

**Ольховик Никита Сергеевич,**

магистрант, МФ МГТУ имени Н.Э. Баумана, г. Мытищи

e-mail: [JLoGuka@yandex.ru](mailto:JLoGuka@yandex.ru)

## **ИССЛЕДОВАНИЕ ВОЗМОЖНОСТЕЙ УСКОРЕНИЯ ПЕРЕДАЧИ ДАНЫХ МЕЖДУ ДВУМЯ УЗЛАМИ СЕТИ НА БОЛЬШИЕ РАССТОЯНИЯ**

**Аннотация.** В статье рассмотрено тестирование альтернатив устоявшимся решениям протоколов, таких как Quiche [3] – реализация QUIC стандарта RFC 9000 [6] от Cloudflare, Warp Data Transfer (WDT) в реализации от Meta [8], а также TCP стандарта RFC 9293 [4], который достаточно давно применяется на практике. Несмотря на основу в качестве протоколов UDP для QUIC и TCP для WDT, последние призваны улучшить характеристики передачи данных по сравнению с протоколами, на которых они базированы.

Цель работы - произвести анализ и сравнение скорости передачи данных по указанным протоколам в контролируемой локальной сети с применением различных искусственно созданных условий (ограничений) для имитации работы сети WAN, а также проанализировать реакцию алгоритмов протоколов на изменяющиеся условия сети, сделать выводы на основе полученных результатов. Основной метод исследования – сравнение.

**Annotation.** The paper examines the utilization of alternative protocols to established solutions, including Quiche (a QUIC implementation of RFC 9000 [6] from Cloudflare) [3], Warp Data Transfer (WDT) from Meta [8], and the TCP standard RFC 9293 [4], which has been widely adopted. Despite QUIC being based on UDP and WDT being based on TCP, both protocols are designed to enhance data transfer characteristics compared to their respective protocols. The objective of the research is to analyze and compare the data transfer rates of the specified protocols in a controlled local network. Various artificial conditions (restrictions) are employed to simulate the operation of a WAN network. Additionally, the response

of protocol algorithms to changing network conditions is evaluated. The findings are synthesized into conclusions based on the results obtained. The primary research method employed is comparison.

**Ключевые слова:** TCP, QUIC, WDT, Warp Data Transfer, скорость, передача, RTT, эффективность

**Keywords:** TCP, QUIC, WDT, Warp Data Transfer, speed, transfer, RTT, efficiency

TCP остаётся доминирующим транспортным протоколом, однако его эффективность сильно зависит от выбранного алгоритма управления перегрузкой. Алгоритмы, такие как TCP CUBIC [5], BBR [2], и Hybla [1], демонстрируют различное поведение при высоких RTT и потерях. При этом большинство реализаций TCP в операционных системах по умолчанию используют CUBIC. QUIC — транспортный протокол, разработанный Google и стандартизированный IETF [6]. Он работает поверх UDP и включает встроенное шифрование, мультиплексирование потоков, а также более быстрое установление соединения по сравнению с TCP (0-RTT). QUIC активно используется в HTTP/3 и показывает хорошие результаты при нестабильных сетевых условиях [7]. Протокол WDT, разработанный в Meta (ранее Facebook) [8], предназначен для быстрой передачи больших файлов между серверами в ЦОДах и удалённых локациях. WDT использует параллельные TCP-соединения и может обходить ограничения одного TCP-потока, увеличивая общую пропускную способность при передаче больших объёмов данных.

В работе было создана программа, которая может с высокой степенью автоматизации произвести подключение к указанному в конфиг-файле разработки компьютеру с помощью заранее или автоматически (с одноразовым предоставлением пароля или ручным запуском скрипта разработки) ключей доступа RSA и провести тестирование, состоящее из запуска мониторинга локальной ОС, запуска мониторинга удаленной ОС, проведения тестовой передачи данных, сохранения данных, собранных во

время тестовой передачи, и визуализации этих данных в виде PDF-отчета, содержащего графики, показывающие загрузку сетевого интерфейса и системы во время передачи данных. Графики содержат скорость передачи в пакетах в секунду, количество ошибок и коллизий для каждого тестируемого протокола, размер пакета данных, скорость передачи полезной нагрузки, количество повторных передач. Данная разработка была использована для того, чтобы провести сравнение работы протоколов в различных условиях сети, сделав упор на эффективность массовой передачи данных. Передача данных проводилась в локальной сети с изменением условий передачи данных с помощью программ tc, netem, iptables.

При проведении тестирования использовались следующие тестовые сценарии передачи между двумя компьютерами: локальная (эталонная) сеть без ограничений, установка изменяющегося со временем RTT, установка фиксированного RTT на длительное время с случайной амплитудой не более 2 мс. В каждом случае происходила передача файла, содержащего белый шум, размер файла варьировался от 1 Гб до 15 Гб, а сеть передачи поддерживала скорость передачи до 1 Гбит/с. В случае с изменяющимся RTT, каждому протоколу давалось 20 секунд для установки максимальной скорости передачи, после чего RTT изменялся на -50 мс, причем изменение происходило от 300 до 50 мс. При этом были получены следующие результаты: QUIC имеет высокую скорость реакции на изменяющиеся условия сети – проходит от 400 до 500 мс на установку максимальной скорости после момента изменения RTT; вместе с высокой скоростью показываются высокие потери пакетов, которые при  $RTT \geq 150$  мс могут нивелировать его высокую пропускную способность; WDT показал в среднем в 2 раза более низкую скорость передачи данных чем QUIC при  $RTT \geq 250$  мс, однако фактически рост до максимальной скорости начался при  $RTT \leq 150$  мс против 50 мс у QUIC. WDT единственный из тестируемых смог достичь максимальной скорости полезной нагрузки в сети со скоростью 1 Гбит/с при  $RTT = 150$  мс и удерживать стабильную скорость передачи при этом. На рисунке 1 показана

скорость передачи пакетов в секунду во время одной из таких тестовых передач, где резкое уменьшение пропускной способности до нуля означает конец передачи файла.

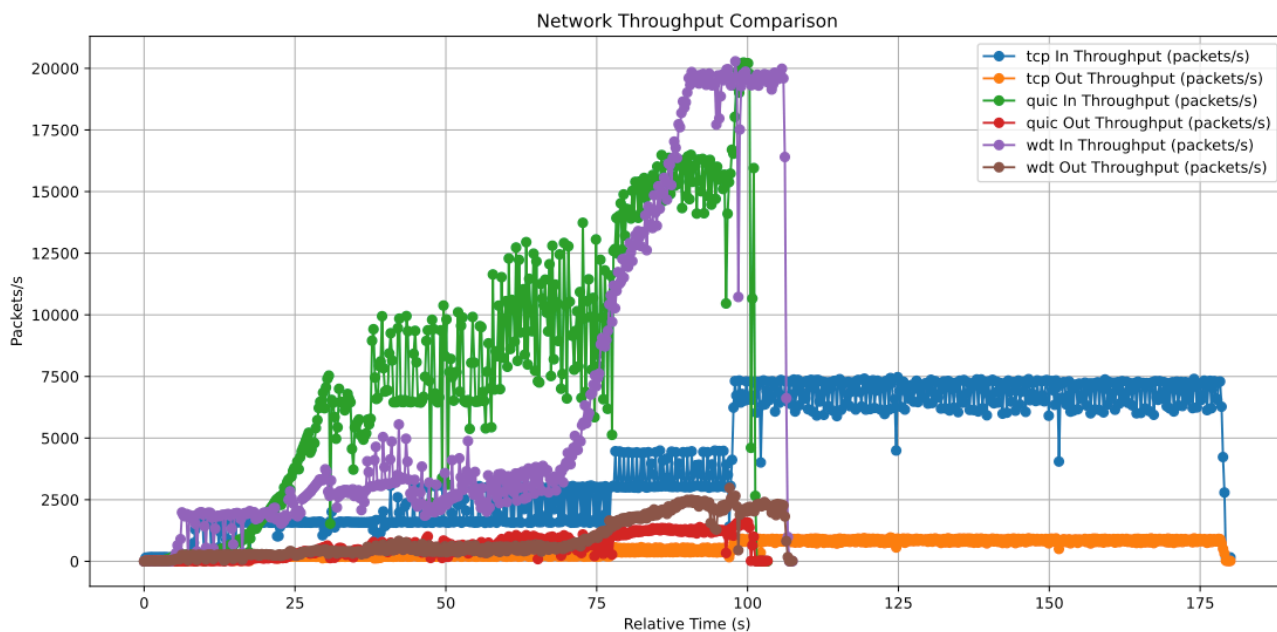


Рисунок 1 – скорость передачи пакетов с изменяющимся RTT

Далее были проведены тесты пропускной способности протоколов TCP в сравнении с WDT при RTT, равном 300 мс для показания возможностей протокола WDT. Тесты показали, что WDT может достигнуть более высокой скорости передачи чем TCP (до 500 Мбит/с в пике) при RTT равном 300 мс и амплитуде в 20 мс, однако на это понадобится достаточно много времени (в среднем по проведенным тестам – 105-120 секунд), более того, большое количество повторных передач пакетов нивелирует реальную полезную нагрузку на этот период. На рисунке 2 показана пропускная способность для полезной нагрузки за период тестовой передачи, на рисунке 3 показано количество повторных передач пакетов за тот же период.

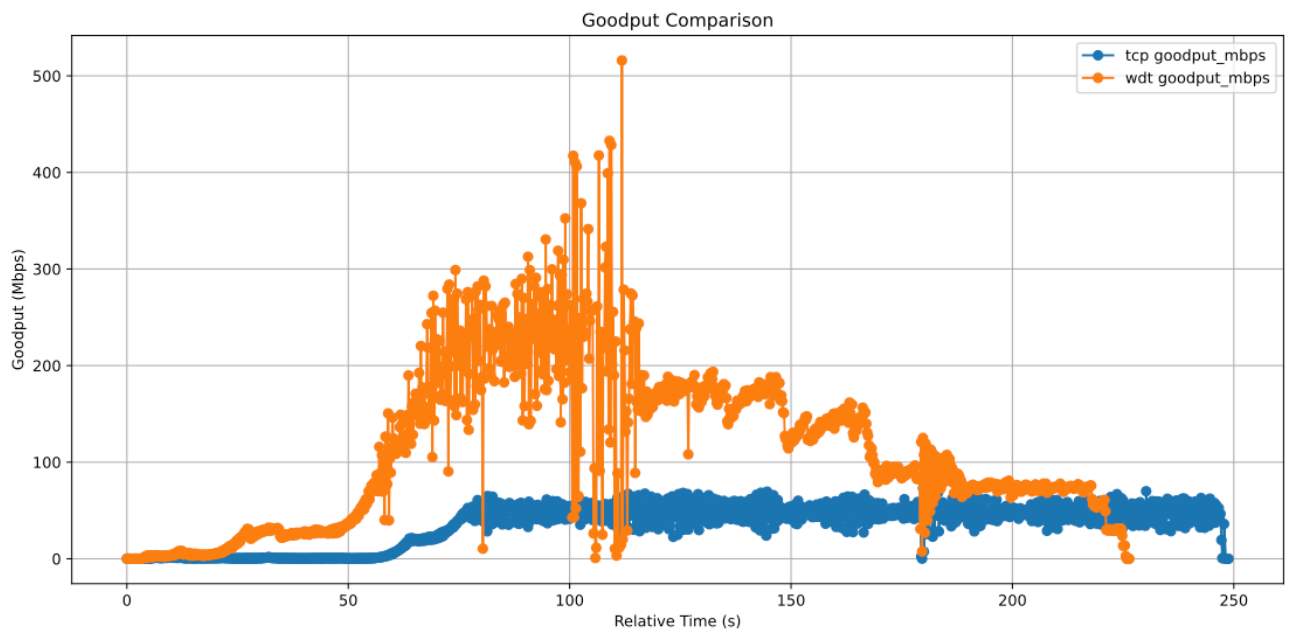


Рисунок 2 - Размер полезной нагрузки при RTT=300мс, Мбит/с

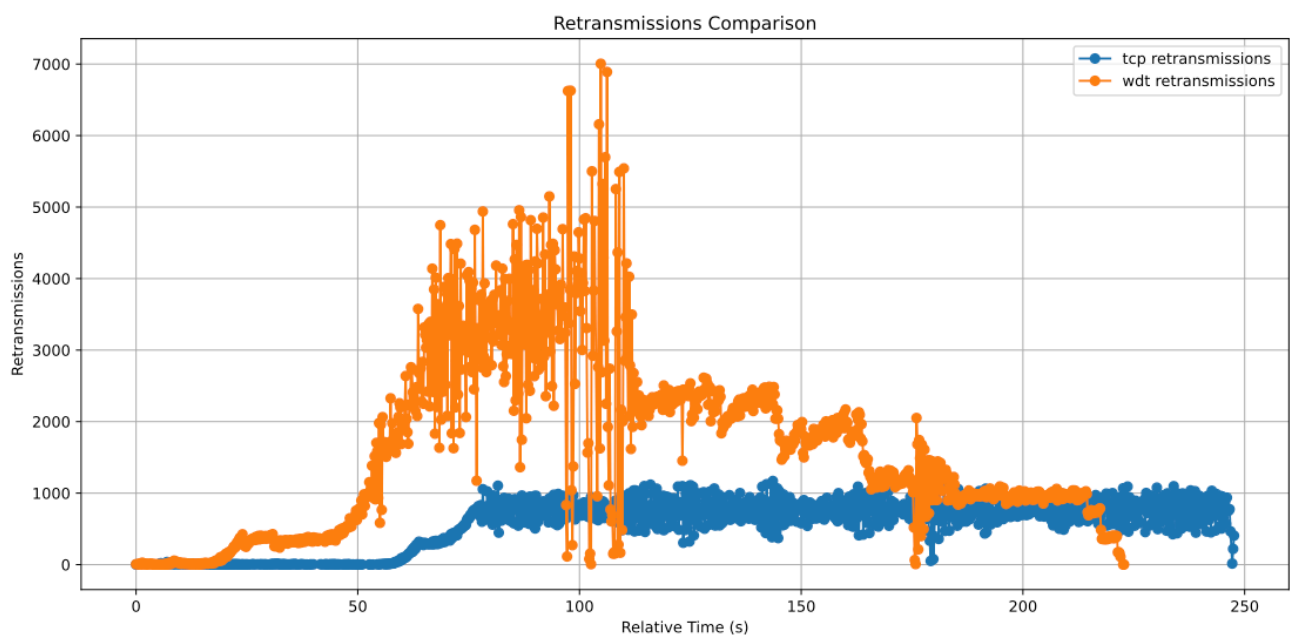


Рисунок 3 - Количество повторных передач пакетов, RTT=300мс

Далее, для рассмотрения эффективности протоколов с точки зрения нагрузки на CPU, были проведены тесты всех протоколов на максимальной скорости передачи данных, в котором стабильной скорости передачи на протяжении всего теста в 940 Мбит/с достигли все протоколы. По итогу тестирования было рассчитано среднее соотношение скорости передачи на 1% загрузки ядра. В итоге расчетов – самым эффективным оказался WDT

(использовался только один поток передачи), причем TCP показал большее отношение чем QUIC. Результаты расчетов показаны в таблице 1.

Таблица 1 – отношение скорости передачи к загрузке ядра

TCP, скорость передачи на 1% загрузки ядра, (Мбит/с)	WDT, скорость передачи на 1% загрузки ядра, (Мбит/с)	QUIC, скорость передачи на 1% загрузки ядра, (Мбит/с)
20.336	69,1345	11,328

**Заключение.** Среди протоколов в тестах самым эффективным для массовой передачи данных, как с точки зрения загрузки CPU, так и с точки зрения времени достижения максимальной скорости передачи, оказался Warp Data Transfer, с точки зрения нагрузки на CPU результат не менялся в том числе при использовании только одного ядра. Проведенная работа позволяет обратить внимание на эффективное решение для передачи данных и при необходимости, дополнить разработанный инструмент тестирования новыми реализациями протоколов для дальнейшего анализа.

### Список литературы

1. Caini, Carlo, and Renato Firrincieli. "TCP Hybla: a TCP enhancement for heterogeneous networks." *International Journal of Satellite Communications and Networking* 22.5 (2004): 547-566. URL: <https://doi.org/10.1002/sat.799> – дата обращения 09.05.2025.

2. Cardwell, Neal, et al. “BBR: Congestion-based congestion control.” *Communications of the ACM* 60.2 (2017): 58-66. URL: <https://queue.acm.org/detail.cfm?id=3022184>.
3. Cloudflare. Quiche: Fast and secure Rust implementation of the QUIC protocol / Cloudflare. – Режим доступа: <https://github.com/cloudflare/quiche>.
4. Eddy W. Transmission Control Protocol (TCP): RFC 9293 / Internet Engineering Task Force. – 2022. – URL: <https://datatracker.ietf.org/doc/html/rfc9293>.
5. Ha, Sangtae, et al. “CUBIC: A new TCP-friendly high-speed TCP variant.”
6. Iyengar J., Thomson M. QUIC: A UDP-Based Multiplexed and Secure Transport: RFC 9000 / Internet Engineering Task Force. – 2021. – URL: <https://datatracker.ietf.org/doc/html/rfc9000>.
7. Langley, Adam, et al. “The QUIC transport protocol: Design and Internet-scale deployment.” *Proceedings of the Conference of the ACM Special Interest Group on Data Communication* (2017): 183–196. [Электронный ресурс] URL: <https://doi.org/10.1145/3098822.3098842> – дата обращения 09.05.2025.
8. Warp Data Transfer Protocol: Fast and reliable data transfer tool / Meta Platforms, Inc. – GitHub. – URL: <https://github.com/facebook/wdt>.

## References

1. Caini, Carlo, and Renato Firrincieli. “TCP Hybla: a TCP enhancement for heterogeneous networks.” *International Journal of Satellite Communications and Networking* 22.5 (2004): 547-566. URL: <https://doi.org/10.1002/sat.799>.

2. Cardwell, Neal, et al. “BBR: Congestion-based congestion control.” *Communications of the ACM* 60.2 (2017): 58-66. URL: <https://queue.acm.org/detail.cfm?id=3022184>.
3. Cloudflare. Quiche: Fast and secure Rust implementation of the QUIC protocol / Cloudflare. – Режим доступа: <https://github.com/cloudflare/quiche>.
4. Eddy W. Transmission Control Protocol (TCP): RFC 9293 / Internet Engineering Task Force. – 2022. – URL: <https://datatracker.ietf.org/doc/html/rfc9293>.
5. Ha, Sangtae, et al. “CUBIC: A new TCP-friendly high-speed TCP variant.”
6. Iyengar J., Thomson M. QUIC: A UDP-Based Multiplexed and Secure Transport: RFC 9000 / Internet Engineering Task Force. – 2021. – URL: <https://datatracker.ietf.org/doc/html/rfc9000>.
7. Langley, Adam, et al. “The QUIC transport protocol: Design and Internet-scale deployment.” *Proceedings of the Conference of the ACM Special Interest Group on Data Communication* (2017): 183–196. URL: <https://doi.org/10.1145/3098822.3098842>.
8. Warp Data Transfer Protocol: Fast and reliable data transfer tool / Meta Platforms, Inc. – GitHub. – URL: <https://github.com/facebook/wdt>.