

Оценка параметров качества сервиса (QoS) сетевого соединения

Необходимо разработать программу на C++ или Python, которая будет оценивать параметры качества сервиса (QoS) сетевого соединения, такие как задержка (latency), потери пакетов (packet loss) и скорость (throughput). Программа должна использовать сетевые утилиты для измерения параметров и проводить расширенный анализ статистики, включая медиану, процентиля и визуализацию данных с помощью гистограмм. Для тестирования программы необходимо использовать эмулятор сети Mininet.

Требования к базовой части (5 баллов):

1. Программа должна измерять следующие параметры QoS каждую секунду:
 - **Задержка (latency):** можно использовать утилиту ping для измерения задержки;
 - **Потери пакетов (packet loss):** можно использовать утилиту ping для измерения потерь пакетов;
 - **Скорость (throughput):** можно использовать утилиту iperf для измерения скорости транспортного потока.

Программа должна автоматизировать вызов соответствующих утилит, разбирать их вывод и сохранять результаты для дальнейшего анализа.

2. Для каждого параметра QoS программа должна провести анализ и вывести таблицу со следующими столбцами:

Параметр QoS, Среднее значение, Медиана, 95-й процентиль, Минимальное значение, Максимальное значение, Стандартное отклонение.

Программа должна уметь выводить таблицу на экран и сохранять в файл (например, CSV или JSON).

3. Для каждого параметра QoS программа должна строить гистограммы. Для этого рекомендуется использовать библиотеку matplotlib или другую подходящую библиотеку для визуализации.
4. Исследование работы программы необходимо провести при помощи эмулятора сети Mininet. Создайте топологию сети в Mininet, состоящую из двух хостов (h1, h2) и одного коммутатора (s1). Настройте параметры QoS для каждого канала: задержка 10 мс, потери пакетов 0,5%, пропускная способность 10 Мбит/с. Запустите программу на эмулируемой сети в течение одной минуты и соберите данные. Сравните результаты измерений с заданными параметрами сети, настроенными в Mininet, и оцените, насколько измеренные значения соответствуют ожидаемым. Результаты сравнения, таблицу с анализом данных из пункта 2 и гистограммы необходимо составить в отдельный отчет в формате pdf, который необходимо будет выслать с программой.

Дополнительные задания:

1. **Анализ трафика с помощью tcpdump (2 балла):**

Используйте утилиту tcpdump для захвата сетевого трафика во время передачи данных. Проанализируйте захваченный трафик для оценки количества повторных передач (retransmissions), например, подсчитывая пакеты с флагом [R]. Визуализируйте результаты анализа в виде график количества повторных передач во времени.

2. Многопоточность (1 балл):

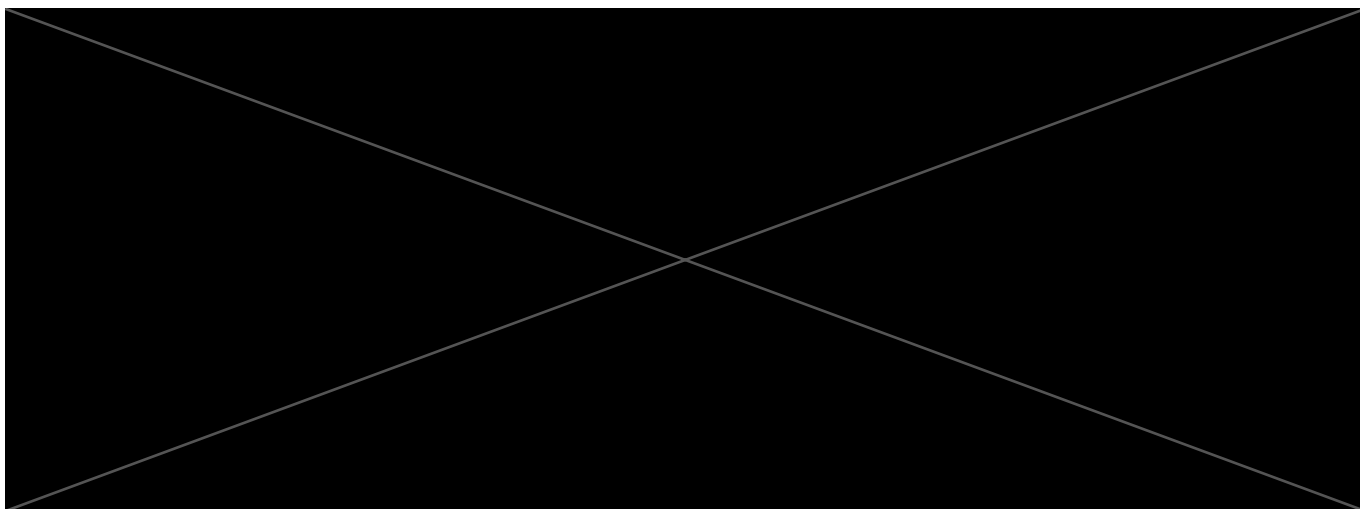
Реализуйте многопоточность для одновременного измерения нескольких параметров QoS: один поток измеряет задержку, второй поток измеряет скорость, третий поток отслеживает потери пакетов. В рамках потока осуществляется не только запуск соответствующей утилиты, но и разбор выходных данных с их анализом.

3. Анализ влияния параметров сети с использованием сетки значений (2 балла):

Проведите эксперименты, изменяя параметры каналов (считать, что все каналы предоставляют одинаковое качество сервиса) в Mininet по следующей сетке (пропускную способность оставить равной 10 Мбит/с):

- **Задержка (latency):** 2 мс, 4 мс, 6 мс, 8 мс, 10 мс, 12 мс, 14 мс, 16 мс, 18 мс, 20 мс.
- **Потери пакетов (packet loss):** 0,2%, 0,4%, 0,6%, 0,8%, 1%, 1,2%, 1,4%, 1,6%, 1,8%, 2%.

Для каждой комбинации параметров запустите программу и соберите данные. Постройте график зависимости скорости от задержки и потери пакетов (например, построив тепловую карту). График включить в отчет.



Полезные ссылки:

1. Mininet:

Официальная документация: <http://mininet.org/docs/>

Руководство по использованию: <http://mininet.org/walkthrough/>

2. Утилита ping:

Документация: man ping (в Linux)

3. Утилита iperf:

Официальный сайт: iperf.fr

Документация: man iperf (в Linux) или <https://iperf.fr/iperf-doc.php>

4. **Утилита tcpdump:**

Документация: `man tcpdump` (в Linux)

или <https://www.tcpdump.org/manpages/tcpdump.1.html>

5. **Matplotlib (визуализация данных):**

Официальная документация: <https://matplotlib.org/stable/contents.html>