

COMP 7005

Applied Computer Science, Network Security Applications Development Option

Jan 2024

This is a pairs assignment. You must have the instructor's permission not to work in a pair.

Objective

- Understand and implement reliability mechanisms over an unreliable protocol (UDP) akin to TCP's.
- To explore the challenges and solutions in ensuring data integrity and delivery in network communication.

Learning Outcomes

- Gain practical experience in socket programming with UDP.
- Understand the principles behind TCP's reliability and how to implement similar mechanisms in UDP.
- Develop skills in creating robust network applications that can handle network unreliability.
- Learn to use command-line arguments for dynamic application configuration.
- Analyze and interpret network behaviour through data visualization.

Details

Components

Client

- Sends data to the server via the proxy.
- Must handle acknowledgments and timeouts.

Server

- Receives data from the client, processes it, and possibly sends a response.
- It must also handle duplicate and out-of-order packets.

Proxy

- Simulates network unreliability.
- Introduces packet loss and delays according to specified probabilities and ranges.

- You must provide options for:
 - % chance to drop packets coming from the client
 - % chance to drop packets coming from the server
 - % chance to drop delay coming from the client
 - % chance to drop delay coming from the server
 - Range (min/max) milliseconds to delay if a packet is delayed coming from the client
 - Range (min/max) milliseconds to delay if a packet is delayed coming from the server

Functionality

- Implement reliability features such as packet acknowledgment, retransmission on timeout, and sequence numbering to handle duplicates and ordering.
- Use command-line arguments to configure all parameters (e.g., IP addresses, ports, packet drop probability, delay range, etc.).
- Bonus: real-time graph.

Visualization

- Generate graphs to visualize packets sent, received, retransmitted packets, etc., for both the client and server.

Documentation

Design Document

- Describe the architecture, components, and algorithms used to ensure UDP reliability.

Testing Document

- Detail the testing methodology, scenarios, and outcomes.
- Include how the proxy's behaviour was varied, how the system was observed, and how it responded to different challenges.

Constraints

- Follow the [guidelines](#).
- Ensure all components are robust against network unreliability scenarios.
- Dynamic configuration through command-line arguments is mandatory; hardcoding of parameters is not allowed.
- **Both students must participate equally in all aspects of the project, or you may fail.**

Resources

- Provide references to UDP and TCP documentation, socket programming tutorials, and any libraries or tools used for graph generation.

Submission

- Follow the assignment submission [requirements](#).
- Be sure you are aware of the [late submission policy](#).

Note: *Please strictly adhere to the submission requirements to ensure you don't lose any marks.*

Evaluation

Topic	Value
Implementation Correctness	20%
Handling of Network Reliability	25%
Dynamic Configuration	5%
Visualization & Analysis	10%
Design Document	20%
Testing Document	20%
Total	100

Hints

- Start with basic UDP client-server communication.
- Incrementally add reliability features, testing each thoroughly before moving on.
- Use external libraries or tools for graph generation to save time and focus on the core assignment objectives.
- Consider the design carefully to ensure the system can be easily adjusted and tested under different conditions.
- You can use a spreadsheet to graph the data, it does not have to be done programmatically.