



Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Московский государственный технический университет
имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИУ «Информатика и системы управления»

КАФЕДРА ИУ-7 «Программное обеспечение ЭВМ и информационные технологии»

РАСЧЕТНО-ПОЯСНИТЕЛЬНАЯ ЗАПИСКА К НАУЧНО-ИССЛЕДОВАТЕЛЬСКОЙ РАБОТЕ НА ТЕМУ:

«Анализ и классификация алгоритмов организации
ретрансляции видеопотоков»

Студент группы ИУ7-76Б

(Подпись, дата)

Мансуров В. М.

(Фамилия И.О.)

Руководитель

(Подпись, дата)

(Фамилия И.О.)

2023 г.

РЕФЕРАТ

Расчетно-пояснительная записка 23 с., 5 рис., 14 лист., 19 ист., 1 прил.

СОДЕРЖАНИЕ

РЕФЕРАТ	3
ОПРЕДЕЛЕНИЯ	5
ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ	6
ВВЕДЕНИЕ	7
1 Аналитическая часть	8
1.1 Ретрансляция видеопотоков	8
1.2 Организация ретрансляции видеопотоков	8
1.3 Модель взаимодействия открытых систем	9
1.4 Обзор существующих решений	9
1.4.1 TCP	9
1.4.2 UDP	9
1.4.3 RTP	10
1.4.4 RTCP	11
1.4.5 RTSP	13
1.4.6 RTMP	15
1.4.7 HLS	16
1.4.8 MPEG-DASH	18
1.4.9 MPTCP	19
ЗАКЛЮЧЕНИЕ	20
СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ	21
ПРИЛОЖЕНИЕ А	23

ОПРЕДЕЛЕНИЯ

Трансляция — непрерывная передача аудио или видеоданных с источника на платформу для дальнейшего распространения.

Ретрансляция — повторная передача видеоданных без изменения его содержания.

Видеопоток — последовательность видеоданных, которая передается по компьютерным сетям или другим каналам связи [1].

Протокол — набор соглашений логического уровня.

Потоковая передача данных (передача потоков) — способ передачи, при котором транспортировка и воспроизведение мультимедийных данных на удаленном компьютере осуществляются в режиме реального времени [1].

Пакет — это блок данных, содержащий информацию, необходимую для доставки [2].

Задержка —

Джиттер —

Клиент RTSP — устройство, которое запрашивает медиапоток данных с сервера RTSP [3].

Соединение — канал транспортного уровня, установленный между двумя программами с целью обмена данными [3].

Сеанс — механизм передачи непрерывного медиапотока или запуск потока с помощью данных управления [3].

Битрейт — количество бит, используемых для передачи или обработки данных в единицу времени [12].

ОБОЗНАЧЕНИЯ И СОКРАЩЕНИЯ

В текущей расчетно-пояснительной записке применяются следующие сокращения и обозначения.

ISO — International Standards Organization, международной организацией по стандартизации.

OSI — Open Systems Interconnect, взаимодействия открытых систем.

TCP — Transmission Control Protocol, протокол управления передачей.

UDP — User Datagram Protocol, протокол пользовательских дейтаграмм.

PAR — Positive Acknowledgment with Retransmission.

RTP — Real-time Transport Protocol, транспортный протокол реального времени.

RTCP — Real-Time Transport Control Protocol, протокол управления передачей в реальном времени.

RTSP — Real-Time Streaming Protocol, потоковый протокол реального времени.

RTMP — Real-Time Messaging Protocol, протокол обмена сообщениями в реальном времени.

HTTP — HyperText Transfer Protocol, протокол передачи гипертекста.

HLS — HTTP Live Streaming,

MPEG-DASH — Dynamic Adaptive Streaming over HTTP, технология адаптивной потоковой передачи данных.

SRT — Secure Reliable Transport.

QoS — Quality of Service, качество обслуживания.

ВВЕДЕНИЕ

TODO: актуальность работы и введение ...

Целью работы является проведение анализа алгоритмов организации ретрансляции видеопотоков.

Чтобы достигнуть поставленной цели, необходимо решить следующие задачи:

- провести анализ предметной области ретрансляции видеопотоков;
- провести обзор существующих алгоритмов организации ретрансляции видеопотоков;
- сформулировать критерии сравнения алгоритмов организации ретрансляции видеопотоков;
- классифицировать существующие алгоритмы организации ретрансляции видеопотоков.

1 Аналитическая часть

1.1 Ретрансляция видеопотоков

Ретрансляция видеопотоков — это повторная передача видеопотоков через компьютерные сети с целью распространения или обработки. Ретрансляция начинается с захвата видеопотока с источника, такого как веб-камера, IP-камера, экран компьютера или другое устройство, которое генерирует видеоданные. Ретрансляция может быть использована для одновременной трансляции видеопотока на несколько устройств, при этом видеопотоки могут подвергаться различным видам обработки: сжатию, кодированию, фильтрации или масштабированию, в зависимости от конкретных потребностей.

1.2 Организация ретрансляции видеопотоков

TODO: добавить задачи и подробностей организации

Для передачи обычно используется один из двух способов.

- Последовательный — видеофайл воспроизводится с жесткого диска компьютера пользователя или сервера провайдера услуг. Как правило, при передаче таким способом качество изображения и звука выше. К недостаткам стоит отнести то, что невозможно переключить ролик с одного момента на другой, не дождавшись его буферизации. То есть, интересующий фрагмент должен быть загружен, чтобы пользователь смог его просмотреть.
- В реальном времени — требует наличия потокового сервера. Этот метод больше подходит для передачи видеофайлов большой длительности. Пользователь может выбрать место, с которого он хочет начать просмотр. Также этот вид потоковой передачи мультимедиа используется для трансляции с веб-камеры или захвата экрана.

1.3 Модель взаимодействия открытых систем

TODO: Описание OSI/ISO

1.4 Обзор существующих решений

Согласно общепринятым нормам взаимодействия открытых систем, ретрансляции видеопотоков регламентируется протоколами транспортного уровня OSI/ISO. Наиболее важными протоколами транспортного уровня являются протоколы TCP и UDP [2].

1.4.1 TCP

TCP — протокол с установлением соединения, обеспечивающий надежную доставку данных со сквозным обнаружением и устранением ошибок. Надежность в TCP обеспечивается благодаря механизму подтверждения приема повторной передачей PAR, т.е. PAR повторяет отправку до тех пор пока не получит подтверждения, что данные успешно получены. Согласно TCP гарантируется доставка данных. Также TCP регулирует скорость отправки в соответствии со своей оценкой пропускной способности сети, потери и задержку, и если они ухудшились, это снизит скорость потока, а если они улучшились, это увеличит скорость потока. В случае потери пакета во время передачи, пакет передается повторно [2, 4, 5].

1.4.2 UDP

UDP — протокол без установления соединения. Поскольку протокол UDP таких проверок не совершает и нет механизма PAR, что обеспечивает более быструю передачу данных, но не предоставляет надежность их передачи. UDP выполняет три действия:

- идентифицирует процесс отправки и получения, используя номера портов;
- запускает проверку на ошибку в заголовке;

— записывает проверку в заголовке.

Поэтому UDP не восстанавливает потерянные пакеты [2, 4, 5].

Таким образом, UDP применяется для передачи видеопотоков в реальном времени, потеря пакета данных не приведет к задержке видео для восстановления пакета, но при этом качество видео будет ухудшаться в зависимости от того, сколько пакетов было потеряно.

Для передачи видеопотока используются протоколы транспортного уровня без установления соединения. Так как UDP не может гарантировать качество видео при потере последовательности пакетов. В сочетании с специальными протоколами, которые преобразуют исходные данные таким образом, что они могут быть переданы в сеть, как непрерывная последовательность, обеспечивается наилучшая потоковую передачу видео, чем TCP. Использование передовых технологий сжатия и буферизации позволяет просматривать потоковый контент с любого места, не дожидаясь его полной загрузки на компьютер пользователя. При этом протоколы формируют полезную нагрузку пакетов, включая необходимые поля и данные [1].

1.4.3 RTP

RTP — транспортный протокол сквозной доставки данных в режиме реального времени, включая видео и аудио.

RTP позволяет компенсировать негативное влияние задержек на качество видео и аудио, но при этом не гарантирует своевременную доставку пакетов, т.е. не обеспечивает QoS, также не предоставляет функции исправления ошибок и управления потоком. RTP совместно используется с UDP используя его функции. RTP предусматривает индикацию типа полезной нагрузки и порядкового номера пакета в потоке, а также применение временных меток. Отправитель помечает каждый RTP-пакет временной меткой, получатель извлекает ее и вычисляет суммарную задержку. Разница в задержке разных пакетов позволяет определить джиттер и смягчить его влияние — все пакеты будут выдаваться приложению с одинаковой задержкой [6, 7, 8],

рис.унок 1.1.

Однако RTP может использоваться с другими подходящими базовыми сетевыми или транспортными протоколами. Также RTP поддерживает передачу данных нескольким адресатам с использованием многоадресной рассылки, если это предусмотрено базовой сетью [7].

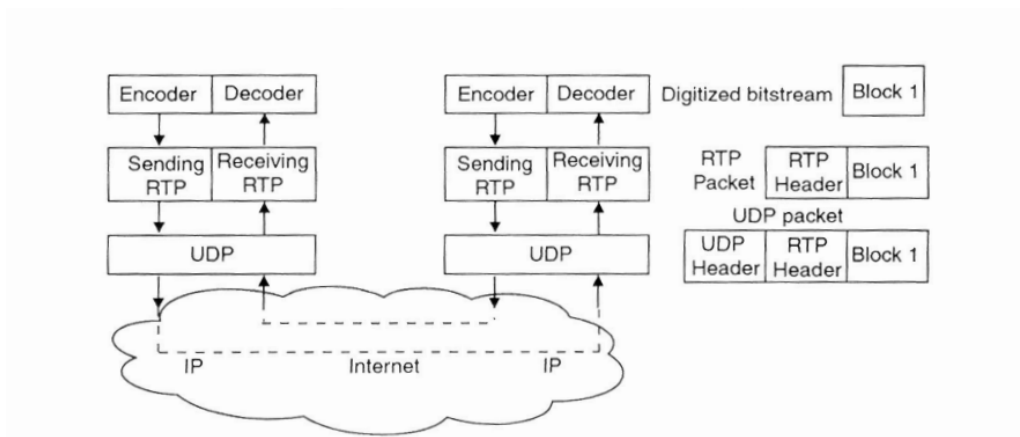


Рисунок 1.1 – Схема алгоритма работы RTP совместно с UDP [9]

На рисунке 1.2 представлена схема заголовка протокола RTP, содержащий ряд полей, которые идентифицируют такие элементы, как формат пакета, порядковый номер, источник информации, границы и тип полезной нагрузки.

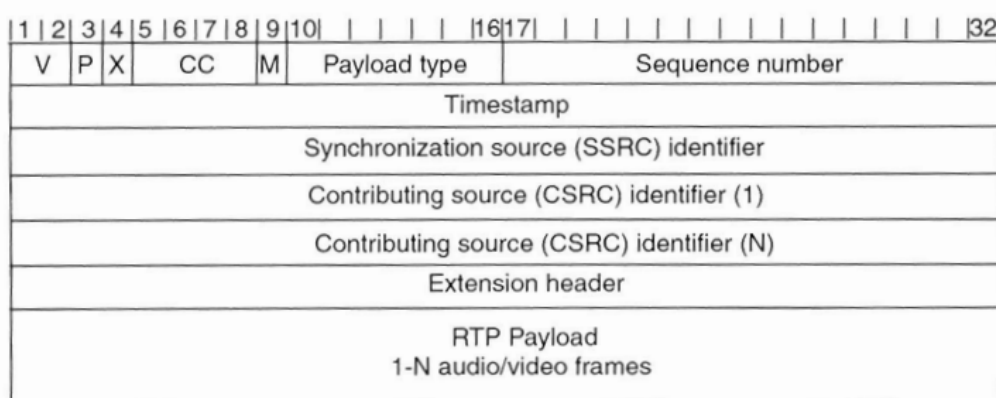


Рисунок 1.2 – Схема заголовка RTP [9]

1.4.4 RTCP

RTSP — протокол управления, созданный для совместной работы с RTP, он помогает осуществлять синхронизацию видео и звука, обеспечивать

QoS, т.е. обратную связь участников сеанса RTP и контроль качества передачи данных. Также RTCP передает сведения о числе переданных и потерянных пакетов, значения джиттера, задержке и т.д.

Базовый протокол должен обеспечивать мультиплексирование пакетов данных и управления, в UDP это обычно реализуется с использованием отдельных номеров портов.

Основными функциями RTCP являются:

- мониторинг качества обслуживания и контроль перегрузки;
- идентификация источника RTP;
- оценка размера сеанса и масштабирование.

Пакеты RTCP содержат прямую информацию для мониторинга качества обслуживания. Отчеты отправителя (SR) и отчеты получателя (RR) обмениваются информацией о потерях пакетов, задержке и джиттере. Эта информация может быть использована для реализации механизма управления потоком, подобного TCP. Средство управления сетью может отслеживать загрузку сети на основе пакетов RTCP без получения фактических данных или обнаруживать неисправные участки сети [7, 9].

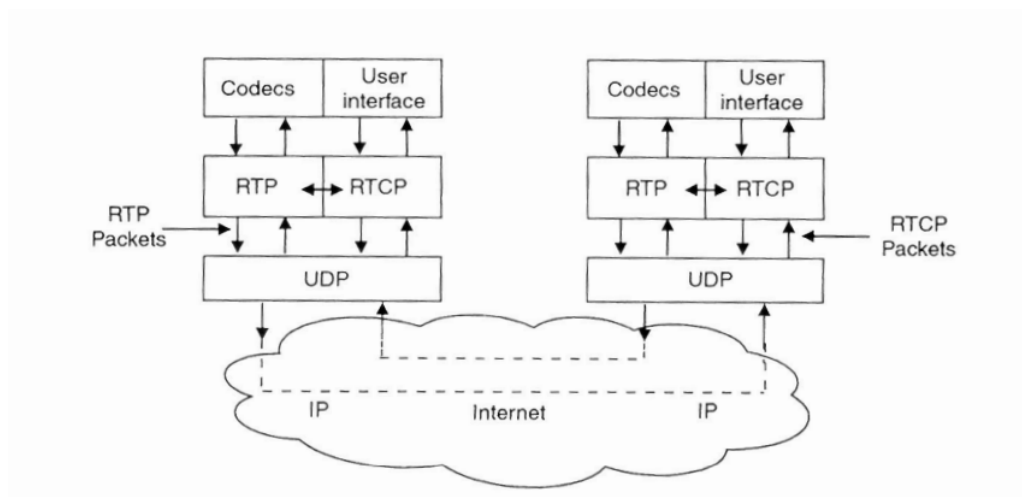


Рисунок 1.3 – Схема алгоритма работы RTCP совместно с RTP и UDP [9]

1.4.5 RTSP

RTSP — протокол управления прикладного уровня для инициализации и направления потоковых данных от видеосервера, реализующий возможности «удаленного управления», который был разработан в 1996 году для управления развлекательными и коммуникационными системами видеотрансляций.

RTSP требует отдельного сервера для приема запросов от нескольких клиентов и потоковой передачи данных, который поддерживает сеанс RTSP постоянного соединения с клиентами RTSP, помеченный идентификатор, при этом во время соединения сеанс не привязывается к использованию транспортного протокола, поскольку он реализует надежность на прикладном уровне. RTSP не навязывает использования определенного формата медиафайлов, Во время сеанса, клиент RTSP может открывать и закрывать множество надежных транспортных подключений к серверу для выдачи запросов RTSP. Таким образом, могут передаваться несколькими различными способами [3]:

- постоянные транспортные соединения, используемые для нескольких транзакций запрос-ответ;
- одно подключение на транзакцию запроса-ответа;
- режим без установления соединения.

RTSP управляет потоком, который может быть отправлен по отдельному протоколу, независимому от канала управления. Так RTSP использует TCP для передачи и приема команд управления, в то время как данные передаются по UDP. При этом RTSP использует RTP и RTCP для передачи потоков в реальном времени, что в свою очередь обеспечивает воспроизведение RTSP-потока во время загрузки потока и получение потока продолжается если сервер не получает запросов, что называется сегментированной потоковой передачей, рисунок 1.4. Кроме того, в течение всего срока службы один медиапоток может управляться запросами RTSP, выдаваемыми последова-

тельно по разным TCP-соединениям, поэтому сервис поддерживать «состояние сеанса», чтобы сопоставлять запросы с потоком [3].

RTSP не конкретизируется на формате передачи данных, включая их кодировки, поэтому позволяет клиенту выбрать подходящую комбинацию видеоданных. Каждый поток идентифицируется по URL-адресу RTSP, который указывает на сервер обрабатывающий этот поток. Помимо параметров видео, необходимо определять сетевой адрес назначения и порт, что выделяет несколько режимов работы [3]:

- многоадресная рассылка и сервер выбирает адрес;
- многоадресная рассылка и клиент выбирает адрес;
- одноадресная рассылка.

Основные методы протокола, который идентифицируется по URI запроса:

- DESCRIBE — запрос описания содержимого, например, в формате SDP;
- OPTIONS — запрос поддерживаемых методов;
- PLAY — запрос начала вещания содержимого;
- PAUSE — запрос временной остановки вещания;
- RECORD — запрос на записывание содержимого сервером;
- REDIRECT — запрос на перенаправление на другое содержимое;
- SETUP — запрос установки транспортного механизма для содержимого;
- ANNOUNCE — запрос на обновление данных описания содержимого;
- TEARDOWN — запрос на остановку потока и освобождение ресурсов.

ц

По функциональности RTSP совпадает с HTTP и может взаимодействовать с ним, только если первоначально взаимодействие с потоком осуществляется через веб-страницу с помощью специального программного обеспечения. Однако RTSP принципиально отличается от HTTP тем, что доставка данных осуществляется вне диапазона по другому протоколу, но для работы с кэшами, прокси-серверами и аутентификацией используется аналогичный

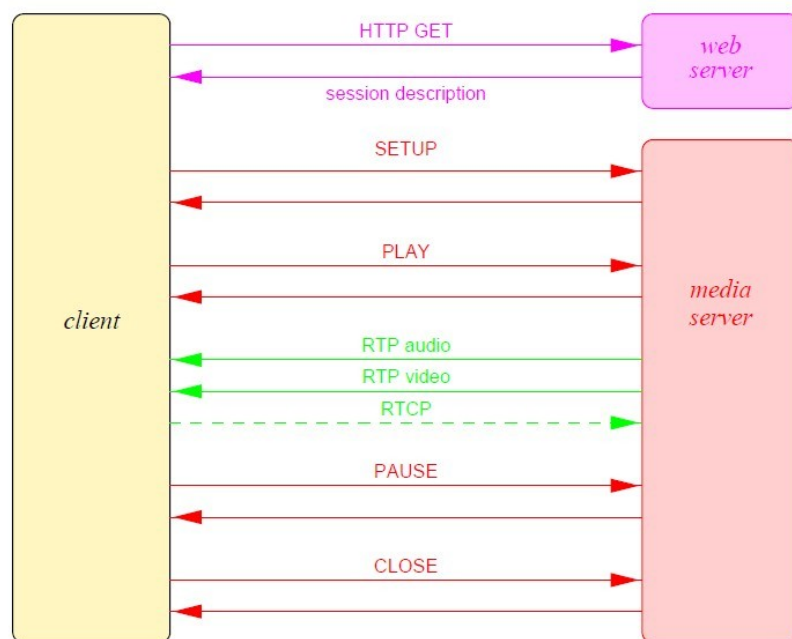


Рисунок 1.4 – Схема алгоритма работы RTSP

функционал HTTP. Таким образом, RTSP несовместим с HTTP [3, 10].

1.4.6 RTMP

RTMP — это протокол прикладного уровня, разработанный Adobe в 2009 году, для высокоэффективной потоковой передачи видео в реальном времени через Интернет между Adobe Flash и сервером. RTMP предоставляет двустороннюю многоканальную службу передачи сообщений с использованием надёжного транспортного потока путем использования протокола TCP, используемого для передачи параллельных потоков видео, аудио и текстовых сообщений с второстепенной тактовой информацией между парой общающихся пользователей. Различные классы сообщений получают разные приоритеты, что влияет на их очерёдность в транспортном потоке, когда пропускная способность ограничена [11].

RTMP нацелена на обеспечение стабильной и плавной передачи увеличивающихся объемов данных, необходимых для передачи и приема видео в реальном времени, что достигается посредством фрагментирования потока данных на небольшие одинаковые части (аудио — 64 байта, видео — 128

байтов), их последовательную передачу через постоянное TCP-соединение на принимающее устройство, которое затем снова собирает их в видеопоток. Для трансляции с устройства-источника видеопоток передается сервер кодировщик, данной этап называется «первой милей». Затем сервер обрабатывает поток и передает его дальше для ретрансляции на устройства клиентов, этап называется «последняя миля». С этого момента поток перехватывается поток другой связи (HLS, MPEG-DASH или WebRTC) или Adobe Flash [11].

Таким образом, постоянное соединение обеспечивает надежность доставки видеопотока, которое не отключается при нестабильной передаче данных между сервером и клиентом, что обеспечивает низкую задержку передачи данных. Благодаря постоянному соединению клиент может подключаться к и отключаться от трансляции, перематывать ее. У RTMP низкая пропускная способность, что может стать причиной потери пакетов без восстановления, что может ухудшить качество видео. RTMP протокол не поддерживается HTML5-плеерами и не совместим с HTTP [11].

1.4.7 HLS

HLS — протокол прикладного уровня для потоковой передачи медиа на основе HTTP, разработанный компанией Apple в 2009 году как часть программного обеспечения QuickTime, Safari, OS X и iOS. В основе работы лежит принцип разбиения цельного потока на небольшие фрагменты, последовательно скачиваемые по HTTP, развертывая контент с помощью обычных веб-серверов и сетевой доставки контента. Поскольку протокол использует стандартный протокол HTTP, рисунок 1.5.

Устройства-источники транслируют видеопоток на сервер, где данные поступают HLS-кодера для передачи потока в реальном времени и кодируются в формате H.264/AVC или HEVC и может соответствовать стандартам спецификации Apple. После чего поток разбивается на сегменты равного размера и создается файл, который называется «списком воспроизведения» в формате M3U, содержащий названия, местоположение и последовательность

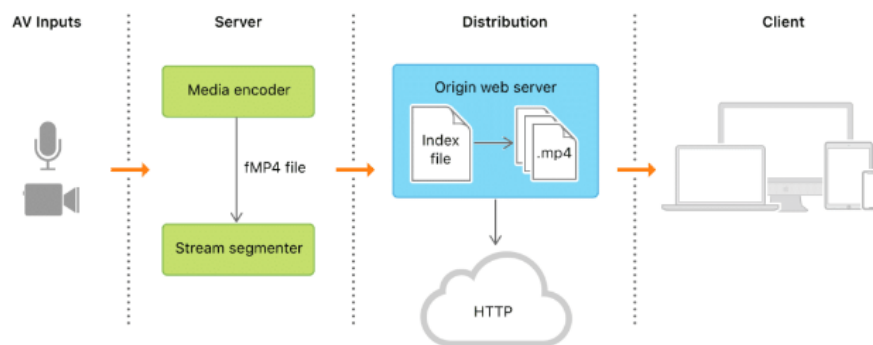


Рисунок 1.5 – Схема работы HLS [12]

воспроизведения сегментов, наряду с метадынными. Данный процесс называется упаковкой пакета для дальнейшей передачи. После упаковки переданного потока, он передается клиенту, где данный поток принимает любой плеер, который принимает HLS, например HTML5-плеер. Воспроизведение начинается с загрузки списка воспроизведения, а затем использования того, чтобы собрать последовательность сегментов пришедших непрерывным потоком. Клиент скачивает первый индексный файл через URL и далее несколько доступных файлов медиа. Программное обеспечение для проигрывания собирает всё в последовательность для воспроизведения. Существует два режима работы HLS — по запросу и трансляции в реальном времени. В режиме по запросу список воспроизведения содержит информацию о всей последовательности сегментов. В режиме трансляции в реальном времени список воспроизведения содержит только ссылки на последние несколько фрагментов, кроме того при последующих обращениях к списку воспроизведения, сегменты будут меняться, отражая текущее состояние трансляции [12].

HLS предусматривает поддержку адаптивного битрейта, который предусматривает наличие нескольких одновременно доступных потоков, каждый из которых может содержать одинаковый контент, закодированный в разных битрейтах, а также имеющий другие отличающиеся характеристики. По мере воспроизведения клиент может выбирать из числа нескольких доступных потоков, что позволяет адаптировать сессию к внешним условиям передачи по

сети, на основе чего происходит переключение потоков в ответ на изменение пропускной способности сети, то есть во время воспроизведения появляется возможность менять качество передаваемого видеопотока. Так как HLS основан на HTTP то ему предоставляется функционал HTTP, что обеспечивает шифрование мультимедиа и аутентификация клиента.

Трансляция по HLS приводит к задержке, что влияет на передачу потоков в реальном времени до 10 секунд. LHLS — это модификация HLS для потоковой передачи в реальном времени, где задержка достигается до 2 секунд [12].

1.4.8 MPEG-DASH

DASH — прокол прикладного уровня на основе HTTP для передачи потокового видео, созданный ISO и MPEG в 2011 году. DASH основан на адаптивной потоковой передаче HTTP, как и в целом работает аналогично HLS. Также поток разбивается на сегменты и формируется список воспроизведения, который уже используется в формате MPEG-TS стандарта MPEG-2. DASH отправляет сегменты потока на основе TCP, которые собираются повторно на устройстве клиента. DASH используется адаптивный битрейт, что способствует переключению между потоками по пропускной способности сети и возможностей устройства клиента, т.е. одно и то же видео может быть представлено в разных вариантах. MPEG-DASH поддерживает кодирования с использованием любого кодека (H.265, H.264, VP9 и т.д.) и разрешения до 4K.

MPEG-DASH предлагает иерархическую структуру организации файлов и использует файлов видеоресурсов на сервере потокового мультимедиа. MDP, файл описания презентации мультимедиа, создается вместе с сегментом потока и содержит все содержимое — видео, аудио, скорость звука, видео и т.д.

Высокая задержка MPEG-DASH обусловлена главным образом потерей сетевых пакетов и методом восстановления, используемым во всех сетях на

основе TCP. И хотя MPEG-DASH предлагает некоторый контроль над размером сегмента мультимедиа, возможность уменьшить задержку ограничена — особенно, если сервер требует загрузки среднего сегмента определённого размера.

1.4.9 MPTCP

MPTCP — протокол транспортного уровня, использующий несколько каналов передачи данных, который является набором расширений к протоколу TCP. Протокол использует обратное мультиплексирование, разбивая один MPTCP-поток на несколько TCP-потоков, каждый из которых проходит через отдельный канал передачи данных и имеет отдельный идентификатор. Каждый TCP-поток может использовать разные пути для передачи данных. При этом если один из путей недоступен или обладает плохим качеством связи, то автоматически переключается на другой путь, что повышает надёжность и обеспечивает адаптацию к пропускной способности сети. На устройстве клиента MPTCP собирает все TCP-потоки в MPTCP-поток, т.е. в единую последовательность.

MPTCP может использовать все доступные сетевые интерфейсы для передачи данных одновременно и балансировать нагрузки между интерфейсами, что позволяет достичь более высокой пропускной способности.

ЗАКЛЮЧЕНИЕ

Список использованных источников

1. Беднаж В.А. Огурцова А.М. Основы протоколы передачи потокового видео [Электронный ресурс]. Режим доступа: <https://vdocuments.mx/-5750a9aa1a28abcf0cd1ffec.html?page=1> (Дата обращения: 05.11.2023).
2. Хант Крэйг. TCP/IP: Сетевое администрирование, 3-е издание. — СПб: Символ-Плюс, 2008. — С. 816.
3. RFC 2326. Real Time Streaming Protocol (RTSP) [Электронный ресурс]. Режим доступа: <https://2rfc.net/2326> (Дата обращения: 13.11.2023).
4. Чепел Л. Титтел Э. TCP/IP: учебный курс, Пер. с англ. — СПб: БХВ-Петербург, 2003. — С. 976.
5. А. Лейкин. Протоколы транспортного уровня UDP, TCP и SCTP: достоинства и недостатки [Электронный ресурс]. Режим доступа: https://www.researchgate.net/publication/336748145_Protokoly_transportnogo_urovna_UDP_TCP_i_SCTP_dostoinstva_i_nedostatki (Дата обращения: 05.11.2023).
6. Гольдштейн Б. С. Гойхман В. Ю. Столповская Ю. В. Протоколы IP-телефонии: RTP, RTCP : учебное пособие. — СПб: СПбГУТ, 2014. — С. 59.
7. K. Tommi. Protocol overview: RTP and RTCP [Электронный ресурс]. Режим доступа: <https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.127.842&rep=rep1&type=pdf> (Дата обращения: 06.11.2023).
8. RFC 1889. RTP: A Transport Protocol for Real-Time Applications [Электронный ресурс]. Режим доступа: <https://www.ietf.org/rfc/rfc1889.txt> (Дата обращения: 06.11.2023).

9. K. Tommi. RTP, RTCP and RTSP — Internet Protocols fro RealTime Multimedia Communication [Электронный ресурс]. Режим доступа: <https://www.cse.wustl.edu/~jain/books/ftp/rtp.pdf> (Дата обращения: 06.11.2023).
10. RFC 2069. An Extension to HTTP : Digest Access Authentication [Электронный ресурс]. Режим доступа: <https://www.ietf.org/rfc/rfc2326.txt> (Дата обращения: 13.11.2023).
11. Adobe RTMP Specification [Электронный ресурс]. Режим доступа: <https://rtmp.veriskope.com/docs/сpec/> (Дата обращения: 13.11.2023).
12. HTTP Live Streaming [Электронный ресурс]. Режим доступа: <https://developer.apple.com/documentation/http-live-streaming?language=data> (Дата обращения: 14.11.2023).

ПРИЛОЖЕНИЕ А

Презентация к курсовой работе

Презентация содержит 14 слайдов.