Statement and Confirmation of Own Work

A signed copy of this form must be submitted with every assignment.
If the statement is missing your work may not be marked.

Student Declaration

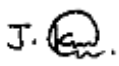
I confirm the following details:

Student Name:	Johnson Kenisha Corera
Student ID Number:	c23020001@cicra.edu.lk
Qualification:	Bachelor in Cyber Security
Unit:	SIT325 Advanced Network Security
Centre:	CICRA Campus
Word Count:	517

I have read and understood both *Deakin Academic Misconduct Policy* and the *Referencing and Bibliographies* document. To the best of my knowledge my work has been accurately referenced and all sources cited correctly.

I confirm that I have not exceeded the stipulated word limit by more than 10%.

I confirm that this is my own work and that I have not colluded or plagiarized any part of it.

Candidate Signature:	
Date:	01/02/2025

Task 6.3D
Table of Content

Table of Figures

Part A

Figure 01: onos and the topology are related.

- I turned on the OS and saw the network.

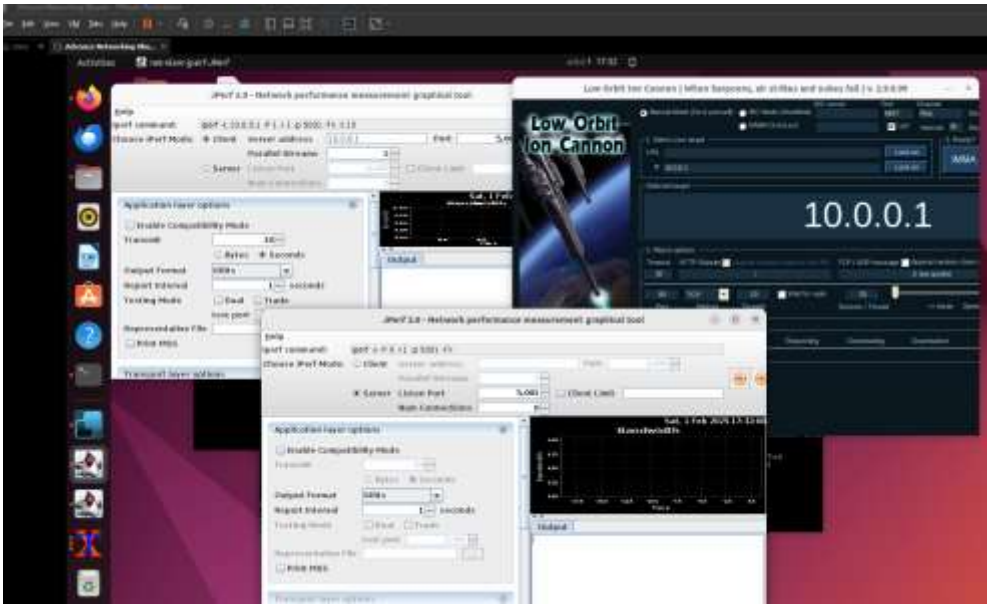


Figure 02: Running Jperf and LOIC

- The client-server setup is correct and the Jperf tool functions well along with LOIC being suitable for attack executions while pre-attack and post-attack CPU and bandwidth usage utilizing TCP and UDP protocols are presented below.

Performance before the attack (while running on TCP)

Figure 03: CPU performance before attack

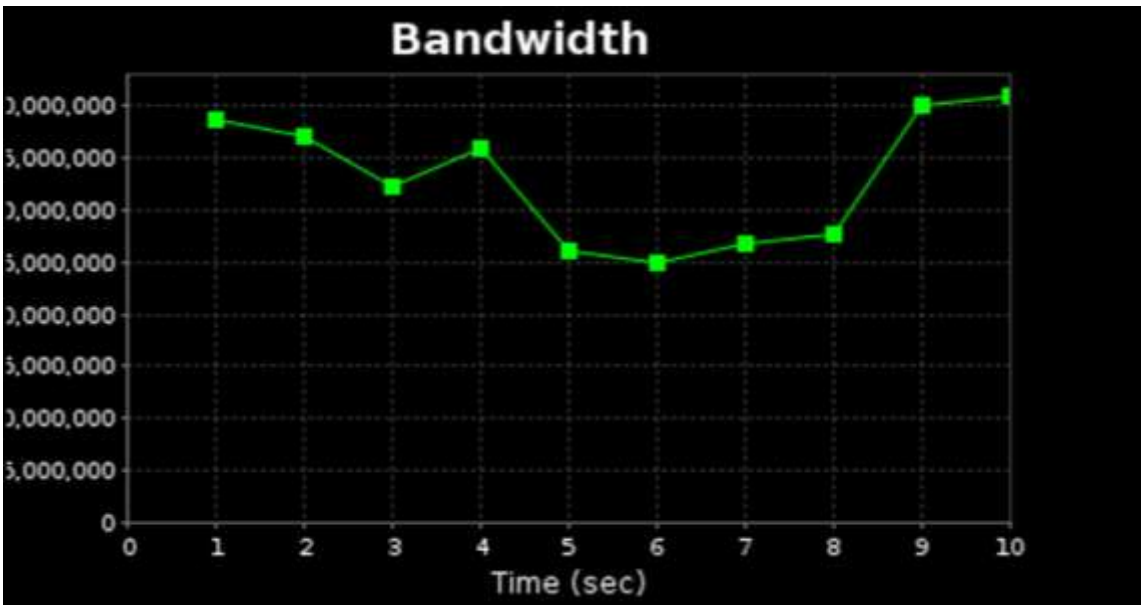


Figure 04: Bandwidth before attack

Performance with the attack (while running on TCP)

Figure 05: CPU while attack

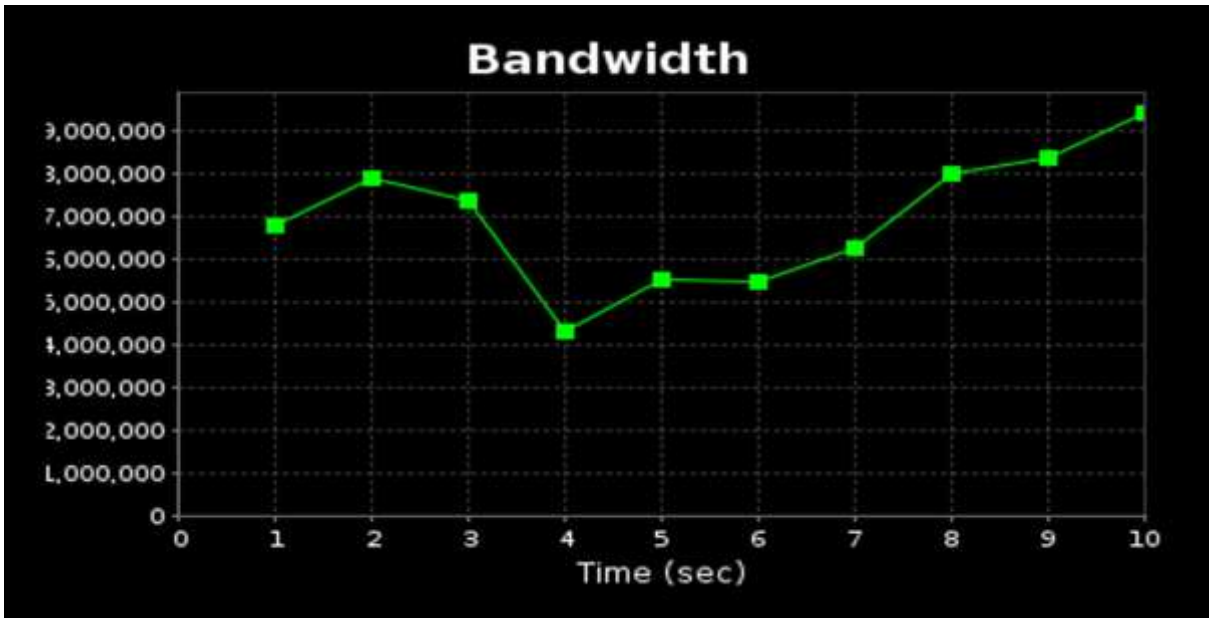


Figure 06: Bandwidth while attack

Performance before the attack (while running on UDP)

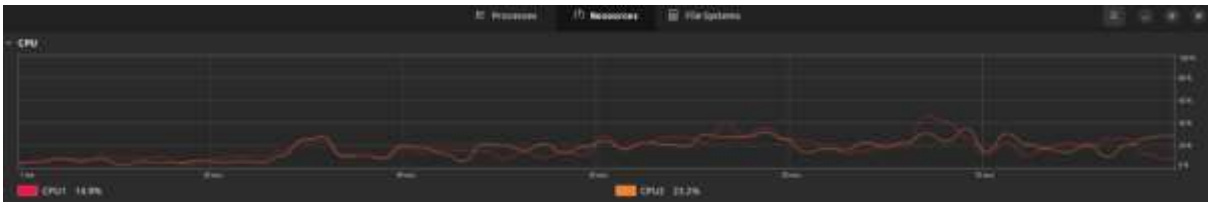


Figure 07: CPU before attack

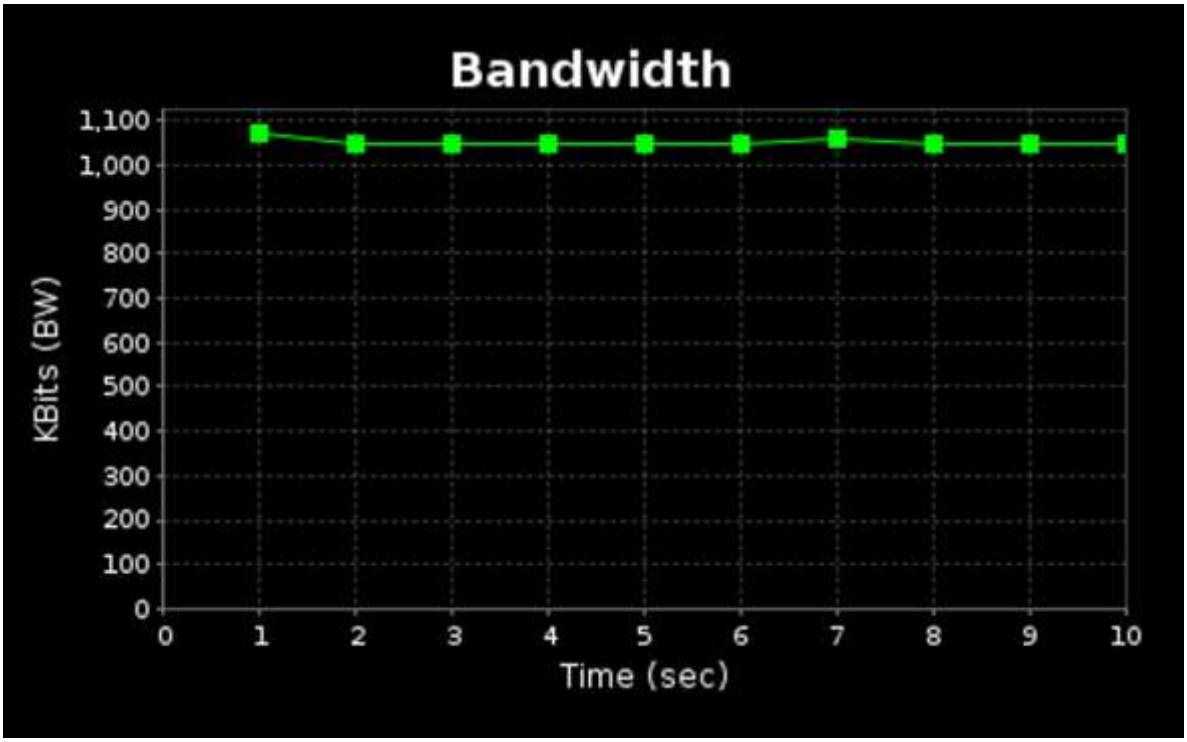


Figure 08: Bandwidth before attack

Performance with the attack (while running on UDP)

Figure 09: CPU while attack

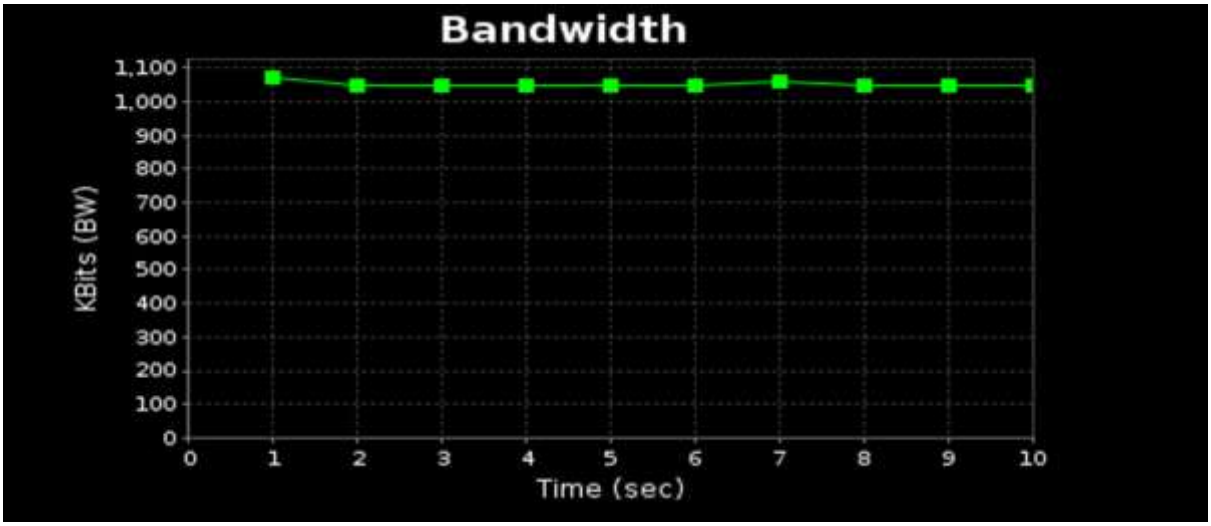


Figure 10: Bandwidth while attack

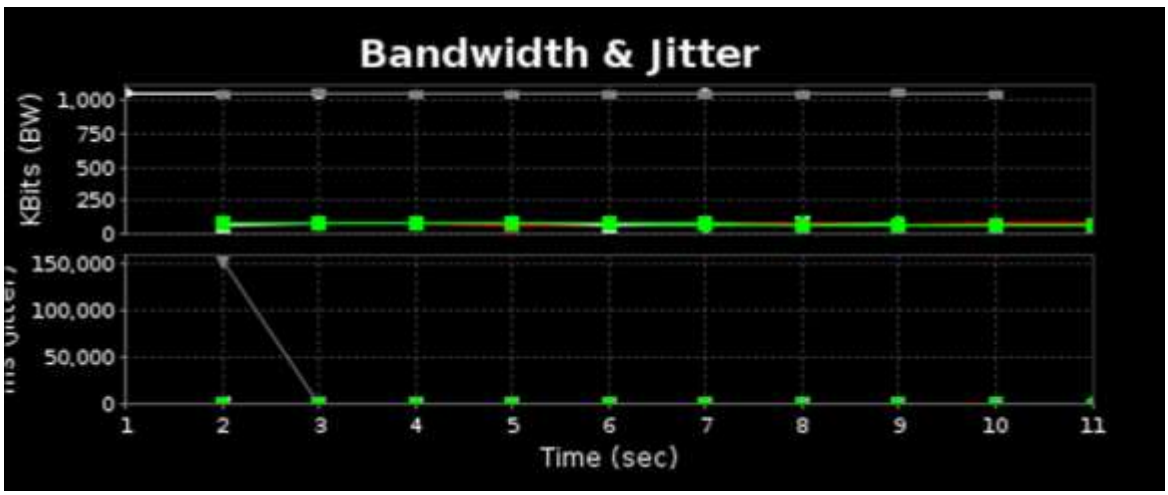


Figure 11: Bandwidth and Jitter

Comparative differences

- Also data transmission through TCP occurs exclusively between connected sending and receiving systems. The specified data transmission rate becomes possible because of these features yet the additional overhead reduces usable bandwidth for data.

- Entity with the information from a previous section verifies that TCP maintains rigid bandwidth restrictions. Under DoS attack scenarios the TCP encounters performance difficulties because its error correction and connection management systems cannot handle the massive number of unauthorized traffic volumes. The legitimate traffic flow area will have a smaller location to travel through as a result of this condition.

- Broadcasting functions best with UDP owing to its connection-less nature among the networking protocols. UDP maintains faster operations and requires lower bandwidth consumption by refraining from creating connections and error correction attempts. The transport protocol that works best for applications requiring minimal latency can be found in UDP since it serves online gaming and video streaming and other related applications. The bandwidth consumption of TCP remains lower than the efficiency of UDP when both protocols are compared.

- The lack of connection maintenance in UDP allows the protocol to manage elevated traffic better than TCP does during DDoS attacks even though UDP bandwidth suffers some impact. However, such environments experience more packet loss and integrity problems when error correction is absent.

CPU Utilization

- The implementation of error-checking together with connection-management and flow-control strategies in TCP networking results in substantial CPU utilization. System CPU power consumption rises high when testing TCP bandwidth because many computational operations need to be done to sustain connection stability. The CPU usage rises to 11 due to the controller and hosts' effort to regulate malicious traffic during a DDoS attack while simultaneously managing rising connection loads. UDP demands less computational power from CPUs because it operates without requiring connections for its operations.

- The reduced CPU power usage of UDP communication exists because the protocol does not need ongoing connection maintenance or complete error monitoring like TCP does. The server CPU utilization becomes elevated even if a DDoS attack seems less damaging than a TCP attack because the system must process additional packets. The systems run less complex processes because UDP does not perform connection maintenance and error checks but this approach exposes them to risks of both data misplacement and decreased data validity.