

М.Г. ВАРДАНЯН

ОБ ОДНОМ МЕТОДЕ МУЛЬТИПЛЕКСИРОВАНИЯ RTP ПОТОКОВ С ЦЕЛЬЮ ЭФФЕКТИВНОГО ИСПОЛЬЗОВАНИЯ СЕТЕВЫХ КАНАЛОВ С ОГРАНИЧЕННОЙ ПРОПУСКНОЙ СПОСОБНОСТЬЮ

Предложен метод повышения эффективности использования сетевых каналов, несущих голосовую информацию, путем мультиплексирования RTP пакетов. Данный метод, в отличие от аналогичных, учитывает особенности протокола SIP, используемого в качестве сигнального протокола для установления RTP соединений. На основе предложенного метода построен прототип программного обеспечения контроллера, позволяющий снизить в типичных случаях расходы на передачу голосовой информации более чем в три раза.

Ключевые слова: транспортный протокол реального времени, протокол инициализации сессии, мультиплексирование, контроллер.

Введение. Использование сетевых каналов в IP телефонии при передаче голосовой информации, определяемой протоколом RTP (Real-Time Transport Protocol [1]), неэффективно из-за большого объема служебной информации в заголовке RTP пакетов. В типичных приложениях соотношение объема служебной информации, передаваемой вместе с голосовыми пакетами, вдвое превышает объем собственно полезной информации. В сетевых каналах с ограниченной пропускной способностью (в спутниковых и других беспроводных каналах) такое неэффективное использование каналов приводит к снижению числа одновременно поддерживаемых голосовых соединений и, следовательно, к удорожанию услуг.

Одним из методов, позволяющим существенным образом повысить эффективность использования каналов связи, является мультиплексирование нескольких RTP потоков в один поток. В работах [2-6] предлагаются различные вариации этого метода, однако ни одна из них не затрагивает в достаточной мере вопросы реализации этого метода в среде SIP (Session Initiation Protocol, [7]) – сигнальном протоколе, получившем широкое распространение в IP телефонии и мобильных сетях третьего поколения (3G) и служащем для установления мультимедийных соединений. В отличие от предыдущих, предлагаемый в данной работе метод ориентирован на использование вместе с SIP (в качестве сигнального протокола для установления соединений) и учитывает все особенности этого протокола. Следует также отметить, что предлагаемый метод легко интегрируется в SIP инфраструктуру, сохраняя при этом совместимость с обычными SIP приложениями.

1. Краткое описание метода. Проиллюстрируем суть предлагаемого метода на примере сетевой топологии, представленной на рис. 1.

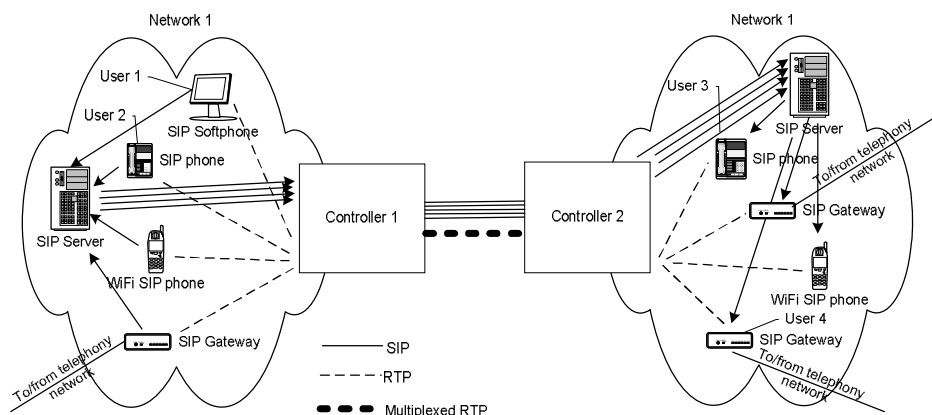


Рис. 1

Предположим, что имеются две локальные сети, соединенные каналом ограниченной пропускной способности. Соединения между SIP клиентами, расположенными в одной и той же сети, устанавливаются традиционными SIP средствами – путь сигнальных сообщений от клиента к клиенту проходит через SIP сервер этой сети, а путь RTP соединений – напрямую между клиентами.

Ввиду ограниченности пропускной способности канала между сетями связь между SIP клиентами в разных сетях предлагается устанавливать через специальные устройства – контроллеры. Все RTP потоки, исходящие из SIP клиентов первой сети, перед тем как поступить на вход канала между сетями, мультиплексируются в один RTP поток контроллером первой сети. На приемном конце этот поток распаковывается, и уже отдельные, восстановленные одноканальные RTP потоки направляются в адрес SIP клиентов второй сети. Мультиплексирование позволяет упаковать множество звуковых фреймов, поступивших с пакетами, принадлежащими к разным RTP потокам, в один RTP пакет, снижая тем самым удельный вес служебной информации, передаваемой с каждым голосовым пакетом.

Предлагаемый метод основан на следующих предпосылках:

- все SIP клиенты одной и той же локальной сети зарегистрированы на одном и том же SIP сервере, обслуживающем клиентов этой сети;
- если SIP клиент запрашивает соединение с другими клиентами той же сети, то сигнальная часть соединения проходит через SIP сервер этой сети, а голосовая часть – напрямую между клиентами;
- если SIP клиент одной сети запрашивает соединение с клиентом другой сети, то SIP сервер первой сети сконфигурирован таким образом, чтобы направлять все входящие SIP сообщения в адрес своего контроллера, расположенного на границе сети;
- принимающий контроллер, расположенный на границе другой сети, направляет все входящие от первого контроллера SIP сообщения в адрес SIP сервера своей сети.

2. Процесс установления соединения с помощью SIP. Пусть SIP клиент U_i первой сети инициирует соединение с клиентом U_j другой сети. Для этого U_i посылает сообщение INVITE (см. сообщение M1 на рис. 2) в адрес SIP сервера S_1 своей сети со следующими значениями полей сообщения (ниже приведены только те поля сообщения INVITE, которые представляют интерес для дальнейшего рассмотрения):

M1: INVITE sip: $U_j@S_2:5060$

To: < sip: $U_j@S_2$ >

From: < sip: $U_i@S_1$ >

Contact: < sip: $U_i@IP_i:P_i$ >

Content-Type: application/sdp

c=IN IP4 IP_i

m=audio p_i RTP/AVP ...,

где IP_i – это IP адрес клиента U_i ; P_i – номер UDP порта, на котором U_i ожидает ответные SIP сообщения; p_i – номер UDP порта, на котором U_i ожидает RTP поток, исходящий из U_j .

Сервер S_1 определяет, что сообщение M1 адресовано клиенту U_j в другой сети и направляет его после некоторой модификации (см. сообщение M2 на рис. 2) контроллеру C_1 своей сети.

Контроллер C_1 после анализа полученного сообщения M2 определяет дальнейшее направление соединения, а именно, адрес контроллера сети назначения и номер порта мультимплексного потока, который предполагается использовать для установления голосового соединения между клиентами U_i и U_j . На основании этой информации C_1 формирует новое INVITE сообщение M3 и отправляет его в адрес контроллера соответствующей сети (в данном примере – C_2). Текст сообщения M3, приведен ниже:

M3: INVITE sip: $U_j@S_2:5060$

To: < sip: $U_j@S_2$ >

From: < sip: $U_i@S_1$ >

Contact: < sip: $U_i@IP_i:P_i$ >

Content-Type: application/sdp

c=IN IP4 $IP(C_1)$

m=audio $p(C_1)$ RTP/AVP ...

ch=n,

где $IP(C_1)$ – IP адрес контроллера C_1 , а $p(C_1)$ – номер UDP порта мультимплексного RTP потока, который будет включать RTP поток между U_i и U_j . Дополнительное поле “ch=” введено для задания номера логического подканала в мультимплексном потоке, используемом RTP потоком между клиентами U_i и U_j .

Наряду с посылкой M3, контроллер C_1 фиксирует в своей базе данных факт создания новой сессии, а также резервирует необходимые системные ресурсы. В случае, если текущий размер мультимплексного потока достиг максимального значения, контроллер создает новый мультимплексный поток. Если и это является невозможным, то запрос клиента U_i на установление соединения с U_j отвергается.

Контроллер C_2 , после анализа полученного сообщения M3 и проверки наличия необходимых системных ресурсов, формирует новое сообщение M4 и отправляет его в адрес SIP сервера своей сети (S_2). Текст сообщения M4, приведен ниже:

M4: INVITE sip: U_j@S₂:5060

To: <sip: U_j@S₂>

From: <sip: U_i@S₁>

Contact: <sip: U_i@IP_i:P_i>

Content-Type: application/sdp

c=IN IP4 IP(C₂)

m=audio p(C₂) RTP/AVP ...

ch=n,

где IP(K₂) – IP адрес контроллера C₂; p(C₂) – номер UDP порта мультимплексного RTP потока на контроллере C₂.

Сервер S₂ определяет в своей базе данных адрес клиента U_j и переадресовывает полученное сообщение в адрес этого клиента (сообщение M₅).

Клиент U_j на полученный запрос отвечает серией сообщений, которые опустим для краткости изложения и остановимся на последнем из них – M₆ (200 OK), отправленном в адрес SIP сервера S₂. Сообщение M₆ (текст сообщения приведен ниже) содержит в себе информацию об IP адресе и номере UDP порта, на котором U_j собирается принимать поток RTP пакетов от контроллера C₂.

M6: 200 OK

To: <sip: U_j@S₂>

From: <sip: U_i@S₁>

Contact: <sip: U_i@IP_i:P_i>

Content-Type: application/sdp

c=IN IP4 IP(U_j)

m=audio p((U_j) RTP/AVP ...

Сервер S₂ ретранслирует это сообщение в адрес контроллера C₂ (см. сообщение M₇ на рис. 2). Как только сообщение M₇, подтверждающее согласие U_j на установление соединения с U_i, поступает на вход C₂, между U_j и C₂ открывается двустороннее RTP соединение. Одновременно с этим C₂ посылает в адрес C₁ сообщение M₈ (текст которого приведен ниже) и открывает логический, двусторонний подканал между двумя контроллерами.

M8: 200 OK

To: <sip: U_j@S₂>

From: <sip: U_i@S₁>

Contact: <sip: U_i@IP_i:P_i>

Content-Type: application/sdp

c=IN IP4 IP(C₂)

m=audio p((K₂) RTP/AVP ...

ch=n.

На следующем шаге контроллер C₁ после получения сообщения M₈ отправляет аналогичное сообщение в адрес сервера S₁, который ретранслирует его в адрес клиента U_i (сообщения M₉ и M₁₀ на рис. 2). После этого открывается двустороннее RTP соединение между U_i и C₁.

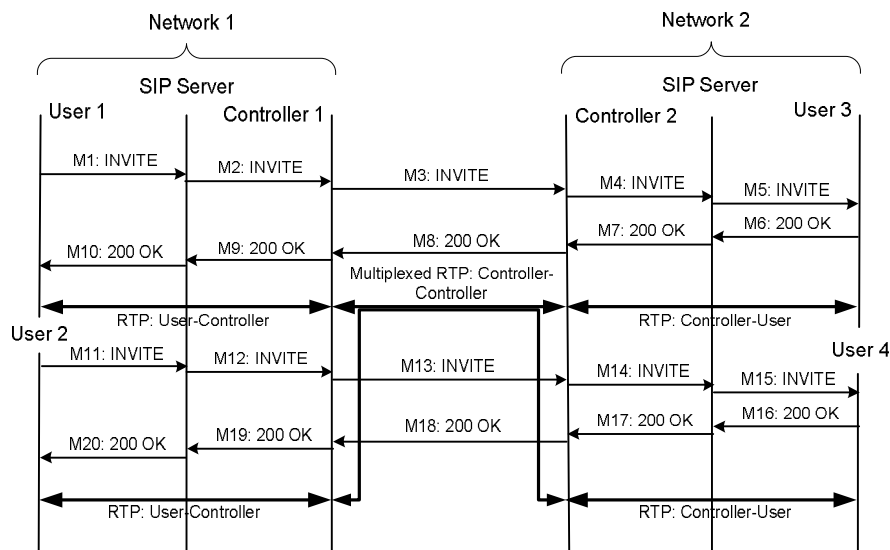


Рис. 2

3. Основные функции контроллера. В предложенной выше схеме обычное RTP соединение, напрямую соединяющее SIP клиентов, заменяется тремя соединениями. Два из них – это одноканальные соединения между конечными пользователями (SIP клиентами) и соответствующими им контроллерами, и, наконец, последнее (многоканальное) соединение – между контроллерами сетей. В точках стыка одноканальных и многоканальных соединений стоят контроллеры, которые ответственны за поддержку этих соединений (в смысле протоколов SIP и RTP), а также мультимплексирование многих одноканальных RTP соединений в одно многоканальное и наоборот.

В процессе установления соединения между клиентами контроллер выполняет следующие основные функции:

- отображение адресов конечных точек соединений с целью построения комплексного соединения между конечными точками – SIP клиентами;
- мультимплексирование пакетов, а именно, объединение пакетов, поступивших от разных клиентов, в один пакет перед его отправкой через канал мультимплексуру другой сети;
- демультимплексирование пакетов, а именно, распаковку пакетов, поступивших от контроллера другой сети, и отправку их отдельным клиентам на принимающей стороне.

Первая функция - отображение адресов, которая реализуется контроллером с помощью протокола SIP, сопоставляет адреса конечных точек отдельных соединений с целью построения комплексного соединения, связывающего SIP клиентов.

Адрес конечной точки одноканального соединения можно представить в виде пары (a, p) , где a – IP адрес конечной точки соединения, а p – номер UDP

порта конечной точки. Адрес конечной точки многоканального соединения можно представить тройкой (a, p, c) , где a – IP адрес конечной точки соединения; p – номер UDP порта соединения, а c – номер подканала многоканального соединения. Пространство адресов конечных точек соединения первого типа представляет собой множество всех пар типа (a, p) , а именно: $U = \{(a_i, p_i) \mid i=1, \dots, n\}$. Пространство адресов конечных точек соединения второго типа представляет собой множество всех троек типа (a, p, c) , а именно: $K = \{(a_i, p_i, c_j) \mid j=1, \dots, m\}$. Таким образом, комплексное соединение между клиентами U_i и U_j в разных сетях строится путем отображения адресов конечных точек отдельных соединений друг в друга, а именно: $U1 \Rightarrow U2 \Rightarrow C1 \Rightarrow C2 \Rightarrow U3 \Rightarrow U4$. Здесь отображение $U1 \Rightarrow U2$ производится контроллером с помощью протокола SIP путем “рукопожатия” с источником RTP пакетов. Отображение $U2 \Rightarrow C1$ производится внутри контроллера без взаимодействия с внешними объектами. Отображение $C1 \Rightarrow C2$ выполняется контроллером с помощью протокола SIP путем “рукопожатия” с контроллером принимающей сети. Отображение $C2 \Rightarrow U3$ выполняется внутри контроллера принимающей сети без взаимодействия с внешними объектами. И, наконец, отображение $U3 \Rightarrow U4$ выполняется контроллером принимающей сети с помощью протокола SIP путем “рукопожатия” с клиентом-приемником RTP пакетов.

Вторая функция – мультиплексирование пакетов, реализуется внутри контроллера следующим образом. Как только адрес очередной конечной точки одноканального соединения становится известным контроллеру (в результате выполнения предыдущей функции), контроллер начинает принимать все RTP пакеты, посылаемые в этот адрес, и включать их в состав мультиплексных пакетов для дальнейшей их отправки в адрес другого контроллера (в другую сеть). Структура мультиплексных пакетов имеет следующий вид:

- IP header – заголовок IP пакета;
- UDP header – заголовок UDP пакета;
- RTP header – заголовок RTP пакета;
- MRTP-1 header – мини-заголовок RTP пакета первого подканала;
- RTP-1 payload – голосовой фрейм (закодированная голосовая информация) 1-го подканала;
- MRTP-n header – мини-заголовок RTP пакета n-го первого подканала;
- RTP-n payload – голосовой фрейм n-го подканала.

Детальное рассмотрение структуры MRTP выходит за рамки данной статьи, однако здесь отметим, что MRTP несет в себе всю информацию, необходимую для восстановления исходных одноканальных соединений на принимающей стороне. Заголовок MRTP, кроме всего прочего, включает поле номера подканала, через который посылается голосовая информация, следующая за заголовком. Этот же номер подканала передается в заголовке SIP сообщения принимающему контроллеру, что позволяет ему сопоставить закодированную голосовую информацию подканала конкретному соединению между конечными клиентами (см. поле “ch” в SIP сообщении МЗ на рис. 2).

И, наконец, третья функция контроллера – распаковка пакетов, реализуется внутри принимающего контроллера и заключается в следующем. Каждый мультиплексированный пакет, полученный контроллером, распаковывается, и затем формируются отдельные RTP пакеты – один пакет на каждое

одноканальное соединение между принимающим контроллером и принимающим SIP клиентом. Для сопоставления отдельных голосовых фреймов с одноканальными соединениями используются значения поля подканала в MRTP и поля “ch” в соответствующем SIP сообщении.

4. Реализация функций контроллера. Вышеперечисленные функции мультиплексирования и отображения адресов реализованы в прототипе контроллера, функционирующем в среде операционной системы Linux. Для обработки SIP сообщений использован “Back-To-Back User Agent” (B2BUA) от Vovida.org¹. Прототип может одновременно поддерживать несколько мультиплексных потоков с числом RTP подканалов не более 256 в каждом из них и позволяет существенно снизить накладные расходы на передачу голосовой информации при использовании в сетях, имеющих топологию, подобную той, которая представлена на рис. 1.

На рис. 3 представлены примерные графики зависимости требуемой полосы пропускания сетевого канала между двумя контроллерами от количества активных соединений при использовании предлагаемого метода мультиплексирования (нижний график) и без него - традиционным способом (верхний график). Приведенные численные значения верны при условии использования одного и того же кодека (G.729) и одного и того же размера голосовых фреймов (20 миллисекунд) на всех соединениях.¹

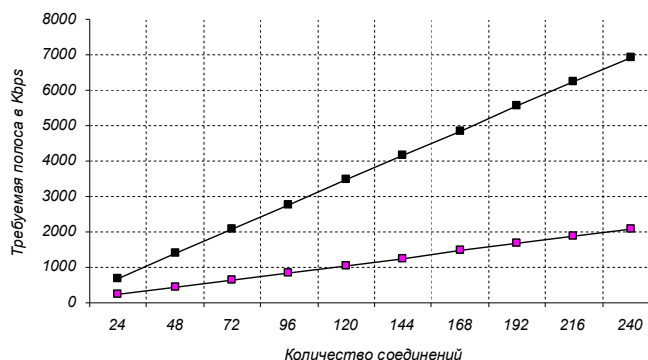


Рис. 3

Заключение. В работе предложен метод повышения эффективности использования сетевых каналов, несущих голосовую информацию, путем мультиплексирования RTP пакетов. В отличие от аналогичных, данный метод учитывает особенности протокола SIP, используемого в качестве сигнального протокола для установления RTP соединений. На основе предложенного метода построен прототип программного обеспечения контроллера, позволяющий снизить в типичных случаях расходы на передачу голосовой информации более чем в три раза.

¹ Vovida.org - a communications community site dedicated to providing a forum for open source software used in datacom and telecom environments (www.vovida.org)

СПИСОК ЛИТЕРАТУРЫ

1. Schulzrinne H., Casner S. RTP: A Transport Protocol for Real-Time Applications // IETF Request for Comments: 3550. - July, 2003.
2. Subbiah B., Sengodan S. User Multiplexing in RTP payload between IP Telephony Gateways // IETF, Work in progress. - August, 1998.
3. Menth M. Carrying wireless traffic in UMTS over IP using realtime transfer protocol multiplexing // In 12th ITC Specialist Seminar (Lillehammer, Norway). - 2000. - P. 13-25.
4. El-Khatib K., Luo G., Bochmann G., Pinjiang Feng Multiplexing Scheme for RTP Flows between Access Routers // IETF, Work in progress. - October, 1999.
5. Vardanyan M. IP voice stream multiplexing method with minimal overhead // CSIT, Proceedings of the conference (Yerevan, Armenia). - September, 2005. - P. 553-557.
6. Sze H. P., Soung C. Liew, Jack Y. B. Lee, and Danny C. S. Yip A Multiplexing Scheme for H.323 Voice-Over-IP Applications // IEEE Journal on Selected Areas in Communications. -2002. -V.20.
7. Rosenberg J., Schulzrinne H. SIP: Session Initiation Protocol // IETF Request for Commnts: 3261, - June, 2002.

ԵրНИИАСУ. Материал поступил в редакцию 10.03.2006.

Մ.Հ. ՎԱՐԴԱՆՅԱՆ

ՍԱՀՄԱՆԱՓԱԿ ԹՈՂՈՒՆԱԿՈՒԹՅԱՄԲ ՑԱՆՑԱՅԻՆ ԿԱՊՈՒՂԻՆԵՐԻ ԱՐԴՅՈՒՆԱՎԵՏ
ՕՏԱՈՐԾՄԱՆ ՆՊԱՏԱԿՈՎ RTP ՀՈՍՔԵՐԻ ՄՈՒԼՏԻՊԼԵՔՍԱՎՈՐՄԱՆ ՄԻ ՄԵԹՈԴԻ ՄԱՍԻՆ

Առաջարկվում է ձայնային ինֆորմացիա փոխանցող ցանցային կապուղիների օգտագործման արդյունավետությունը մեծացնելու նպատակով RTP փաթեթների մուլտիպլեքսավորման մեթոդ: Ի տարբերություն նմանօրինակ մեթոդների՝ առաջարկվող մեթոդը հաշվի է առնում SIP արձանագրության առանձնահատկությունները, որոնք օգտագործվում են որպես ազդանշանային արձանագրություն՝ RTP միացումների հաստատման ժամանակ: Հիմնվելով առաջարկված մեթոդի վրա կառուցվել է ծրագրային կոնտրոլլերի նախատիպ, որը թույլ է տալիս ավելի քան երեք անգամ փոքրացնել ձայնային ինֆորմացիայի փոխանցման ծախսերը տիպային դեպքերում:

Առանցքային բաներ. ռեալ ժամանակի տրանսպորտային արձանագրություն, սեսիայի հաստատման արձանագրություն, մուլտիպլեքսավորում, կոնտրոլլեր:

M.H. VARDANYAN

ON A METHOD OF MULTIPLEXING RTP FLOWS FOR EFFECTIVE USAGE OF NETWORK CHANNELS WITH LIMITED CAPACITY

A method for increasing the efficiency of network channel usage bearing voice information by multiplexing RTP packets is proposed. Unlike the analogous ones, this method takes into account SIP protocol, used as a signaling protocol for establishment of RTP connections. Based on the proposed method, a prototype of software controller allowing to reduce the expenses in typical cases to transfer voice information of provision for more than three times is discussed.

Keywords: Real-time Transport Protocol, Session Initiation Protocol, multiplexing, controller.