COMP8005: Assignment 2

epoll Client Server Implementation

Kevin Lo, A00952922

2-28-2021

Table of Contents

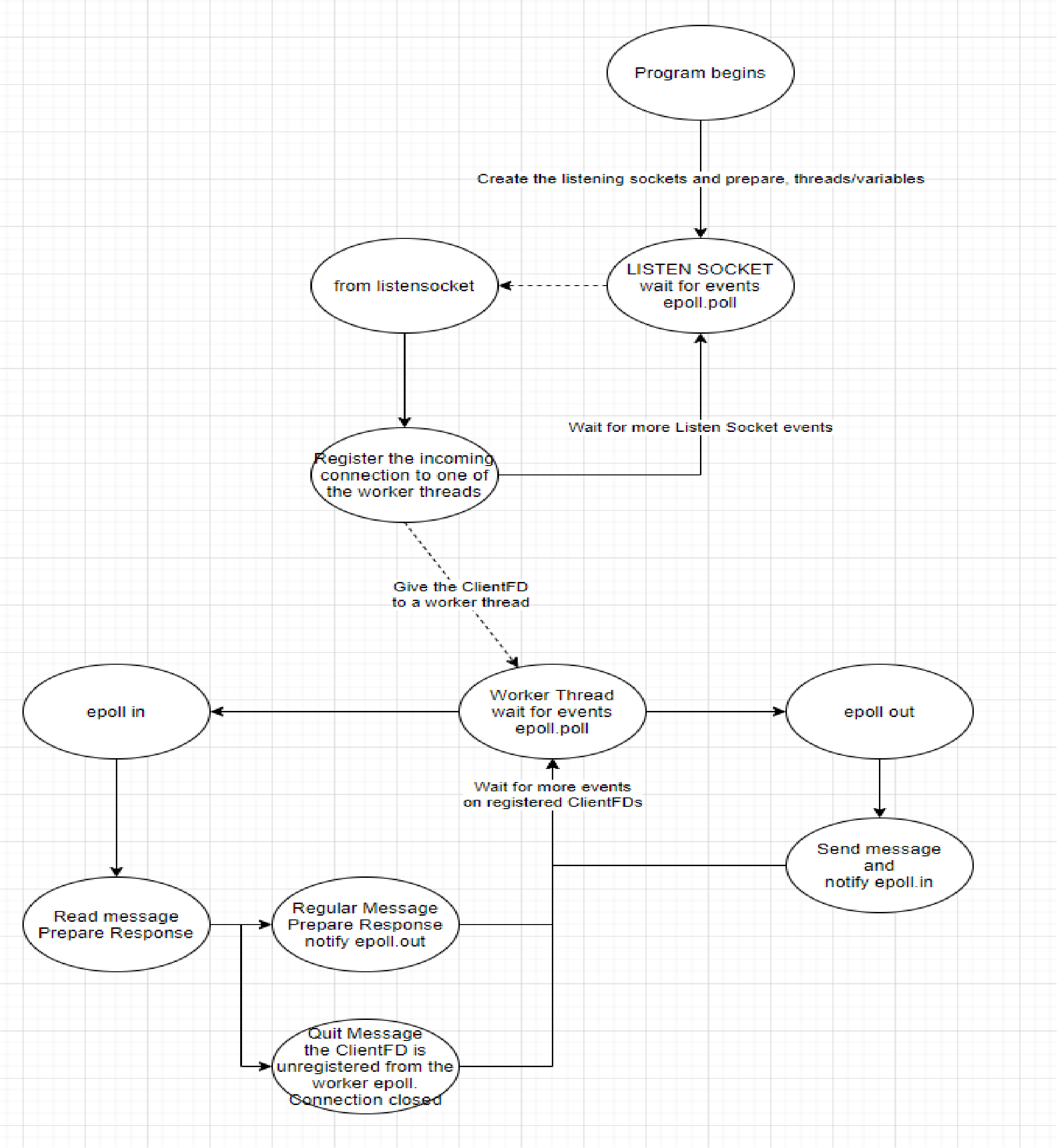
1 Introduction

Epoll servers are known for their performance over the traditional method of TCP servers using one thread per connection model. In today's standards the traditional method is no longer very effective due to the high-speed nature of the internet and large userbases in the millions. Epoll servers work by waiting for a notification that data has arrived on a socket and do work with the data from the socket.

The compiled result of the tests from the clients can be found in the ***RawData/FullCombined_clientstats.xlsx*** spreadsheet which is all the results from both clients and tests put together into one spreadsheet.

1.1 Design

The epoll server begins by initializing all necessary variables and worker threads. The main thread listens to the listening socket with epoll for new connections. When a new connection arrives, the new connection's is accepted, and the file descriptor of the connection is handed to one of the worker threads evenly. The worker thread is now able to monitor the connection and act accordingly to send, receive or close the connection.


Program begins
Create the listening sockets and prepare, threads/variables
LISTEN SOCKET wait for events epoll.poll
from listensocket
Register the incoming connection to one of the worker threads
Wait for more Listen Socket events
Give the ClientFD to a worker thread
Worker Thread wait for events epoll.poll
epoll in
epoll out
Wait for more events on registered ClientFDs
Send message and notify epoll.in
Read message Prepare Response
Regular Message Prepare Response notify epoll.out
Quit Message the ClientFD is unregistered from the worker epoll. Connection closed

2 How to Use

The following demonstrates how to use the two .py scripts.
Both scripts operate over the port number 8000

2.1 basic_client.py

The basic client takes three arguments, server IP address, number of threads, number of requests.

Example:
python basic_client.py 192.168.1.250 500 100

This command will create 500 threads to send 100 requests each thread to the server to simulate 500 connections to the server and requesting 100 times to the server.

2.2 epoll_svr.py

The epoll sever can just be run normally as

python epoll_svr.py

The server will continuously wait for new connections to pass to it's worker threads to register and serve the requests through the worker threads.

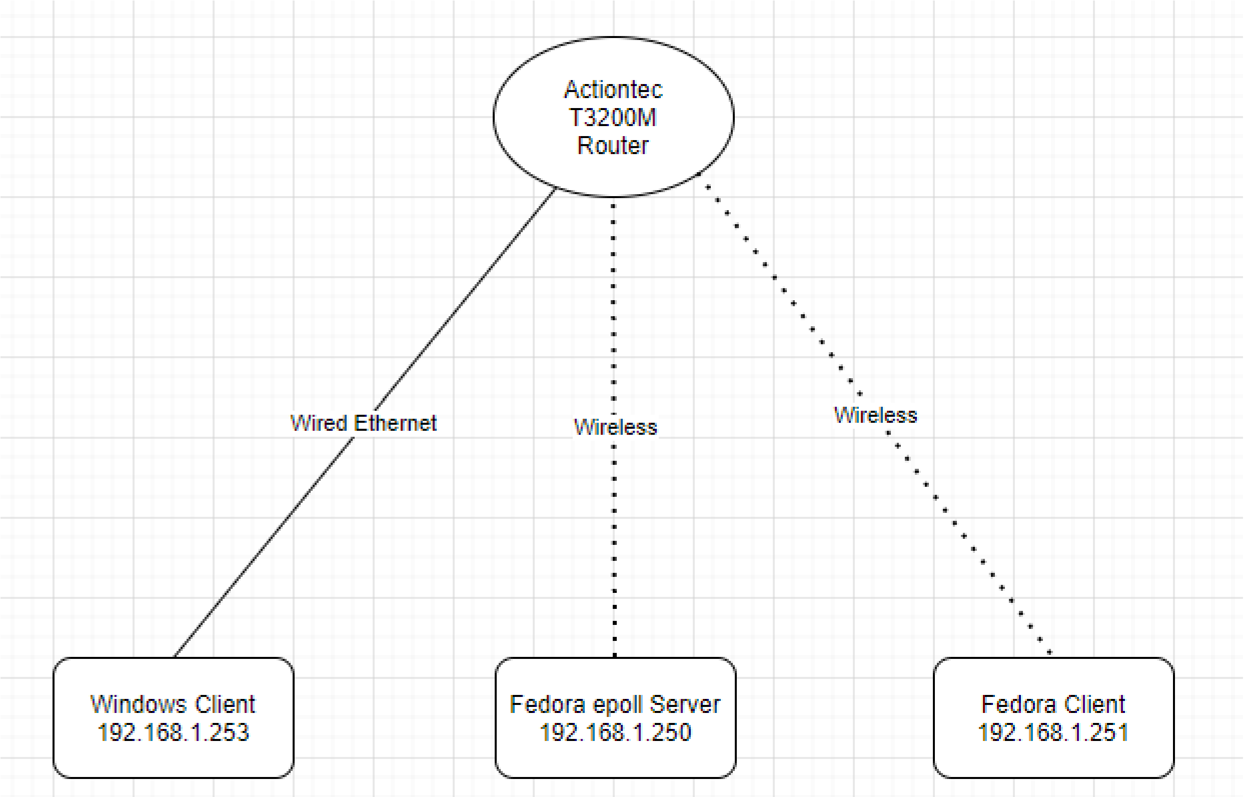
Note: Most servers will require you to use ***ulimit -n 100000*** or any big number, this is because by default, Linux only allows 1024 file descriptors at any one time. In this case, each client FD registered will take an additional file descriptor which will start throwing errors once you go beyond the file descriptor limit set by ***ulimit -n.***

3 Testbench

The following image is my testbench for this assignment.

The Windows client is directly connected to the router using an ethernet cable.
The Fedora client is connected to the router using WIFI
The Fedora epoll server is connected to the router using WIFI

Windows Client: 32GB Ryzen 3800x, Gigabit ethernet connection
Fedora Client: 16GB i5 4570k, Wireless N connection
Fedora Server: 12GB i7 6500U Asus UX303UB Laptop, Wireless N connectio

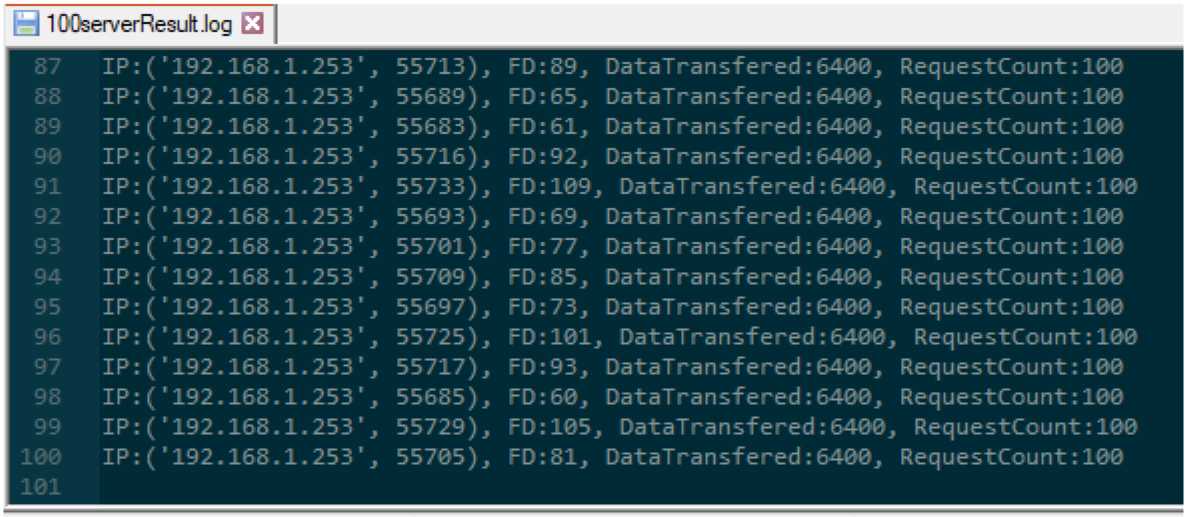


4 Tests

Test #	Test Description	Expected Result	Result	Pass/Fail
1	100 Client Connections	The clients/server will successfully run without any errors.	Client/Server successfully runs and generates logs	Pass. Detailed Resulted attached.
2	200 Client Connections	The clients/server will successfully run without any errors.	Client/Server successfully runs and generates logs	Pass. Detailed Resulted attached.
3	500 Client Connections	The clients/server will successfully run without any errors.	Client/Server successfully runs and generates logs	Pass. Detailed Resulted attached.
4	1000 Client Connections	The clients/server will successfully run without any errors.	Client/Server successfully runs and generates logs	Pass. Detailed Resulted attached.
5	1500 Client Connections	The clients/server will successfully run without any errors.	Client/Server successfully runs and generates logs	Pass. Detailed Resulted attached.
6	2000 Client Connections	The clients/server will successfully run without any errors.	Client/Server successfully runs and generates logs	Pass. Detailed Resulted attached.
7	2500 Client Connections	The clients/server will successfully run without any errors.	Client/Server successfully runs and generates logs	Pass. Detailed Resulted attached.
8	3000 Client Connections	The clients/server will successfully run without any errors.	Client/Server successfully runs and generates logs	Pass. Detailed Resulted attached.
9	4000 Client Connections	The clients/server will successfully run without any errors.	Client has timed out socket due to slow response, runs and generates logs	Fail. Detailed Resulted attached.

4.1 Test 1, 100 Connections

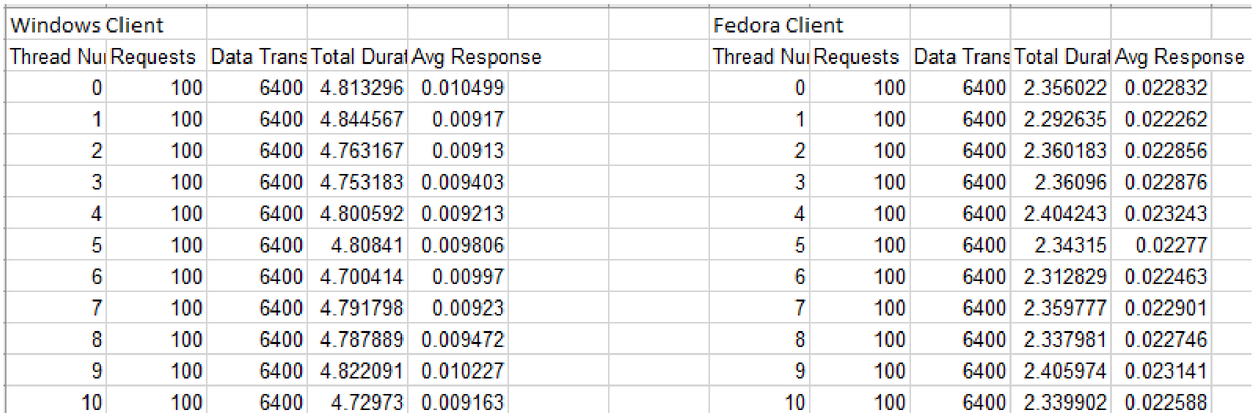
Server log output



100serverResult.log

```
87   IP:('192.168.1.253', 55713), FD:89, DataTransfered:6400, RequestCount:100
88   IP:('192.168.1.253', 55689), FD:65, DataTransfered:6400, RequestCount:100
89   IP:('192.168.1.253', 55683), FD:61, DataTransfered:6400, RequestCount:100
90   IP:('192.168.1.253', 55716), FD:92, DataTransfered:6400, RequestCount:100
91   IP:('192.168.1.253', 55733), FD:109, DataTransfered:6400, RequestCount:100
92   IP:('192.168.1.253', 55693), FD:69, DataTransfered:6400, RequestCount:100
93   IP:('192.168.1.253', 55701), FD:77, DataTransfered:6400, RequestCount:100
94   IP:('192.168.1.253', 55709), FD:85, DataTransfered:6400, RequestCount:100
95   IP:('192.168.1.253', 55697), FD:73, DataTransfered:6400, RequestCount:100
96   IP:('192.168.1.253', 55725), FD:101, DataTransfered:6400, RequestCount:100
97   IP:('192.168.1.253', 55717), FD:93, DataTransfered:6400, RequestCount:100
98   IP:('192.168.1.253', 55685), FD:60, DataTransfered:6400, RequestCount:100
99   IP:('192.168.1.253', 55729), FD:105, DataTransfered:6400, RequestCount:100
100  IP:('192.168.1.253', 55705), FD:81, DataTransfered:6400, RequestCount:100
101
```

Part of the results generated by the two clients.



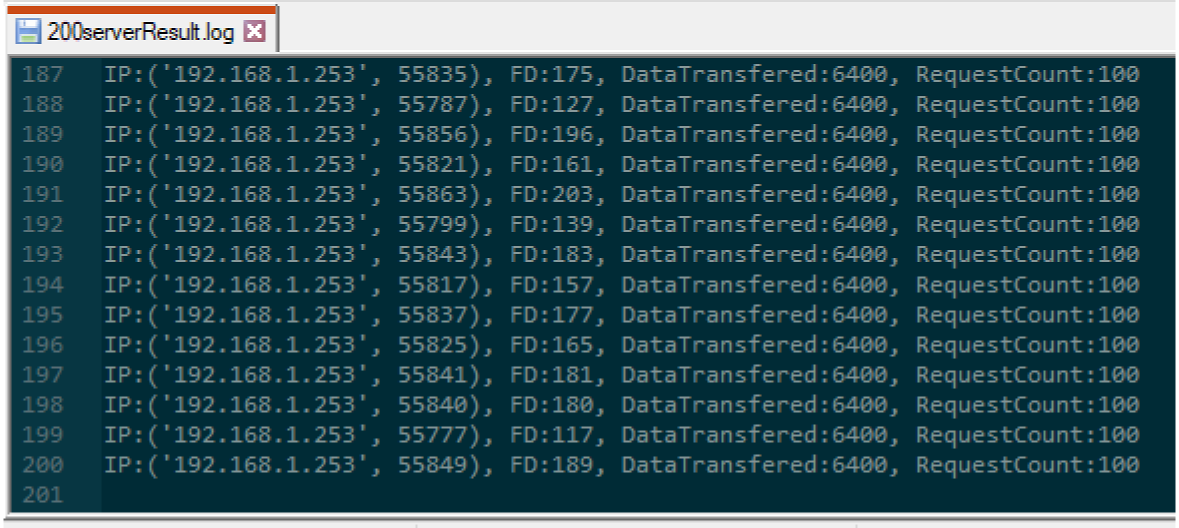
Windows Client						Fedora Client				
Thread Nu	Requests	Data Trans	Total Durat	Avg Response		Thread Nu	Requests	Data Trans	Total Durat	Avg Response
0	100	6400	4.813296	0.010499		0	100	6400	2.356022	0.022832
1	100	6400	4.844567	0.00917		1	100	6400	2.292635	0.022262
2	100	6400	4.763167	0.00913		2	100	6400	2.360183	0.022856
3	100	6400	4.753183	0.009403		3	100	6400	2.36096	0.022876
4	100	6400	4.800592	0.009213		4	100	6400	2.404243	0.023243
5	100	6400	4.80841	0.009806		5	100	6400	2.34315	0.02277
6	100	6400	4.700414	0.00997		6	100	6400	2.312829	0.022463
7	100	6400	4.791798	0.00923		7	100	6400	2.359777	0.022901
8	100	6400	4.787889	0.009472		8	100	6400	2.337981	0.022746
9	100	6400	4.822091	0.010227		9	100	6400	2.405974	0.023141
10	100	6400	4.72973	0.009163		10	100	6400	2.339902	0.022588

By looking at the server log, there are 100 entries in the file which matches the 100 connections. This is because the entry is logged only when the connection with the client is closed properly. If the client’s socket times out, the number of lines will not match the number of clients because the client will not close that connection properly.

The average response time for the connections is fast, roughly an average about 0.016 seconds. The Fedora client has a higher average response time per connection most likely because its operating on wifi as opposed to ethernet.

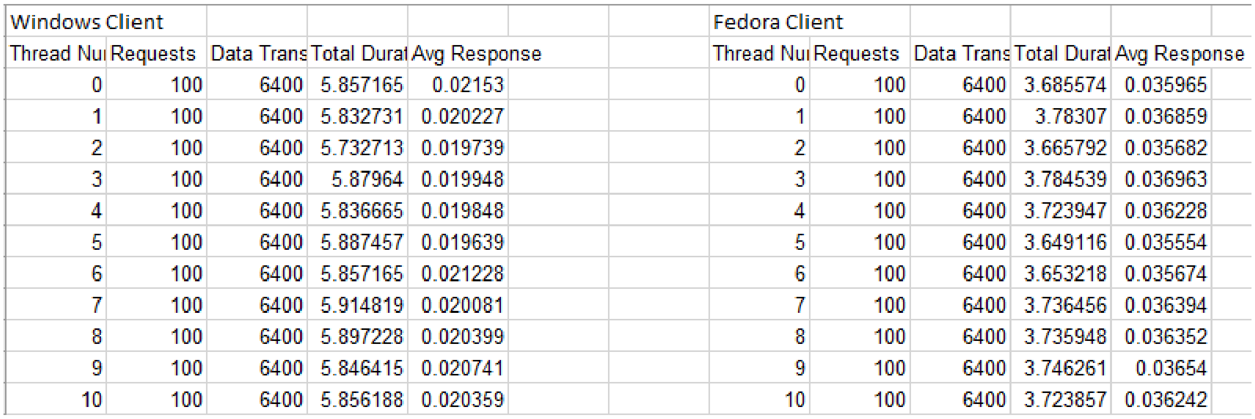
4.2 Test 2, 200 Connection

Server log output



```
187  IP:('192.168.1.253', 55835), FD:175, DataTransfered:6400, RequestCount:100
188  IP:('192.168.1.253', 55787), FD:127, DataTransfered:6400, RequestCount:100
189  IP:('192.168.1.253', 55856), FD:196, DataTransfered:6400, RequestCount:100
190  IP:('192.168.1.253', 55821), FD:161, DataTransfered:6400, RequestCount:100
191  IP:('192.168.1.253', 55863), FD:203, DataTransfered:6400, RequestCount:100
192  IP:('192.168.1.253', 55799), FD:139, DataTransfered:6400, RequestCount:100
193  IP:('192.168.1.253', 55843), FD:183, DataTransfered:6400, RequestCount:100
194  IP:('192.168.1.253', 55817), FD:157, DataTransfered:6400, RequestCount:100
195  IP:('192.168.1.253', 55837), FD:177, DataTransfered:6400, RequestCount:100
196  IP:('192.168.1.253', 55825), FD:165, DataTransfered:6400, RequestCount:100
197  IP:('192.168.1.253', 55841), FD:181, DataTransfered:6400, RequestCount:100
198  IP:('192.168.1.253', 55840), FD:180, DataTransfered:6400, RequestCount:100
199  IP:('192.168.1.253', 55777), FD:117, DataTransfered:6400, RequestCount:100
200  IP:('192.168.1.253', 55849), FD:189, DataTransfered:6400, RequestCount:100
201
```

Part of the results generated by the two clients.



Windows Client							Fedora Client					
Thread Nu	Requests	Data Trans	Total Durat	Avg Response			Thread Nu	Requests	Data Trans	Total Durat	Avg Response	
0	100	6400	5.857165	0.02153			0	100	6400	3.685574	0.035965	
1	100	6400	5.832731	0.020227			1	100	6400	3.78307	0.036859	
2	100	6400	5.732713	0.019739			2	100	6400	3.665792	0.035682	
3	100	6400	5.87964	0.019948			3	100	6400	3.784539	0.036963	
4	100	6400	5.836665	0.019848			4	100	6400	3.723947	0.036228	
5	100	6400	5.887457	0.019639			5	100	6400	3.649116	0.035554	
6	100	6400	5.857165	0.021228			6	100	6400	3.653218	0.035674	
7	100	6400	5.914819	0.020081			7	100	6400	3.736456	0.036394	
8	100	6400	5.897228	0.020399			8	100	6400	3.735948	0.036352	
9	100	6400	5.846415	0.020741			9	100	6400	3.746261	0.03654	
10	100	6400	5.856188	0.020359			10	100	6400	3.723857	0.036242	

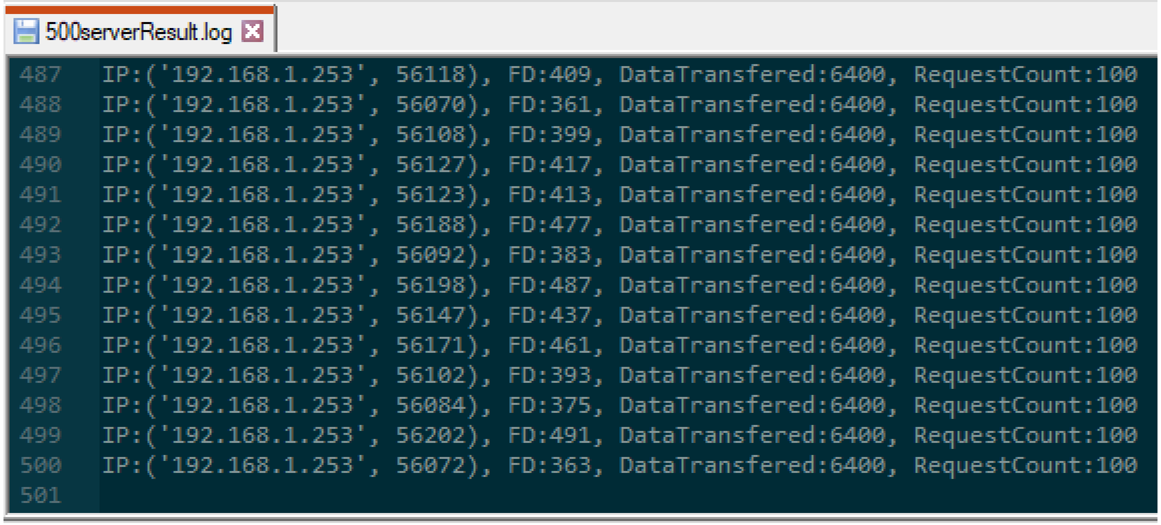
By looking at the server log, there are 200 entries in the file which matches the 200 connections. This is because the entry is logged only when the connection with the client is closed properly. If the client's socket times out, the number of lines will not match the number of clients because the client will not close that connection properly.

The average response time for the connections is fast, roughly an average about 0.028 seconds, roughly 1.74x of the 100 connections result. The Fedora client has a higher average response time per connection most likely because its operating on wifi as opposed to ethernet.

The total duration increased by about 1.34x compared to 100 connections.
The avg response time increased by about 1.74x compared to 100 connections.

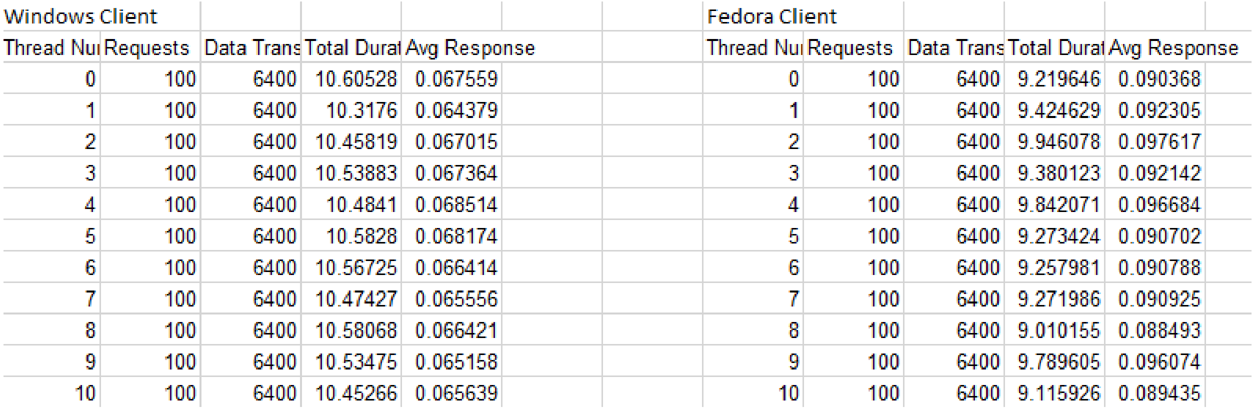
4.3 Test 3, 500 Connections

Server log output



```
IP:('192.168.1.253', 56118), FD:409, DataTransfered:6400, RequestCount:100
IP:('192.168.1.253', 56070), FD:361, DataTransfered:6400, RequestCount:100
IP:('192.168.1.253', 56108), FD:399, DataTransfered:6400, RequestCount:100
IP:('192.168.1.253', 56127), FD:417, DataTransfered:6400, RequestCount:100
IP:('192.168.1.253', 56123), FD:413, DataTransfered:6400, RequestCount:100
IP:('192.168.1.253', 56188), FD:477, DataTransfered:6400, RequestCount:100
IP:('192.168.1.253', 56092), FD:383, DataTransfered:6400, RequestCount:100
IP:('192.168.1.253', 56198), FD:487, DataTransfered:6400, RequestCount:100
IP:('192.168.1.253', 56147), FD:437, DataTransfered:6400, RequestCount:100
IP:('192.168.1.253', 56171), FD:461, DataTransfered:6400, RequestCount:100
IP:('192.168.1.253', 56102), FD:393, DataTransfered:6400, RequestCount:100
IP:('192.168.1.253', 56084), FD:375, DataTransfered:6400, RequestCount:100
IP:('192.168.1.253', 56202), FD:491, DataTransfered:6400, RequestCount:100
IP:('192.168.1.253', 56072), FD:363, DataTransfered:6400, RequestCount:100
```

Part of the results generated by the two clients.



Windows Client						Fedora Client				
Thread Nu	Requests	Data Trans	Total Durat	Avg Response		Thread Nu	Requests	Data Trans	Total Durat	Avg Response
0	100	6400	10.60528	0.067559		0	100	6400	9.219646	0.090368
1	100	6400	10.3176	0.064379		1	100	6400	9.424629	0.092305
2	100	6400	10.45819	0.067015		2	100	6400	9.946078	0.097617
3	100	6400	10.53883	0.067364		3	100	6400	9.380123	0.092142
4	100	6400	10.4841	0.068514		4	100	6400	9.842071	0.096684
5	100	6400	10.5828	0.068174		5	100	6400	9.273424	0.090702
6	100	6400	10.56725	0.066414		6	100	6400	9.257981	0.090788
7	100	6400	10.47427	0.065556		7	100	6400	9.271986	0.090925
8	100	6400	10.58068	0.066421		8	100	6400	9.010155	0.088493
9	100	6400	10.53475	0.065158		9	100	6400	9.789605	0.096074
10	100	6400	10.45266	0.065639		10	100	6400	9.115926	0.089435

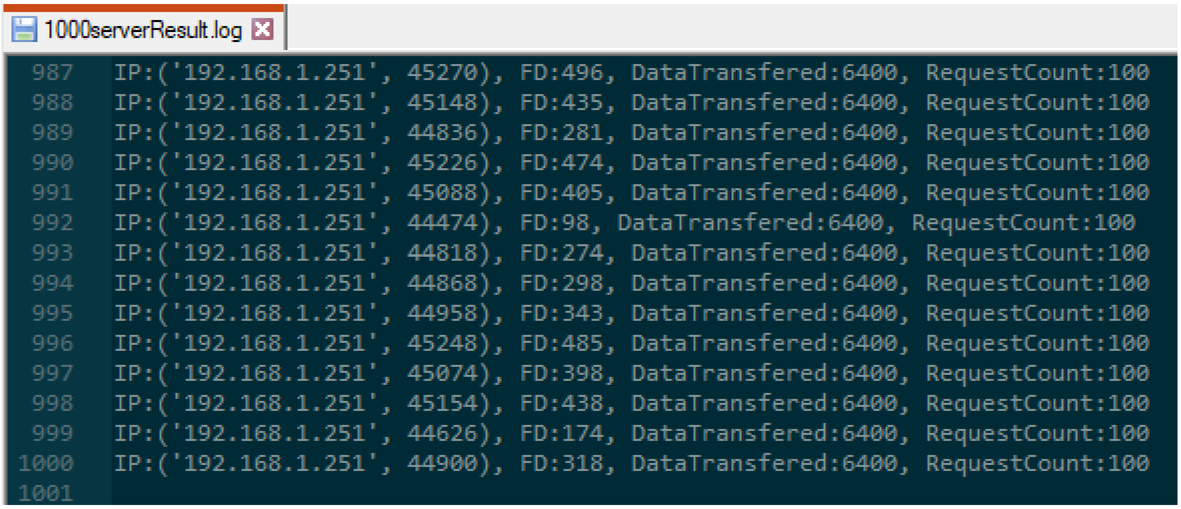
By looking at the server log, there are 500 entries in the file which matches the 500 connections. This is because the entry is logged only when the connection with the client is closed properly. If the client's socket times out, the number of lines will not match the number of clients because the client will not close that connection properly.

The average response time for the connections is getting slower by about 4.9x compared to 100 connections result. The Fedora client has a higher average response time per connection most likely because its operating on wifi as opposed to ethernet.

The total duration increased by about 2.67x compared to 100 connections.
The avg response time increased by about 4.90x compared to 100 connections.

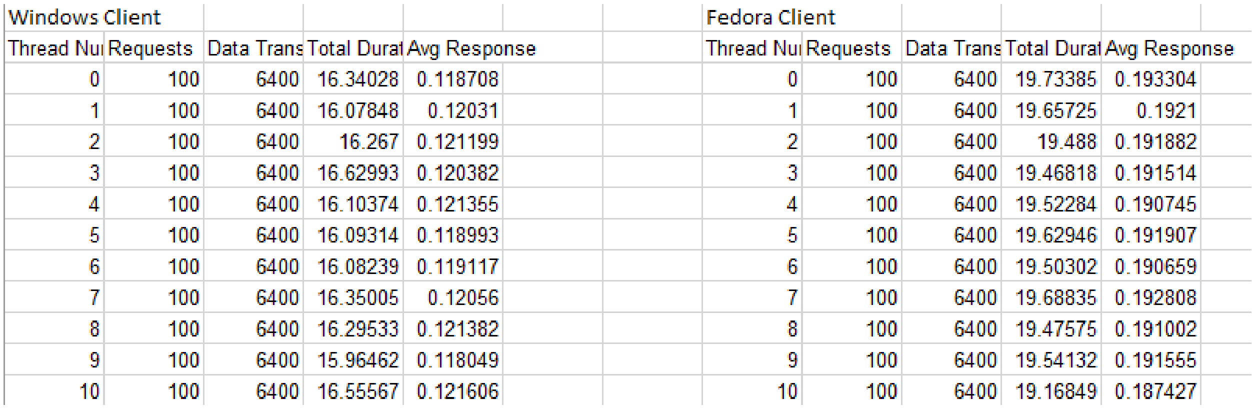
4.4 Test 4, 1000 Connections

Server log output



```
 987  IP:('192.168.1.251', 45270), FD:496, DataTransfered:6400, RequestCount:100
 988  IP:('192.168.1.251', 45148), FD:435, DataTransfered:6400, RequestCount:100
 989  IP:('192.168.1.251', 44836), FD:281, DataTransfered:6400, RequestCount:100
 990  IP:('192.168.1.251', 45226), FD:474, DataTransfered:6400, RequestCount:100
 991  IP:('192.168.1.251', 45088), FD:405, DataTransfered:6400, RequestCount:100
 992  IP:('192.168.1.251', 44474), FD:98, DataTransfered:6400, RequestCount:100
 993  IP:('192.168.1.251', 44818), FD:274, DataTransfered:6400, RequestCount:100
 994  IP:('192.168.1.251', 44868), FD:298, DataTransfered:6400, RequestCount:100
 995  IP:('192.168.1.251', 44958), FD:343, DataTransfered:6400, RequestCount:100
 996  IP:('192.168.1.251', 45248), FD:485, DataTransfered:6400, RequestCount:100
 997  IP:('192.168.1.251', 45074), FD:398, DataTransfered:6400, RequestCount:100
 998  IP:('192.168.1.251', 45154), FD:438, DataTransfered:6400, RequestCount:100
 999  IP:('192.168.1.251', 44626), FD:174, DataTransfered:6400, RequestCount:100
1000  IP:('192.168.1.251', 44900), FD:318, DataTransfered:6400, RequestCount:100
1001
```

Part of the results generated by the two clients.



Windows Client						Fedora Client				
Thread Nu	Requests	Data Trans	Total Durat	Avg Response		Thread Nu	Requests	Data Trans	Total Durat	Avg Response
0	100	6400	16.34028	0.118708		0	100	6400	19.73385	0.193304
1	100	6400	16.07848	0.12031		1	100	6400	19.65725	0.1921
2	100	6400	16.267	0.121199		2	100	6400	19.488	0.191882
3	100	6400	16.62993	0.120382		3	100	6400	19.46818	0.191514
4	100	6400	16.10374	0.121355		4	100	6400	19.52284	0.190745
5	100	6400	16.09314	0.118993		5	100	6400	19.62946	0.191907
6	100	6400	16.08239	0.119117		6	100	6400	19.50302	0.190659
7	100	6400	16.35005	0.12056		7	100	6400	19.68835	0.192808
8	100	6400	16.29533	0.121382		8	100	6400	19.47575	0.191002
9	100	6400	15.96462	0.118049		9	100	6400	19.54132	0.191555
10	100	6400	16.55567	0.121606		10	100	6400	19.16849	0.187427

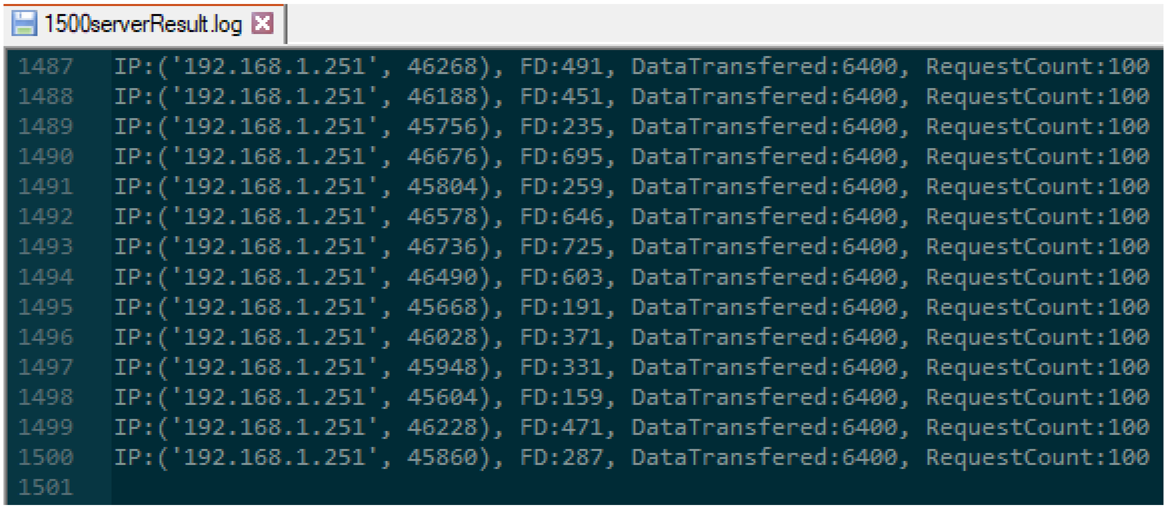
By looking at the server log, there are 1000 entries in the file which matches the 1000 connections. This is because the entry is logged only when the connection with the client is closed properly. If the client's socket times out, the number of lines will not match the number of clients because the client will not close that connection properly.

The average response time for the connections is slower than before by about double, roughly an average about 0.156 seconds which is about 9.6x compared to the results of 100 connections. The Fedora client has a higher average response time per connection most likely because its operating on wifi as opposed to ethernet.

The total duration increased by about 5.02x compared to 100 connections.
The avg response time increased by about 9.60x compared to 100 connections.

4.5 Test 5, 1500 Connections

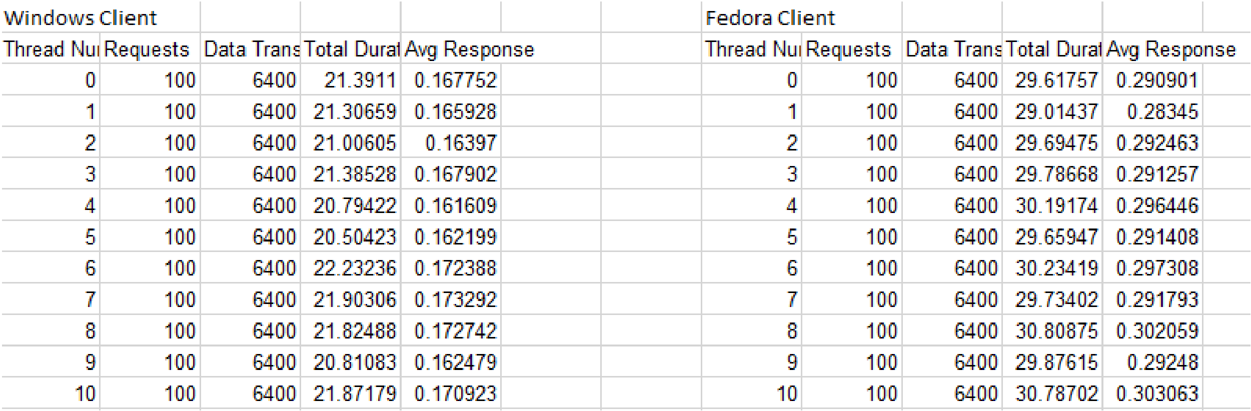
Server log output



1500serverResult.log

```
1487  IP:('192.168.1.251', 46268), FD:491, DataTransfered:6400, RequestCount:100
1488  IP:('192.168.1.251', 46188), FD:451, DataTransfered:6400, RequestCount:100
1489  IP:('192.168.1.251', 45756), FD:235, DataTransfered:6400, RequestCount:100
1490  IP:('192.168.1.251', 46676), FD:695, DataTransfered:6400, RequestCount:100
1491  IP:('192.168.1.251', 45804), FD:259, DataTransfered:6400, RequestCount:100
1492  IP:('192.168.1.251', 46578), FD:646, DataTransfered:6400, RequestCount:100
1493  IP:('192.168.1.251', 46736), FD:725, DataTransfered:6400, RequestCount:100
1494  IP:('192.168.1.251', 46490), FD:603, DataTransfered:6400, RequestCount:100
1495  IP:('192.168.1.251', 45668), FD:191, DataTransfered:6400, RequestCount:100
1496  IP:('192.168.1.251', 46028), FD:371, DataTransfered:6400, RequestCount:100
1497  IP:('192.168.1.251', 45948), FD:331, DataTransfered:6400, RequestCount:100
1498  IP:('192.168.1.251', 45604), FD:159, DataTransfered:6400, RequestCount:100
1499  IP:('192.168.1.251', 46228), FD:471, DataTransfered:6400, RequestCount:100
1500  IP:('192.168.1.251', 45860), FD:287, DataTransfered:6400, RequestCount:100
1501
```

Part of the results generated by the two clients



Windows Client							Fedora Client				
Thread Nu	Requests	Data Trans	Total Durat	Avg Response			Thread Nu	Requests	Data Trans	Total Durat	Avg Response
0	100	6400	21.3911	0.167752			0	100	6400	29.61757	0.290901
1	100	6400	21.30659	0.165928			1	100	6400	29.01437	0.28345
2	100	6400	21.00605	0.16397			2	100	6400	29.69475	0.292463
3	100	6400	21.38528	0.167902			3	100	6400	29.78668	0.291257
4	100	6400	20.79422	0.161609			4	100	6400	30.19174	0.296446
5	100	6400	20.50423	0.162199			5	100	6400	29.65947	0.291408
6	100	6400	22.23236	0.172388			6	100	6400	30.23419	0.297308
7	100	6400	21.90306	0.173292			7	100	6400	29.73402	0.291793
8	100	6400	21.82488	0.172742			8	100	6400	30.80875	0.302059
9	100	6400	20.81083	0.162479			9	100	6400	29.87615	0.29248
10	100	6400	21.87179	0.170923			10	100	6400	30.78702	0.303063

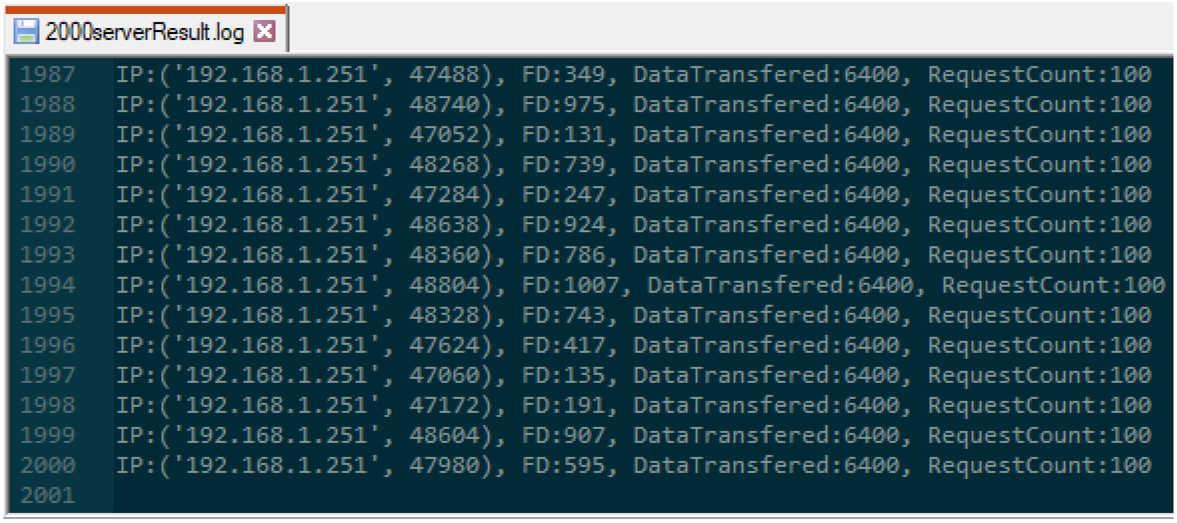
By looking at the server log, there are 1500 entries in the file which matches the 1500 connections. This is because the entry is logged only when the connection with the client is closed properly. If the client's socket times out, the number of lines will not match the number of clients because the client will not close that connection properly.

The average response time for the connections is slower than before by about double, roughly an average about 0.231 seconds which is about 14.26x longer compared to the 100 connections results. The Fedora client has a higher average response time per connection most likely because its operating on wifi as opposed to ethernet.

The total duration increased by about 7.23x compared to 100 connections.
The avg response time increased by about 14.26x compared to 100 connections.

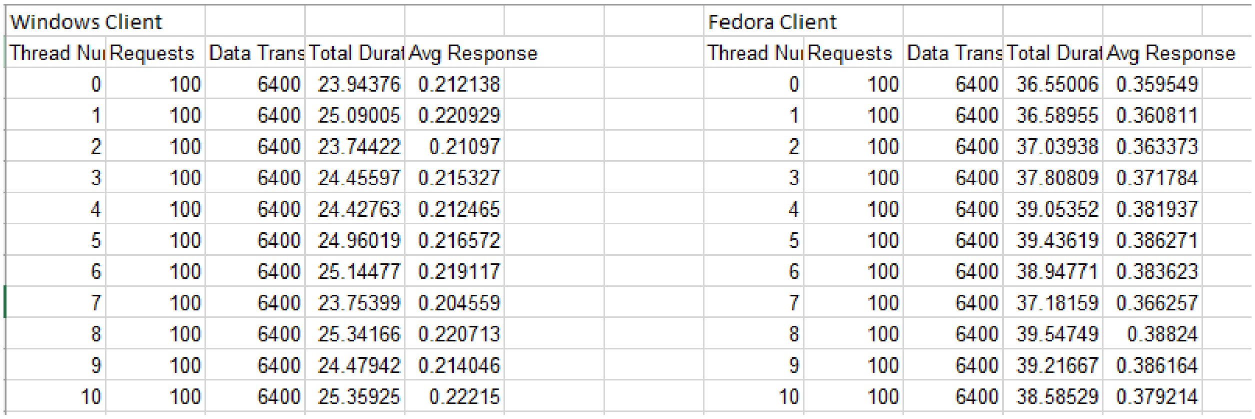
4.6 Test 6, 2000 Connections

Server log output



```
1987  IP:('192.168.1.251', 47488), FD:349, DataTransfered:6400, RequestCount:100
1988  IP:('192.168.1.251', 48740), FD:975, DataTransfered:6400, RequestCount:100
1989  IP:('192.168.1.251', 47052), FD:131, DataTransfered:6400, RequestCount:100
1990  IP:('192.168.1.251', 48268), FD:739, DataTransfered:6400, RequestCount:100
1991  IP:('192.168.1.251', 47284), FD:247, DataTransfered:6400, RequestCount:100
1992  IP:('192.168.1.251', 48638), FD:924, DataTransfered:6400, RequestCount:100
1993  IP:('192.168.1.251', 48360), FD:786, DataTransfered:6400, RequestCount:100
1994  IP:('192.168.1.251', 48804), FD:1007, DataTransfered:6400, RequestCount:100
1995  IP:('192.168.1.251', 48328), FD:743, DataTransfered:6400, RequestCount:100
1996  IP:('192.168.1.251', 47624), FD:417, DataTransfered:6400, RequestCount:100
1997  IP:('192.168.1.251', 47060), FD:135, DataTransfered:6400, RequestCount:100
1998  IP:('192.168.1.251', 47172), FD:191, DataTransfered:6400, RequestCount:100
1999  IP:('192.168.1.251', 48604), FD:907, DataTransfered:6400, RequestCount:100
2000  IP:('192.168.1.251', 47980), FD:595, DataTransfered:6400, RequestCount:100
2001
```

Part of the results generated by the two clients



Windows Client							Fedora Client				
Thread Nu	Requests	Data Trans	Total Durat	Avg Response			Thread Nu	Requests	Data Trans	Total Durat	Avg Response
0	100	6400	23.94376	0.212138			0	100	6400	36.55006	0.359549
1	100	6400	25.09005	0.220929			1	100	6400	36.58955	0.360811
2	100	6400	23.74422	0.21097			2	100	6400	37.03938	0.363373
3	100	6400	24.45597	0.215327			3	100	6400	37.80809	0.371784
4	100	6400	24.42763	0.212465			4	100	6400	39.05352	0.381937
5	100	6400	24.96019	0.216572			5	100	6400	39.43619	0.386271
6	100	6400	25.14477	0.219117			6	100	6400	38.94771	0.383623
7	100	6400	23.75399	0.204559			7	100	6400	37.18159	0.366257
8	100	6400	25.34166	0.220713			8	100	6400	39.54749	0.38824
9	100	6400	24.47942	0.214046			9	100	6400	39.21667	0.386164
10	100	6400	25.35925	0.22215			10	100	6400	38.58529	0.379214

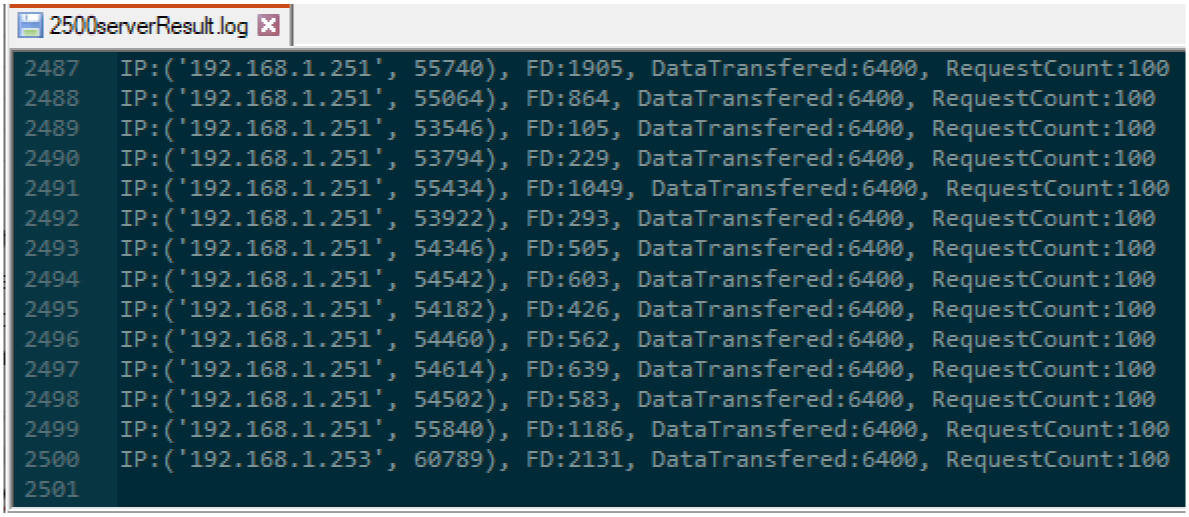
By looking at the server log, there are 2000 entries in the file which matches the 2000 connections. This is because the entry is logged only when the connection with the client is closed properly. If the client's socket times out, the number of lines will not match the number of clients because the client will not close that connection properly.

The average response time for the connections is slower than before, roughly an average about 0.298 seconds which is about 18.36x higher than the 100 connections results. The Fedora client has a higher average response time per connection most likely because its operating on wifi as opposed to ethernet.

The total duration increased by about 8.905x compared to 100 connections.
The avg response time increased by about 18.36x compared to 100 connections.

4.7 Test 7, 2500 Connections

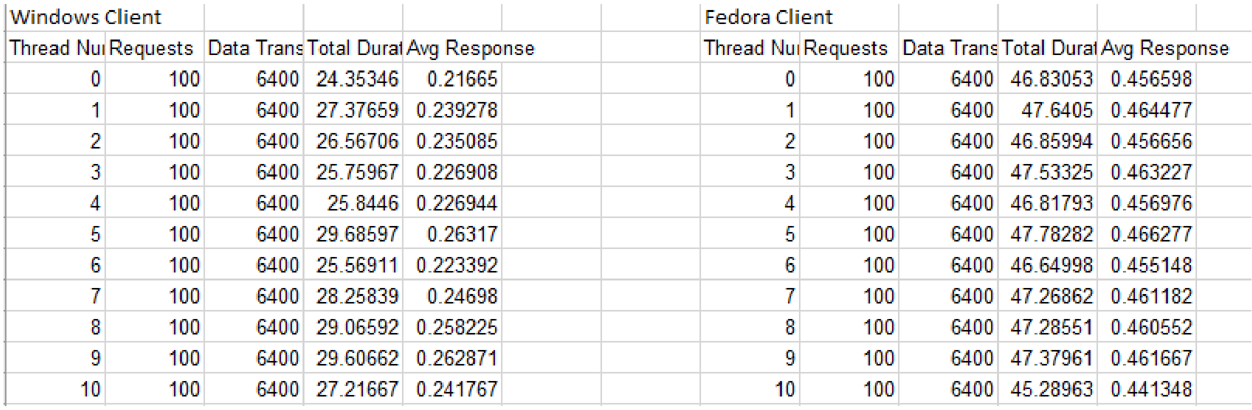
Server log output



2500serverResult.log

```
2487    IP:('192.168.1.251', 55740), FD:1905, DataTransfered:6400, RequestCount:100
2488    IP:('192.168.1.251', 55064), FD:864, DataTransfered:6400, RequestCount:100
2489    IP:('192.168.1.251', 53546), FD:105, DataTransfered:6400, RequestCount:100
2490    IP:('192.168.1.251', 53794), FD:229, DataTransfered:6400, RequestCount:100
2491    IP:('192.168.1.251', 55434), FD:1049, DataTransfered:6400, RequestCount:100
2492    IP:('192.168.1.251', 53922), FD:293, DataTransfered:6400, RequestCount:100
2493    IP:('192.168.1.251', 54346), FD:505, DataTransfered:6400, RequestCount:100
2494    IP:('192.168.1.251', 54542), FD:603, DataTransfered:6400, RequestCount:100
2495    IP:('192.168.1.251', 54182), FD:426, DataTransfered:6400, RequestCount:100
2496    IP:('192.168.1.251', 54460), FD:562, DataTransfered:6400, RequestCount:100
2497    IP:('192.168.1.251', 54614), FD:639, DataTransfered:6400, RequestCount:100
2498    IP:('192.168.1.251', 54502), FD:583, DataTransfered:6400, RequestCount:100
2499    IP:('192.168.1.251', 55840), FD:1186, DataTransfered:6400, RequestCount:100
2500    IP:('192.168.1.253', 60789), FD:2131, DataTransfered:6400, RequestCount:100
2501
```

Part of the results generated by the two clients



Windows Client							Fedora Client				
Thread Nu	Requests	Data Trans	Total Durat	Avg Response			Thread Nu	Requests	Data Trans	Total Durat	Avg Response
0	100	6400	24.35346	0.21665			0	100	6400	46.83053	0.456598
1	100	6400	27.37659	0.239278			1	100	6400	47.6405	0.464477
2	100	6400	26.56706	0.235085			2	100	6400	46.85994	0.456656
3	100	6400	25.75967	0.226908			3	100	6400	47.53325	0.463227
4	100	6400	25.8446	0.226944			4	100	6400	46.81793	0.456976
5	100	6400	29.68597	0.26317			5	100	6400	47.78282	0.466277
6	100	6400	25.56911	0.223392			6	100	6400	46.64998	0.455148
7	100	6400	28.25839	0.24698			7	100	6400	47.26862	0.461182
8	100	6400	29.06592	0.258225			8	100	6400	47.28551	0.460552
9	100	6400	29.60662	0.262871			9	100	6400	47.37961	0.461667
10	100	6400	27.21667	0.241767			10	100	6400	45.28963	0.441348

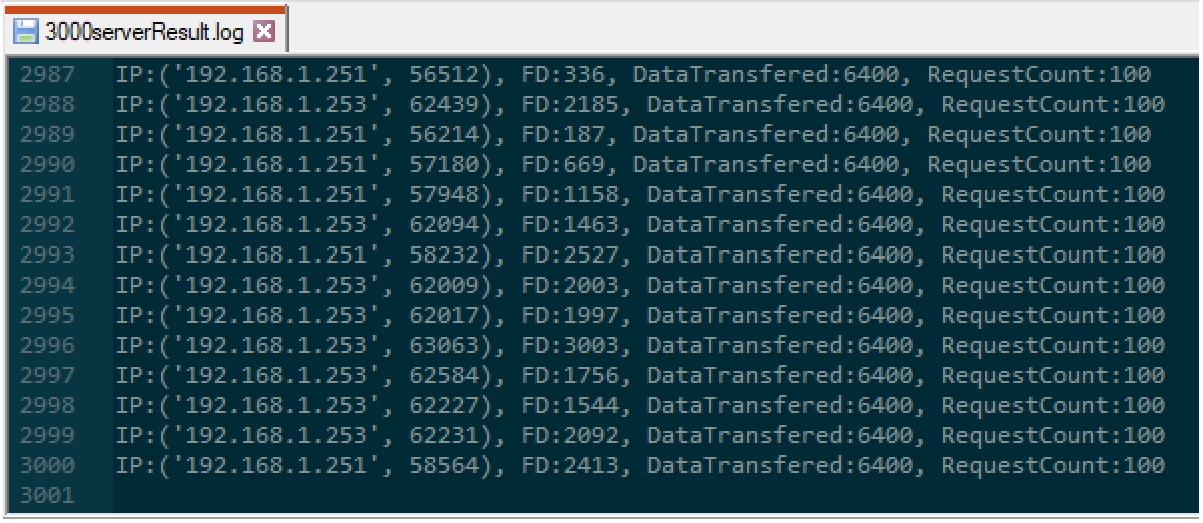
By looking at the server log, there are 2500 entries in the file which matches the 2500 connections. This is because the entry is logged only when the connection with the client is closed properly. If the client's socket times out, the number of lines will not match the number of clients because the client will not close that connection properly.

The average response time for the connections is slower than before, roughly an average about 0.354 seconds which is 21.83x more than the results of 100 connections. The Fedora client has a higher average response time per connection most likely because its operating on wifi as opposed to ethernet.

The total duration increased by about 10.43x compared to 100 connections.
The avg response time increased by about 21.83x compared to 100 connections.

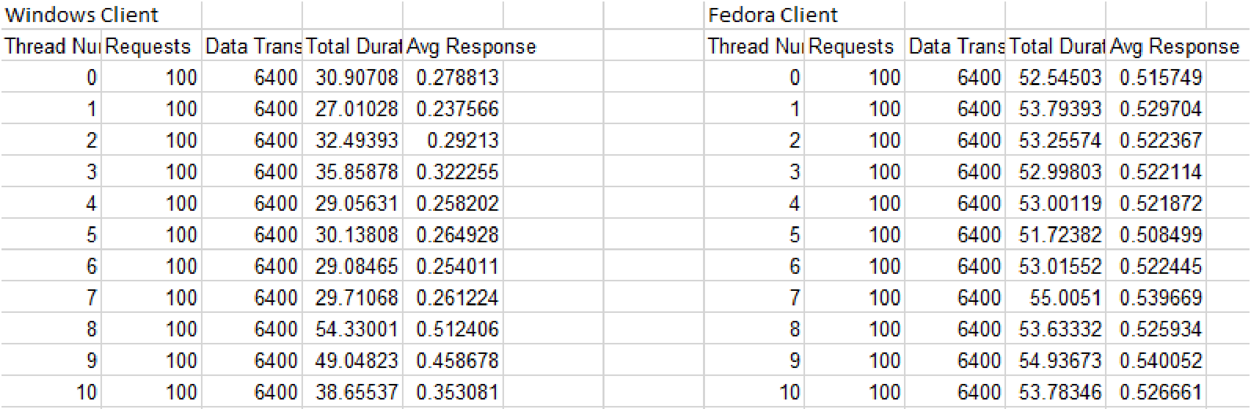
4.8 Test 8, 3000 Connections

Server log output



```
2987  IP:('192.168.1.251', 56512), FD:336, DataTransfered:6400, RequestCount:100
2988  IP:('192.168.1.253', 62439), FD:2185, DataTransfered:6400, RequestCount:100
2989  IP:('192.168.1.251', 56214), FD:187, DataTransfered:6400, RequestCount:100
2990  IP:('192.168.1.251', 57180), FD:669, DataTransfered:6400, RequestCount:100
2991  IP:('192.168.1.251', 57948), FD:1158, DataTransfered:6400, RequestCount:100
2992  IP:('192.168.1.253', 62094), FD:1463, DataTransfered:6400, RequestCount:100
2993  IP:('192.168.1.251', 58232), FD:2527, DataTransfered:6400, RequestCount:100
2994  IP:('192.168.1.253', 62009), FD:2003, DataTransfered:6400, RequestCount:100
2995  IP:('192.168.1.253', 62017), FD:1997, DataTransfered:6400, RequestCount:100
2996  IP:('192.168.1.253', 63063), FD:3003, DataTransfered:6400, RequestCount:100
2997  IP:('192.168.1.253', 62584), FD:1756, DataTransfered:6400, RequestCount:100
2998  IP:('192.168.1.253', 62227), FD:1544, DataTransfered:6400, RequestCount:100
2999  IP:('192.168.1.253', 62231), FD:2092, DataTransfered:6400, RequestCount:100
3000  IP:('192.168.1.251', 58564), FD:2413, DataTransfered:6400, RequestCount:100
3001
```

Part of the results generated by the two clients



Windows Client							Fedora Client				
Thread Nu	Requests	Data Trans	Total Durat	Avg Response			Thread Nu	Requests	Data Trans	Total Durat	Avg Response
0	100	6400	30.90708	0.278813			0	100	6400	52.54503	0.515749
1	100	6400	27.01028	0.237566			1	100	6400	53.79393	0.529704
2	100	6400	32.49393	0.29213			2	100	6400	53.25574	0.522367
3	100	6400	35.85878	0.322255			3	100	6400	52.99803	0.522114
4	100	6400	29.05631	0.258202			4	100	6400	53.00119	0.521872
5	100	6400	30.13808	0.264928			5	100	6400	51.72382	0.508499
6	100	6400	29.08465	0.254011			6	100	6400	53.01552	0.522445
7	100	6400	29.71068	0.261224			7	100	6400	55.0051	0.539669
8	100	6400	54.33001	0.512406			8	100	6400	53.63332	0.525934
9	100	6400	49.04823	0.458678			9	100	6400	54.93673	0.540052
10	100	6400	38.65537	0.353081			10	100	6400	53.78346	0.526661

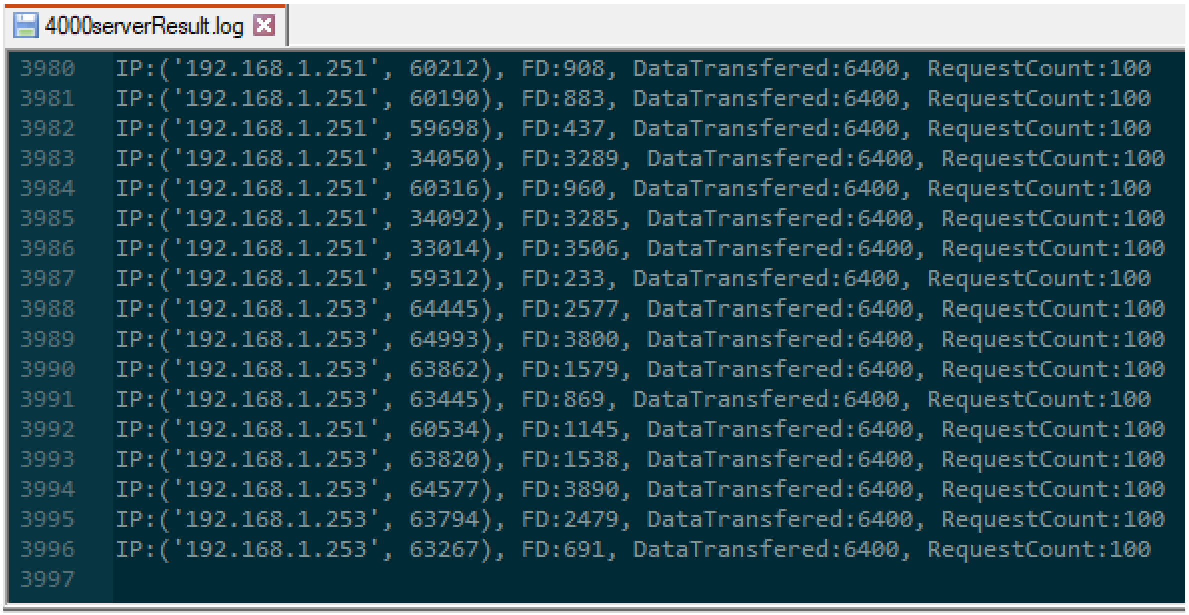
By looking at the server log, there are 3000 entries in the file which matches the 3000 connections. This is because the entry is logged only when the connection with the client is closed properly. If the client's socket times out, the number of lines will not match the number of clients because the client will not close that connection properly.

The average response time for the connections is slower than before, roughly an average about 0.4064 seconds which is about 25.03x higher than the 100 connections results. The Fedora client has a higher average response time per connection most likely because its operating on wifi as opposed to ethernet.

The total duration increased by about 11.99x compared to 100 connections.
The avg response time increased by about 25.03x compared to 100 connections.

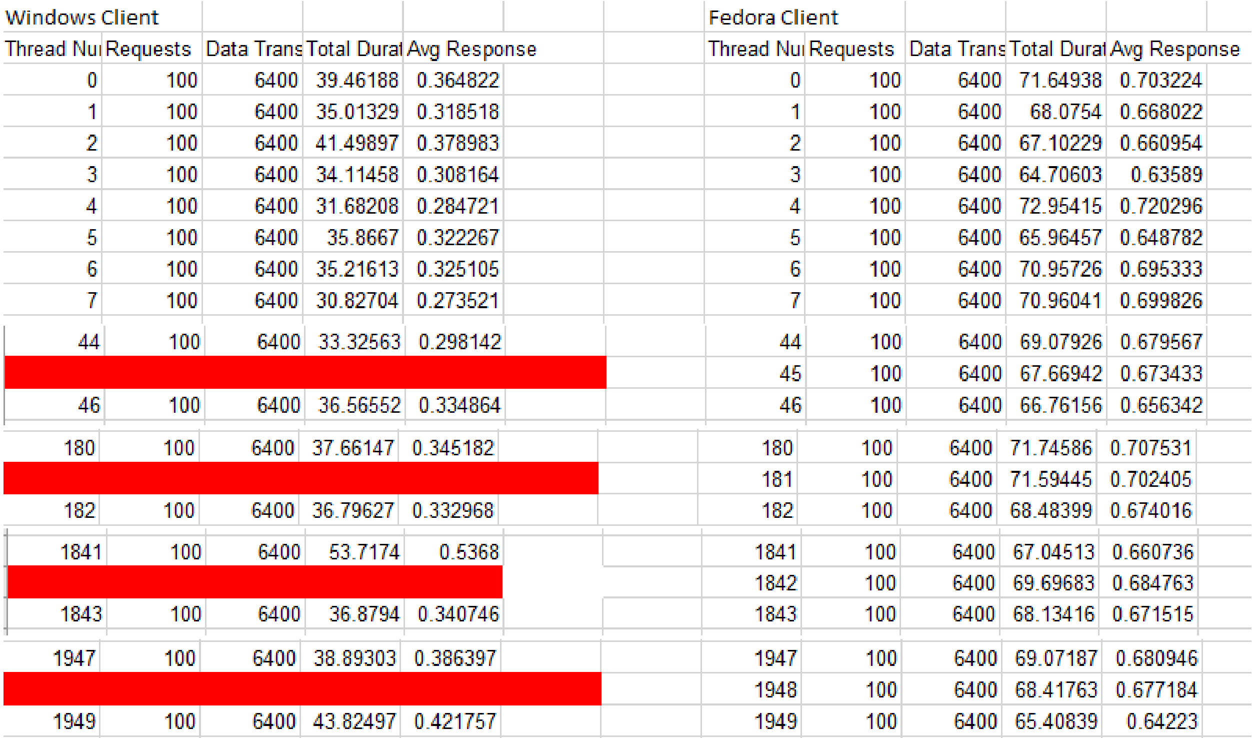
4.9 Test 9, 4000 Connections

Server log output



```
3980   IP:('192.168.1.251', 60212), FD:908, DataTransfered:6400, RequestCount:100
3981   IP:('192.168.1.251', 60190), FD:883, DataTransfered:6400, RequestCount:100
3982   IP:('192.168.1.251', 59698), FD:437, DataTransfered:6400, RequestCount:100
3983   IP:('192.168.1.251', 34050), FD:3289, DataTransfered:6400, RequestCount:100
3984   IP:('192.168.1.251', 60316), FD:960, DataTransfered:6400, RequestCount:100
3985   IP:('192.168.1.251', 34092), FD:3285, DataTransfered:6400, RequestCount:100
3986   IP:('192.168.1.251', 33014), FD:3506, DataTransfered:6400, RequestCount:100
3987   IP:('192.168.1.251', 59312), FD:233, DataTransfered:6400, RequestCount:100
3988   IP:('192.168.1.253', 64445), FD:2577, DataTransfered:6400, RequestCount:100
3989   IP:('192.168.1.253', 64993), FD:3800, DataTransfered:6400, RequestCount:100
3990   IP:('192.168.1.253', 63862), FD:1579, DataTransfered:6400, RequestCount:100
3991   IP:('192.168.1.253', 63445), FD:869, DataTransfered:6400, RequestCount:100
3992   IP:('192.168.1.251', 60534), FD:1145, DataTransfered:6400, RequestCount:100
3993   IP:('192.168.1.253', 63820), FD:1538, DataTransfered:6400, RequestCount:100
3994   IP:('192.168.1.253', 64577), FD:3890, DataTransfered:6400, RequestCount:100
3995   IP:('192.168.1.253', 63794), FD:2479, DataTransfered:6400, RequestCount:100
3996   IP:('192.168.1.253', 63267), FD:691, DataTransfered:6400, RequestCount:100
3997
```

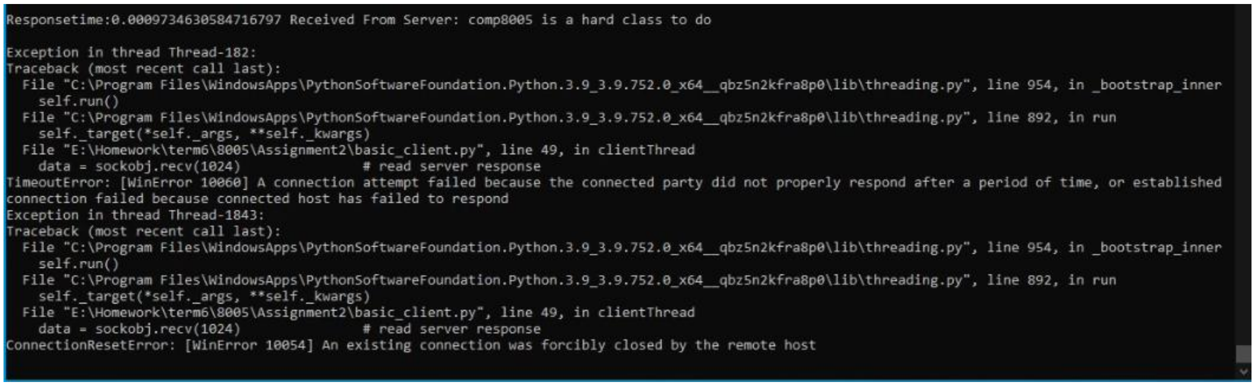
Part of the results generated by the two clients



Windows Client						Fedora Client				
Thread Nu	Requests	Data Trans	Total Durat	Avg Response		Thread Nu	Requests	Data Trans	Total Durat	Avg Response
0	100	6400	39.46188	0.364822		0	100	6400	71.64938	0.703224
1	100	6400	35.01329	0.318518		1	100	6400	68.0754	0.668022
2	100	6400	41.49897	0.378983		2	100	6400	67.10229	0.660954
3	100	6400	34.11458	0.308164		3	100	6400	64.70603	0.63589
4	100	6400	31.68208	0.284721		4	100	6400	72.95415	0.720296
5	100	6400	35.8667	0.322267		5	100	6400	65.96457	0.648782
6	100	6400	35.21613	0.325105		6	100	6400	70.95726	0.695333
7	100	6400	30.82704	0.273521		7	100	6400	70.96041	0.699826
44	100	6400	33.32563	0.298142		44	100	6400	69.07926	0.679567
🟥	🟥	🟥	🟥	🟥		45	100	6400	67.66942	0.673433
46	100	6400	36.56552	0.334864		46	100	6400	66.76156	0.656342
180	100	6400	37.66147	0.345182		180	100	6400	71.74586	0.707531
🟥	🟥	🟥	🟥	🟥		181	100	6400	71.59445	0.702405
182	100	6400	36.79627	0.332968		182	100	6400	68.48399	0.674016
1841	100	6400	53.7174	0.5368		1841	100	6400	67.04513	0.660736
🟥	🟥	🟥	🟥	🟥		1842	100	6400	69.69683	0.684763
1843	100	6400	36.8794	0.340746		1843	100	6400	68.13416	0.671515
1947	100	6400	38.89303	0.386397		1947	100	6400	69.07187	0.680946
🟥	🟥	🟥	🟥	🟥		1948	100	6400	68.41763	0.677184
1949	100	6400	43.82497	0.421757		1949	100	6400	65.40839	0.64223

By looking at the server log, there are 3996 entries in the file which does not match the 4000 connections. This is because the entry is logged only when the connection with the client is closed properly. If the client's socket times out, the number of lines will not match the number of clients because the client will not close that connection properly

Looking at my results generated by my clients, my Windows client had their sockets time out on 4 connections which is why there are 4 missing connections as I have marked them as red on the ***FullCombined_clientstats.xlsx***



```
Responsetime:0.0009734630584716797 Received From Server: comp8005 is a hard class to do

Exception in thread Thread-182:
Traceback (most recent call last):
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.752.0_x64__qbz5n2kfra8p0\lib\threading.py", line 954, in _bootstrap_inner
    self.run()
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.752.0_x64__qbz5n2kfra8p0\lib\threading.py", line 892, in run
    self._target(*self._args, **self._kwargs)
  File "E:\Homework\term6\8005\Assignment2\basic_client.py", line 49, in clientThread
    data = sockobj.recv(1024)               # read server response
TimeoutError: [WinError 10060] A connection attempt failed because the connected party did not properly respond after a period of time, or established
connection failed because connected host has failed to respond
Exception in thread Thread-1843:
Traceback (most recent call last):
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.752.0_x64__qbz5n2kfra8p0\lib\threading.py", line 954, in _bootstrap_inner
    self.run()
  File "C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.9_3.9.752.0_x64__qbz5n2kfra8p0\lib\threading.py", line 892, in run
    self._target(*self._args, **self._kwargs)
  File "E:\Homework\term6\8005\Assignment2\basic_client.py", line 49, in clientThread
    data = sockobj.recv(1024)               # read server response
ConnectionResetError: [WinError 10054] An existing connection was forcibly closed by the remote host
```

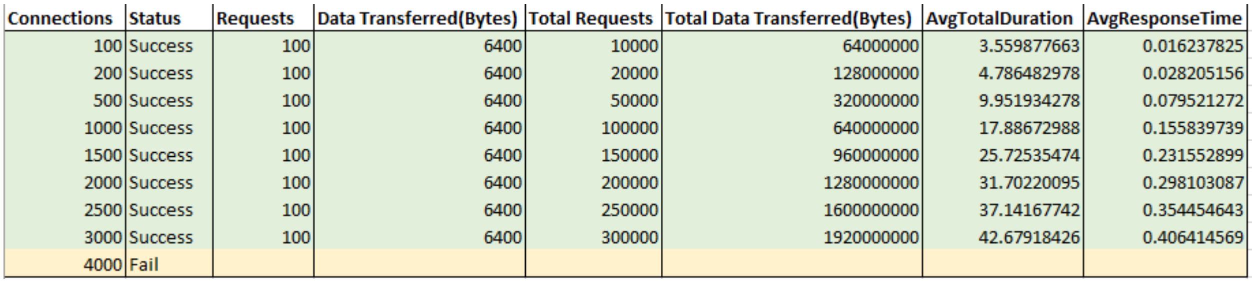
This screenshot from the video of attempting 4000 connections matches the timed-out socket because threads are referred starting from 1, in my spreadsheet threads start at 0 so the missing 181 entry matches according to the video.

Other than the socket timing out, the total duration and average response times increased drastically compared to 100 connections. The Avg response time should average to about 0.52 seconds with a duration of around 55 seconds.

5 Conclusions

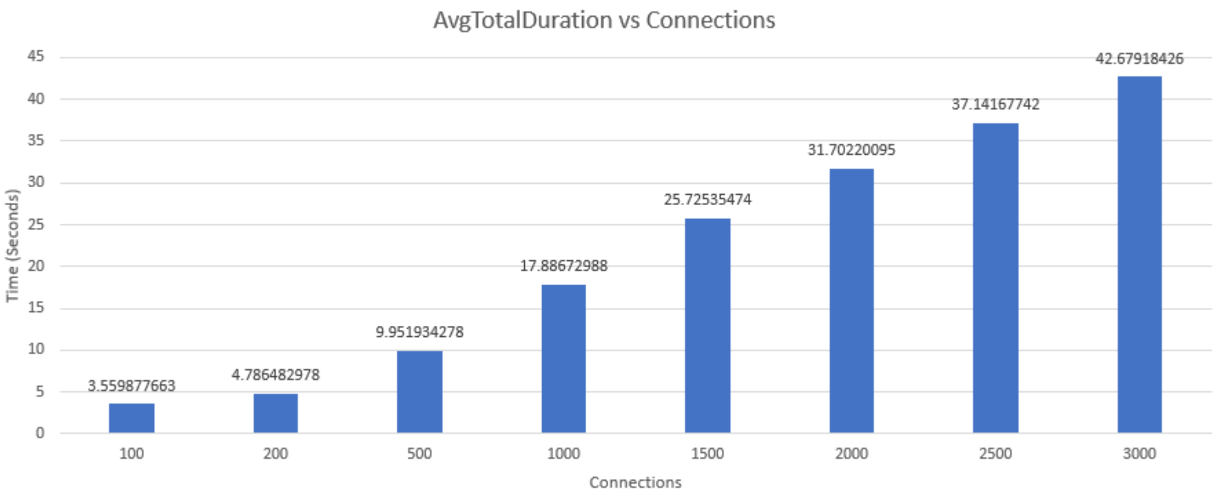
From all the data collected, I can infer that the more connections there are, the average connection duration needed to complete the 100 requests increases greatly along with the average response time when there are many clients connected to the server.

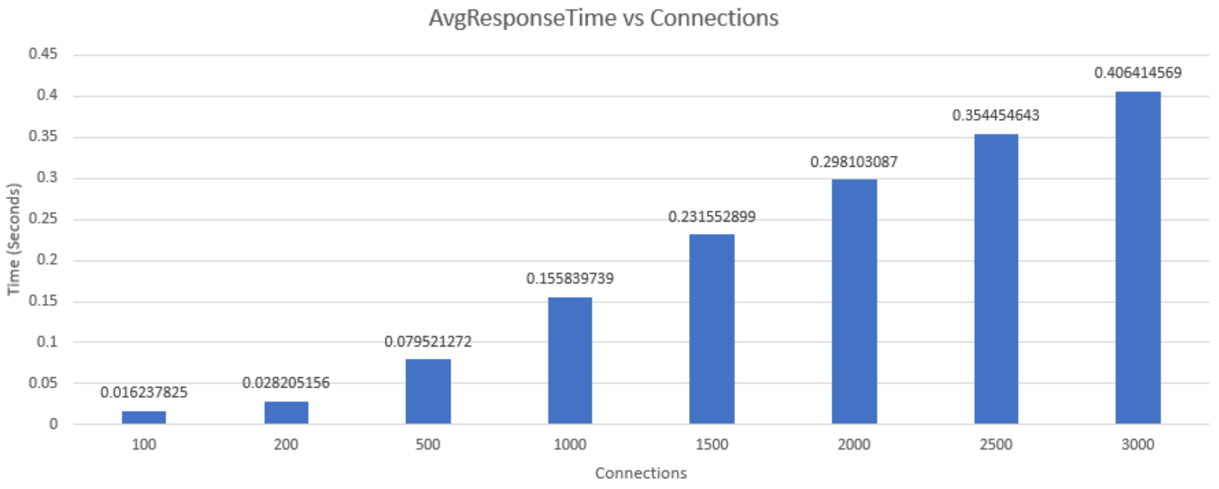
By compiling all the data gathered by the client, the results are the following tables which can viewed as the "Overall" sheet in ***FullCombined_clientstats.xlsx***



Connections	Status	Requests	Data Transferred(Bytes)	Total Requests	Total Data Transferred(Bytes)	AvgTotalDuration	AvgResponseTime
100	Success	100	6400	10000	64000000	3.559877663	0.016237825
200	Success	100	6400	20000	128000000	4.786482978	0.028205156
500	Success	100	6400	50000	320000000	9.951934278	0.079521272
1000	Success	100	6400	100000	640000000	17.88672988	0.155839739
1500	Success	100	6400	150000	960000000	25.72535474	0.231552899
2000	Success	100	6400	200000	1280000000	31.70220095	0.298103087
2500	Success	100	6400	250000	1600000000	37.14167742	0.354454643
3000	Success	100	6400	300000	1920000000	42.67918426	0.406414569
4000	Fail						

This table represents the results of each tests and the average durations and response time as an average across both clients. I used this data to generate the following two bar graphs, specifically the ***AvgTotalDuration*** and ***<u>AvgResponseTime</u>***





Looking closely at both graphs, it is visible that with increasing connections, the increase in time is linear. Both the response time and the duration needed to complete the 100 requests made by the client grow linearly as you can draw a straight diagonal line through both bar graphs, and it will match.

From testing, I believe that Windows times out its sockets much faster than on Fedora because during the test with 4000 connections by doing 2000 connections on my Windows client and 2000 connections on my Fedora client. My Fedora client had a much higher connection duration and response time for each client, and it all completed successfully however, my Windows client had a shorter total duration and average response time, but its connection was reset as evidenced by my test data.

As there more clients connect to the sever, the server's performance drops drastically until the point some connections may potentially drop and difference in operating system may make a connection reset prematurely.