems @SiteID, @ContainerRFID, @ItemRFID, @Status, @User, @ScannedOperation, @LastScanedTime, @DeviceID, @UpdatedAt

לשם סימולציה הנתונים שיוכנסו מקונפגים כ- Json

[

{

    "siteID": 1,

    "ContainerRFID": "1111",

    "User": "Tomer",

    "ScanedTime": "2021-03-23T18:25:43.511Z",

    "operationType": 3,

    "DeviceID": "123",

    "ScannedTags": [{"ItemRFID": "11111","ItemStatus" : 0},

                    {"ItemRFID": "22222","ItemStatus" : 0},

                    {"ItemRFID": "33333","ItemStatus" : 1}]

}

]

# אופן ההעמסה

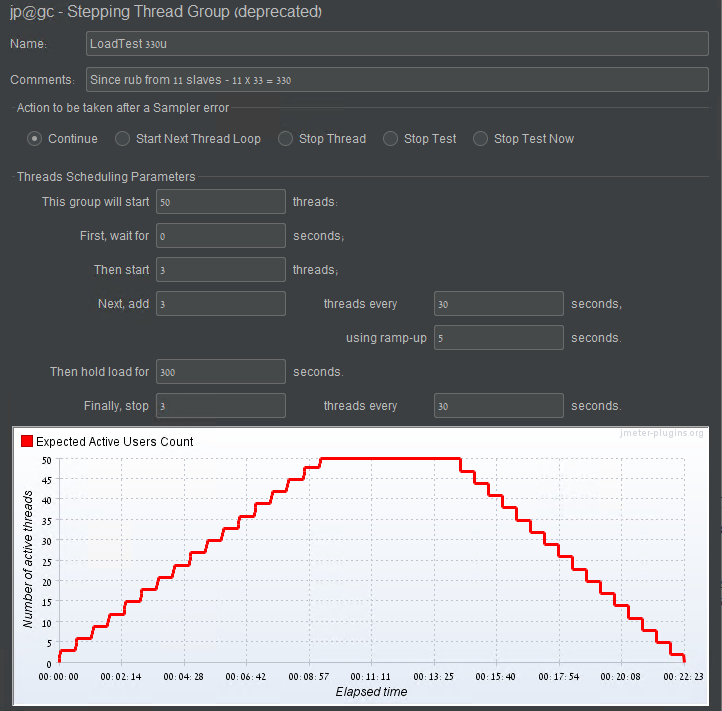
הרצת התסריט תבוצע תחת שתי העמסות – עומס מרבי של 330 תהליכים (לעדכון בסיס נתונים) במקביל (בו זמנית), עומס מרבי של 550 תהליכים (לעדכון בסיס נתונים) במקביל (בו זמנית).

עומס מרבי של 330 בקשות בו זמנית

* העמסה של 3 משתמשים כל חצי דקה.
* העמסה מרבית של 330 משתמשים למשך 3 דקות.
* הורדה (הגרעה של...) 3 משתמשים כל חצי דקה.

**הערה:**

צילום מסך נלקח ממחשב ה- Master. ההרצה נעשית מ- 11 Slave ולכן יש להכפיל ב- 11 את מספר ה- Threads.

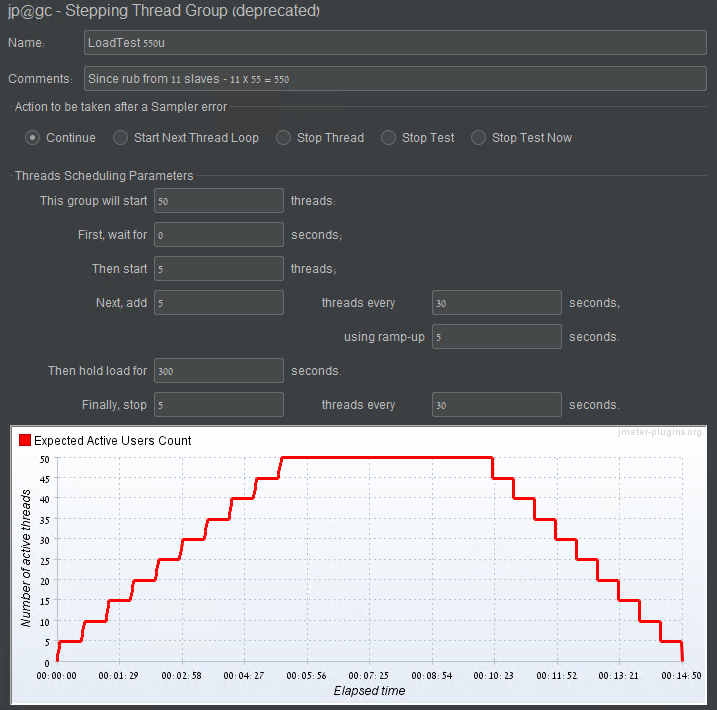


עומס מרבי של 550 בקשות בו זמנית

* העמסה של 5 משתמשים כל חצי דקה.
* העמסה מרבית של 550 משתמשים למשך 3 דקות.
* הורדה (הגרעה של...) 5 משתמשים כל חצי דקה.

**הערה:**

צילום מסך נלקח ממחשב ה- Master. ההרצה נעשית מ- 11 Slave ולכן יש להכפיל ב- 11 את מספר ה- Threads.



# תוצאות

## הרצה בתאריך 18/03/2021

### שעה 14:39:46

**העמסה של מספר משתמשים מרבי:** 550

שעת התחלה: 2021/03/18 14:39:46

שעת סיום: 2021/03/18 14:54:31

זמני תגובה:

שרתי אפליקציה (פניה מ- LB)

דו"ח סיכום

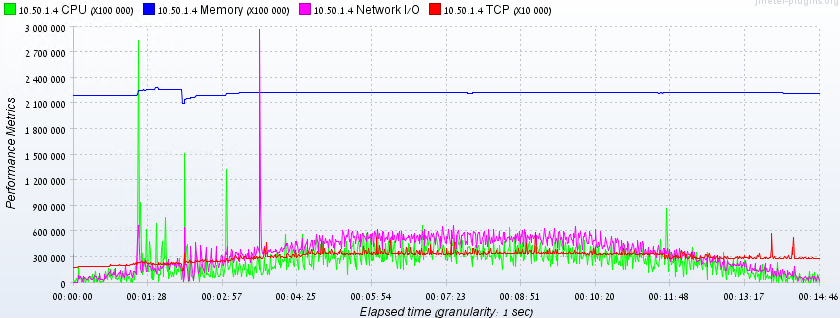
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Samples** | **Average** | **Min** | **Max** | **Std.Dev** | **Error %** | **Throughput** | **Received KB/sec** | **Sent KB/sec** | **Avg.Bytes KB/sec** |
| UploadScannedTags | 54062 | 27 | 1 | 3152 | 94.50944981433683 | 0.0017017498427731124 | 61.0861083396515 | 7.017836973157456 | 34.95747996780838 | 117.64155969072546 |
| TOTAL | 54062 | 27 | 1 | 3152 | 94.50944981433683 | 0.0017017498427731124 | 61.0861083396515 | 7.017836973157456 | 34.95747996780838 | 117.64155969072546 |

קיבוץ נתונים

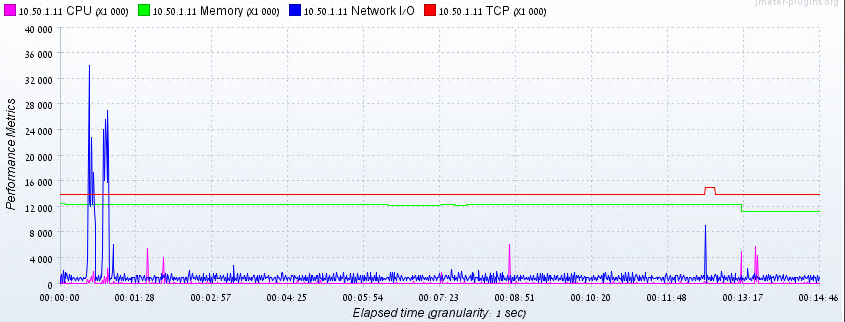
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Samples** | **Average** | **Median** | **90% Line** | **95% Line** | **99% Line** | **Min** | **Maximum** | **Error %** | **Throughput** | **Received KB/sec** | **Sent KB/sec** |
| UploadScannedTags | 54062 | 27 | 11 | 53 | 75 | 165 | 1 | 3152 | 0.0017017498427731124 | 61.08610833965151 | 7.017836973157456 | 34.95747996780838 |
| TOTAL | 54062 | 27 | 11 | 53 | 75 | 165 | 1 | 3152 | 0.0017017498427731124 | 61.08610833965151 | 7.017836973157456 | 34.95747996780838 |

**צריכת משאבים בשרתים**

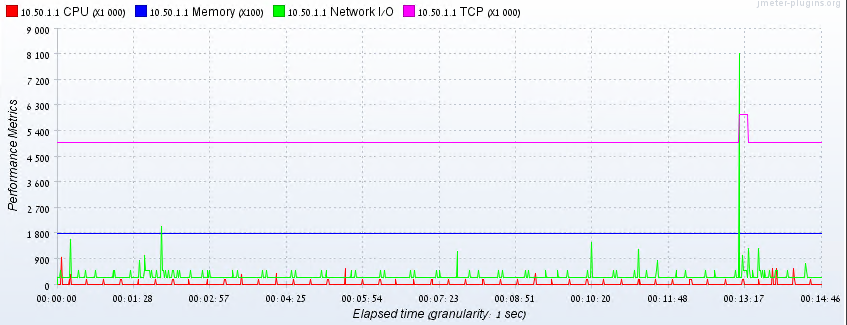
**IIS Server**



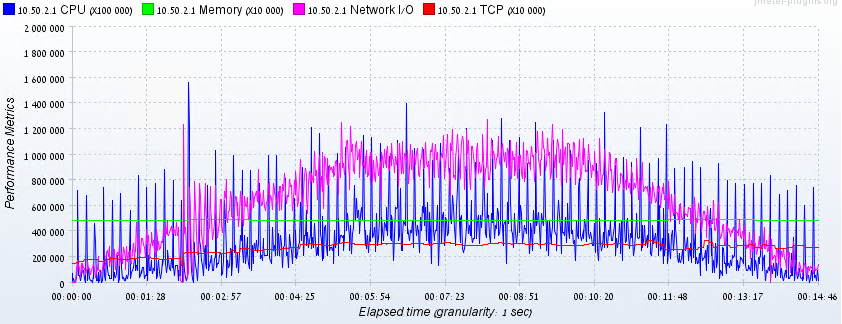
**IIS02 Server**



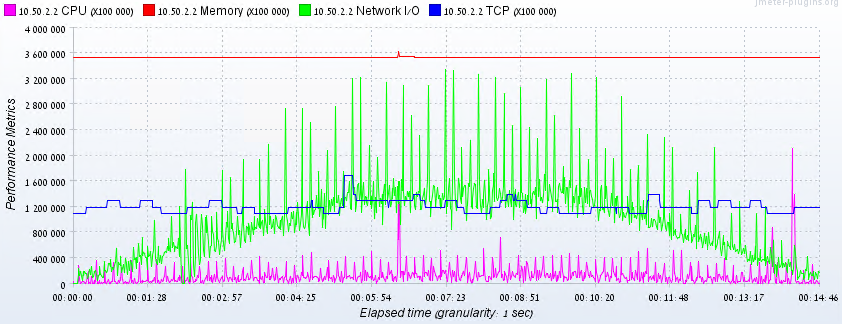
**Report Server**



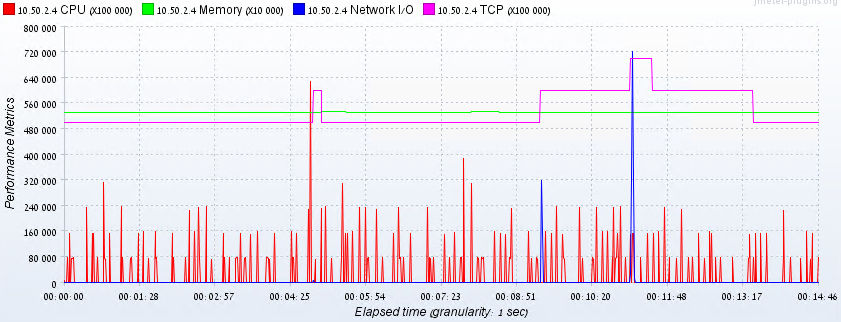
**SQL01 Server**



**SQL02 Server**



**File Server**



**Jmeter Log**

2021-03-18 14:44:00,109 INFO o.a.j.r.Summariser: summary + 6800 in 00:01:59 = 57.2/s Avg: 41 Min: 1 Max: 3152 Err: 92 (1.35%) Active: 440 Started: 432 Finished: 0

2021-03-18 14:44:00,110 INFO o.a.j.r.Summariser: summary = 8714 in 00:04:19 = 33.6/s Avg: 35 Min: 1 Max: 3152 Err: 92 (1.06%)

2021-03-18 14:44:30,025 INFO o.a.j.r.Summariser: summary + 2300 in 00:00:30 = 76.9/s Avg: 31 Min: 7 Max: 172 Err: 0 (0.00%) Active: 495 Started: 487 Finished: 0

2021-03-18 14:44:30,026 INFO o.a.j.r.Summariser: summary = 11014 in 00:04:49 = 38.1/s Avg: 34 Min: 1 Max: 3152 Err: 92 (0.84%)

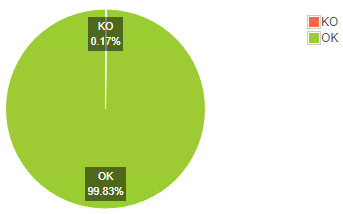
משורה 552 כשפקודת ה- API מופעלת ע"י 440 רכיבים/משתמשים נוצר מצב שלא כל הנתונים מוכנסים לבסיס הנתונים, ויש זיהוי חוזר של המספר 92 .

**HTML Report**

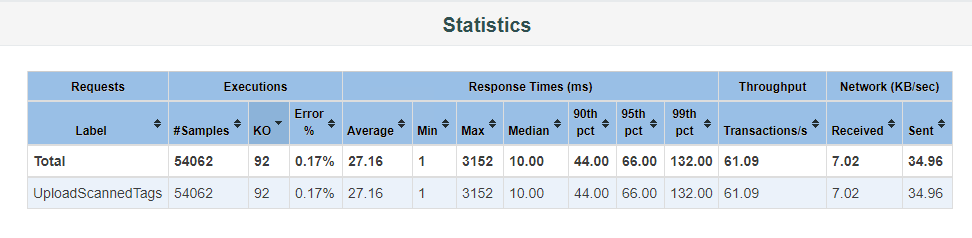
בסיכום כולל של הפניות למערכת נראה ש 0.17% פניות נכשל.

KO פירושו לא תקין נכשל.

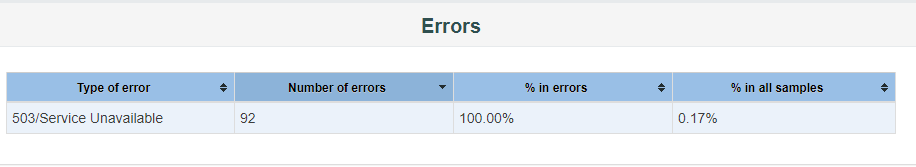
הוא מופיע בקונפיגורציה של המערכת הבודקת (JMeter) כערך של reportgenerator\_summary\_statistics\_error\_count (reportgenerator\_summary\_statistics\_error\_count=KO)



ניתן לראות ב- HTML Report בטבלת סטטיסטיקות שקיימות 92 שגיאות בפניה למערכת

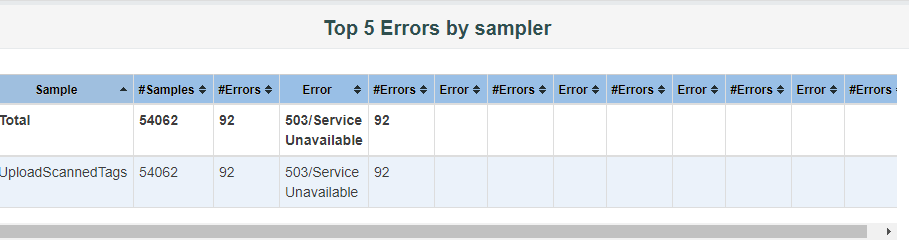


ניתן לראות ב- HTML Report בטבלת השגיאות שהסיבה לשגיאה היא 503.



ניתן לראות ב- HTML Report בטבלת **Top 5 Errors By Sampler** סיכום ש כל הנתונים לעיל

* הסיבה לשגיאה היא 503.
* כמות השגיאות 92.
* הפניה היא פניה אחת יחידה בתסריט המזוהה כמספר 54062.



ניתוח תוצאות ניתן לראות בקובץ המצורפים



**סיכום כלל התוצאות:**

שגיאה 503 (המופיעה ב- HTML Report) מציינת שהשרת אינו פנוי לטפל בפניה.

ע"פ Jmeter Log, הפניה לשרת שבה מופיעה שגיאה 503 מתחילה להיות אירוע חוזר החל מכמות פניות של 440 בו זמנית ולמעלה מזה.

### שעה 15:16:28

**העמסה של מספר משתמשים מרבי:** 330

שעת התחלה: 2021/03/18 15:16:28

שעת סיום: 2021/03/18 15:38:46

זמני תגובה:

שרתי אפליקציה (פניה מ- LB)

דו"ח סיכום

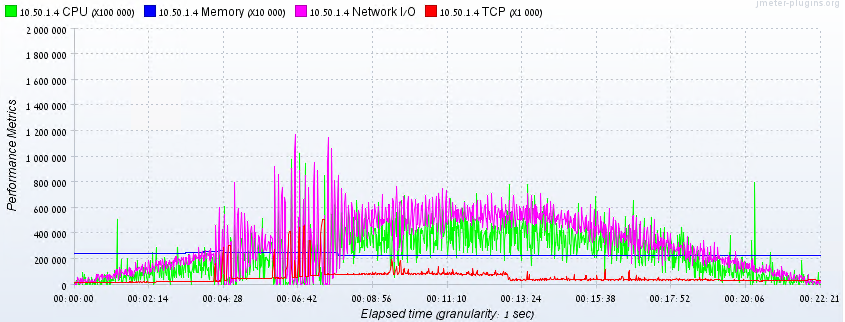
|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Samples** | **Average** | **Min** | **Max** | **Std.Dev** | **Error %** | **Throughput** | **Received KB/sec** | **Sent KB/sec** | **Avg.Bytes KB/sec** |
| UploadScannedTags | 72104 | 247 | 5 | 11093 | 975.0329906052966 | 0.0 | 53.88833998741429 | 6.157163846218235 | 30.838444563110134 | 117.0 |
| TOTAL | 72104 | 247 | 5 | 11093 | 975.0329906052966 | 0.0 | 53.88833998741429 | 6.157163846218235 | 30.838444563110134 | 117.0 |

קיבוץ נתונים

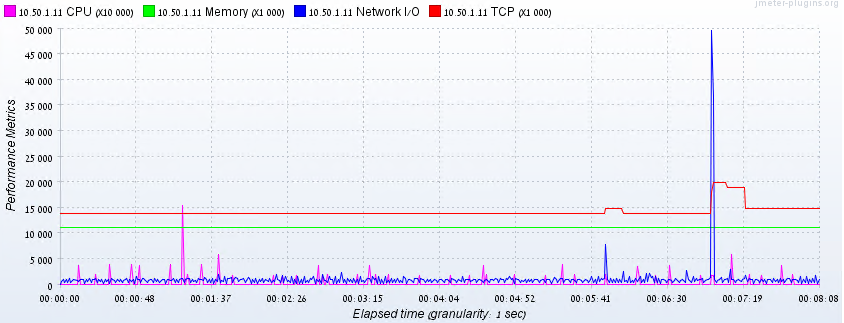
|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Label** | **Samples** | **Average** | **Median** | **90% Line** | **95% Line** | **99% Line** | **Min** | **Maximum** | **Error %** | **Throughput** | **Received KB/sec** | **Sent KB/sec** |
| UploadScannedTags | 72104 | 247 | 41 | 243 | 684 | 6232 | 5 | 11093 | 0.0 | 53.8883399874143 | 6.157163846218235 | 30.838444563110134 |
| TOTAL | 72104 | 247 | 41 | 243 | 684 | 6232 | 5 | 11093 | 0.0 | 53.8883399874143 | 6.157163846218235 | 30.838444563110134 |

**צריכת משאבים בשרתים**

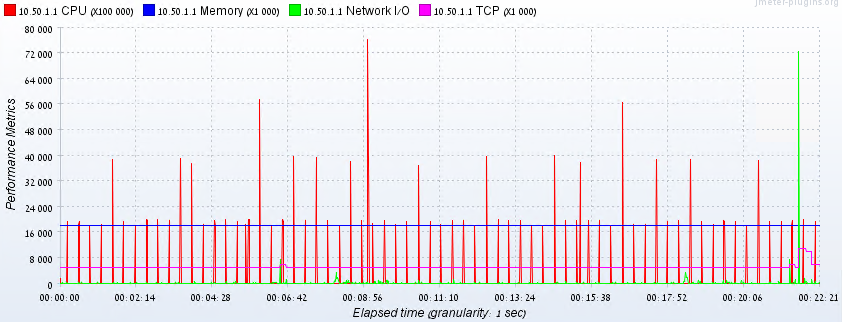
**IIS Server**



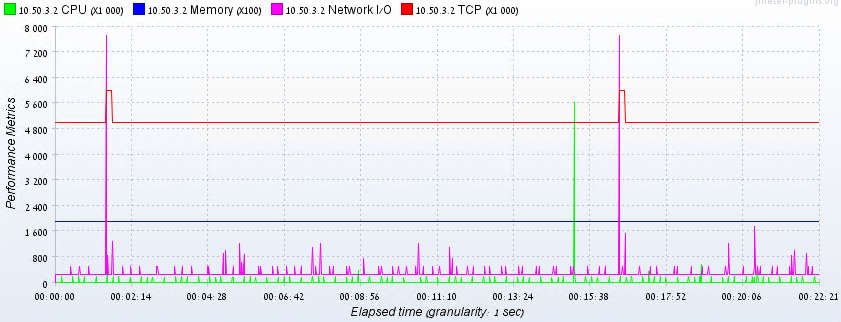
**IIS02 Server**



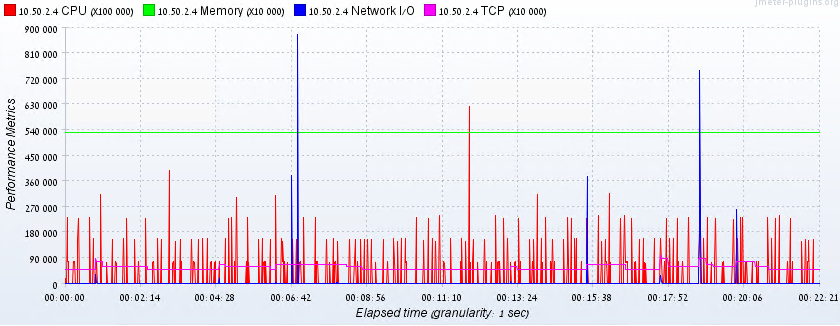
**Report Server**



**Log Server**



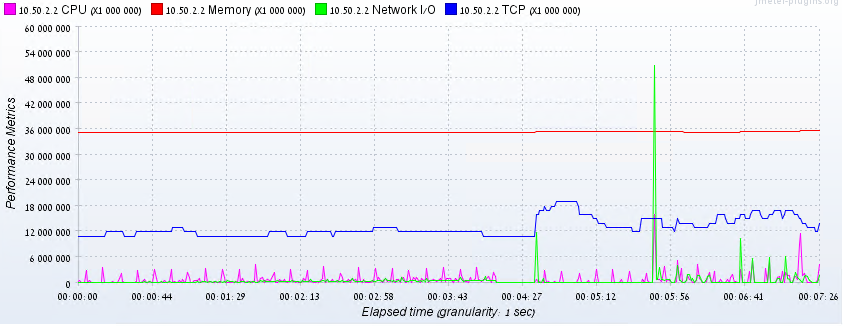
**File Server**



**SQL01 Server**



**SQL02 Server**



**Jmeter Log**

2021-03-18 15:16:24,500 INFO o.a.j.s.SampleResult: sampleresult.nanoThreadSleep=5000

2021-03-18 15:16:30,063 INFO o.a.j.r.Summariser: summary + 8 in 00:00:07 = 1.1/s Avg: 14 Min: 9 Max: 29 Err: 0 (0.00%) Active: 33 Started: 27 Finished: 0

2021-03-18 15:17:30,204 INFO o.a.j.r.Summariser: summary + 178 in 00:01:00 = 3.0/s Avg: 12 Min: 8 Max: 65 Err: 0 (0.00%) Active: 66 Started: 60 Finished: 0

2021-03-18 15:17:30,205 INFO o.a.j.r.Summariser: summary = 186 in 00:01:07 = 2.8/s Avg: 12 Min: 8 Max: 65 Err: 0 (0.00%)

2021-03-18 15:18:30,016 INFO o.a.j.r.Summariser: summary + 493 in 00:01:00 = 8.2/s Avg: 13 Min: 7 Max: 69 Err: 0 (0.00%) Active: 132 Started: 126 Finished: 0

2021-03-18 15:18:30,016 INFO o.a.j.r.Summariser: summary = 679 in 00:02:07 = 5.3/s Avg: 13 Min: 7 Max: 69 Err: 0 (0.00%)

2021-03-18 15:22:01,605 INFO o.a.j.r.Summariser: summary + 7462 in 00:03:32 = 35.3/s Avg: 393 Min: 5 Max: 7639 Err: 0 (0.00%) Active: 330 Started: 324 Finished: 0

2021-03-18 15:22:01,606 INFO o.a.j.r.Summariser: summary = 8141 in 00:05:39 = 24.0/s Avg: 361 Min: 5 Max: 7639 Err: 0 (0.00%)

2021-03-18 15:23:30,450 INFO o.a.j.r.Summariser: summary + 4500 in 00:01:29 = 50.7/s Avg: 936 Min: 6 Max: 11093 Err: 0 (0.00%) Active: 429 Started: 423 Finished: 0

2021-03-18 15:23:30,450 INFO o.a.j.r.Summariser: summary = 12641 in 00:07:07 = 29.6/s Avg: 566 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:25:03,178 INFO o.a.j.r.Summariser: summary + 6500 in 00:01:33 = 70.1/s Avg: 878 Min: 6 Max: 10456 Err: 0 (0.00%) Active: 495 Started: 489 Finished: 0

2021-03-18 15:25:03,178 INFO o.a.j.r.Summariser: summary = 19141 in 00:08:40 = 36.8/s Avg: 672 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:25:30,113 INFO o.a.j.r.Summariser: summary + 2300 in 00:00:27 = 85.4/s Avg: 44 Min: 16 Max: 319 Err: 0 (0.00%) Active: 528 Started: 522 Finished: 0

2021-03-18 15:25:30,113 INFO o.a.j.r.Summariser: summary = 21441 in 00:09:07 = 39.2/s Avg: 605 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:26:32,282 INFO o.a.j.r.Summariser: summary + 5400 in 00:01:02 = 86.9/s Avg: 216 Min: 14 Max: 1623 Err: 0 (0.00%) Active: 550 Started: 544 Finished: 0

2021-03-18 15:26:32,282 INFO o.a.j.r.Summariser: summary = 26841 in 00:10:09 = 44.1/s Avg: 526 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:27:33,354 INFO o.a.j.r.Summariser: summary + 5500 in 00:01:01 = 90.1/s Avg: 103 Min: 14 Max: 790 Err: 0 (0.00%) Active: 550 Started: 544 Finished: 0

2021-03-18 15:27:33,369 INFO o.a.j.r.Summariser: summary = 32341 in 00:11:10 = 48.2/s Avg: 454 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:28:34,267 INFO o.a.j.r.Summariser: summary + 5500 in 00:01:01 = 90.3/s Avg: 91 Min: 14 Max: 697 Err: 0 (0.00%) Active: 550 Started: 544 Finished: 0

2021-03-18 15:28:34,267 INFO o.a.j.r.Summariser: summary = 37841 in 00:12:11 = 51.7/s Avg: 402 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:29:00,262 INFO o.a.j.r.Summariser: summary + 2900 in 00:00:26 = 111.6/s Avg: 91 Min: 14 Max: 457 Err: 0 (0.00%) Active: 550 Started: 544 Finished: 0

2021-03-18 15:29:00,262 INFO o.a.j.r.Summariser: summary = 40741 in 00:12:37 = 53.8/s Avg: 380 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:29:35,024 INFO o.a.j.r.Summariser: summary + 2600 in 00:00:35 = 74.8/s Avg: 95 Min: 15 Max: 547 Err: 0 (0.00%) Active: 550 Started: 544 Finished: 0

2021-03-18 15:29:35,024 INFO o.a.j.r.Summariser: summary = 43341 in 00:13:12 = 54.7/s Avg: 362 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:30:00,212 INFO o.a.j.r.Summariser: summary + 2300 in 00:00:25 = 91.3/s Avg: 83 Min: 15 Max: 528 Err: 0 (0.00%) Active: 550 Started: 544 Finished: 0

2021-03-18 15:30:00,212 INFO o.a.j.r.Summariser: summary = 45641 in 00:13:37 = 55.9/s Avg: 348 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:30:35,019 INFO o.a.j.r.Summariser: summary + 3200 in 00:00:35 = 91.9/s Avg: 90 Min: 15 Max: 1053 Err: 0 (0.00%) Active: 550 Started: 544 Finished: 0

2021-03-18 15:30:35,019 INFO o.a.j.r.Summariser: summary = 48841 in 00:14:12 = 57.3/s Avg: 331 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:31:00,241 INFO o.a.j.r.Summariser: summary + 2200 in 00:00:25 = 87.2/s Avg: 75 Min: 16 Max: 497 Err: 0 (0.00%) Active: 517 Started: 544 Finished: 33

2021-03-18 15:31:00,241 INFO o.a.j.r.Summariser: summary = 51041 in 00:14:37 = 58.2/s Avg: 320 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:31:30,048 INFO o.a.j.r.Summariser: summary + 3000 in 00:00:30 = 100.6/s Avg: 68 Min: 15 Max: 777 Err: 0 (0.00%) Active: 484 Started: 544 Finished: 66

2021-03-18 15:31:30,048 INFO o.a.j.r.Summariser: summary = 54041 in 00:15:07 = 59.6/s Avg: 306 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:33:00,319 INFO o.a.j.r.Summariser: summary + 5900 in 00:01:30 = 65.4/s Avg: 85 Min: 12 Max: 1433 Err: 0 (0.00%) Active: 385 Started: 544 Finished: 165

2021-03-18 15:33:00,319 INFO o.a.j.r.Summariser: summary = 59941 in 00:16:37 = 60.1/s Avg: 285 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:33:34,931 INFO o.a.j.r.Summariser: summary + 2100 in 00:00:35 = 60.7/s Avg: 136 Min: 14 Max: 1415 Err: 0 (0.00%) Active: 352 Started: 544 Finished: 198

2021-03-18 15:33:34,931 INFO o.a.j.r.Summariser: summary = 62041 in 00:17:12 = 60.1/s Avg: 279 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:35:05,069 INFO o.a.j.r.Summariser: summary + 4400 in 00:01:30 = 48.8/s Avg: 58 Min: 15 Max: 1019 Err: 0 (0.00%) Active: 253 Started: 544 Finished: 297

2021-03-18 15:35:05,069 INFO o.a.j.r.Summariser: summary = 66441 in 00:18:42 = 59.2/s Avg: 265 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:35:33,948 INFO o.a.j.r.Summariser: summary + 1100 in 00:00:29 = 38.1/s Avg: 34 Min: 15 Max: 475 Err: 0 (0.00%) Active: 220 Started: 544 Finished: 330

2021-03-18 15:35:33,948 INFO o.a.j.r.Summariser: summary = 67541 in 00:19:11 = 58.7/s Avg: 261 Min: 5 Max: 11093 Err: 0 (0.00%)

2021-03-18 15:38:41,213 INFO o.a.j.JMeter: Finished remote host: 10.34.167.8 (1616074721213)

מספר התשובות שמתקבלות מהשרת שוות למספר הבקשות.

אין דיווח על שגיאות (מספר בקשות שווה מול מספר התשובות).

**HTML Report**

אין דיווח על שגיאות (מספר בקשות שווה מול מספר התשובות).



ניתוח תוצאות ניתן לראות בקובץ המצורפים



**סיכום כלל התוצאות:**

ניתן לראות שמספר הבקשות זהה למספר התשובות.

שרתי בסיס נתונים עומדים בעומס, שאר השרתים גם עומדים בעומס .

זמני התגובה מהירים/תקינים.

# מסקנות ופעולות להמשך

## מסקנות 18/03/2021

מסקנות לאחר סיכום הרצת שתי הבדיקות:

ניתן לומר שהמערכת תעמוד בעומס של עד לערך 400 פניות במקביל מסוג UploadScannedTags.

נקודת הכשל של המערכת (נקודה שבה המערכת תהיה לא יציבה) תהיה במצב של לערך 400 עד 440 פניות במקביל מסוג UploadScannedTags.

היות והצפי הוא ל- 200 רכיבים/משתמשים הפונים למערכת בו זמנית ניתן לשער שהמערכת תדע להתנהל בצורה יציבה וללא כשל כאשר מדובר בפניה מסוג UploadScannedTags.

**הערה**:

במערכת הבודקת התגלו Errors בזמן ההעמסה, אולם בלוגים של האפליקציה לא הופיעו Errors. הנושא הובא לידיעת מפתחי המערכת.

בשלב ההרצה עדיין לא הופיעו Errors . המסקנות שהתקבלו הנם על בסיס המערכת הבודקת ולא נכללה התייחסות ללוגים של האפליקציה.

# נספחים

## נספח א' – תצורת אפליקציה

בנוסף לברירת המחדל של האפליקציה בשעת ההתקנה נוספו פלאגינים (תוספים) לשם הרחבת יכולות האפליקציה, להלן רשימת הפלאגינים (תוספים) שנוספו:

* Servers Performance Monitoring

במהלך בדיקת העומסים, חשוב לדעת את תקינות השרתים שנטענו. באמצעות תוסף זה ניתן לבדוק את התנהגות משאבי השרתים הנטענים בזמן הריצה. ניתן לנטר על CPU, Memory, Swap, Disks I/O Networks I/O .

* Console Status Logger

תוסף מאזין המדפיס יומן סיכום קצר למסוף בזמן JMeter שפועל במצב שאינו GUI. התוסף גם כותב את אותו מידע ל- jmeter.log במצב GUI. המידע הנו ממוצע של זמן התגובה וההשהיה.

* Stepping Thread Group

תוסף זה מאפשר הגדלת יכולת הקונפיגורציה של המשתמשים (Thread) בתהליך ההעמסה.

להלן האפשרויות:

1. מאפשר תצוגה מקדימה של גרף המציג עומס משוער (ראה דוגמה למסך למטה)
2. מאפשר השהייה של קבוצת העמסה ראשונית לשם שילוב של פעילויות העמסה קבוצתיות
3. מאפשר הגדלת העומס על ידי ביזור המשתמשים לחלקים נפרדים בזמן הריצה
4. מאפשר הקפאה של אותה פעילות למשך זמן קצוב
5. הפחת של עומס לפי מנות

כדי למדוד התנהגות משאבים של שרתי המערכת הנבדקת יורץ יישום Agent על השרת עצמו בזמן הרצת הבדיקה. יישום זה משדר בפורט 4444 אל המערכת הבודקת האוספת ומעבדת את הנתונים הנשלחים.

## נספח ב' – מילון מונחים

להלן הסבר על המושגים המופיעים בדו"ח סיכום תוצאת הריצה (Summary Report)

### Summary Report

Label

שם / כתובת ה- URL לבקשת ה- HTTP הספציפית.

**Label:** It is the name/URL for the specific HTTP(s) Request. If you have selected “Include group name in label?” option then the name of the Thread Group is applied as the prefix to each label.

Samples

מציין את מספר המשתמשים הווירטואליים לכל בקשה.

**#Samples:** This indicates the number of virtual users per request.

Average

הזמן הממוצע שלוקחים כל הדגימות לביצוע תווית ספציפית.

**Average:** It is the average time taken by all the samples to execute specific label..

Min

הזמן הקצר ביותר שלוקח מדגם לתווית ספציפית.

**Min:** The shortest time taken by a sample for specific label

Max

הזמן הארוך ביותר שלוקח מדגם לתווית ספציפית.

**Max:** The longest time taken by a sample for specific label.

Std.Dev

מראה את מכלול המקרים החריגים שסטו מהערך הממוצע של זמן התגובה, לדוגמא: ככל שערך זה פחות עקבי יותר. סטיית התקן צריכה להיות פחות או שווה למחצית מהזמן הממוצע לתווית

**Std. Dev.:** This shows the set of exceptional cases which were deviating from the average value of sample response time. The lesser this value more consistent the data. Standard deviation should be less than or equal to half of the average time for a label.

Error%

אחוז הבקשות שנכשלו לכל תווית.

**Error%:** Percentage of Failed requests per Label.

Throughput

התפוקה היא מספר הבקשות שעובדות על ידי השרת ליחידת זמן (שניות, דקות, שעות). זמן זה מחושב מתחילת המדגם הראשון ועד סוף המדגם האחרון. תפוקה גדולה יותר עדיפה.

**Throughput:** Throughput is the number of request that are processed per time unit(seconds, minutes, hours) by the server. This time is calculated from the start of first sample to the end of the last sample. Larger throughput is better.

KB/Sec

תפוקה הנמדדת בקילובייט לשנייה.

**KB/Sec:** This indicates the amount of data downloaded from server during the performance test execution. In short, it is the Throughput measured in Kilobytes per second.

Received KB/sec

כמות הנתונים שהורדן מהשרת במהלך החזרת תשובה ע"י השרת.

Sent KB/sec

כמות הנתונים שנשלחו כבקשה אל השרת.

Avg. Bytes

ממוצע כלל הנתונים.

מקור: <https://octoperf.com/blog/2017/10/19/how-to-analyze-jmeter-results/#installation>

### Aggregate Report

ע"י קיבוץ כלל הנתונים ב- Summary Report ניתן לראות פרמטרים נוספים

קו 90%:

90% מהדגימות לקחו לא יותר מהפעם. הדגימות שנותרו ארכו לפחות זמן זה. (אחוזון 90)

כדי לחשב 90%, יש לרשום את כל ערכי העסקה ולסדר מחדש לפי ערכיהם בסדר יורד. לטחר מכן אין לכלול עסקאות 10% מובילות ברשימה הכוללת.

הערך הגבוה ביותר שנותר הוא האחוזון ה -90.

***95% Line:*** 95% of the samples took no more than this time. The remaining samples took at least as long as this. (95th percentile)

To calculate 95%, list down all the transaction values and re-order by their values in descending order. Now exclude top 5% transactions of your total list.

The highest value left is the 95th percentile.

קו 95%:

95% מהדגימות לקחו לא יותר מהפעם. הדגימות שנותרו ארכו לפחות זמן זה. (אחוזון 95)

כדי לחשב 95%, רשום את כל ערכי העסקה וסדר מחדש לפי ערכיהם בסדר יורד. כעת אל תכלול את העסקאות המובילות של 5% מכלל הרשימה שלך.

הערך הגבוה ביותר שנותר הוא האחוזון ה -95.

95% Line: 95% of the samples took no more than this time. The remaining samples took at least as long as this. (95th percentile)

To calculate 95%, list down all the transaction values and re-order by their values in descending order. Now exclude top 5% transactions of your total list.

The highest value left is the 95th percentile.

קו 99%:

99% מהדגימות לקחו לא יותר מהפעם. הדגימות שנותרו ארכו לפחות זמן זה. (אחוזון 99)

כדי לחשב 99%, רשום את כל ערכי העסקה וסדר מחדש לפי ערכיהם בסדר יורד. כעת אל תכלול עסקאות מובילות של 1% מכלל הרשימה שלך.

הערך הגבוה ביותר שנותר הוא אחוזון 99.

***99% Line:*** 99% of the samples took no more than this time. The remaining samples took at least as long as this. (99th percentile)

To calculate 99%, list down all the transaction values and re-order by their values in descending order. Now exclude top 1% transactions of your total list.

The highest value left is the 99th percentile.

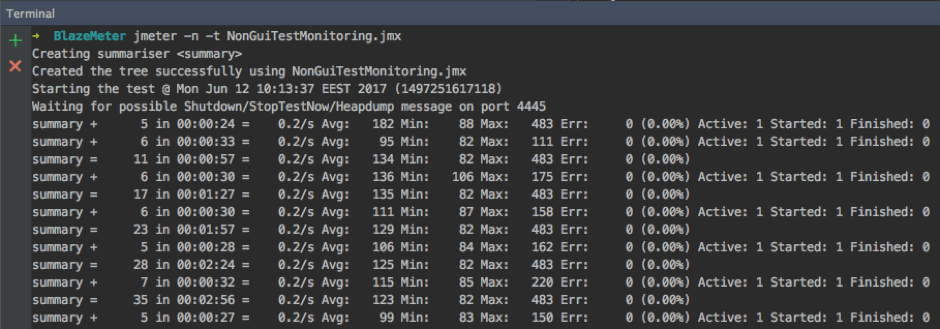
חציון

חציון: הזמן באמצע קבוצה של תוצאת דוגמאות. מצביע על כך ש- 50% מהדגימות לקחו לא יותר מהפעם כלומר השאר לקח זמן ממושך לפחות.

**Median:** It is the time in the middle of a set of samples result. It indicates that 50% of the samples took no more than this time i.e the remainder took at least as long.

### non-gui-test

ניתוח דוגמא שבצילום מסך



* **“5 in 00:00:24 = 0.2/s”** means that in 24 seconds we have sent 5 requests to the server with an average throughput of 0.2 requests per second
* **“Avg: 182”** means that the average response time at that moment is 182 milliseconds
* **“Min: 88”** means that the minimum response time from send requests for that period was 88 milliseconds
* **“Max: 483”** means that the maximum response time from send requests for that period was 483 milliseconds
* **“Err: 0 (0.00%)”** means that we didn’t have any errors in requests for that period and the percentage of errors from total requests is 0 accordingly
* **Active:** shows the number of active users who were performing requests for this period
* **Started:** shows the total number of started threads since the beginning of the tests
* **Finished:** shows the total number of threads that already finished execution since the beginning of tests