CSC 450 - Computer Networks

Dr. Majd Tahat

Group Project

TCP Congestion Control: Tahoe vs. Reno

Project Overview

Your group will implement and analyze two TCP congestion control algorithms: TCP Tahoe and TCP Reno. You will start by converting a provided pcap file into a CSV format, detecting congestion events and simulating how each algorithm responds to these events. The objective is to compare the performance of TCP Tahoe and TCP Reno in managing network congestion and document your findings in a comprehensive report.

Project Tasks

- **Convert pcap to CSV:** Extract relevant features from the given pcap file and convert it into a CSV format suitable for analysis.
- **Implement TCP Tahoe and TCP Reno Algorithms:** Simulate their responses to congestion events detected from the CSV data.
- **Analyze Network Traffic:** Use the converted CSV file and identify congestion events objectively.
- **Apply and Compare:** Generate separate response CSV files for both TCP Tahoe and TCP Reno and compare their performance metrics.
- **Report Findings:** Document your methodology, results, and insights in a detailed report, including graphs and performance comparisons.

Project Sequence and Deliverables

1. Setup and Preparation

- Understand TCP Tahoe and TCP Reno congestion control mechanisms.
- Familiarize yourself with pcap file analysis and CSV conversion tools.

2. pcap to CSV Conversion

- Use tools such as Wireshark or Python libraries like PyShark to analyze the provided pcap file.
- Extract relevant TCP features such as ACK numbers, sequence numbers, and timestamps, and convert this data into a CSV format.

3. Congestion Detection

- Analyze the converted CSV file to detect congestion events, such as triple duplicate ACKs, TCP Window size, or retransmissions that signify packet loss and network congestion.

4. Algorithm Implementation

- TCP Tahoe & TCP Reno: Simulate TCP Tahoe's and TCP Reno's response to congestion individually.

5. Generate Response CSV Files

- Alter copies of the original CSV files for TCP Tahoe and TCP Reno showing how each algorithm responds to detected congestion events.
- Only alter rows corresponding to congestion events from the original CSV file; the rest of the CSV should remain unchanged.

6. Performance Comparison

- Compare the performance of TCP Tahoe and TCP Reno using the following metrics:
    - **Throughput:** Measure how efficiently data is transmitted under each algorithm.
    - **Recovery Time:** Analyze the time taken to recover from congestion.
    - **Congestion Window Behavior:** Examine the dynamics of `cwnd` during congestion and recovery phases.

7. Reporting and Presentation

- Compile your findings into a report that includes:
    - Introduction and objectives of the project.
    - Methodology for pcap to CSV conversion, congestion detection, and algorithm implementation.
    - Analysis results with graphs comparing Tahoe and Reno.
    - Discussion on which algorithm performed better and why.
- Prepare a brief presentation summarizing your project, results, and insights.

Assessment

Criterion	Description	Weight
Methodology	Evaluate the completeness of your project's implementation.	25%
Results & Analysis	Focuses on the accuracy and the depth of your analysis.	25%
Report Quality	Assesses the clarity, organization, and completeness of your report.	25%
Presentation	Evaluates your ability to effectively present your project findings.	25%

Submission Guidelines

- Submit the final report as a PDF document along with the TCP Tahoe and TCP Reno CSV files and the code used to generate them, and the original CSV.
- Include your source code in a separate folder within your submission.

Academic Integrity

- All group members are expected to contribute equitably. Plagiarism or unauthorized collaboration will result in disciplinary action as per the university's academic integrity policy. All submissions will be checked *(Including the code)* with the AI detection tool; if any are detected, all the group members will automatically get a zero for the project

----