

Транспортный протокол QUIC как способ надежной передачи трафика реального времени¹

В. М. Антонова^{***}, И. Г. Бужин^{***}, Н. А. Гречишкина^{**}, Е. Д. Корочкин^{*},
Н. А. Кузнецов^{**}

^{*}Московский технический университет связи и информатики, Москва, Россия;

^{**}Институт радиотехники и электроники (ИРЭ) им. В. А. Котельникова РАН, Москва, Россия;

^{***}Национальный исследовательский центр "Курчатовский институт", Москва, Россия;

Поступила в редколлегию 29.07.2025 г. Принята 10.10.2025 г.

Аннотация—В статье проведен анализ методов обеспечения надежности передачи трафика реального времени на транспортном уровне сети связи. Рассмотрены традиционные подходы, реализованные в протоколах TCP и UDP, а также современное решение — протокол QUIC. Показано, что QUIC сочетает низкие задержки UDP с надежностью TCP и дополнительными преимуществами, такими как встроенное шифрование и многопоточность. Приведены результаты сравнительного анализа производительности QUIC и TCP, демонстрирующие преимущества нового протокола в различных сетевых условиях.

КЛЮЧЕВЫЕ СЛОВА: протоколы, quic, tcp, udp, транспортный уровень OSI.

DOI: 10.53921/18195822_2025_25_3_490

1. ВВЕДЕНИЕ

Современные сети связи активно используются для передачи трафика реального времени, включая видеоконференцсвязь (ВКС), потоковое видео, онлайн-игры и другие интерактивные приложения. Эти виды трафика критичны к задержкам и потерям пакетов, что требует эффективных механизмов обеспечения надежности передачи данных.

Традиционно транспортный уровень сети представлен двумя основными протоколами: UDP (User Datagram Protocol) и TCP (Transmission Control Protocol). UDP обеспечивает минимальную задержку, но не гарантирует доставку пакетов, что приводит к возможным потерям и ухудшению качества обслуживания. TCP, в свою очередь, предлагает надежную передачу данных за счет механизмов управления потоком и повторной передачи, но его высокая задержка и проблемы с блокировкой последовательности пакетов делают его неэффективным для приложений реального времени.

Одним из наиболее значимых решений, появившихся в последние годы, является QUIC (Quick UDP Internet Connections) — транспортный протокол, разработанный компанией Google. Он сочетает в себе преимущества UDP (низкие задержки) с надежностью и управлением потоком, аналогичными TCP. Благодаря встроенному шифрованию, многопоточности и механизму 0-RTT установления соединения, QUIC демонстрирует высокую эффективность в передаче данных в реальном времени.

Целью данного исследования является анализ протокола QUIC как инструмента обеспечения надежности передачи трафика реального времени по UDP. В рамках исследования рассматриваются следующие аспекты:

¹ Антоновой В. М., Бужиным И. Г. работа выполнена в НИЦ "Курчатовский Институт" в рамках государственного задания по теме FNEF-2024-0014, Гречишкиной Н. А., Кузнецовым Н. А. работа выполнена в ИРЭ РАН в рамках государственного задания по теме FFZW-2025-0015

- Ограничения и проблемы транспортного уровня при передаче трафика реального времени;
- Существующие методы обеспечения надежности в UDP и TCP;
- Архитектура и механизмы работы протокола QUIC (детальный анализ);
- Сравнительный анализ QUIC с традиционными транспортными протоколами (TCP и UDP);
- Перспективы применения QUIC для различных типов реального трафика.

Протокол QUIC является относительно новым стандартом, и его применение в транспортных системах требует детального изучения. В отличие от традиционных решений, он предлагает гибридный подход к обеспечению надежности передачи данных. Исследование его характеристик позволит:

- Повысить качество передачи потокового контента и ВКС;
- Разработать новые стратегии управления перегрузкой и восстановлением потерь;
- Улучшить маршрутизацию трафика в условиях ограниченных ресурсов.

2. ОБЗОР ПРОБЛЕМ ТРАНСПОРТНОГО УРОВНЯ ПРИ ПЕРЕДАЧЕ ТРАФИКА РЕАЛЬНОГО ВРЕМЕНИ

Для понимания проблем, возникающих при передаче трафика реального времени, необходимо рассмотреть основы модели OSI (Open Systems Interconnection) и стека TCP/IP. Эти концепции определяют уровневую структуру взаимодействия сетевых устройств.

Модель OSI включает семь уровней: физический, канальный, сетевой, транспортный, сеансовый, представления и прикладной. Каждый уровень выполняет специфические функции, обеспечивающие надежную передачу данных.

Стек TCP/IP более практичен и состоит из четырех уровней: сетевой доступ, интернет, транспортный и прикладной. Он является основой для большинства современных сетевых технологий, включая интернет. На рисунке 1 представлено соответствие уровней модели OSI и стека TCP/IP.

Трафик реального времени (например, видеоконференцсвязь, потоковое видео, онлайн-игры) требует низких задержек, минимальных потерь пакетов и стабильности передачи. Основные проблемы, с которыми сталкиваются такие приложения:

- Задержки и джиттер: вызваны изменяющимся временем прохождения пакетов по сети.
- Потери пакетов: в UDP отсутствуют механизмы повторной передачи, что приводит к артефактам в видео- и аудиопотоках.
- Перегрузка сети: при нехватке пропускной способности могут происходить буферизация, разрывы соединений.
- Проблемы работы через NAT и файрволлы: UDP-потоки могут блокироваться, так как не содержат явного установления соединения.
- Отсутствие надежных механизмов управления потоком в UDP: TCP использует контроль потока и перегрузки, чего нет в стандартном UDP.

TCP обеспечивает:

- Гарантированную доставку данных через механизмы подтверждений (ACK) и повторных передач (retransmission).
- Контроль перегрузки (алгоритмы AIMD, CUBIC и др.).
- Управление потоком (регулируемое окно передачи).

Однако, у TCP есть недостатки для трафика реального времени, например задержки из-за установления соединения (3-way handshake). Помимо этого существует head-of-line blocking:

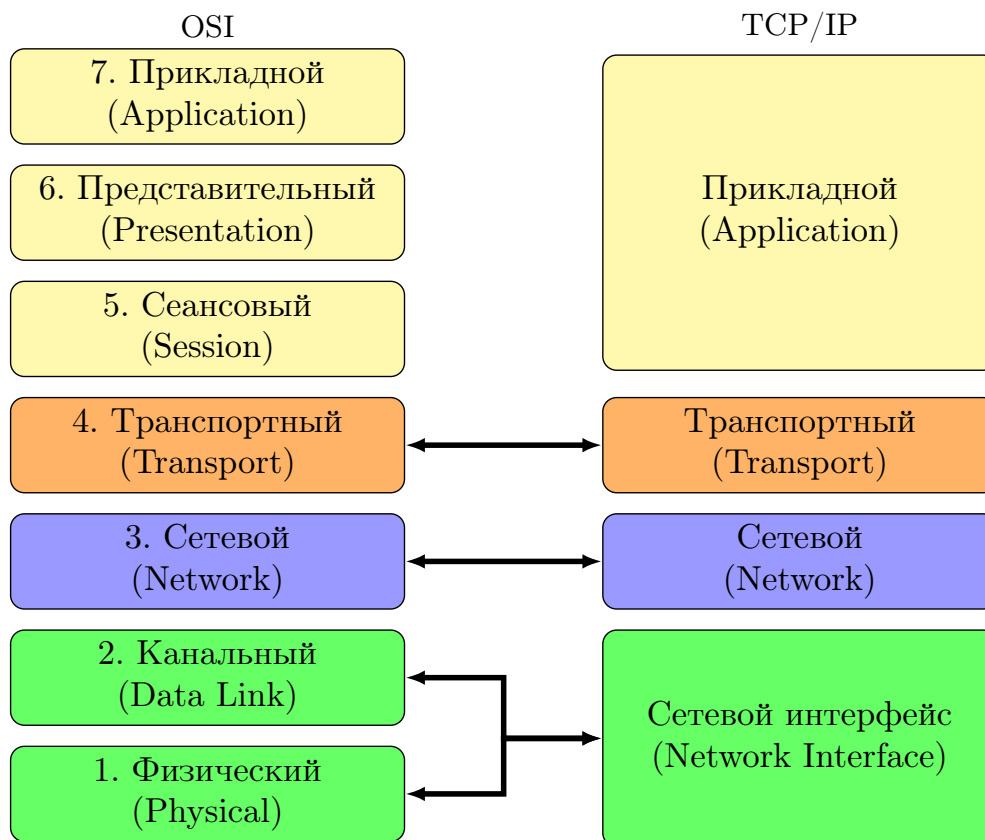


Рис. 1. Соответствие стеков протоколов OSI и TCP/IP

если один пакет теряется, все последующие задерживаются. Также, TCP свойственна избыточная коррекция потерь, даже если потери вызваны не перегрузкой, а сетевыми особенностями (например, Wi-Fi).

UDP широко применяется в потоковой передаче из-за низких задержек, но у него нет встроенной гарантии доставки пакетов. Это приводит к проблемам с качеством обслуживания (QoS), особенно при высокой загрузке сети.

Некоторые протоколы попытались объединить преимущества TCP и UDP:

- DCCP (Datagram Congestion Control Protocol) добавляет контроль перегрузки к UDP, но достаточно сложен в реализации.
- SCTP (Stream Control Transmission Protocol) — многопоточный транспортный протокол с надежностью TCP, не получивший массового распространения.
- MPTCP (Multipath TCP) позволяет использовать несколько маршрутов для передачи данных, однако требует поддержки на стороне сети.

Проблемы надежности трафика реального времени обусловлены отсутствием в UDP механизмов контроля доставки, а использование TCP ведет к увеличению задержек. Альтернативные решения, такие как DCCP и SCTP, не получили широкого распространения. В связи с этим возникла необходимость в новом транспортном протоколе, объединяющем надежность TCP и скорость UDP — таким решением стал QUIC.

3. QUIC: АЛЬТЕРНАТИВА TCP ДЛЯ НАДЕЖНОСТИ UDP

Протокол QUIC (Quick UDP Internet Connections) был разработан Google как альтернатива TCP с целью обеспечения высокой скорости и надежности передачи данных. Он использует UDP, но добавляет функции, характерные для TCP, включая контроль потерь, управление потоком и установление соединения. QUIC стал основой для HTTP/3 и уже используется в таких сервисах, как Google, YouTube и Facebook.

Основные преимущества QUIC:

- Минимизация задержек за счет быстрого установления соединения (0-RTT, 1-RTT).
- Надежность доставки благодаря встроенным механизмам повторной передачи утраченных пакетов.
- Многопоточность, исключая head-of-line blocking.
- Шифрование по умолчанию (TLS 1.3) для защиты передаваемых данных.
- Гибкое управление перегрузкой на основе современных алгоритмов.

QUIC устраняет трехэтапное рукопожатие TCP (3-way handshake), используя 0-RTT (нулевая задержка) и 1-RTT (одноразовая задержка) механизмы. Это значительно ускоряет соединение по сравнению с TCP (рисунок 2).

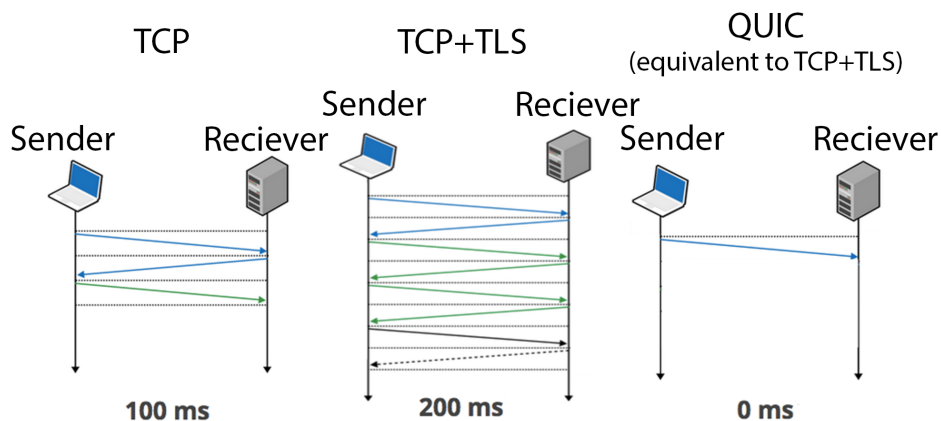


Рис. 2. Сравнение времени установления соединения для различных сетевых протоколов

Несмотря на то, что QUIC работает на UDP, он включает механизм подтверждений (ACKs) и повторных передач и фрейминг данных, позволяющий передавать несколько потоков одновременно. Многопоточность, исключает блокировку передачи данных при потере одного пакета.

Помимо значительного перераспределения сетевого трафика с протокола TCP на UDP, внедрение QUIC характеризуется обязательным использованием шифрования. В отличие от других протоколов, QUIC не поддерживает незащищённые соединения. Для установления ключей сессии QUIC использует протокол TLS 1.3, который также обеспечивает последующее шифрование каждого передаваемого пакета. Однако, в силу того что QUIC базируется на протоколе UDP, ряд данных о сессии и метаданных, которые в TCP обычно передаются в открытом виде, в QUIC подвергаются шифрованию. Структуры заголовков TCP, UDP и QUIC приведены на рисунке 3.

Сравнение основных характеристик TCP, UDP и QUIC приведено в Таблице 1.

На текущий момент QUIC уже активно применяется в повседневной жизни, например в:

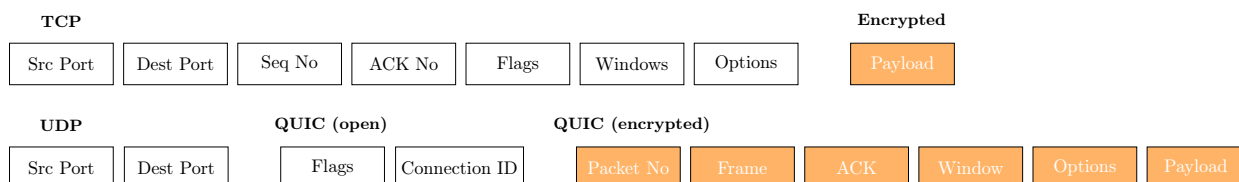


Рис. 3. Параметры для оптимизации виртуализации

Таблица 1.

Характеристика	UDP	TCP	QUIC
Надежность	Нет	Да	Да
Задержка	Минимальная	Высокая	Низкая
Установление соединения	Не требуется	Долгое (3-way handshake)	Быстрое (0-RTT, 1-RTT)
Поддержка многопоточности	Нет	Нет	Да
Защита от потерь	Нет	Да	Да
Работа через NAT	Проблемы	Да	Да

- Google, YouTube, Facebook для быстрого загрузки контента.
- HTTP/3, который основан на QUIC.
- VPN и облачные сервисы для повышения безопасности и скорости передачи данных.
- Игровых и VoIP-приложениях, где критична минимальная задержка.

HTTP/3 полностью отказался от TCP и перешел на QUIC, так как протокол HTTP/2 из-за особенностей TCP сталкивался с проблемой head-of-line blocking, когда потеря одного пакета задерживала всю передачу данных в соединении. Переход на QUIC позволил избежать head-of-line blocking благодаря независимой передаче потоков данных, сократить задержки за счет 0-RTT установления соединения и уменьшить накладные расходы на обработку соединений, так как QUIC управляет потерями на уровне пакетов, а не соединений. Помимо этого, переход обеспечил встроенную безопасность с использованием обязательного шифрования TLS 1.3.

HTTP/3 уже поддерживается современными веб-браузерами, сетевой инфраструктурой (Cloudflare, Google, Facebook) и серверными технологиями (NGINX, Apache с поддержкой QUIC).

QUIC сочетает гибкость UDP и надежность TCP, обеспечивая высокую скорость, безопасность и минимальные задержки. Это делает его перспективным транспортным протоколом для трафика реального времени, особенно в условиях перегруженных и ненадежных сетей.

4. СРАВНЕНИЕ QUIC И TCP

В рамках данной работы были произведены собственные тесты, анализирующие производительность TCP и QUIC с точки зрения:

- Общего времени передачи данных
- Средней пропускной способности (bandwidth)
- Накладные расходы на обработку (overhead in bytes)

Был организован лабораторный стенд в среде виртуализации, запущенной на ЭВМ типа ПК со следующими основными техническими характеристиками:

- ЦП: AMD Ryzen 7 7900X (для ВМ выделено 8 потоков, ограниченных 3500 МГц)
- ОЗУ: DDR5 16GB 6400 МГц

- SSD: Kingston KC3000
- Программная среда:
 - Arch Linux (репозитории ПО CachyOS x86_64-v3) x86_64, Linux 6.12.6
 - GCC 5.4
 - Python 2.7.12

Для работы с протоколом QUIC используется библиотека `aioquic`, которая предоставляет асинхронную поддержку QUIC. Для работы с TCP используется стандартная библиотека Python (`asyncio`), обеспечивающая асинхронные сетевые операции.

Эксперимент проводится в локальной виртуальной сети. Для настройки локального интерфейса обратной связи (`loopback interface`) используются `tc netem` и `tbpf`.

Файл размером 33 МБ сгенерирован в директории `/var/www/html/`, и передан с серверов QUIC и Apache2, используя клиенты QUIC и `wget`.

Принцип работы тестового стенда приведён на рисунке 4.

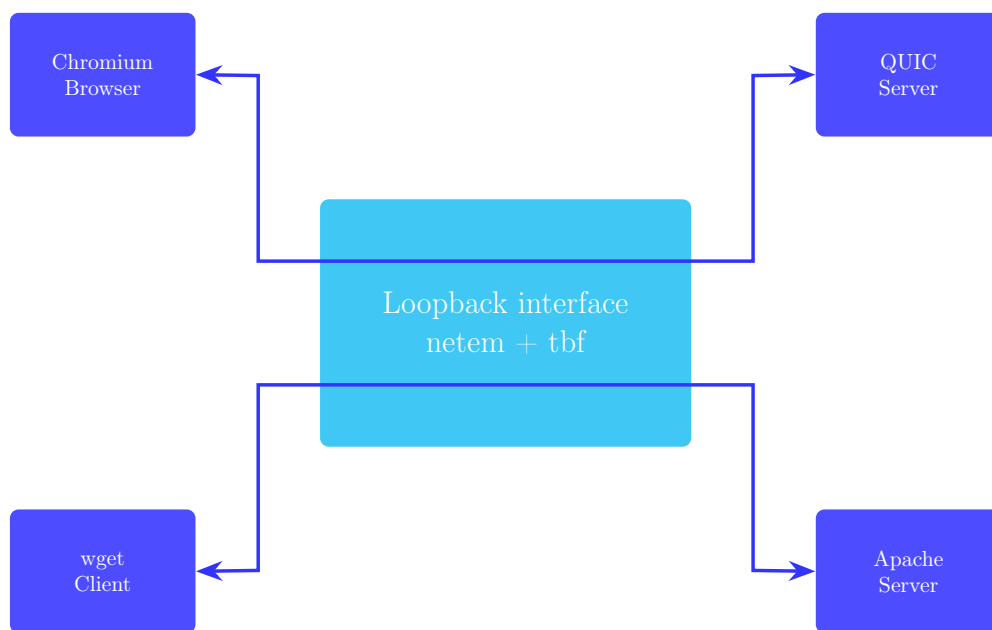


Рис. 4. Схема тестового стенда

Графики зависимостей, построенные на основе полученных данных, приведены на рисунках 5–16.

Из графиков следует, что несмотря на более высокие накладные расходы, QUIC превосходит TCP по времени передачи данных и средней используемой пропускной способности. Особенно заметно это преимущество проявляется при высоких задержках, потерях пакетов и большой пропускной способности сети, где QUIC демонстрирует значительно лучшие результаты как по времени передачи, так и по производительности. В благоприятных условиях QUIC также показывает большую стабильность по сравнению с TCP, что видно из графиков в разделе временных рядов. При наличии потерь пакетов QUIC также опережает TCP: при отсутствии потерь его производительность существенно выше, а при 5% потерь, хотя показатели протоколов становятся близки, QUIC все же остается лидером. Однако при возникновении джиттера TCP может превзойти QUIC, поскольку данный протокол пока недостаточно развит для эффективной работы в таких условиях.

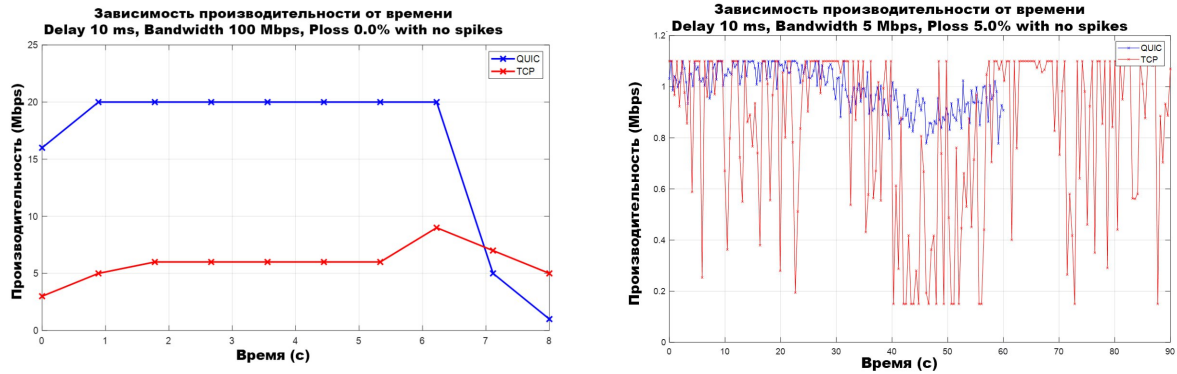


Рис. 5. Зависимость производительности от времени с разной пропускной способностью

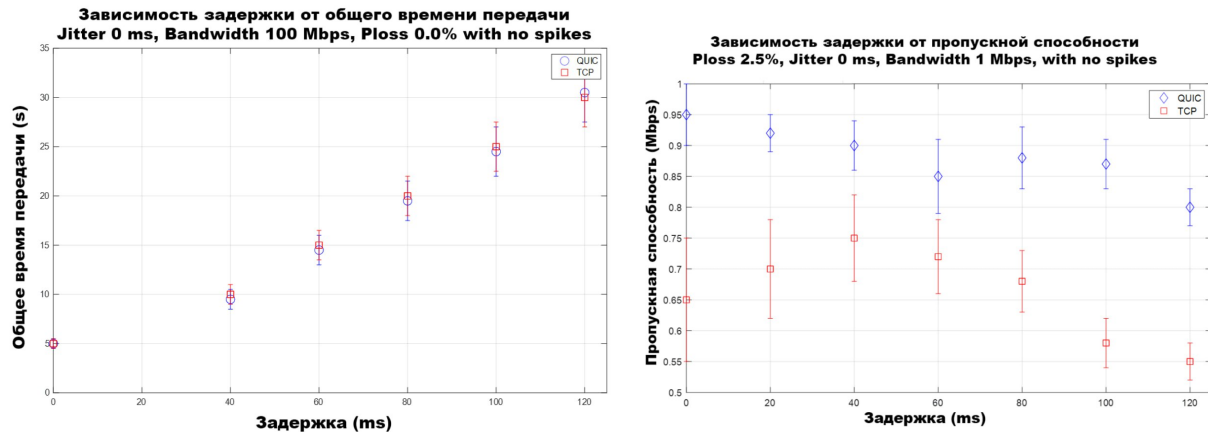


Рис. 6. Зависимость задержки от времени и пропускной способности

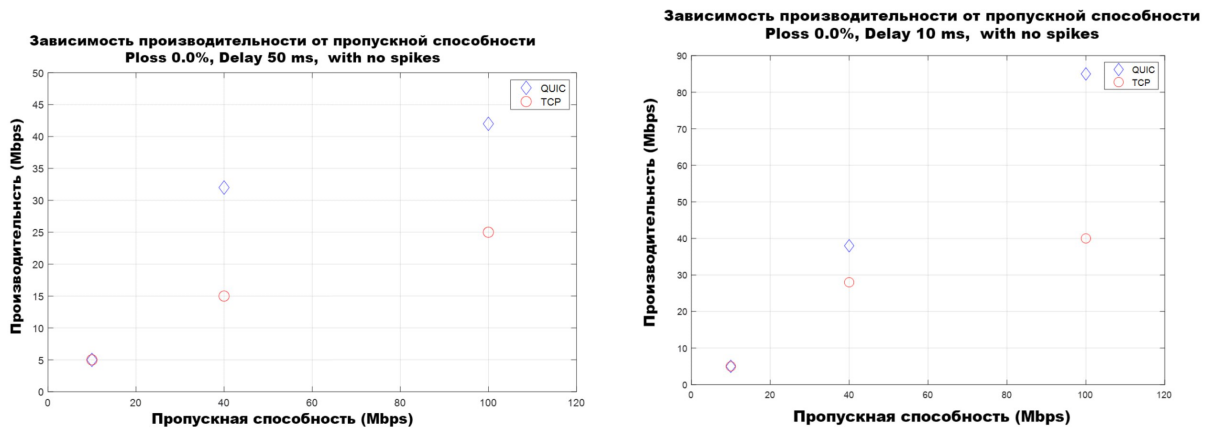


Рис. 7. Зависимость производительности от пропускной способности при разной задержке

5. ПЕРСПЕКТИВЫ РАЗВИТИЯ ПРОТОКОЛА QUIC

Протокол QUIC является одним из наиболее значимых достижений в области транспортных протоколов за последние десятилетия. Его преимущества перед TCP и UDP, включая

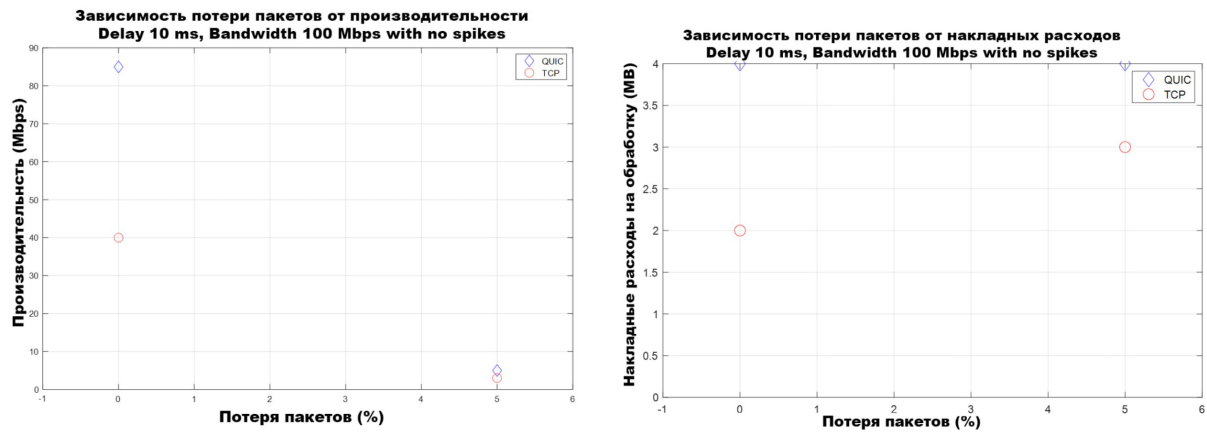


Рис. 8. Зависимость потери пакетов от накладных расходов и производительности

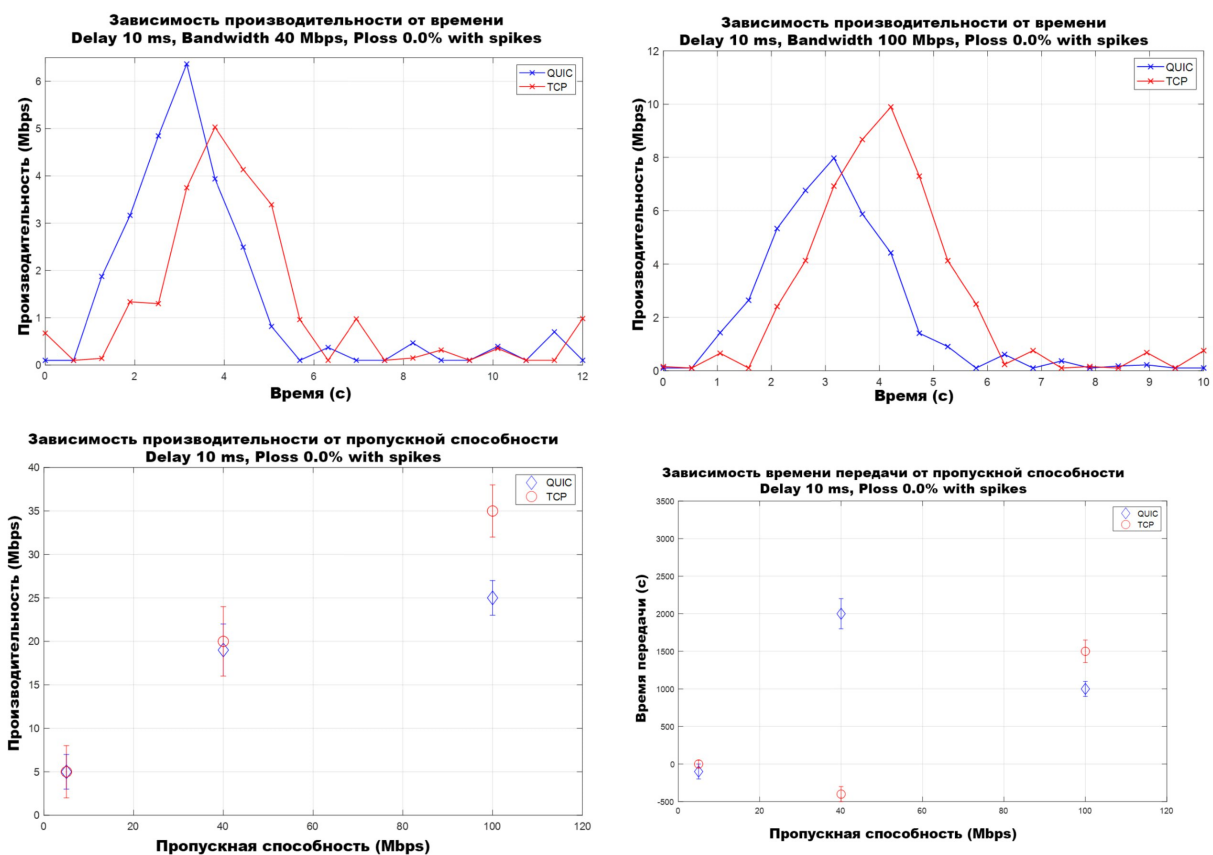


Рис. 9. Зависимость характеристик с учётом джиттера

минимизацию задержек, встроенное шифрование, многопоточность и адаптивный контроль перегрузки, делают его объектом активных исследований. Современные научные работы посвящены оптимизации алгоритмов работы QUIC, адаптации к новым условиям сетей связи и интеграции с перспективными технологиями, такими как 5G и IoT.

Один из ключевых аспектов дальнейшего развития QUIC — совершенствование алгоритмов контроля перегрузки. Современные механизмы, такие как BBR (Bottleneck Bandwidth and

RTT), демонстрируют значительное улучшение по сравнению с традиционными алгоритмами TCP (CUBIC, Reno), но требуют дальнейшей адаптации для сложных сетевых сред, таких как мобильные сети 5G и 6G и сети с высоким уровнем потерь, включая спутниковые и беспроводные сети, где необходимы более эффективные стратегии повторных передач.

Многопутевая передача данных является перспективным направлением, которое уже реализовано в Multipath TCP (MPTCP). Включение многопутевой маршрутизации в QUIC позволит:

- Увеличить устойчивость соединений за счет использования нескольких маршрутов передачи данных.
- Оптимизировать балансировку нагрузки в современных сетях с высокой изменчивостью пропускной способности.
- Улучшить пользовательский опыт в мобильных средах, обеспечивая плавный переход между Wi-Fi и сотовыми сетями без разрыва соединения.

Промышленные сети, высоконагруженные облачные платформы и системы управления требуют детерминированных характеристик задержек и надежности. Исследования в данной области направлены на:

- Адаптацию QUIC для низкозадержных сетей (Ultra-Low Latency Networking, ULLN).
- Оптимизацию работы в распределенных облачных средах, включая поддержку междатационных соединений.
- Интеграцию с архитектурами SDN (Software-Defined Networking) для динамического управления трафиком.

В контексте использования QUIC совместно с TLS 1.3 для шифрования трафика особое внимание уделяется адаптации протокола к криптографическим стандартам постквантовой эпохи. Основное направление исследований включает разработку квантово-устойчивых алгоритмов шифрования, которые могут быть эффективно интегрированы в QUIC. Помимо этого, значительные усилия направлены на оптимизацию процесса шифрования с целью минимизации накладных расходов, что особенно важно для поддержания высокой производительности протокола. Кроме того, активно изучаются механизмы защиты метаданных, обеспечивающие предотвращение анализа трафика и повышающие уровень конфиденциальности пользовательских данных.

6. ЗАКЛЮЧЕНИЕ

В данной работе был проведен анализ проблем надежности передачи трафика реального времени на транспортном уровне сети, рассмотрены традиционные решения, основанные на протоколах TCP и UDP, а также исследован перспективный подход, реализованный в протоколе QUIC. Применение UDP для трафика реального времени позволяет минимизировать задержки, но не гарантирует надежность доставки, в то время как TCP обеспечивает контроль целостности данных, но приводит к увеличению задержек и снижению эффективности в условиях нестабильных сетей. QUIC представляет собой современное решение, объединяющее достоинства обоих подходов, обеспечивая низкие задержки, встроенную надежность передачи и высокий уровень безопасности.

Научные перспективы в области транспортных протоколов включают дальнейшее совершенствование механизмов управления перегрузками, особенно в условиях высокоскоростных сетей нового поколения, таких как 5G и 6G. Современные исследования направлены на адаптацию QUIC и аналогичных технологий к многопутевым системам передачи, которые позволят

более эффективно распределять нагрузку между несколькими каналами связи, снижая вероятность перегрузок и потерь пакетов. Одним из ключевых направлений является интеграция квантово-устойчивых криптографических алгоритмов, что позволит обеспечить долгосрочную безопасность транспортных протоколов перед новыми угрозами кибербезопасности.

Развитие сетевых технологий также связано с внедрением QUIC и подобных решений в промышленные сети и системы интернета вещей (IoT). Для этого требуется разработка легковесных версий протоколов, способных работать в условиях ограниченных вычислительных ресурсов и низкой пропускной способности. Кроме того, перспективным направлением является улучшение механизмов адаптации протоколов к динамически изменяющимся условиям сети, что особенно важно для беспроводных сред передачи данных и мобильных устройств.

Проведенный анализ показывает, что протокол QUIC представляет собой значительный шаг вперед в обеспечении надежности передачи трафика реального времени на транспортном уровне. Однако, несмотря на многочисленные преимущества QUIC перед традиционными протоколами TCP и UDP, в данной области остаются открытые вопросы и перспективы для дальнейших исследований:

- Совершенствование механизмов управления перегрузками для работы в сложных сетевых средах.
- Оптимизацию работы протокола в условиях джиттера и нестабильного соединения.
- Разработку эффективных алгоритмов многопутевой передачи данных.
- Адаптацию протокола для работы в сетях с ограниченными ресурсами и IoT-устройствах.
- Интеграцию квантово-устойчивых криптографических алгоритмов.

Таким образом, современные транспортные технологии, включая QUIC, демонстрируют значительный прогресс в повышении надежности и эффективности передачи данных. Однако дальнейшее развитие требует решения ряда научных и технических задач, направленных на повышение адаптивности, безопасности и устойчивости к перегрузкам. Исследования в этой области будут способствовать созданию новых стандартов транспортных протоколов, способных обеспечить высокий уровень качества сервиса для всех типов трафика, включая критически важные системы реального времени.

СПИСОК ЛИТЕРАТУРЫ

1. Tanenbaum A. S., Wetherall D. *Computer Networks* (5th Edition). — Pearson Education, 2011.
2. Stallings W. *Data and Computer Communications* (10th Edition). — Pearson, 2013.
3. RFC 9000 — QUIC: A UDP-Based Multiplexed and Secure Transport. IETF, May 2021.
4. RFC 8999 — Version-Independent Properties of QUIC. IETF, May 2021.
5. RFC 9001 — Using TLS to Secure QUIC. IETF, May 2021.
6. Langley A. et al. *The QUIC Transport Protocol: Design and Internet-Scale Deployment*. In: *Proceedings of the Conference of the ACM Special Interest Group on Data Communication (SIGCOMM)*, 2017.
7. Roskind J. *QUIC design document*, Google, 2012.
8. Kakhki A. M. et al. Taking a Long Look at QUIC: An Approach for Rigorous Evaluation of Rapidly Evolving Transport Protocols. In: *Proceedings of the 2017 ACM Internet Measurement Conference (IMC)*.
9. Yu H., Zhao J., Peng Y. *Performance Comparison of QUIC and TCP for HTTP/3*. In: *IEEE Access*, 2021.
10. Смирнов С. Н., Петухов В. И. *Обеспечение надежности транспортного уровня в сетях нового поколения* // Вестник связи, 2020.

11. Ключев С. В. *Методы повышения отказоустойчивости TCP/IP* // Вестник СибГУТИ, 2019.
12. Cisco Systems. *Quality of Service Design Overview*. Cisco Press, 2020.
13. Куракин Д. С. *Сравнительный анализ надёжности TCP, UDP и QUIC* // Современные технологии, 2022.
14. Google Developers. *QUIC and HTTP/3 Explained* — <https://developers.google.com>
15. Yassir F., Raji M. *Impact of QUIC in Real-Time Applications: A Comparative Study with TCP* // *International Journal of Computer Applications*, 2022.

QUIC transport protocol as reliable transmission method for real-time traffic

V. M. Antonova, I. G. Buzhin, N. A. Grechishkina, E. D. Korochkin, N. A. Kuznetsov

This article analyzes methods for ensuring reliable real-time traffic transmission at the transport layer of a communication network. Traditional approaches implemented in the TCP and UDP protocols are considered, as well as a modern solution – the QUIC protocol. It is shown that QUIC combines the low latency of UDP with the reliability of TCP and additional advantages such as built-in encryption and multithreading. The results of a comparative performance analysis of QUIC and TCP are presented, demonstrating the advantages of the new protocol in various network conditions.

KEYWORDS: protocols, quic, tcp, udp, OSI transport layer