RTCP (Real-Time Transport Control Protocol

- Протокол управления передачей в реальном времени, используемый совместно с RTP.
- Протокол описан в RFC 3550.
- RTCP базируется на периодической передаче управляющих пакетов всем участникам сессии, используя тот же механизм рассылки, что и для пакетов данных.
- Протокол RTCP используется для передачи информации о задержках и потерях медиа-пакетов, джиттер-буфере, уровне звукового сигнала. Также передаются метрика качества сигнала (Call Quality Metrics) и Echo Return Loss.

- Определены следующие типы сообщений RTCP:
-
- SR - Sender Report - отчёт отправителя по отправленным медиа-пакетам RTP
- RR - Receiver Report - отчёт получателя по полученным медиа-пакетам RTP
- SDES - элементы описания источника, включая cname
- BYE - Отмечает прекращение участия в группе
- APP - Специфические функции приложения
-
- В рекомендации RFC 3611 определёно также сообщение XR - Extended Report, которое позволяет отправлять большее число параметров, по сравнению со стандартными отчётами, а именно:
-
- Время получения пакета
- Порядковые номера потерянных пакетов
- Порядковые номера повторяющихся пакетов
- Ожидаемое время доставки
- Задержка с момента приема последнего отчета RTCP Receiver Report
- Общая статистика медиа-пакетов
- Оценка VoIP - направления (MOS и R Factor - параметр характеризующий качество сигнала)

Протокол **RTP** (англ. *Real-time Transport Protocol*)

- работает на транспортном уровне и используется при передаче трафика реального времени. Протокол был разработан Audio-Video Transport Working Group в IETF и впервые опубликован в 1996 году какRFC 1889, и заменён RFC 3550 в 2003 году.
- Протокол RTP переносит в своём заголовке данные, необходимые для восстановления голоса или видеоизображения в приёмном узле, а также данные о типе кодирования информации (JPEG, MPEG и т. п.). В заголовке данного протокола, в частности, передаются временная метка и номер пакета. Эти параметры позволяют при минимальных задержках определить порядок и момент декодирования каждого пакета, а также интерполировать потерянные пакеты.

- RTP не имеет стандартного зарезервированного номера порта. Единственное ограничение состоит в том, что соединение проходит с использованием чётного номера, а следующий нечётный номер используется для связи по протоколу RTCP. Тот факт, что RTP использует динамически назначаемые адреса портов, создаёт ему трудности для прохождения межсетевых экранов, для обхода этой проблемы, как правило, используется STUN-сервер.
- Установление и разрыв соединения не входит в список возможностей RTP, такие действия выполняются сигнальным протоколом(например, RTSP или SIP протоколом).

- RTP был разработан как протокол реального времени, из конца в конец (end-to-end), для передачи потоковых данных. В протокол заложена возможность компенсации джиттера и детектирования нарушения последовательности пакетов данных — типичных событий при передаче через IP-сети. RTP поддерживает передачу данных для нескольких адресатов через Multicast. RTP рассматривается как основной стандарт для передачи голоса и видео в IP-сетях и совместно с кодеками.
- Приложения, формирующие потоки реального времени, требуют своевременной доставки информации и для достижения этой цели могут допустить некоторую потерю пакетов. Например, потеря пакета в аудио-приложении может привести к доле секунды тишины, которая может быть незаметна при использовании подходящих алгоритмов скрытия ошибок. Протокол TCP, хотя и стандартизирован для передачи RTP, как правило не используется в RTP-приложениях, так как надежность передачи в TCP формирует временные задержки. Вместо этого, большинство реализаций RTP базируется на UDP. Кроме этого, существуют другие спецификации для транспортных протоколов SCTP и DCCP, но они мало распространены.

- Компоненты протокола
- Спецификация RTP описывает два под-протокола:
- Протокол передачи данных, RTP, который взаимодействует с передачей данных реального времени. Информация, предоставляемая посредством этого протокола включает тайм-стемп (для синхронизации), последовательный номер (для детектирования потери и дублирования пакетов) и формат полезной нагрузки, который определяет формат кодирования данных.
- Протокол контроля, RTCP, используемый для определения качества обслуживания (QOS), обратной связи и синхронизации между медиа-потоками. Занимаемая полоса пропускания RTCP — мала в сравнении с RTP, обычно около 5 %.
- Управляющий сигнальный протокол, такой как SIP, H.323, MGCP или H.248. Сигнальные протоколы управляют открытием, модификацией и закрытием RTP-сессий между устройствами и приложениями реального времени.
- Управляющий протокол описания медиа, такой как Session Description Protocol.

- Сессии
- RTP-сессия устанавливается для каждого потока мультимедиа. Сессия состоит из IP-адреса и пары портов для RTP и RTCP. Например, аудио и видео потоки будут иметь различные RTP-сессии, позволяющие приемнику для этого выделить конкретный поток. Порты, которые образуют сессию, связываются друг с другом средствами других протоколов, таких как SIP (содержащий в своих сообщениях протокол SDP) и RTSP (используя SDP в методе Setup). В соответствии со спецификацией, RTP не имеет стандартного зарезервированного номера порта. Единственное ограничение состоит в том, что соединение проходит с использованием чётного номера, а следующий нечётный номер используется для связи по протоколу RTCP. RTP и RTCP обычно используют непривилегированные UDP-порты (16k-32k), но могут использовать и другие протоколы, поскольку сам протокол RTP независим от транспортного уровня.

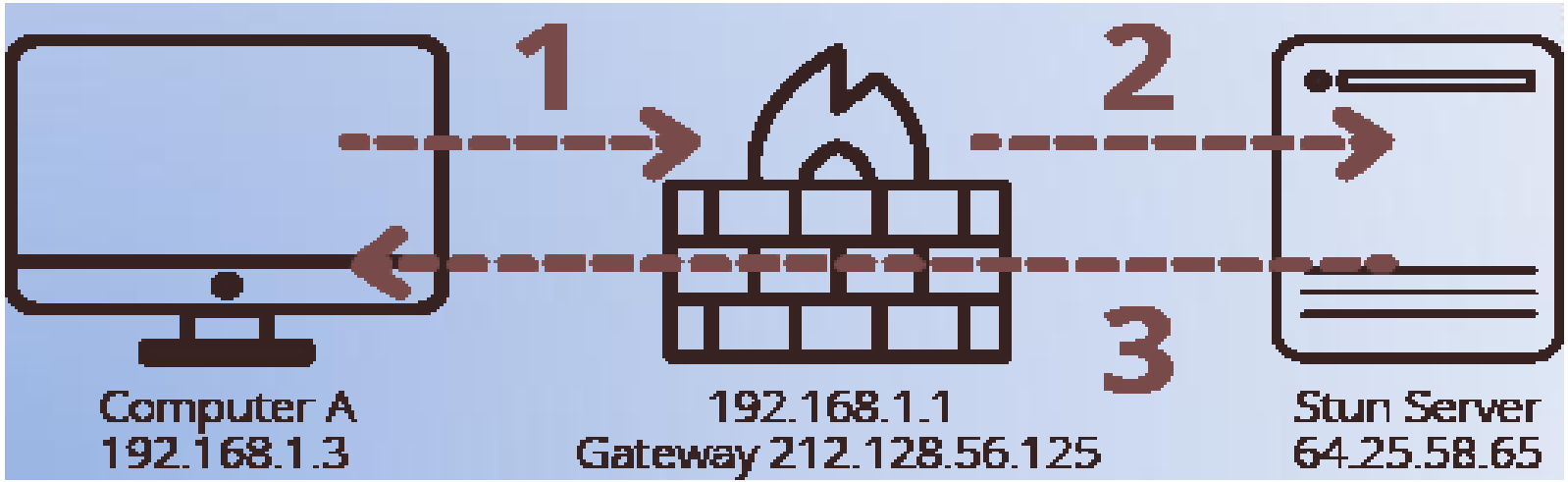
- 0-1 — Ver. (2 бита) указывает версию протокола. Текущая версия — 2.
- 2 — P (один бит) используется в случаях, когда RTP-пакет дополняется пустыми байтами на конце.
- 3 — X (один бит) используется для указания расширений протокола, задействованных в пакете.
- 4-7 — CC (4 бита) содержит количество CSRC-идентификаторов, следующих за постоянным заголовком.
- 8 — M (один бит) используется на уровне приложения и определяется профилем. Если это поле установлено, то данные пакета имеют какое-то особое значение для приложения.
- 9-15 — PT (7 бит) указывает формат полезной нагрузки и определяет её интерпретацию приложением.
- Порядковый номер (16 бит)
- Метка времени (32 бита)
- 64-95 — SSRC указывает источник синхронизации.
- EHL (Extension Header Length) — количество 32-битных слов в блоке данных расширения заголовка.
- Данные
- L — последний байт в пакете, определяющий длину области заполнения в байтах (используется для выравнивания в последнем пакете).

STUN

- STUN (Session Traversal Utilities for NAT, Утилиты прохождения сессий для NAT, ранее англ. Simple Traversal of UDP through NATs, Простое прохождение UDP через серверы NAT)

- STUN-сервер позволяет клиентам находить свой адрес общего доступа, тип NAT, за которым они находятся и порт Интернета, связываемый NAT с конкретным локальным портом, а так же тип NAT. Эта информация используется для настройки связи UDP между клиентом и провайдером VOIP и организации сеанса. Протокол STUN определяется стандартом RFC 3489.
- Соединение с сервером STUN устанавливается через UDP-порт 3478, однако сервер предлагает клиентам выполнить проверку также и альтернативного IP и номера порта (у серверов STUN есть два IP-адреса). В RFC говорится, что выбор порта и IP является произвольным.

- STUN — это клиент-серверный протокол.
- VoIP-клиент может включать в себя реализацию клиента STUN, который отправляет запрос серверу STUN.
- Затем сервер STUN отправляет клиенту обратно информацию о том, каков внешний адрес маршрутизатора NAT, и какой порт открыт на NAT для приема входящих запросов обратно во внутреннюю сеть.



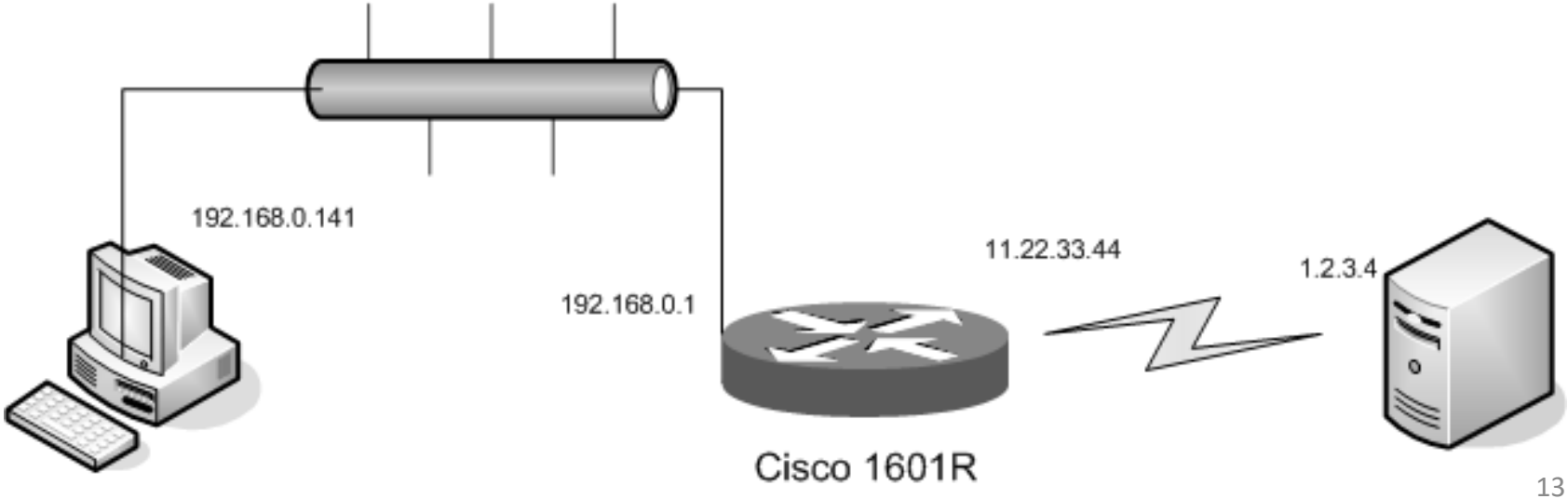
- Ответ сервера STUN позволяет клиенту STUN определить, какой тип трансляции адреса используется, (разные типы маршрутизаторов NAT обрабатывают входящие UDP пакеты по-разному).
- STUN работает с тремя из четырёх основных типов:
- Full Cone NAT, Address Restricted NAT и Port Restricted NAT (четвёртый - Symmetric NAT).

- В случае ограничивающего NAT клиент должен отправить пакет на удаленный узел, прежде чем NAT начнет пропускать пакеты от удаленного узла к клиенту. STUN не будет работать с симметричным NAT'ом (также называемым «двусторонний NAT»), который часто встречается в сетях больших компаний. При симметричном NAT IP-адрес сервера STUN отличается от конечного адреса, и из-за этого адрес NAT, который видит STUN-сервер, отличается от конечного адреса, который будет использоваться для отправки пакетов клиенту.
-
- Как только клиент обнаружил свой внешний адрес, он может передать его узлу, с которым проходит соединение. Если на пути встречаются трансляторы типа «полный конус», любая из двух сторон может начать общение. Если же выполняется трансляция типа «ограниченный конус» или «порт ограниченного конуса», обе стороны должны начать передачу данных совместно.

- Классификация NAT, часто встречающаяся в связи с VoIP. Термин «соединение» использован в значении «последовательный обмен пакетами UDP».
-
- Симметричный NAT (Symmetric NAT) — трансляция, при которой каждое соединение, инициируемое парой «внутренний адрес: внутренний порт» преобразуется в свободную уникальную случайно выбранную пару «публичный адрес: публичный порт». При этом инициация соединения из публичной сети невозможна. Не пропускает несовпадающие по адресам и портам отправителя (ориентируется только на адрес и порт получателя).
-
- Cone NAT, Full Cone NAT — однозначная (взаимная) трансляция между парами «внутренний адрес: внутренний порт» и «публичный адрес: публичный порт». Любой внешний хост может инициировать соединение с внутренним хостом (если это разрешено в правилах межсетевого экрана). Например, любой внешний адрес и порт отправителя пройдет во внутреннюю сеть, если адрес назначения совпал с хранимой в таблице NAT информацией.
-
- Address-Restricted cone NAT, Restricted cone NAT — постоянная трансляция между парой «внутренний адрес: внутренний порт» и «публичный адрес: публичный порт». Любое соединение, инициированное с внутреннего адреса, позволяет в дальнейшем получать ему пакеты с любого порта того публичного хоста, к которому он отправлял пакет(ы) ранее.
-
- Port-Restricted cone NAT — трансляция между парой «внутренний адрес: внутренний порт» и «публичный адрес: публичный порт», при которой входящие пакеты проходят на внутренний хост только с одного порта публичного хоста — того, на который внутренний хост уже посылал пакет.

-
- Пусть компьютер с адресом IL 192.168.0.141 отправляет DNS-запрос на внешний хост 1.2.3.4 (порт 53, протокол UDP). Как следует из конфигурации, наш внешний адрес IG – 11.22.33.44.
-
- В результате этого в таблице NAT появится примерно такая запись:

Proto	Inside global	Inside local	Outside local	Outside global
UDP	11.22.33.44:1053	192.168.0.141:1053	1.2.3.4:53	1.2.3.4:53



Secure Real-time Transport Protocol (сокр. SRTP, рус. Безопасный протокол передачи данных в реальном времени)

- Определяет профиль протокола RTP и предназначен для шифрования, установления подлинности сообщения, целостности, защиты от подмены данных RTP в однонаправленных и multicast передачах медиа и приложениях.
- Впервые опубликован в IETF в марте 2004 как RFC 3711.

- У SRTP как и у RTP также есть родственный протокол, названный Secure RTCP (или SRTCP).
-
- Для шифрования медиа потока (в целях конфиденциальности голосового соединения), SRTP (вместе с SRTCP) стандартизирует использование только единственного шифра, AES, который может использоваться в двух режимах, превращающих изначально блочный шифр AES в потоковый шифр:
- Сегментированный целочисленный счётчик — типичный режим, который осуществляет произвольный доступ к любым блокам, что является существенным для трафика RTP, передающегося в публичных сетях с непредсказуемым уровнем надежности и возможной потерей пакетов. В общем случае почти любая функция может использоваться в роли «счётчика», предполагая, что эта функция не повторяется для большого числа итераций. Но стандарт для шифрования данных RTP — только обычное целочисленное значение счётчика. Используется AES с длиной шифровального ключа по умолчанию в 128 бит и ключом сессии длиной в 112 бит.
- f8-режим — вариант режима способа обратной связи, расширенного, чтобы быть доступным с изменённой функцией инициализации. Так же используется AES.

ZRTP — криптографический шифрования, используемый в системах протокол согласования ключей ередачи голоса по IP-сетям (VoIP).

- ZRTP описывает метод получения ключей по алгоритму Диффи — Хелмана для организации передачи SRTP. ZRTP осуществляет согласование ключей в том же потоке RTP, по которому установлена аудио/видео связь после инициализации вызова, например по протоколу SIP (Session Initiation Protocol), то есть не требует отдельного канала связи. Разработан в 2006 году.
- Во время посыла звонка создаётся публичный ИД, используемый при создании ключей, которыми будет шифроваться медиа поток разговора. Таким образом ключ действителен только в течение одного разговора, образуя таким образом сессию Secure RTP (SRTP). При разрыве соединения ключ и весь криптографический контекст уничтожается, что обеспечивает совершенную прямую секретность (PFS). Таким образом существует потенциал встраивания этого механизма в уже существующие программные VoIP программные продукты, шлюзы и ИП телефоны.

- Протокол не требует заранее сгенерированных ключей, или поддержки инфраструктуры обмена ключей (PKI), или центра сертификации (CA). Это избавляет от сложностей создания структуры авторизации, основанной на доверенной поддержке, которая, например, применяется в шифровании SSL. Главной целью организации шифроканала, будь то голосовая сессия, или https соединение с интернет-банком — избежать возможности присутствия Человека_посередине (man in the middle), обеспечивая единую криптозащиту между любыми двумя точками в мире ИП.
-
- ZRTP может теоретически применяться совместно с любыми сигнальными протоколами, использующими RTP для передачи медиа потока, включая SIP, H.323, SCCP, MGCP Unistim и Jingle, так как в теории ZRTP не зависит от сигнализации, осуществляя обмен ключей в медиа сессии RTP. Таким образом ZRTP может стать открытым стандартом де-факто в мире IP-телефонии.

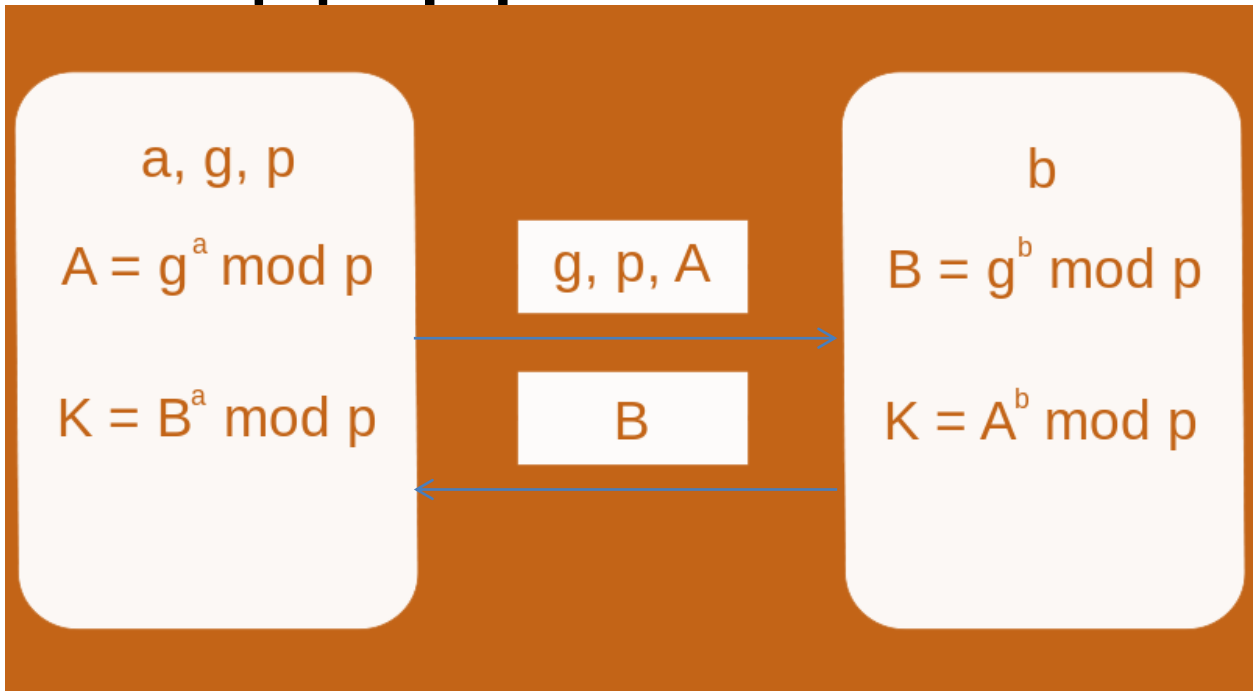
SAS

-
- Сам по себе алгоритм обмена ключами Диффи — Хелмана не может обеспечить защиту от присутствия человека посередине (man in the middle).
- Для аутентификации ZRTP использует Short Authentication String (SAS) "короткую строку аутентификации", являющуюся сокращённым представлением криптографического хеша полученных ключей Диффи — Хелмана.
- Значения SAS вычисляются на каждой стороне соединения, абоненты передают их друг другу голосом для сверки. Если значения не совпали, то с большой уверенностью можно предположить присутствие Человека_посередине.
- Использование алгоритма Диффи — Хелмана даёт потенциальному Человеку_посередине всего одну попытку сгенерировать правильную SAS при попытке атаки.
- Так как SAS получается из старших 32 битов хеша и имеет две формы представления (16-битную B256 в виде пары слов из списка PGP, и 20-битную B32 в виде четырёх символов), она является очень короткой, вероятность необнаружения атаки при использовании SAS в формате B256 равна 1/65536. Применение SAS в формате B32 понижает вероятность необнаружения атаки в 16 раз (по сравнению с B256), до 1/1048576.

Непрерывность ключевого материала

- Вторым уровнем защиты ZRTP от атак "человек посередине" является непрерывность ключевого материала. Хеш ключевой информации предыдущего вызова подмешивается к параметрам алгоритма Диффи - Хеллмана при следующем вызове (между этими же абонентами), что придаёт протоколу ZRTP сходство с SSH. Если "человека посередине" не было при первом вызове, он исключается из всех последующих.

Диффи Хеллмана



$K = A^b \mod p = (g^a \mod p)^b \mod p = g^{ab} \mod p = (g^b \mod p)^a \mod p = B^a \mod p$

DCCP (англ. *Datagram Congestion Control Protocol*)

- протокол транспортного уровнямодели OSI, разрабатываемый IETF. Принят в качестве стандарта в марте 2006 года. Он предоставляет механизмы для отслеживания перегрузок в сети, избегая возможности использования механизмов прикладного уровня. Этот протокол не гарантирует доставку информации в нужном порядке.
- DCCP очень эффективен для приложений, в которых данные, пришедшие не вовремя, становятся бесполезными. Например: потоковое медиа-вещание, онлайн игры и интернет-телефония. Главная особенность этих приложений состоит в том, что старые сообщения очень быстро становятся бесполезными, поэтому лучше получить новое сообщение, чем пытаться переслать старое. Но на данный момент большинство таких приложений самостоятельно реализуют отслеживание перегрузок, а в качестве протоколов передачи используются TCP или UDP.
- Протокол DCCP доступен в ядре Linux с версии 2.6.14 и улучшается с каждым выпуском.

Протокол **RDP** (англ. *Reliable Data Protocol*)

- разработан для обеспечения надежной передачи данных между пакетно-ориентированными приложениями. Изначально он был разработан для приложений, реализующих удаленную загрузку данных и удаленное устранение неполадок, однако его можно использовать и в других приложениях, требующих надежной передачи сообщений. Существуют две версии RDP, описанные в спецификациях RFC-908 и RFC-1151 соответственно.

Протокол **RUDP** (англ. *Reliable User Datagram Protocol*)

- основанный на протоколе RDP, разработан для передачи телефонных сигналов через IP-сети. Этот протокол не стандартизирован, он не имеет официальной спецификации.
- Протоколы RDP и RUDP используются в тех случаях, когда нельзя использовать UDP из-за его ненадежности, а использование TCP влечет за собой слишком высокую сложность процесса передачи данных.
- В отличие от UDP, RDP и RUDP поддерживают следующие функции:
- подтверждение доставки пакетов
- повторная отправка потерянных пакетов
- управление потоками передачи данных
- RDP обеспечивает прикладной уровень надежной службой передачи сообщений. Интерфейс протокола преобразует данные пользователя в сообщения. Сообщения, в свою очередь, в ходе обмена данными между RDP и IP преобразуются в сегменты данных и далее в дейтаграммы.