**Author: Trung Lê Chí**

**Contact number: 0902712727**

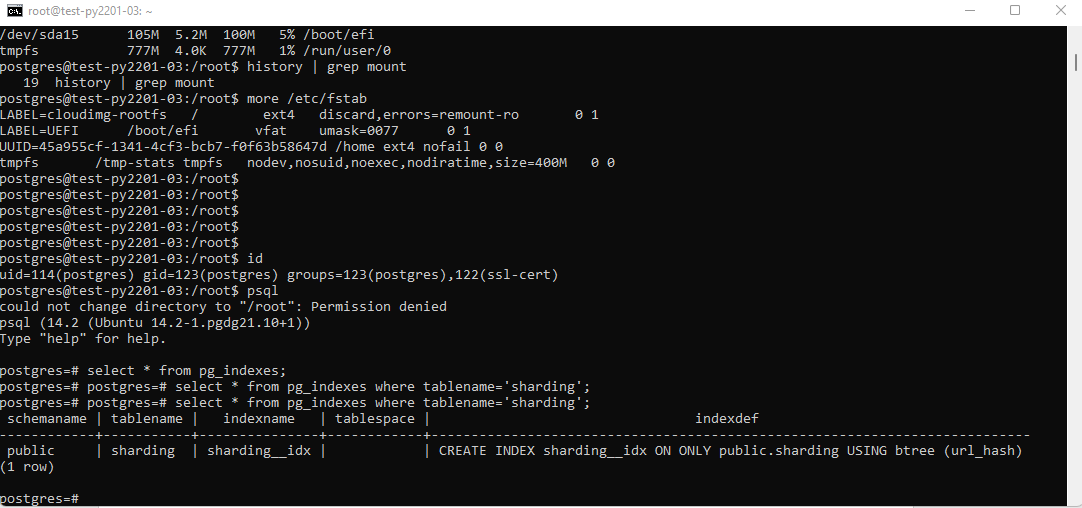
**Email:** [**lectrung@gmail.com**](mailto:lectrung@gmail.com)

**Test date: 15th Mar 2022**

+Step 1: Examine the database

select \* from pg\_indexes where tablename='sharding';

The table sharding already has a index for the column hash\_url



The table sharding is also partioned. Maybe, this table should be stored in separated tablespace such as “SHARDING” for separate IO on the disk storage.

<https://github.com/lectrung/python_test/>

Please read the readme.txt and follows instructions

+ Step 2: Write scipt and collect data

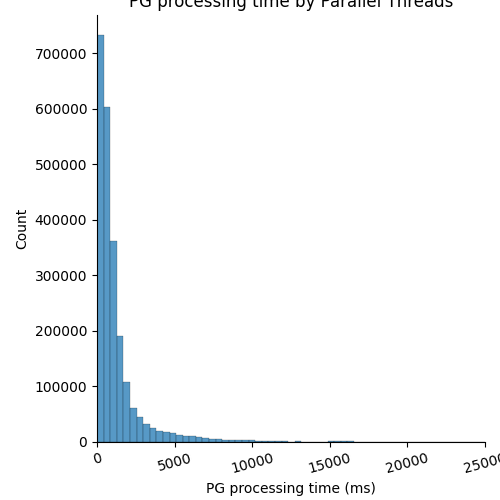
My script will simulate N parallel threads. The N number is set in the file input/config.txt

(500,400001,80,80,70,'/trunglc/data/summary.csv','/trunglc/data/os.csv','/trunglc/data/time.csv')

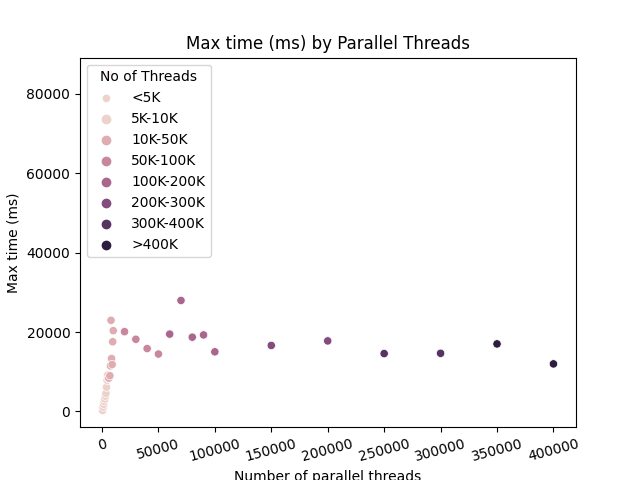
The script will generated 500 – 400K parallel threads and store the result in 03 files summary.csv, os.csv and time.csv

|  |  |  |
| --- | --- | --- |
| **File** | **Columns** | **Note** |
| summary.csv | threads,  errors,  overloads,  max,  min,  avg,  cpu\_count,  cpu\_percent,  memory\_percent,  disk\_io\_read,  disk\_io\_write | Store the summary information of the test |
| os.csv | threads,  cpu\_count,  cpu\_percent,  memory\_percent,  disk\_io\_read,  disk\_io\_write | Store the server benchmark metrics |
| time.csv | threads,time | Store all processing time of thread |

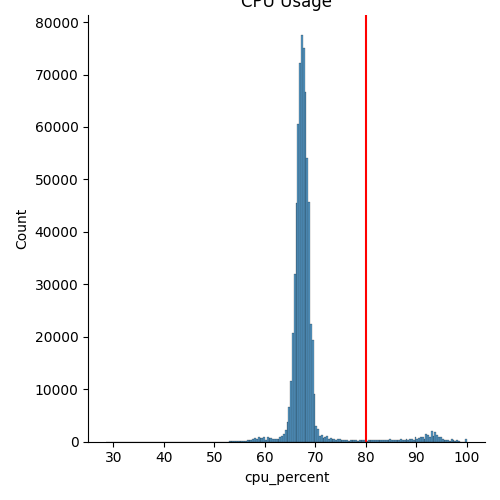
+ Step 3 GRAPH RESULT:



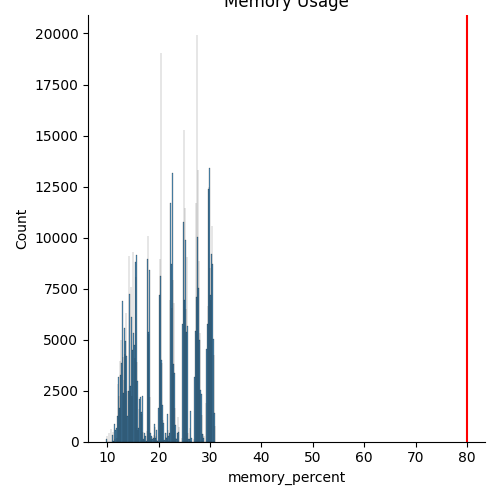
The histogram graph has a right skew. So that most of requests are done less than 4 seconds. So the PG database performance may be optimized well.



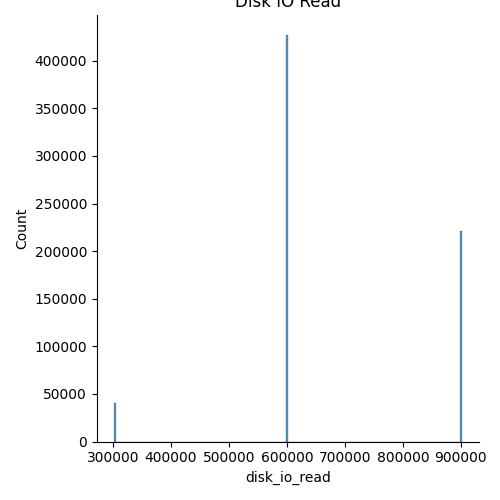
However, when I increase the number of parallel simulation threads the maximum wait time will increased. The PG database has 300 millions rows, each row uses about 64Mb. So I suggest it should be stored in memory (using Redis in this case may be a reasonable choice).



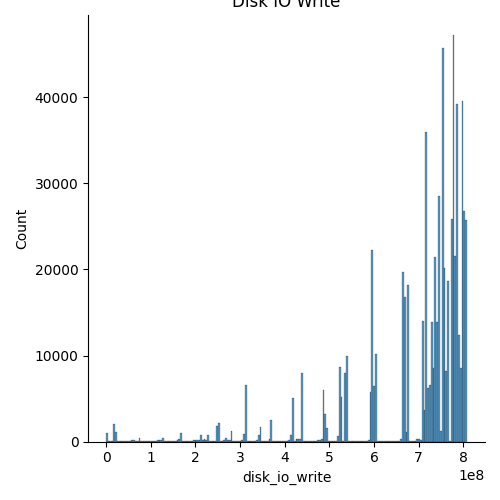
During the test, the CPU usage percent is focus in range 62% to 75%. The red vertical line is the alarm threshold. The CPU unit is 2 so that we need to deploy on a stronger server to avoid overload.



The memory usage is almost focus in range [10-20%]. Maybe the test use less data than realistic scenarios. So I think the test shoud simulate to store some objects to memory.



During the test, the read IO is switched between some disret discrete values.



But the write process is increased rapidly. Maybe we need to optimize the OS (turn off necessary port, services, programs, …)

With 400K parallel threads =>

+ Write process = 800M

+ Max RAM: 30%

+ Max CPU: 90%

+ Max time: 20 seconds

Base on the maximum CPU and memory threshold, the server is can serve for 400K requests at a time.

**I think we can start with some number of threads and then we calculate the server’s health. Try to collect many information as we can and store all in to csv file format.**

**After that, visualize the data and explore some ideas about the trends of the server limitation. Machine learning is a good solution for predict the maximum threshold instead of trying a very big number of threshold and … crash the customer’s server. We have to spend more time and must restore the data if there is a lost or currupted. Linear Regession is a reasonale approach.**