Network Working Group
Request for Comments: 3286
Category: Informational

L. Ong
Ciena Corporation
J. Yoakum
Nortel Networks

May 2002

Введение в SCTP

An Introduction to the Stream Control Transmission Protocol (SCTP)

Статус документа

Этот документ содержит информацию, предназначенную для сообщества Internet, и не задает каких-либо стандартов Internet. Документ может распространяться без ограничений.

Авторские права

Аннотация

В данном документе проводится обзорное рассмотрение возможностей протокола SCTP[1]. Документ подготовлен в качестве руководства для потенциальных пользователей SCTP как транспортного протокола общего назначения.

1. Введение

SCTP представляет собой новый транспортный протокол для сетей IP, работающий на одном уровне с протоколами UDP[2] и TCP[3] и обеспечивающий транспортные функции для различных приложений Internet. Протокол SCTP был одобрен IETF в качестве предварительного стандарта[4] [1]. После выхода предварительного стандарта был обновлен алгоритм контроля ошибок [2]. Новые изменения протокола будут отражаться в поддерживаемом IETF списке RFC.

Подобно TCP, протокол SCTP обеспечивает транспортный сервис с гарантией доставки и сохранением порядка следования пакетов. Протокол SCTP (как и TCP) использует сеансовые механизмы, что означает создание ассоциаций[5] SCTP между конечными точками обмена данными до начала реальной передачи данных и поддержка таких отношений в течение всего сеанса обмена информацией.

В отличие от TCP протокол SCTP обеспечивает множество функций, которые имеют важное значение для организации телефонной связи и в то же время могут обеспечивать ряд преимуществ для других приложений, которым требуется транспорт с высоким уровнем производительности и надежности. Первоначальное назначение протокола SCTP описано в работе [3].

2. Основные функции SCTP

SCTP представляет собой unicast-протокол[6], который обеспечивает обмен данными между двумя конечными точками. Отметим, что каждая из этих точек может быть представлена более чем 1 адресом IP.

Протокол SCTP обеспечивает гарантию доставки, детектирование случаев отбрасывания данных, изменения порядка доставки, повреждения или дублирования данных, а также повторную передачу пакетов при возникновении необходимости. Обмен данными по протоколу SCTP происходит в полнодуплексном режиме.

Работа SCTP основана на обмене сообщениями и протокол поддерживает кадрирование границ отдельных сообщений. Протокол TCP ориентирован на обмен байтами и не хранит никаких неявных структур в передаваемых потоках данных.

Скорость передачи по протоколу SCTP может адаптироваться подобно тому, как это происходит в TCP, - в зависимости от загрузки сети. Протокол SCTP может использоваться одновременно с TCP на общих каналах передачи данных.

3. Поддержка множества потоков в SCTP

Название протокола SCTP обусловлено его многопотоковой природой передачи данных. Поддержка множества одновременных потоков позволяет распределить между этими потоками передаваемую информацию так, чтобы каждый из потоков обеспечивал независимую упорядоченную доставку данных. При потере сообщения в любом из потоков это оказывает влияние лишь на данный поток, не затрагивая работу других потоков данных.

Протокол TCP работает с одним потоком данных и обеспечивает для такого потока сохранение порядка доставки байтов из потока. Такой подход удобен для доставки файлов или записей, но он может приводить к дополнительным задержкам при потере информации в сети или нарушении порядка доставки пакетов. При возникновении таких ситуаций протокол TCP должен дожидаться доставки всех данных, требуемых для восстановления порядка.

Для многих приложений строгое сохранение порядка доставки не является обязательным. В системах передачи телефонной сигнализации достаточно поддерживать порядок передачи для последовательности сообщений, которые

[1] Stream Control Transmission Protocol - протокол управления потоковой передачей.
[2] User Datagram Protocol - протокол пользовательских дейтаграмм.
[3] Transmission Control Protocol - протокол управления передачей.
[4] Proposed Standard - предложенных стандарт
[5] Прямой связи. *Прим. перев.*
[6] Т. е. протокол, не поддерживающий групповой адресации. *Прим. перев.*

воздействуют на один ресурс (например, для одного вызова или одного канала). Остальные сообщения слабо коррелируют между собой и могут доставляться с нарушением порядка.

Другим примером является возможность использования множества потоков для доставки multimedia-документов (например, web-страниц) в рамках одной сессии. Поскольку такие документы состоят из разнотипных объектов разных размеров, многопотоковая доставка таких компонент не требует строгой упорядоченности. Однако использование множества потоков при доставке существенно повышает для потребителей привлекательность сервиса.

В то же время транспорт в рамках одного соединения[1] SCTP обеспечивает единый механизм управления потоком и контроля насыщения, что существенно снижает нагрузку на транспортный уровень.

Протокол SCTP поддерживает многопотоковую передачу за счет устранения зависимости между передачей и доставкой данных. В частности, каждый блок[2] полезной информации типа DATA (данные) использует два набора порядковых номеров. Номер TSN[3] управляет передачей сообщений и детектированием их потери, а пара "идентификатор потока Stream ID - номер SSN[4]" используется для управления порядком доставки потребителю полученных данных.

Такая независимость механизмов нумерации позволяет получателю незамедлительно обнаруживать пропуски данных (например, в результате потери сообщения), а также видеть влияние потерянных данных на поток. Утрата сообщения вызывает появление пропуска в порядковых номерах SSN для потока, на который это сообщение оказывает влияние и не вызывает такого пропуска для других потоков. Следовательно, получатель может продолжать доставку незатронутых потоков, не дожидаясь повтора передачи утраченного сообщения.

4. Поддержка многодомных хостов в SCTP

Другим важным качеством SCTP является поддержка многодомных хостов, позволяющая создавать конечные точки SCTP с множеством IP-адресов. Поддержка многодомных хостов повышает уровень "живучести" сессий в случаях возникновения сбоев в сети. В традиционных однодомных сеансах отказ в соединении с ЛВС может изолировать конечную точку, а сбой в работе магистральной сети может привести к временным проблемам на транспортном уровне, пока протокол маршрутизации IP не найдет пути в обход сбойного участка. При использовании многодомных узлов SCTP могут быть организованы резервные (избыточные) соединения с ЛВС и поддерживаются различные варианты преодоления сложностей, связанных с отказами в магистральных сетях. Использование адресов с различными префиксами может обеспечить автоматическую маршрутизацию пакетов к другому оператору. Можно использовать методы route-pinning или даже резервировать соединения с магистральными сетями, если обеспечивается контроль над сетевой архитектурой и протоколами.

Действующий вариант SCTP не поддерживает распределения нагрузки (load sharing), поэтому многодомные хосты обеспечивают лишь избыточность соединений для повышения уровня надежности. Один из адресов многодомного хоста указывается в качестве основного (primary) и используется как адрес получателя для всех блоков DATA при нормальной передаче. При передаче повторных блоков DATA используется один из дополнительных адресов с целью повышения вероятности доставки в конечную точку. При повторяющихся неоднократно повторах передачи принимается решение об отправке всех блоков DATA с использованием альтернативного адреса, пока системе мониторинга не удастся увидеть доступность основного адреса.

Для поддержки множества интерфейсов конечные точки SCTP обмениваются списками своих адресов в процессе создания ассоциации. Каждая из конечных точек должна быть способна принимать сообщения с любого адреса, связанного с удаленным партнером; на практике некоторые ОС могут использовать в пакетах циклический перебор адресов отправителя и в таких случаях прием пакетов с различных адресов является нормальной ситуацией. Для всего списка адресов конечной точки в данной сессии используется один номер порта.

Для повышения уровня безопасности требуется, чтобы некоторые отклики передавались по адресу, указанному в поле отправителя сообщения, вызвавшего отклик. Например, когда сервер получает блок INIT от клиента для инициирования SCTP-ассоциации, сервер всегда будет передавать блок INIT ACK по адресу отправителя в заголовке IP блока INIT.

5. Процедура создания ассоциаций SCTP

Процедура создания ассоциаций SCTP[5] состоит из 4 последовательных сообщений. Два последних сообщения стартовой последовательности могут включать блок DATA, поскольку эти сообщения передаются после того, как проверена корректность ассоциации. Для предотвращения некоторых типов атак на службы в стартовую последовательность встроен механизм cookie.

5.1 Механизм Cookie

Механизм cookie обеспечивает защиту от атак вслепую, когда атакующий генерирует множество блоков INIT с целью перегрузки сервера SCTP за счет расхода памяти и иных ресурсов, требуемых для обработки новых запросов INIT. Взамен выделения памяти для TCB[6] сервер создает параметр Cookie с информацией TCB, корректным временем жизни и сигнатурой для аутентификации. Этот параметр передается клиенту в сообщении INIT ACK. Поскольку сообщения INIT ACK всегда возвращаются по адресу отправителя запросов INIT, атакующий вслепую не будет получать Cookie. Легитимный клиент SCTP получит Cookie и вернет серверу блок COOKIE ECHO, по которому сервер SCTP может проверить корректность Cookie и использовать это значение для создания TCB. Поскольку параметр Cookie создается сервером и на сервере же проверяется, формат и секретный ключ Cookie знает только сервер и не возникает необходимости в передаче их через сеть клиенту.

---

[1] В английском языке для соединений SCTP используется термин association (ассоциация, связь). *Прим. перев.*
[2] В английском языке используется термин chunk. *Прим. перев.*
[3] Transmission Sequence Number - порядковый номер при передаче.
[4] Stream Sequence Number - порядковый номер потока.
[5] SCTP Initiation Procedure
[6] Transmission Control Block - блок управления передачей.

Прочие компоненты процедуры создания ассоциаций SCTP соответствуют большинству соглашений, используемых для соединений TCP - конечные точки обмениваются информацией о размере приемного окна, стартовых порядковых номерах и т. п. Кроме того, конечные точки могут обмениваться упомянутыми выше списками адресов, а также согласовывать количество потоков, открываемых каждой из сторон.

5.2. Пазрешение конфликтов INIT

Поддержка многодомных хостов может приводить к изменению порядка доставки сообщений, для которых использовались разные пути. Эта проблема возникает на этапе создания ассоциации, когда работа без процедуры разрешения конфликтов может с легкостью привести к созданию множества параллельных ассоциаций между парой конечных точек. Для предотвращения этого SCTP включает множество процедур связывания параллельных попыток создания ассоциации с одной ассоциацией SCTP.

6. Функции обмена данными в SCTP

За обменом блоками DATA в SCTP следует процедура, подобная TCP ACK. Получение блока DATA подтверждается передачей блока SACK, который показывает не только диапазон принятых TSN[1], но и некумулятивные номера TSN, указывающие пропуски в принятой последовательности TSN. Как и в процедурах TCP подтверждения SACK передаются с использованием метода delayed ack (обычно одно сообщение SACK на каждый полученный пакет с ограничением задержки между передачей сообщений SACK и увеличением задержки на 1 при получении каждого пакета с обнаруженным пропуском).

Алгоритмы управления трафиком и предотвращения насыщения[2] соответствуют аналогичным алгоритмам TCP. Анонсируемый размер окна показывает емкость буфера на приемной стороне, а поддерживаемое для пути окно насыщения позволяет управлять пакетами “на лету”. Алгоритмы Congestion avoidance, Fast recovery и Fast retransmit[3] встроены в протокольные процедуры в соответствии с RFC 2581. Единственным отличием является то, что конечные точки должны обеспечивать преобразование между количеством принятых/переданных байтов и номерами TSN, поскольку нумерация TSN осуществляется для транка в целом.

Приложение может указать для передаваемых данных время жизни и, если по истечении заданного времени данные еще не переданы, они будут отбрасываться (например, связанные со временем сигнальные сообщения). Если данные передаются, их доставка должна продолжаться во избежание пропусков в последовательности TSN.

7. Завершение работы ассоциаций SCTP

Процедура SCTP использует последовательность из 3 сообщений для разрыва существующей ассоциации. Конечные точки до завершения существования ассоциации подтверждают получение переданных им блоков DATA. Процедура Abort используется для аварийного завершения работы ассоциации, когда требуется незамедлительное прекращение обмена пакетами.

Отметим, что протокол SCTP не поддерживает «полуоткрытых» соединений, которые могут возникать в TCP, когда одна сторона показывает другой отсутствие данных для передачи, а вторая сторона может неограниченно долго продолжать передачу данных. Протокол SCTP предполагает, что после начала процедуры shutdown обе стороны прекращают передачу новых данных через ассоциацию и требуется лишь обмен пакетами, подтверждающими прием отправленных ранее данных.

8. Формат сообщений SCTP

Сообщения SCTP включают общий заголовок, за которым следует один или несколько блоков (chunk), которые могут содержать данные или управляющую информацию. В заголовке указываются номера портов отправителя и получателя, что позволяет мультиплексировать различные ассоциации SCTP на одном адресе. 32-битовый тег верификации предотвращает возможность включения в ассоциацию SCTP устаревших или фальсифицированных сообщений, а 32-битовая контрольная сумма, рассчитанная на основе полиномиального алгоритма CRC-32c [2] служит для детектирования ошибок.

Каждый блок содержит поля типа, флагов, размера, а также значение. Управляющие блоки включают различные параметры и флаги, зависящие от типа блока. Блоки данных (DATA) включают флаг управления сегментацией и сборкой, а также параметры TSN, Stream ID, Stream Sequence Number и Payload Protocol Identifier.

Параметр Payload Protocol ID включен для обеспечения возможности расширения в новых версиях протокола. Если предположить, что функции идентификации протокола и мультиплексирования по портам в будущем перестанут играть столь важную роль, как сейчас, Payload Protocol ID будет обеспечивать идентификацию протоколов, передаваемых с помощью SCTP без использования номера порта.

Формат сообщений SCTP обеспечивает механизм связывания множества блоков данных и управления в одно сообщение для повышения эффективности транспорта. Использованием такой группировки (bundling) управляет приложение, поэтому группировка стартовой передачи невозможна. Связывание естественным образом осуществляется при повторе передачи блоков DATA в целях снижения вероятности насыщения.

9. Обработка ошибок

9.1. Повтор передачи

Повторная передача блоков DATA может быть обусловлена (a) тайм-аутом, определяемым таймером повтора (retransmission timer) или (b) получением SACK, показывающих что блок DATA не был получен адресатом. Для снижения вероятности насыщения повтор передачи блоков DATA ограничивается. Значение тайм-аута для повтора (RTO) устанавливается на основе оценки времени кругового обхода и уменьшается экспоненциально с ростом частоты потери сообщений.

[1] Transmission Sequence Number - порядковый номер при передаче.
[2] Flow and Congestion Control.
[3] Предотвращение насыщения, ускоренное восстановление и ускоренный повтор передачи, соответственно.

Для активных ассоциаций с почти постоянным уровнем трафика DATA причиной повтора скорей всего будут сообщения SACK, а не тайм-аут. Для снижения вероятности ненужных повторов используется правило 4 SACK, в соответствии с которым повтор передачи происходит только по четвертому SACK, указывающему на пропуск блока данных. Это позволяет предотвратить повторы передачи, вызванные нарушением порядка доставки.

9.2. Сбой в пути

Поддерживается счетчик для числа повторов передачи по конкретному адресу получателя без подтверждения успешной доставки. Когда значение этого счетчика достигает заданного порога (конфигурационный параметр), адрес объявляется неактивным и протокол SCTP начинает использовать другой адрес для передачи блоков DATA.

Кроме того, по всем неиспользуемым (дополнительным) адресам периодически передаются специальные блоки Heartbeat и поддерживается счетчик числа блоков Heartbeat, переданных без возврата соответствующего Heartbeat Ack. Когда значение счетчика достигает заданного порога (параметр конфигурации), соответствующий адрес объявляется неактивным.

Блоки Heartbeat передаются по неактивным адресам до тех пор, пока не будет получено сообщение Ack, говорящее о восстановлении активности адреса. Частота передачи блоков Heartbeat определяется значение RTO и дополнительной задержкой, которая позволяет передавать блоки Heartbeat без помех для пользовательского трафика.

9.3. Отказ в конечной точке

Для всех адресов получателя поддерживается общий счетчик числа повторов или блоков Heartbeat, переданных удаленной точке без получения от нее соответствующего подтверждения (Ack). Когда значение счетчика достигает заданного порога (параметр конфигурации) конечная точка декларируется как недостижимая и ассоциация SCTP закрывается.

10. Интерфейс API

Спецификация протокола включает модель примитивов, используемых при обмене информацией между приложениями и уровнем SCTP. Эта модель является скорее информационным материалом, нежели формальной спецификацией API. Для упрощения переноса приложений TCP или UDP на протокол SCTP в спецификации определен интерфейс API на базе сокетов.

11. Вопросы безопасности

В дополнение к тегам верификации и механизму cookie протокол SCTP поддерживает интеграцию с механизмами шифрования и обеспечения целостности IPSec. Спецификация SCTP не определяет новых протоколов или процедур обеспечения безопасности.

Рассматриваются вопросы расширения IPSec для поддержки многодомных хостов. Ведутся работы по использованию TLS[1] на основе протокола SCTP [4].

12. Расширения

Формат SCTP позволяет определять новые типы блоков (chunk), поля флагов и параметров для расширения возможностей протокола. Любое расширение должно базироваться на стандартном соглашении с IETF, следовательно фирменные расширения не поддерживаются протоколом.

Значения Chunk Type разбиты на 4 группы, чтобы позволить расширениям использовать предопределённые процедуры для откликов на нераспознанные типы блоков. Варианты откликов включают: отбрасывание пакета целиком, игнорирование блока, а также игнорирование с передачей отчета. Аналогичные предопределённые отклики поддерживаются и для неопознанных значений Parameter Type.

Значения Chunk Parameter Type используют независимые диапазоны для каждого Chunk Type. На практике значения, определенные в спецификации SCTP, скоординированы таким образом, что конкретный параметр будет использовать одинаковые

значения Chunk Parameter Type для всех Chunk Type. Необходимость такого выбора подтвердит практика использования протокола.

13. Литература

[1] Stewart, R., Xie, Q., Morneault, K., Sharp, C., Schwarzbauer, H., Taylor, T., Rytina, I., Kalla, M., Zhang, L. and V. Paxson, "Stream Control Transmission Protocol", RFC 2960, October 2000.

[2] Stewart, Sharp, et. al., "SCTP Checksum Change", Work in Progress[2].

[3] Ong, L., Rytina, I., Garcia, M., Schwarzbauer, H., Coene, L., Lin, H., Juhasz, I., Holdrege, M. and C. Sharp, "Framework Architecture for Signaling Transport", RFC 2719, October 1999.

[4] Jungmeier, Rescorla and Tuexen, "TLS Over SCTP", Work in Progress[3].

14. Адреса авторов

Lyndon Ong

Ciena Corporation

10480 Ridgeview Drive

[1]Transport Layer Security - защита на транспортном уровне.
[2]Работа завершена и опубликована в RFC 3309. *Прим. перев.*
[3]Работа завершена и опубликована в RFC 3436. *Прим. перев.*

Cupertino, CA 95014

EMail: lyong@ciena.com

John Yoakum

Emerging Opportunities

Nortel Networks

EMail: yoakum@nortelnetworks.com

Перевод на русский язык

Николай Малых

nmalykh@protokols.ru

15. Полное заявление авторских прав

Подтверждение

Финансирование функций RFC Editor в настоящее время обеспечивается Internet Society.