

Homework was done using python 3.9

server.py -> run: python server.py and choose the protocol

client.py -> run: python client.py with the following arguments:

- protocol: tcp or udp (default: tcp)
- mechanism: streaming or stop-and-wait (default streaming)

| Protocol | Total dimension of transmission (mb) | Mecanism | Elapsed Time | Client: Messages sent | Client: Bytes sent | Server: Messages received | Server: Bytes received |
|----------|--------------------------------------|---------------|--------------|-----------------------|--------------------|---------------------------|------------------------|
| TCP | 100 | Streaming | 0.48239 | 10300 | 10547200000 | 10296 | 102430900 |
| TCP | 500 | Streaming | 2.37152 | 51500 | 52736000000 | 51487 | 512154500 |
| TCP | 1000 | Streaming | 4.784084 | 103000 | 52736000000 | 102974 | 1024309000 |
| TCP | 100 | Stop-and-wait | 0.76817 | 10300 | 10547200000 | 10300 | 102430900 |
| TCP | 500 | Stop-and-wait | 3.62518 | 51500 | 52736000000 | 51500 | 512154500 |
| TCP | 1000 | Stop-and-wait | 7.61130 | 103000 | 105472000000 | 103000 | 1024309000 |
| UDP | 100 | Streaming | 1.19785 | 10300 | 10547200000 | 10300 | 102430900 |
| UDP | 500 | Streaming | 5.94265 | 51500 | 52736000000 | 51500 | 512154500 |
| UDP | 1000 | Streaming | 15.01539 | 103000 | 105472000000 | 102998 | 1024288994 |
| UDP | 100 | Stop-and-wait | 2.52983 | 10300 | 10547200000 | 10300 | 102430900 |
| UDP | 500 | Stop-and-wait | 8.17720 | 51500 | 52736000000 | 51500 | 512154500 |
| UDP | 1000 | Stop-and-wait | 9.09627 | 103000 | 105472000000 | 103000 | 1024309000 |

The data presented in this report shows the performance of the TCP and UDP protocols using different congestion control mechanisms for transmission of data over a network. The protocols were tested using three different transmission sizes of 100MB, 500MB, and 1000MB, and two different congestion control mechanisms: stop-and-wait and streaming.

The results indicate that the TCP protocol performs better than the UDP protocol in terms of reliability and congestion control. Specifically, the TCP protocol with streaming mechanism was able to achieve the highest throughput and lowest elapsed time across all transmission sizes. The increase was more pronounced for the larger transmission sizes, indicating that the TCP protocol is more suitable for high-bandwidth networks.

The UDP protocol, on the other hand, showed lower throughput and longer elapsed time compared to TCP, particularly for larger transmission sizes. However, the use of the stop-and-wait mechanism in UDP was able to improve its performance, particularly for larger transmission sizes.

Conclusion:

In conclusion, the choice of protocol and congestion control mechanism plays a critical role in the performance of data transmission over a network. The TCP protocol, with its streaming mechanism, proved to be the most efficient in terms of reliability and congestion control, particularly for large

transmission sizes. However, the use of stop-and-wait mechanism in UDP can also be an effective option for larger transmission sizes.

It is important for network administrators and developers to consider the requirements of their network when choosing the appropriate protocol and congestion control mechanism. The TCP protocol may be more suitable for high-bandwidth networks, while UDP with stop-and-wait may be more appropriate for applications where reliability is less important than speed. By selecting the right protocol and mechanism, network performance can be optimized and potential issues with congestion and reliability can be avoided.