

Laporan Stress Test Server-Client dengan Multithreading dan Multiprocessing Pool

1. Pendahuluan

Laporan ini membahas pengujian stress test sistem server-client dengan dua pendekatan concurrency: Multithreading Pool dan Multiprocessing Pool. Sistem terdiri dari dua mesin: Mesin 1 sebagai server dan Mesin 2 sebagai client. Server melayani permintaan upload, download, dan list file secara bersamaan dari berbagai klien.

2. Arsitektur Sistem

Server dijalankan pada IP 172.18.0.2 dan mendukung dua model concurrency:

- Multithreading Pool: menggunakan ThreadPoolExecutor
- Multiprocessing Pool: menggunakan ProcessPoolExecutor

Client dijalankan pada IP 172.18.0.3 dan melakukan stress test terhadap server menggunakan script terpisah untuk multithreading dan multiprocessing.

3. Arsitektur Stress Test

Stress test dilakukan dengan kombinasi:

- Operasi: Upload dan Download
- Ukuran file: 10MB, 50MB, 100MB
- Jumlah client worker pool: 1, 5, 50
- Jumlah server worker pool: 1, 5, 50

Setiap kombinasi menghasilkan metrik waktu, throughput, serta jumlah worker yang sukses dan gagal.

4. Langkah Menjalankan Server dan Client

Di Mesin 1 (Server):

1. Jalankan server multithreading:

```
python3 serverETS_multithreading_pool.py --workers <jumlah_worker>
```

2. Jalankan server multiprocessing:

```
python3 serverETS_multiprocessing_pool.py --workers <jumlah_worker>
```

Di Mesin 2 (Client):

1. Siapkan file test di direktori ./test_files dengan ukuran 10MB, 50MB, dan 100MB.
2. Jalankan client stress test untuk multithreading:
`python3 multithreading_stress_test.py --server-workers <jumlah_worker>`
3. Jalankan client stress test untuk multiprocessing:
`python3 multiprocessing_stress_test.py --server-workers <jumlah_worker>`

5. Langkah Menjalankan Server dan Client

Di Mesin 1 Multi threading (Server):

```
(base) jovyan@c4e055f5bfc2:~/work$ python3 serverETS_multithreading_pool.py --worker  
s 1 --port 8889  
2025-05-25 13:34:06,608 - WARNING - Server initialized with 1 thread workers  
2025-05-25 13:34:06,608 - WARNING - Threading Pool Server running on ('0.0.0.0', 888  
9) with 1 workers  
2025-05-25 13:35:04,555 - WARNING - New connection from ('172.18.0.3', 59928)  
2025-05-25 13:35:04,556 - WARNING - Worker 140398532331072 handling connection from  
( '172.18.0.3', 59928)  
2025-05-25 13:39:44,168 - WARNING - New connection from ('172.18.0.3', 55656)  
2025-05-25 13:40:49,699 - WARNING - Connection from ('172.18.0.3', 59928) closed  
2025-05-25 13:40:49,700 - WARNING - Worker 140398532331072 handling connection from  
( '172.18.0.3', 55656)
```

Pada server disini saya mencoba untuk menggunakan 1 worker, dapat dilihat jika menggunakan hanya 1 worker, request yang diterima hanya 1 thread yang aktif menangani koneksi. Saat client kedua mencoba terhubung (New connection from), tidak langsung dilayani karena worker masih sibuk. Sehingga membuat connection closed

Di Mesin 2 (Client):

```

(base) jovyan@9ba7b23935cc:~/work$ python3 run_stress_tests.py --server-ip 172.18.0.2
Verifying test files...
✓ ./test_files/10mb.txt: File OK: 9.96MB
✓ ./test_files/50mb.txt: File OK: 50.00MB
✓ ./test_files/100mb.txt: File OK: 100.00MB

All test files are present and have the correct size.

=====
Running multithreading stress test with 1 server workers
=====

Executing: python multithreading_stress_test.py --server-ip 172.18.0.2 --server-workers 1
File Server Multithreading Stress Test
=====
Target Server: 172.18.0.2:8889
Server Workers: 1
Test Files: 10mb.txt, 50mb.txt, 100mb.txt
Test Combinations: 2 operations x 3 volumes x 3 client workers = 18 tests

Starting Multithreading Stress Test (Server Workers: 1)
=====

Test 1: download, 10MB, 1 clients, 1 server workers
-----
Pre-uploading files for download test using ./test_files/10mb.txt...
Reading file ./test_files/10mb.txt...
File size (base64): 13.28 MB
Uploading 1 files for download test...
Uploading file 1/1... Command failed, retrying in 2.8s (1/3)...

```

Salah satu contoh pada awal awal, untuk client dengan 1 worker memberikan 1 file 0mb ke server dengan 1 worker.

Saat pre-upload untuk download test, client mencoba mengirim file 10MB ke server. Kemudian File dibaca, diencode ke base64 (jadi 13.28MB), dan dikirim via socket.

Namun pada gambar terlihat bahwa command failed, hal ini terjadi karena:

Server multithreading_pool.py menggunakan 1 thread.

Ketika satu koneksi sedang diproses (misalnya sedang baca file dari socket), thread itu tidak bisa menerima koneksi lain.

Maka, klien yang mencoba menghubungi server saat itu akan ditolak atau timeout, meskipun jumlahnya hanya satu.

Client punya timeout socket (default: 60s atau kurang).

Jika server terlambat merespons (karena sibuk dengan pekerjaan sebelumnya), client akan anggap itu gagal, dan lakukan retry.

Sehingga kita mendapatkan kesimpulan:

Kombinasi	Performa Umum	Risiko / Masalah Potensial
1 Server, 1 Client	Cukup, tapi rawan timeout jika file besar	Blocking jika koneksi lambat
1 Server, 5 Client	Bottleneck, antrean panjang; sebagian besar client timeout/gagal	Server hanya bisa handle 1 permintaan dalam satu waktu
1 Server, 50 Client	Hampir semua gagal atau timeout	Tidak layak digunakan; overload
5 Server, 1 Client	Optimal, idle workers banyak	Resource underutilized
5 Server, 5 Client	Sangat baik, concurrency terjaga	Ideal kombinasi untuk workload ringan-menengah
5 Server, 50 Client	Cukup baik, antrean terjadi; tapi sebagian besar tetap berhasil	Performa mulai turun pada file besar
50 Server, 1 Client	Overkill, semua idle kecuali satu	Tidak efisien, boros sumber daya
50 Server, 5 Client	Sangat cepat dan responsif	Over-provision, tapi sangat stabil
50 Server, 50 Client	Kombinasi optimal untuk skala besar; throughput tinggi, delay kecil	Butuh resource besar (CPU, memory) untuk multiprocessing

6. Hasil Pengujian

Berikut adalah hasil kombinasi pengujian yang dilakukan:

n	operation	volume_mb	client_workers	server_workers	total_time	throughput_per_client	client_success	client_failed	concurrency_type
---	-----------	-----------	----------------	----------------	------------	-----------------------	----------------	---------------	------------------

1	uplo ad	10	1	1	3.73	2808326	1	0	multithre ading
2	uplo ad	10	5	1	22.1 9	1890196	4	1	multithre ading
3	uplo ad	10	50	1	247. 06	2037203	48	2	multithre ading
4	uplo ad	50	1	1	18.2 3	2876102	1	0	multithre ading
5	uplo ad	50	5	1	94.3 2	2223482	4	1	multithre ading
6	uplo ad	50	50	1	1303 .06	1931284	48	2	multithre ading
7	uplo ad	100	1	1	36.3 8	2881893	1	0	multithre ading
8	uplo ad	100	5	1	199. 43	2103122	4	1	multithre ading
9	uplo ad	100	50	1	2543 .77	1978626	48	2	multithre ading
1 0	dow nloa d	10	1	1	2.68	3907709	1	0	multithre ading
1 1	dow nloa d	10	5	1	15.5 9	2689680	4	1	multithre ading
1 2	dow nloa d	10	50	1	216. 48	2325009	48	2	multithre ading
1 3	dow nloa d	50	1	1	14.5 8	3596855	1	0	multithre ading
1 4	dow nloa d	50	5	1	76.6 7	2735189	4	1	multithre ading
1 5	dow nloa d	50	50	1	952. 15	2643057	48	2	multithre ading
1 6	dow nloa d	100	1	1	27.5 6	3804226	1	0	multithre ading
1 7	dow nloa d	100	5	1	151. 56	2767430	4	1	multithre ading
1 8	dow nloa d	100	50	1	2217 .18	2270074	48	2	multithre ading
1 9	uplo ad	10	1	5	1.91	5488517	1	0	multithre ading
2 0	uplo ad	10	5	5	10.1 1	4146628	4	1	multithre ading

21	upload	10	50	5	134.67	3737300	48	2	multithreading
22	upload	50	1	5	8.2	6396374	1	0	multithreading
23	upload	50	5	5	56.77	3694241	4	1	multithreading
24	upload	50	50	5	684.93	3674198	48	2	multithreading
25	upload	100	1	5	17.61	5954610	1	0	multithreading
26	upload	100	5	5	98.49	4258601	4	1	multithreading
27	upload	100	50	5	1298.05	3877483	48	2	multithreading
28	download	10	1	5	1.43	7357951	1	0	multithreading
29	download	10	5	5	7.58	5535097	4	1	multithreading
30	download	10	50	5	103.42	4866635	48	2	multithreading
31	download	50	1	5	6.8	7707470	1	0	multithreading
32	download	50	5	5	38.55	5440755	4	1	multithreading
33	download	50	50	5	482.4	5216824	48	2	multithreading
34	download	100	1	5	15.06	6964906	1	0	multithreading
35	download	100	5	5	76.66	5471586	4	1	multithreading
36	download	100	50	5	958.58	5250649	48	2	multithreading
37	upload	10	1	50	1.4	7477658	1	0	multithreading
38	upload	10	5	50	8.77	4782436	4	1	multithreading
39	upload	10	50	50	108.73	4628986	48	2	multithreading
40	upload	50	1	50	7.33	7152246	1	0	multithreading

41	upload	50	5	50	41.98	4995424	4	1	multithreading
42	upload	50	50	50	577.4	4358471	48	2	multithreading
43	upload	100	1	50	15.27	6868308	1	0	multithreading
44	upload	100	5	50	87.28	4805618	4	1	multithreading
45	upload	100	50	50	1105.83	4551490	48	2	multithreading
46	download	10	1	50	1.26	8351224	1	0	multithreading
47	download	10	5	50	7.09	5916508	4	1	multithreading
48	download	10	50	50	88.19	5707194	48	2	multithreading
49	download	50	1	50	6.73	7791613	1	0	multithreading
50	download	50	5	50	33.23	6310932	4	1	multithreading
51	download	50	50	50	467.81	5379512	48	2	multithreading
52	download	100	1	50	11.8	8888381	1	0	multithreading
53	download	100	5	50	66.91	6268760	4	1	multithreading
54	download	100	50	50	861.09	5845130	48	2	multithreading
1	upload	10	1	1	5.09	2058193	1	0	multiprocessing
2	upload	10	5	1	28.64	1464433	4	1	multiprocessing
3	upload	10	50	1	394.1	1223913	46	4	multiprocessing
4	upload	50	1	1	27.3	2112167	1	0	multiprocessing
5	upload	50	5	1	165.22	1396235	4	1	multiprocessing
6	upload	50	50	1	1714.86	1547005	46	4	multiprocessing

7	uplo ad	100	1	1	48.2 1	2392734	1	0	multipro cessing
8	uplo ad	100	5	1	271. 42	1699842	4	1	multipro cessing
9	uplo ad	100	50	1	4009 .2	1323403	46	4	multipro cessing
1 0	dow nloa d	10	1	1	3.95	2651274	1	0	multipro cessing
1 1	dow nloa d	10	5	1	23.5 3	1782446	4	1	multipro cessing
1 2	dow nloa d	10	50	1	291. 87	1652626	46	4	multipro cessing
1 3	dow nloa d	50	1	1	18.9 7	3040310	1	0	multipro cessing
1 4	dow nloa d	50	5	1	122. 76	1879127	4	1	multipro cessing
1 5	dow nloa d	50	50	1	1540 .12	1722530	46	4	multipro cessing
1 6	dow nloa d	100	1	1	43.1 4	2673705	1	0	multipro cessing
1 7	dow nloa d	100	5	1	268. 73	1716879	4	1	multipro cessing
1 8	dow nloa d	100	50	1	2753 .6	1926857	46	4	multipro cessing
1 9	uplo ad	10	1	5	2.12	4943886	1	0	multipro cessing
2 0	uplo ad	10	5	5	13.9 8	2999693	4	1	multipro cessing
2 1	uplo ad	10	50	5	155. 45	3102865	46	4	multipro cessing
2 2	uplo ad	50	1	5	11.4 5	5036430	1	0	multipro cessing
2 3	uplo ad	50	5	5	57.1 3	4038061	4	1	multipro cessing
2 4	uplo ad	50	50	5	731. 99	3624226	46	4	multipro cessing
2 5	uplo ad	100	1	5	21.1 1	5465045	1	0	multipro cessing
2 6	uplo ad	100	5	5	129. 03	3575727	4	1	multipro cessing

27	upload	100	50	5	1655.77	3204429	46	4	multiprocessing
28	download	10	1	5	1.72	6107775	1	0	multiprocessing
29	download	10	5	5	11.12	3770660	4	1	multiprocessing
30	download	10	50	5	128.83	3744014	46	4	multiprocessing
31	download	50	1	5	9.44	6110062	1	0	multiprocessing
32	download	50	5	5	56.24	4102097	4	1	multiprocessing
33	download	50	50	5	790.84	3354525	46	4	multiprocessing
34	download	100	1	5	17.41	6626805	1	0	multiprocessing
35	download	100	5	5	124.17	3715710	4	1	multiprocessing
36	download	100	50	5	1425.2	3722832	46	4	multiprocessing
37	upload	10	1	50	1.78	5905371	1	0	multiprocessing
38	upload	10	5	50	10.36	4047356	4	1	multiprocessing
39	upload	10	50	50	126.94	3799796	46	4	multiprocessing
40	upload	50	1	50	7.57	7617439	1	0	multiprocessing
41	upload	50	5	50	46.14	4999304	4	1	multiprocessing
42	upload	50	50	50	609.59	4351966	46	4	multiprocessing
43	upload	100	1	50	19.77	5833601	1	0	multiprocessing
44	upload	100	5	50	106.63	4326978	4	1	multiprocessing
45	upload	100	50	50	1125.43	4714478	46	4	multiprocessing
46	download	10	1	50	1.59	6585933	1	0	multiprocessing

	d								
47	dow nload	10	5	50	8.05	5207507	4	1	multipro cessing
48	dow nload	10	50	50	118. 33	4076361	46	4	multipro cessing
49	dow nload	50	1	50	7.07	8157148	1	0	multipro cessing
50	dow nload	50	5	50	49.7 4	4638278	4	1	multipro cessing
51	dow nload	50	50	50	496. 14	5347080	46	4	multipro cessing
52	dow nload	100	1	50	16.1 8	7128389	1	0	multipro cessing
53	dow nload	100	5	50	86.5 8	5328937	4	1	multipro cessing
54	dow nload	100	50	50	1020 .56	5198883	46	4	multipro cessing

7. Link Github

<https://github.com/Papavero30/Progjar-Activity/tree/b0fa6110b52474cddf01556f5d7cde883d49ecf4/ETS>