Министерство образования Республики Беларусь
Учреждение образования
«Белорусский государственный университет
информатики и радиоэлектроники»

Кафедра сетей и устройств телекоммуникаций

М. Ю. Хоменок

СИГНАЛИЗАЦИЯ НА СЕТЯХ ПЕРЕДАЧИ ДАННЫХ С ПАКЕТНОЙ КОММУТАЦИЕЙ. ПРОТОКОЛ SIP

Методическое пособие
по курсу
«Сетевые технологии и сигнализация в телекоммуникациях»
для студентов специальностей
1-45 01 03 «Сети телекоммуникаций»,
1-45 01 05 «Системы распределения мультимедийной информации»,
1-98 01 02 «Защита информации в телекоммуникациях»
всех форм обучения

Библиотека БГУИР

Минск БГУИР 2010

УДК 654.9+004.724.4(076)
ББК 32.885я73
Х76

Р е ц е н з е н т:

доцент кафедры «Системы телекоммуникаций»
учреждения образования «Белорусский государственный университет информатики и радиоэлектроники», кандидат технических наук В. Н. Урядов

Хоменок, М. Ю.

Х76 Сигнализация на сетях передачи данных с пакетной коммутацией. Протокол SIP : метод. пособие по курсу «Сетевые технологии и сигнализация в телекоммуникациях» для студ. спец. 1-45 01 03 «Сети телекоммуникаций», 1-45 01 05 «Системы распределения мультимедийной информации», 1-98 01 02 «Защита информации в телекоммуникациях» всех форм обуч. / М. Ю. Хоменок. – Минск : БГУИР, 2011. – 84 с. : ил.

ISBN 978-985-588-613-8.

В методическом пособии рассмотрены общие принципы построения протокола SIP, формат и структура сигнальных сообщений, а также основные сценарии установления соединений (8 приложений).

Пособие предназначено для практических занятий студентам специальностей 1-45 01 03 «Сети телекоммуникаций», 1-45 01 05 «Системы распределения мультимедийной информации» и 1-98 01 02 «Защита информации в телекоммуникациях» при изучении особенностей взаимодействия на основе сообщений протокола SIP терминального и сетевого оборудования сетей передачи данных с пакетной коммутацией.

УДК 654.9+004.724.4(076)
ББК 32.885я73

ISBN 978-985-588-613-8

© Хоменок М. Ю., 2011
© УО «Белорусский государственный университет информатики и радиоэлектроники», 2011

Библиотека БГУИР

ВВЕДЕНИЕ

Сети с коммутацией каналов (ISDN-сети) и сети с коммутацией пакетов (IP-сети) длительное время существовали практически независимо друг от друга и использовались для различных целей: телефонные сети – в основном для передачи голосовой информации, а IP-сети – для передачи данных.

Значительным ускорителем в конвергенции телекоммуникационных сетей явилась технология VoIP (Voice over IP), позволившая передавать «голос» поверх IP-сетей, которая с помощью специальных устройств – шлюзов, объединила телефонные сети и сети передачи данных в единую инфокоммуникационную сеть, определяющую современную объединенную транспортную платформу для передачи различного вида пользовательского трафика: данных, речи или изображений.

В то же время выход за границы возможностей качества обслуживания мультимедийного трафика в традиционных сетях привел к качественному преобразованию всей сетевой структуры, в результате чего появилась концепция мультисервисной сети передачи данных следующего поколения NGN (Next Generation Network).

Общими характеристиками NGN, определенными ITU и ETSI, являются разделение функций переноса информации и функций управления переносом информации через сеть, а также отделение функций услуг и приложений от функций транспортной сети передачи, в качестве инфраструктуры которой используется сеть IP/MPLS на базе магистральных IP-маршрутизаторов и коммутаторов. Это определило распределенную архитектуру сети NGN, организованную по функциональному принципу, в которой связь между компонентами осуществляется через открытые интерфейсы.

Сегодня в IP-сетях предоставление мультисервисных услуг осуществляется конкурирующими между собой тремя основными семействами протоколов: H.323, SIP и MGCP. Протоколы всех трех перечисленных семейств регламентируют передачу медиатрафика в IP-сетях и управление мультимедиавызо-

Библиотека БГУИР

вами, но при этом реализуют три различных подхода к построению систем сигнализации.

Исторически первым был набор рекомендаций H.xxx, предложенный ITU-T для организации мультимедийной связи. Соответствующие работы велись с начала 90-х г. XX столетия, когда технология IP-телефонии была неизвестна. Рекомендации H.xxx являются частью стандартов, которые рассматривают возможности организации мультимедийной связи по множеству сетей. Одной из первых была разработана рекомендация H.320, описывающая системы видеоконференц-связи в цифровых сетях с коммутацией каналов (ISDN).

В 1996 г. была предложена рекомендация H.323 для сетей с коммутацией пакетов, которая создавалась под сильным влиянием H.320, и по сути это было попыткой перенести телефонную сигнализацию ISDN Q.931 на IP-соединения путем «наложения» традиционной телефонии на сети передачи данных. Вследствие этого H.323 является в определенной степени чужеродным включением для сетей на базе TCP/IP. Она определяет способы организации мультимедийных конференций, охватывая сервисы передачи голоса, видео и компьютерных данных в локальных сетях с негарантированной доставкой. Большинство современных сетей относится именно к такому типу. Примерами могут служить сети на базе протоколов TCP/IP, IPX в средах Ethernet, Fast Ethernet и Token Ring.

Протокол SIP (Session Initiation Protocol) – протокол инициирования сессии, является протоколом прикладного уровня, разработанным рабочей группой MMUSIC (Multiparty Multimedia Session Control) по управлению многоточечными сеансами мультимедийной связи целевого комитета IETF (Internet Engineering Task Force) по инженерным проблемам сети Интернет, – главного органа стандартизации Интернета. Первые спецификации протокола SIP представлены IETF в 1999 г. в рабочих предложениях RFC 2543 (Requests for Comments).

SIP похож на протокол HTTP (Hypertext Transfer Protocol – протокол передачи гипертекста), поскольку разрабатывался по образу широко известных

Библиотека БГУИР

спецификаций HTTP и SMTP (Simple Mail Transfer Protocol – протокол передачи электронной почты). По существу это клиент-серверный протокол, работа которого состоит из последовательности запросов и ответов, причем все заголовки сообщений SIP передаются в формате ASCII-текста (American Standard Code for Information Interchange – американский стандартный код информационного обмена), а потому легко читаются.

Структура протокола SIP включает несколько комплементарных протоколов, которые служат для реализации дополнительных возможностей. Наиболее важный из них протокол описания сессий SDP (Session Description Protocol, RFC 2327), выполняющий согласование таких параметров сеанса связи, как виды кодеков, номера UDP-портов и т. д. SDP обеспечивает изменение параметров сеанса связи «на ходу» во время сеанса. Перенос сообщений SDP основан на протоколе анонса сервисов SAP (Service Announcement Protocol RFC 2974).

Другой пример комплементарного протокола – SIMPLE (SIP for Instant Messaging and Presence Levering Extension). Фактически это расширение SIP, служащее для рассылки «мгновенных» сообщений (instant messaging) и для предоставления информации о событиях (presence). Первое обеспечивает обмен в реальном времени короткими сообщениями (как ICQ на ПК или SMS в сетях GSM), второе позволяет определять состояние абонента, например, свободен, занят и т. д. Благодаря этим двум функциям SIP позволяет реагировать на события, а также рассылать сообщения «по событию».

Поддержка SIP языка CPL (Call Processing Language – язык обработки звонков) на основе XML (Extensible Markup Language – расширяемая спецификация языка для создания Web-страниц), используемого для написания телефонных скриптов, позволяет определить действия в зависимости от вызывающего абонента (например, кто, кому, когда и по каким вопросам устанавливает соединение) или от состояния линии вызываемого абонента (например, не ответил, ответил не тот и т. д).

Библиотека БГУИР

Это дает возможность, используя интеллектуальные системы, с помощью SIP настроить свое SIP-окружение и управлять доступом других абонентов к своему номеру, осуществляя персональную маршрутизацию звонков, например, в зависимости от времени суток, дней недели, приоритетности вызова или местоположения, по одному из доступных средств – рабочему, домашнему или мобильному терминалам, отправку сообщений по электронной почте, факсу или на автоответчик.

Кроме того, функциональность протокола, конвергенция сетей и мультимедийность трафика позволяют организовать универсальный почтовый ящик, который будет хранить факсимильные, речевые, текстовые и видеосообщения. Управление такими услугами осуществляется самим абонентом – достаточно получить доступ к «личному кабинету», имеющему Web-интерфейс, и весь спектр необходимых услуг станет незаменимым инструментом в построении персональной коммуникационной сети пользователя. При этом SIP как универсальное средство для создания, модификации и завершения сессий работает независимо от используемых транспортных протоколов и типа устанавливаемой сессии, а взаимодействие сетей с различной транспортной платформой осуществляется на основе протокола MGCP.

Протокол MGCP (Media Gateway Control Protocol) – протокол управления медиашлюзом, включает группу спецификаций SGCP, IPDC, MGCP, MEGACO, H.248. Его формирование началось с создания двух протоколов - SGCP (Simple Gateway Control Protocol, разработанного компаниями Bellcore и Cisco Systems) и IPDC (Internet Protocol for Device Control, разработанного компанией Level 3 при участии многих производителей). Затем SGCP и IPDC были объединены в один протокол, получивший название MGCP, первая версия которого (RFC 2705) опубликована в октябре 1999 г. В дальнейшем эволюция MGCP и сотрудничество над его разработкой IETF и МСЭ привели к появлению протоколов MEGACO (в рамках IETF) и H.248 (в рамках МСЭ).

В этом смысле MGCP – единственный из трех протоколов, работа над которым выполнялась обеими организациями совместно. В то же время сущест-

Библиотека БГУИР

вуют и другие реализации протоколов, подобных MGCP, например, фирменный протокол Cisco Systems SSCP (Skinny Station Control Protocol), с помощью которого УАТС Cisco Call Manager управляет IP-телефонами.

Алгоритм взаимодействия на основе MGCP состоит в том, что управление сигнализацией (Call Control) сосредоточено на центральном управляющем устройстве, называемом контроллером сигнализаций (Call Agent, CA), и полностью отделено от медиапотоков. Эти потоки обрабатываются «неинтеллектуальными» шлюзами, которые способны исполнять лишь ограниченный набор команд от управляющего устройства. Манипулируя наборами команд, можно получать специализированные шлюзы: транковые (Trunking gateways, TGW), абонентские (Residential gateways, RGW), шлюзы доступа (Access gateways, AGW) и т. д.

Взаимодействие абонентов с различной сетевой платформой требует сопряжения систем сигнализаций, в частности передачи сообщений сигнализации ОКС№7 (SS7) по протоколу IP. Описание взаимодействия SIP с SS№7 выполнено в SIP-T (SIP extention for Telephony), определяющем перенос сообщений SS7 между контроллерами сигнализации в виде объектов MIME (Multipurpose Internet Mail Extensions), используемых для передачи мультимедийной информации посредством электронной почты. Для решения этого рода задач Intel предлагает работающий по стандарту SIGTRAN (Signaling Transport) сигнальный шлюз SS7 over IP - SIGTRAN gateway (SGW).

Различия функциональных особенностей трех протоколов обусловлены изменениями представлений о пути развития телекоммуникаций в разное время. При этом H.323 – это технологически устоявшийся, широко распространенный протокол IP-телефонии для операторских сетей и межоператорского обмена, т. е. может рассматриваться как «транзитный» протокол. В свою очередь SIP – протокол предоставления расширенных мультисервисных услуг в IP-сетях, быстро развивается как «абонентский» протокол. Что касается MGCP, то он ориентирован прежде всего на организацию больших операторских узлов сопряжения IP-сетей с ТфОП и сетями SS7.

Библиотека БГУИР

1. ОБЩИЕ ПРИНЦИПЫ ПРОТОКОЛА SIP

В Интернет-сети существует множество приложений, которые требуют создания сессии и управления ею для осуществления обмена информацией между участниками сессии. Техническая реализация таких приложений осложняется необходимостью учёта всех возможностей пользователей: пользователи могут перемещаться от одной оконечной точки доступа к сети к другой, они могут быть адресуемы одновременно по нескольким направлениям и/или одновременно обмениваться информацией различного типа.

Для передачи разнотипной мультимедийной информации, например, такой как аудио-, видео- и текстовые сообщения, в масштабе реального времени разработано большое число протоколов. Протокол SIP работает в сочетании с ними, позволяя оконечным точкам сети Интернет (называемыми агентами пользователя) обнаружить друг друга и согласовать параметры сессии, которую они желают установить. Для обнаружения местоположения будущих участников сессии и других функций протокол SIP предусматривает создание инфраструктуры сетевых узлов (называемых прокси-серверами), с которыми агенты пользователя взаимодействуют, отсылая сообщения регистрации, приглашения к сеансу связи и другие запросы.

SIP является управляющим протоколом прикладного уровня, который предназначен для установления, модификации и завершения мультимедийных сеансов связи с одним или несколькими участниками. Эти сеансы могут включать в себя мультимедиа-конференции, дистанционное обучение, телефонные звонки по Интернет и распространение мультимедийного информационного контента. Протокол SIP также может приглашать участников к уже существующим сессиям, таким как мультикаст-конференции.

SIP прозрачно поддерживает преобразование адресов и сервисы переадресации, которые обеспечивают персональную мобильность на основе единого идентификатора в независимости от местоположения пользователей в сети.

Библиотека БГУИР

Средства протокола SIP обеспечивают следующие аспекты по работе с мультимедийными сеансами связи:

– определение оконечной системы, которая будет использована для коммуникаций;

– определение готовности вызываемого пользователя принять вызов;

– установление типа мультимедийной информации и её параметров для последующего использования;

– согласование участниками параметров сессии;

– управление сеансом связи путем включения, переключения и завершения сессии, изменения сессионных параметров и предоставления сервисов.

Протокол SIP является одним из компонентов, который может быть использован совместно с другими протоколами IETF для построения законченной коммуникационной архитектуры, специализированной для передачи мультимедийного трафика. Как правило, эти архитектуры включают такие протоколы, как RTP – для передачи трафика реального времени, RTSP – для контроля доставки потоковой информации, MEGACO – для управления шлюзами с ТфОП и SDP – для описания мультимедийных сессий, в сочетании с которыми SIP предоставляет сервисы пользователям. Однако функциональность и работа SIP в широком смысле не зависит ни от одного из этих протоколов.

Протокол SIP обеспечивает следующие системные характеристики:

– персональную мобильность пользователей, определяемую возможностью перемещения пользователя и получения сервиса вне зависимости от его местоположения (как, например, электронная почта) на основе персонального идентификатора пользователя, по которому он может быть найден;

– масштабируемость сети, характеризуемую возможностью увеличения количества элементов сети при её расширении. Серверная структура сети, построенная на базе протокола SIP, в полной мере отвечает этому требованию;

Библиотека БГУИР

– расширяемость протокола, обеспечиваемую возможностью дополнения протокола для введения новых услуг и адаптации протокола для работы с различными приложениями;

– интеграцию в стек существующих протоколов Интернет путем включения протокола SIP в глобальную архитектуру мультимедиа, разработанную комитетом IETF. Эта архитектура включает в себя также протокол резервирования ресурсов (Resource Reservation Protocol – RSVP, RFC 2205), транспортный протокол реального времени (Real-Time Transport Protocol – RTP, RFC 1889), протокол передачи потоковой информации в реальном времени (Real-Time Streaming Protocol – RTSP, RFC 2326), протокол описания параметров связи (Session Description Protocol – SDP, RFC 2327). Однако функции протокола SIP не зависят ни от одного из этих протоколов;

– взаимодействие с другими протоколами сигнализации. Протокол SIP может быть использован совместно с протоколом H.323. Возможно также взаимодействие протокола SIP с системами сигнализации ISDN: цифровой абонентской DSS1 и межстанционной общеканальной CCS№7. Причем для облегчения процедуры взаимодействия сигнальные сообщения протокола SIP могут переносить не только специфичный SIP-адрес, но и телефонный номер формата E.164 или любого другого формата.

Кроме того, протокол SIP наравне с протоколами H.323 и ISUP/IP может применяться для синхронизации работы устройств управления шлюзами. В этом случае он должен взаимодействовать с протоколом MGCP. Другой важной особенностью протокола SIP является то, что он приспособлен к организации доступа пользователей сетей IP-телефонии к услугам интеллектуальных сетей.

Структура сообщений SIP не зависит от выбранной транспортной технологии. Но в то же время предпочтение отдается технологии маршрутизации пакетов IP и протоколу UDP. При этом необходимо создать дополнительные механизмы для надежной доставки сигнальной информации. К таким механизмам

Библиотека БГУИР

относятся: повторная передача информации при ее потере, подтверждение приема и другие.

Следует отметить, что сигнальные сообщения могут переноситься не только протоколом транспортного уровня UDP, но и протоколом TCP. Протокол UDP позволяет быстрее, чем TCP, доставлять сигнальную информацию (даже с учетом повторной передачи неподтвержденных сообщений), а также вести параллельный поиск местоположения пользователей и передавать приглашения к участию в сеансе связи в режиме многоадресной рассылки. В свою очередь протокол TCP упрощает работу с межсетевыми экранами (firewall), а также гарантирует надежную доставку данных. При использовании протокола TCP разные сообщения, относящиеся к одному вызову, либо могут передаваться по одному TCP-соединению, либо для каждого запроса и ответа на него может открываться отдельное TCP-соединение. В табл. 1.1 показано место, занимаемое протоколом SIP в стеке протоколов TCP/IP.

Таблица 1.1

Протокол SIP в стеке протоколов TCP/IP

Протокол	Уровень модели IOS
SIP	Прикладной
TCP и UDP	Транспортный
IP4 IP6	Сетевой
PPP, ATM, Ethernet	Канальный
UTP5, SDH, DDH, V.34 др.	Физический

По сети с маршрутизацией пакетов IP может передаваться пользовательская информация любого вида. Поэтому при организации связи между терминалами пользователей необходимо известить встречную сторону о типе информации, которая может приниматься/передаваться, алгоритме ее кодирования и адрес, на который следует передавать информацию. Таким образом, одним из обязательных условий организации связи при помощи протокола SIP является обмен

Библиотека БГУИР

между сторонами данными об их функциональных возможностях. Для этой цели чаще всего используется протокол описания сеансов связи SDP. Поскольку в течение сеанса связи может производиться его модификация, предусмотрена передача сообщений SIP с новыми описаниями сеанса средствами SDP.

Для передачи речевой информации комитет IETF предлагает использовать протокол RTP, но сам протокол SIP не исключает возможность применения для этих целей других протоколов.

В протоколе SIP не реализованы механизмы управления потоками информации и предоставления гарантированного качества обслуживания. Кроме того, протокол SIP не предназначен для передачи пользовательской информации, в его сообщениях может переноситься информация лишь ограниченного объёма. При переносе через сеть слишком большого сообщения SIP не исключена его фрагментация на уровне IP, что может повлиять на качество передачи информации.

Протокол SIP предусматривает организацию конференций трех видов:

– в режиме многоадресной рассылки (мультикастинг), когда информация передается на один мультикаст-адрес, а затем доставляется сетью конечным адресатам;

– в режиме «точка-точка» участников конференции с устройством управления конференцией (MCU), которое обрабатывает принимаемую информацию (т. е. смешивает или коммутирует), а затем рассылает участникам конференции;

– режиме «точка-точка» путем соединения пользователей «каждого с каждым».

При этом протокол SIP дает возможность присоединения новых участников к уже существующему сеансу связи и соответственно двусторонний сеанс может перейти в конференцию.

Библиотека БГУИР

2. ФУНКЦИОНАЛЬНАЯ МОДЕЛЬ ПРОТОКОЛА SIP

2.1. Архитектура сети и адресация SIP

Архитектура протокола SIP в значительной степени определяется заимствованными принципами протокола переноса гипертекста (HTTP) и взаимодействия сторон по типу «клиент-сервер» или «запрос/ответ». Клиент выдает запросы, в которых указывает, что он желает получить от сервера. Сервер принимает запрос, обрабатывает его и выдает ответ, который может содержать уведомление об успешном выполнении запроса, уведомление об ошибке или информацию, затребованную клиентом.

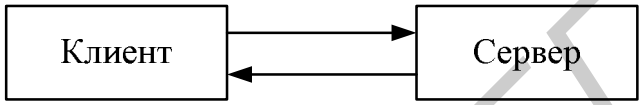


Рис. 2.1. Принцип взаимодействия «клиент-сервер»

Сети SIP строятся из следующих основных элементов: SIP-терминалов, прокси-серверов, серверов переадресации, регистрации и определения местоположения. Фрагмент типовой архитектуры сети SIP приведен на рис. 2.2.

Рис. 2.2. Архитектура сети SIP

Библиотека БГУИР

SIP-серверы, представленные на рис. 2.2, являются отдельными функциональными элементами сети. Физически они могут быть реализованы на базе серверов локальной сети, которые, помимо выполнения своих основных функций, будут также обрабатывать SIP-сообщения.

В свою очередь терминалы могут быть двух типов: ПК, оснащённый необходимыми аппаратными средствами и программным модулем SIP (UA-user agent), или SIP-телефон, подключающийся непосредственно к ЛВС.

Управление процессом обслуживания вызова распределено между разными элементами сети SIP, но основным функциональным элементом, реализующим функции управления соединением, является терминал. Остальные элементы сети отвечают за маршрутизацию вызовов, а в некоторых случаях предоставляют дополнительные услуги.

В случае если клиент и сервер реализованы в оконечном оборудовании пользователя, они называются соответственно клиентом агента пользователя – User Agent Client (UAC) – и сервером агента пользователя – User Agent Server (UAS). Если в устройстве присутствуют и сервер UAS, и клиент UAC, то оно называется агентом пользователя – User Agent (UA) и по своей сути представляет собой терминальное оборудование SIP.

Прокси-сервер представляет интересы пользователя в сети. Он принимает запросы, обрабатывает их и в зависимости от типа запроса выполняет определенные действия. Это может быть поиск и вызов пользователя, маршрутизация запроса, предоставление услуг и т. д. Прокси-сервер состоит из клиентской и серверной частей, поэтому может принимать вызовы, инициировать собственные запросы и возвращать ответы. Предусмотрено два типа прокси-серверов – с сохранением состояний (stateful) и без сохранения состояний (stateless).

Так как пользователь может перемещаться в пределах сети, то необходим механизм определения его местоположения в текущий момент времени. Для хранения текущего адреса пользователя служит сервер определения местоположения пользователей, представляющий собой базу данных адресной ин-

Библиотека БГУИР

формации, с которым прокси-сервер взаимодействует. Кроме постоянного адреса пользователя, в этой базе данных может храниться один или несколько текущих адресов.

Для регистрации пользователей в сети и внесения изменений в базу данных сервера определения местоположения конкретного домена по инициативе пользователя используется регистрирующий сервер, называемый сервер-регистр. Путем обращения к нему пользователь может указать адрес (адреса), по которому его можно найти в текущее время. Как правило, сервер-регистер совмещается с прокси-сервером домена и выполняется в виде модуля регистрации при прокси-сервере.

Сервер переадресации предназначен для определения текущего адреса вызываемого пользователя. Для реализации этой функции сервер переадресации должен воспользоваться сервисом определения местоположения. Прокси-серверы при необходимости могут работать в качестве серверов переадресации.

Для организации взаимодействия с существующими приложениями и для обеспечения мобильности пользователей IP-сетей протокол SIP использует адрес, подобный адресу электронной почты. В качестве адресов рабочих станций используются специальные универсальные указатели ресурсов – URL (Universal Resource Locators), так называемые SIP URL.

SIP-адреса бывают четырех типов: имя@домен, имя@хост, имя@IP-адрес, номер телефона@шлюз. Таким образом, адрес состоит из двух частей. Первая часть – это имя пользователя, зарегистрированного в домене или на рабочей станции. Если вторая часть адреса идентифицирует какой-либо шлюз, то в первой указывается телефонный номер абонента. Во второй части адреса указывается имя домена, рабочей станции или шлюза.

Обычно UAC осуществляет запрос, используя не IP-адрес, а доменное имя UAS. Для определения IP-адреса прокси-сервера зоны места назначения используется служба доменных имен – Domain Name Service (DNS).

Библиотека БГУИР

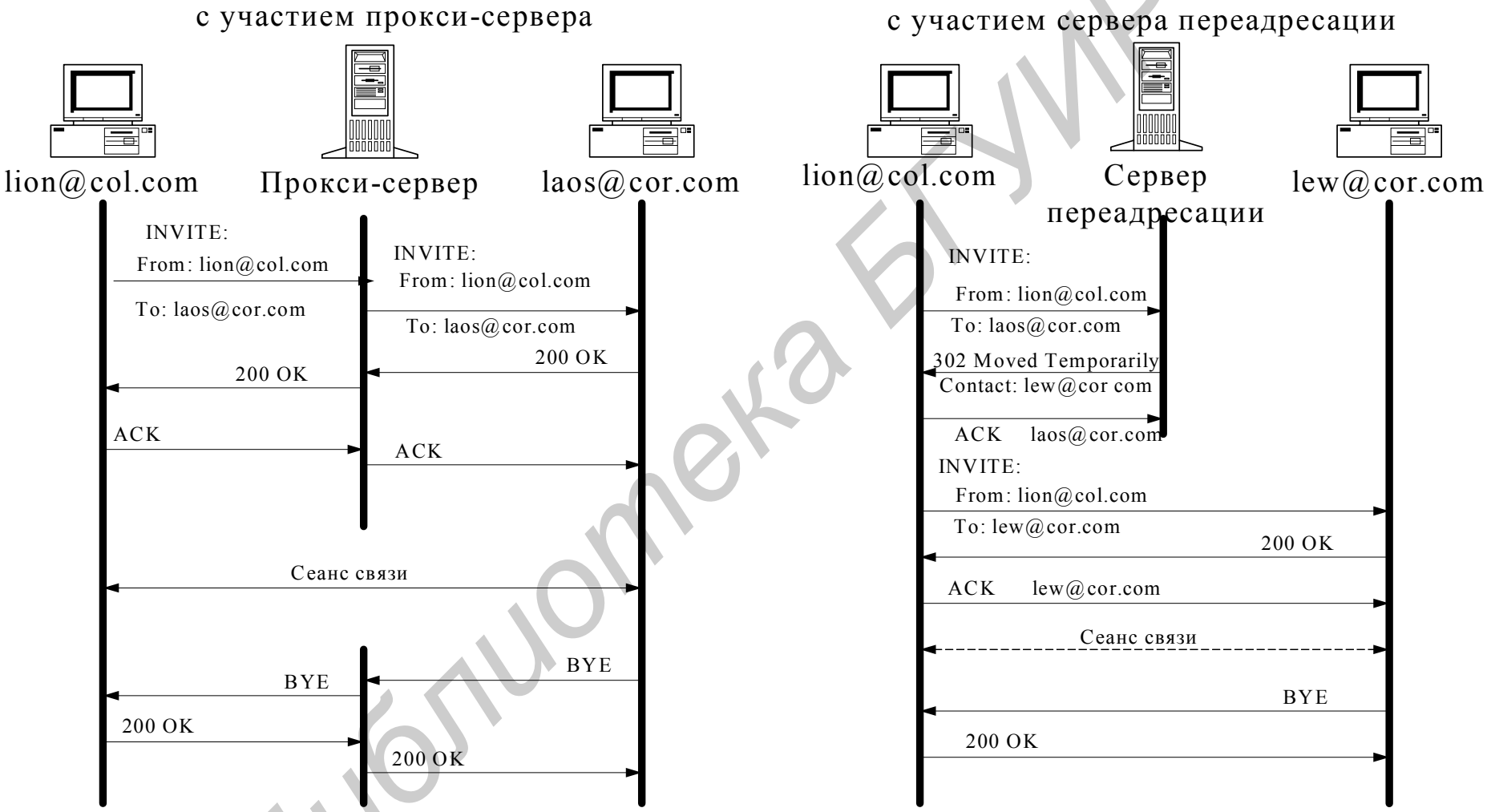


Рис. 2.3. IP-адресация сообщений протокола SIP с участием прокси-сервера и сервера переадресации

Библиотека БГУИР

Если же во второй части SIP-адреса размещается IP-адрес, то с рабочей станцией можно связаться напрямую. В начале SIP-адреса ставится слово «sip:», указывающее, что это именно SIP-адрес.

Синтаксически SIP-адреса представляются в виде:

sip: als@rts.loniis.by;

sip: user1@192.168.100.175;

sip: 294-75-45@gateway.by.

IP-адресация сообщений показана на рис. 2.3.

2.2. Сообщения протокола SIP

2.2.1. Формат сообщений SIP

Протокол SIP является текстовым протоколом, использующим набор символов ISO 10646 в кодировке UTF-8. Сообщения SIP представляют собой либо запрос от клиента серверу, либо ответ сервера клиенту. Запросы и ответы используют один базовый формат сообщения несмотря на различия в наборе символов и синтаксисе.

Формат сообщений обоих типов показан на рис. 2.4 и содержит стартовую строку, одно или несколько полей заголовков (message-header), пустую строку для обозначения конца поля заголовков и при необходимости тело сообщения.

Стартовая строка представляет собой начальную строку любого сообщения SIP. Если сообщение является запросом, в этой строке указывается тип запроса, адресат и номер версии протокола. Если сообщение является ответом на запрос, в стартовой строке указывается номер версии протокола, тип ответа и его короткая расшифровка, предназначенная только для пользователя.

Заголовки сообщений служат для передачи информации об отправителе, адресате, пути следования и других сведений, т. е. переносят необходимую для обслуживания данного сообщения информацию.

Сообщения протокола SIP могут также содержать тело сообщения, формат и синтаксис полей, который определяется протоколом описания сессии SDP в соответствии с RFC2327. Описание сессии включает параметры мульти-

Библиотека БГУИР

медиа-сеанса связи в приглашении к началу сеанса связи и в других мультимедийных сеансах при установки связи и согласовании параметров, используя три типа полей, включающих атрибуты их описания:

1. Описание сеанса:

   v= (версия протокола);

   o= (идентификаторы создателя/владельца и сессии);

   s= (имя сессии);

   i=* (информация о сессии);

   u=* (URI-описания);

   e=* (email адрес);

   p=* (номер телефона);

   c=* (информация для соединения: не требуется, если она есть в описании всех медиаданных);

   b=* (информация о занимаемой полосе пропускания канала связи).

   Одна и более строк с описанием параметров времени.

   z=* (корректировка часового пояса/установка для временной зоны);

   k=* (ключ шифрования);

   a=* (одна или несколько строк с описанием атрибутов сессии).

2. Описание параметров времени:

   t= (время активности сеанса);

   r=* (число попыток повторов, от нуля и больше).

3. Описание данных передачи мультимедиа:

   m= (название медиаданных и адрес их передачи);

   i=* (заголовок медиаданных);

   c=* (информация для соединения: не обязательна, если описана в параметрах сеанса);

   b=* (информация о занимаемой полосе пропускания канала связи);

   k=* (ключ шифрования);

   a=* (от нуля и более строк с описанием атрибутов медиаданных).

Необязательные элементы отмечены символом «*».

Библиотека БГУИР

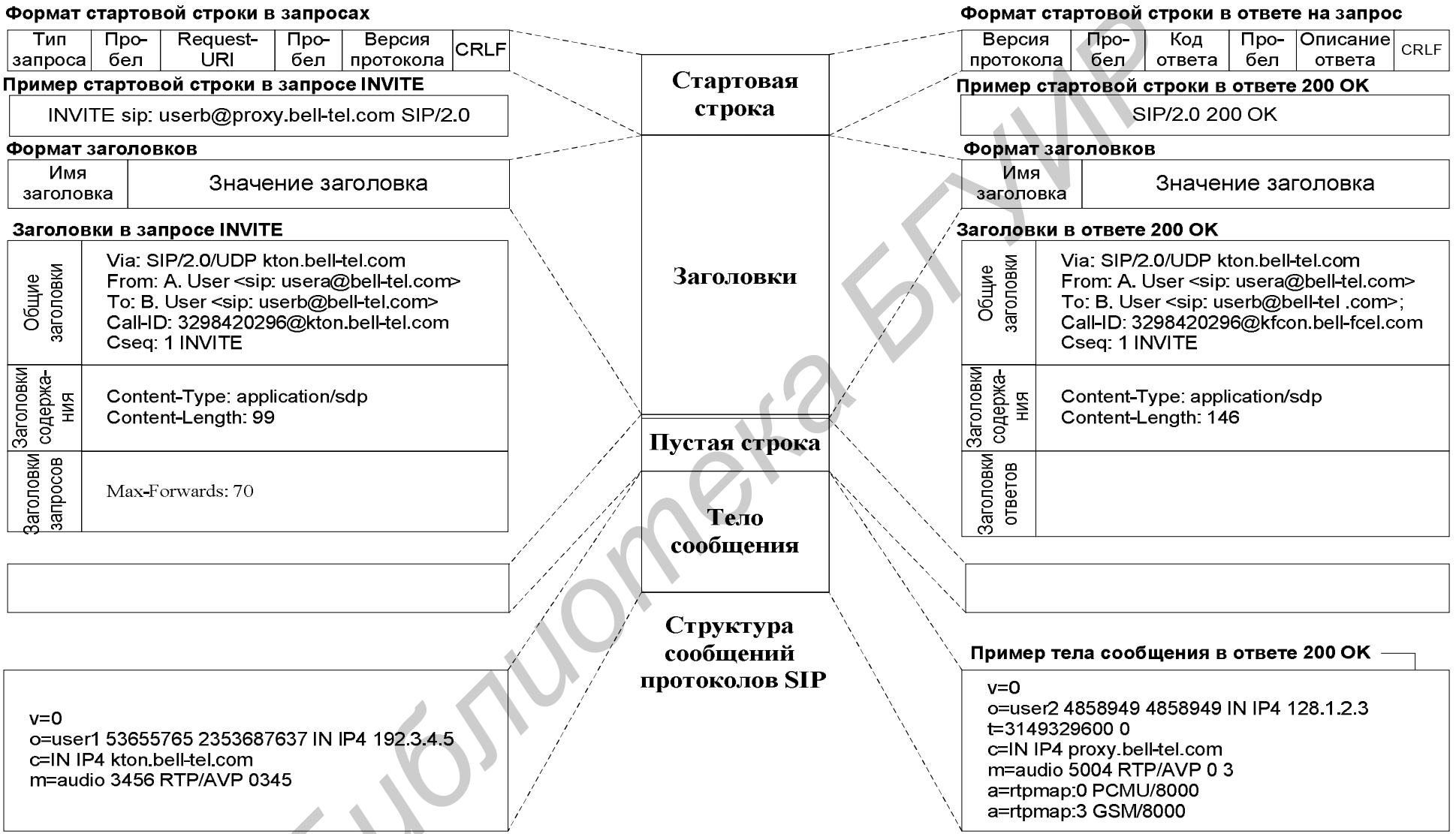


Рис. 2.4. Структура сообщений протокола SIP в запросах и ответах

Библиотека БГУИР

2.2.2. Назначение и формат запросов

SIP-запросы характеризуются наличием строки Request-Line в стартовой строке. Request-Line состоит из названия типа запроса, Request-URI и версии протокола, разделённых пробелом (например ACK sip:anton@niits.ru SIP/2.0). Request-Line заканчивается символами возврата каретки и перевода строки (CRLF). Оба символа, вместе или по одиночке, не должны встречаться в других частях строки. Использование линейного пробела как произвольного сочетания пробелов и символов горизонтальной табуляции не допускается.

Тип запроса	Пробел	Request URI	Пробел	Версия протокола	CRLF

Рис 2.5 Структура строки Request-Line

- **Тип запроса.** В базовой рекомендации IETF RFC 3261 определено 6 типов запросов: REGISTER – для регистрации контактной информации, INVITE, ACK и CANCEL – для установления сеансов, BYE – для завершения сеансов и OPTION – для запроса информации о функциональных возможностях сервера. Каждый из них предназначен для выполнения широкого круга задач, что является достоинством протокола SIP, так как число сообщений, которыми обмениваются терминалы и серверы, сведено к минимуму. Сервер определяет тип принятого запроса по названию, указанному в стартовой строке.

- **Request-URI.** Request-URI – это SIP или SIPS URI (SIP Secured – SIP с функциями безопасности). Он указывает на пользователя или на услугу, которой адресован запрос. Элементы сети SIP могут поддерживать поля Request-URI со схемами, отличными от «sip» и «sips», например «tel».

Содержание полей To и Request-URI может быть различным, например, в поле To может быть указан списочный адрес получателя, а в Request-URI – адрес прокси-сервера, через который проходит запрос.

Запрос INVITE приглашает вызываемого пользователя принять участие в сеансе связи. Сообщение обычно содержит описание сеанса, указывающее

Библиотека БГУИР

вид принимаемой информации и параметры (список возможных вариантов параметров), необходимые для приема информации, а также может указываться вид информации, которую вызываемый пользователь имеет право передавать. В ответе на запрос INVITE указывается вид информации, которая будет приниматься вызываемым пользователем, а кроме того, может указываться вид информации, которую вызываемый пользователь собирается передавать (возможные параметры передачи информации).

В этом сообщении могут содержаться также данные, необходимые для аутентификации абонента и, следовательно, доступа клиентов к SIP-серверу. В случае необходимости изменения уже установленных характеристик передается запрос INVITE с новым описанием сеанса. Для приглашения нового участника к уже установленному соединению также используется сообщение INVITE, которое передается его агенту пользователя.

Запрос ACK подтверждает прием ответа на команду INVITE. Следует отметить, что подтверждение ACK используется только совместно с запросом INVITE, т. е. этим сообщением оборудование вызывающего пользователя показывает, что оно получило окончательный ответ на свой запрос INVITE. В запросе ACK может содержаться окончательное описание сеанса, передаваемое вызывающим пользователем.

Запрос CANCEL отменяет обработку ранее переданных запросов с такими же, как и в запросе CANCEL, значениями полей Call-ID, To, From и CSeq, но не влияет на те запросы, обработка которых уже завершена. Например, запрос CANCEL применяется тогда, когда прокси-сервер размножает запросы для поиска пользователя по нескольким направлениям и находит его по одному из них. Обработку запросов, разосланных по всем остальным направлениям, сервер отменяет при помощи команды CANCEL.

Сообщением BYE оборудование вызываемого или вызывающего пользователей завершает соединение. Сторона, получившая запрос BYE, должна

Библиотека БГУИР

прекратить передачу речевой (мультимедийной) информации и подтвердить выполнение запроса ответом 200 (OK).

При помощи команды REGISTER пользователи сообщают свое текущее местоположение.

Сообщением OPTIONS вызываемый пользователь запрашивает информацию о возможностях терминального оборудования вызываемого пользователя. В ответ на запрос оборудование вызываемого пользователя сообщает требуемую информацию. Применение запроса ограничено теми случаями, когда существует необходимость узнать о поддерживаемых возможностях оборудования до установления соединения. Для установления соединения запрос не используется.

После экспериментальной проверки протокола SIP в реальных сетях оказалось, что для решения ряда задач рассмотренных выше шести запросов недостаточно. Поэтому организацией IETF введены новые сообщения. Ими являются: INFO, PRACK, UPDATE, SUBSCRIBE, NOTIFY, REFER, MESSAGE.

Запрос INFO предназначен для обмена сигнальной информацией по сигнальному тракту SIP в процессе установления и поддержания соединения. Запрос INFO не изменяет состояния процесса обработки SIP-вызовов, как не изменяет и состояния сеансов связи, инициированных при помощи протокола SIP. Однако он обеспечивает передачу дополнительной информации прикладного уровня, которая в дальнейшем может способствовать более производительному функционированию приложений, использующих протокол SIP для доставки информации.

Возможными применениями запроса INFO являются:

- перенос текущих сигнальных сообщений ТфОП между шлюзами ТфОП в течение сеансов связи;
- перенос DTMF-сигналов, созданных в ходе сеанса;
- перенос защищённой сигнальной информации беспроводных систем для поддержки мобильности приложений;

Библиотека БГУИР

- перенос информации об остатке на счёте (билинговой информации);
- перенос изображений и другой не потоковой информации между участниками сеанса связи.

SIP определяет два типа ответов – предварительные и окончательные. Окончательные ответы передают результат обработки запроса и отсылаются надёжно. Предварительные ответы обеспечивают информацию о текущей стадии обработки запроса, но отсылаются ненадёжно. Однако в некоторых случаях, например, при взаимодействии с ТфОП, необходим механизм обеспечения надёжной передачи предварительных ответов. В этом случае тип запроса PRACK играет ту же роль, что и ACK, но для предварительных ответов.

Часто возникают случаи, когда необходимо изменить некоторые параметры сессии до прихода окончательного ответа на начальное сообщение INVITE. После установления диалога, подтверждённого или на ранней стадии, вызывающая сторона может создать запрос UPDATE, который содержит информацию «offer» – предложение с описанием сеанса связи в формате SDP, предназначенное для обновления параметров сессии. Ответ на этот запрос переносит информацию «answer» – ответ на предложение с указанием принятых параметров также в формате SDP. Подобным образом после установления диалога вызываемая сторона может отослать запрос UPDATE с информацией «offer», а вызывающая поместить «answer» в ответ класса «2xx» на UPDATE.

Для большинства реализованных на базе протокола SIP услуг, которые требуют взаимодействия между конечными точками, представляет интерес возможность запрашивать асинхронное уведомление о событиях. Эти услуги включают: услуги автоматического обратного вызова (связанные с событиями изменения состояния терминала пользователя), интерактивные списки контактов «buddy lists» (связанные с событиями присутствия пользователя), оповещение об ожидающем сообщении (связанные с событиями изменения состояния почтового ящика) и передачу информации о состоянии вызова при взаимодействии сетей Интернет и ТфОП.

Библиотека БГУИР

Объекты сети SIP могут подписаться на предоставление информации о состоянии определённого ресурса или вызова в сети, и объекты, располагающие этими сведениями (или объекты, действующие от их лица), будут отсылать уведомления каждый раз, когда это состояние изменится.

Для запроса информации о текущем состоянии и информации об обновлениях состояния удалённого узла используется запрос SUBSCRIBE. Запрос должен быть подтверждён окончательным ответом. После того как подписка была успешно создана или обновлена, уведомитель должен незамедлительно отослать сообщение NOTIFY, чтобы сообщить подписчику текущее состояние ресурса. Когда происходит изменение в состоянии, на которое была открыта подписка, подписчику также направляется запрос NOTIFY. Получив уведомление, подписчик должен возвратить ответ с кодом 200 (OK).

Запрос REFER предписывает получателю связаться с третьей стороной, используя контактную информацию, которая содержится в сообщении. Такой механизм может быть использован для многих целей, включая передачу вызова (Call Transfer). В запрос REFER включается заголовок Refer-To, содержащий адрес третьей стороны.

Интерактивный обмен текстовыми сообщениями (Instant Messaging) происходит между группой участников в режиме, близком к реальному времени. В SIP запрос типа MESSAGE предназначен для передачи мгновенных текстовых сообщений. Тело сообщения будет включать текстовое сообщение, которое необходимо доставить.

Синтаксическое содержание структуры запроса SIP может быть пояснено на примере сообщения INVITE:

INVITE sip: alexander@serv1.loniis.ru SIP/2.0
Via: SIP/2.0/UDP kton.loniis.ru
From: Anton <sip: anton@loniis.ru>
To: Alexander <sip: alexander@loniis.ru>
Call-ID: 3298420296@kton.loniis.ru
Cseq: 1 INVITE
Content-Type: application/sdp
Content-Length: ...

Библиотека БГУИР

v=0
o=bell 53655765 2353687637 IN IP4 128.3.4.5
c=IN IP4 kton.loniis.ru
m=audio 3456 RTP/AVP 0 3 4

В примере пользователь Anton (anton@loniis.ru) вызывает пользователя Alexander (alexander@loniis.ru). Запрос передается на прокси-сервер (serv1.loniis.ru). В полях To и From перед адресом стоит надпись, которую вызывающий пользователь желает вывести на дисплей вызываемого пользователя.

Таблица 2.1

Типы запросов протокола SIP

Название запроса	Описание запроса
INVITE	Приглашает пользователя к сеансу связи
ACK	Подтверждает прием окончательного ответа на запрос INVITE
BYE	Завершает сеанс. Посылается любой из сторон, участвующих в соединении
CANCEL	Отменяет обработку запросов с такими же заголовками Call-ID, To, From и Cseq, как и в запросе CANCEL
REGISTER	Переносит адресную информацию для регистрации пользователя на сервере определения местоположения
OPTION	Запрашивает информацию о функциональных возможностях сервера
INFO	Переносит по сигнальному тракту управляющую и прочую информацию
PRACK	Используется для надёжной транспортировки предварительных ответов
UPDATE	Служит для изменения параметров сеанса до прихода окончательного ответа (без воздействия на состояние диалога)
SUBSCRIBE	Запрашивает информацию о текущем состоянии и информацию об обновлении состояния удалённого ресурса
NOTIFY	Сообщает подписчику текущее состояние ресурса и уведомляет о том, что интересующее его событие произошло
REFER	Предписывает получателю связаться с третьей стороной, используя контактную информацию, которая содержится в сообщении
MESSAGE	Переносит мгновенные текстовые сообщения (instant massages)

Библиотека БГУИР

В теле сообщения оборудование вызывающего пользователя указывает в формате протокола SDP, что оно может принимать в порту 3456 упакованную в пакеты RTP речевую информацию, закодированную по одному из алгоритмов кодирования: 0 – PCMU, 3 – GSM и 4 – G.723.

2.2.3. Назначение и формат ответов на запросы

Характерное отличие SIP-ответов от запросов – это наличие строки Status-Line в стартовой строке. Status-Line составляют: версия протокола и код ответа (Status-Code) со связанной с ним текстовой расшифровкой (Reason-Phrase), разделённые пробелом (SP). Символы возврата каретки (CR) и перевода строки (LF) могут использоваться только совместно в завершающей строке последовательности CRLF.

Версия протокола	Пробел	Status-Code	Пробел	Reason-Phrase	CRLF

Рис. 2.6 Структура строки Status-Line

Код ответа Status-Code – это целое трёхзначное число, отражающее результат обработки запроса сервером. Reason-Phrase даёт краткое описание кода ответа и предназначена для визуального восприятия пользователем в отличие от Status-Code, который служит для оповещения технических устройств. К формулировке Reason-Phrase не предъявляется жестких требований: фирмы-производители вправе выбрать другой текст на произвольном национальном языке, указанном в поле заголовка Accept-Language запроса.

Первая цифра кода ответа определяет класс ответа. Оставшиеся две цифры носят дополнительный характер и служат для упорядочения кодов в пределах категории. В некоторых случаях оборудованию даже необязательно знать все коды ответов, но оно обязательно должно интерпретировать первую цифру ответа.

После приема и интерпретации запроса адресат (прокси-сервер) передает ответ на полученный запрос. Назначение ответов бывает разным, в том числе подтверждение установления соединения, передача запрашиваемой информации, сообщение о неисправностях и т. д. Структуру и виды ответов протокол

Библиотека БГУИР

SIP унаследовал от протокола HTTP. Определено шесть классов ответов, которые несут разную функциональную нагрузку. Все ответы делятся на два типа: информационные и окончательные.

Информационные ответы кодируются трехзначным числом, начинающимся с единицы «1xx» и показывают, что запрос находится в обработке.

Таблица 2.2

Информационные ответы: «1xx»

Код	Назначение ответов
100	**Trying.** Запрос обрабатывается. Например, сервер обращается к базе данных, но местоположение вызываемого пользователя в настоящий момент не определено
180	**Ringing.** Местоположение вызываемого пользователя определено. Вызываемый пользователь получает сигнал о входящем вызове от своего UA
181	**Call Is Being Forwarded.** Прокси-сервер переадресует вызов к другому пользователю
182	**Queued.** Вызываемый пользователь временно не доступен, но входящий вызов поставлен в очередь. Когда вызываемый пользователь станет доступен, он передаст окончательный ответ
183	**Session Progress.** Этот ответ используется для того, чтобы заранее получить описание сессии информационного обмена от шлюзов на пути к вызываемому пользователю таким образом, чтобы мог быть проключен ранний речевой тракт ещё до того, как вызывающий пользователь получит сигнал КПВ

Таблица 2.3

Ответы успешной обработки запроса: «2xx»

Код	Назначение ответов
200	**OK.** Запрос успешно выполнен: Ответ **200** на запрос INVITE означает, что вызываемый пользователь согласен принять участие в сеансе связи, в теле ответа указываются возможности оборудования вызываемого пользователя Ответ **200** на запрос BYE означает завершение сеанса связи, в теле ответа не переносится никакой информации Ответ **200** на запрос CANCEL означает отмену поиска, в теле ответа не переносится никакой информации Ответ **200** на запрос REGISTER означает, что регистрация прошла успешно Ответ **200** на запрос OPTIONS означает согласие вызываемого пользователя сообщить функциональные возможности своего оборудования, которые содержаться в теле ответа
202	**Accepted.** Запрос был принят для обработки, но обработка еще не завершена. Неизвестно, будет ли выполнен запрос, поскольку после завершения обработки запрос может быть отклонён

Библиотека БГУИР

Информационные или предварительные ответы содержат информацию о том, что запрашиваемый сервер находится на стадии выполнения действий по обработке запроса и не может в данный момент выдать окончательный ответ. Сервер посылает «1xx» ответ, если он предполагает, что формирование финального запроса займёт более 200 мс.

Таблица 2.4

Ответы перенаправления вызова: «3xx»

Код	Назначение ответов
300	**Multiple Choices.** Вызываемый пользователь доступен по нескольким адресам. Эти адреса передаются вызывающему пользователю, и тот может выбрать один из них и направить вызов по этому адресу
301	**Moved Permanently.** Вызываемый пользователь больше не находится по указанному в запросе адресу и вызывающий пользователь должен направлять запросы на новый адрес, указанный в заголовке Contact ответа
302	**Moved Temporarily.** Вызываемый пользователь временно изменил свое местоположение и может быть найден по адресу, указанному в заголовке Contact-ответа
305	**Use Proxy**. Вызываемый пользователь не доступен напрямую, входящий вызов должен обязательно пройти через прокси-сервер
380	**Alternative Service.** Запрошенная услуга недоступна, но доступны альтернативные варианты обслуживания, которые описаны в теле сообщения ответа

Окончательные ответы кодируются трехзначными числами, начинающимися с цифр 2, 3, 4, 5 и 6. Все они означают завершение обработки запроса, а каждый из них в отдельности – результат обработки запроса. Ответы «2xx» означают, что запрос был успешно обработан.

Ответы «3xx» информируют оборудование вызывающего пользователя о новом местоположении вызываемого пользователя или об альтернативных сервисах, с помощью которых может быть обслужен вызов пользователя.

Таблица 2.5

Ответы на ошибки в запросе: «4xx»

Код	Назначение ответов
1	2
400	**Bad Request.** В запросе обнаружена синтаксическая ошибка
401	**Unauthorized.** Запрос требует проведения процедуры аутентификации пользователя

Библиотека БГУИР

Продолжение табл. 2.5

1	2
402	**Payment Required.** Требуется предварительная оплата услуг
403	**Forbidden.** Запрещенный запрос – запрос не будет обрабатываться сервером и не должен передаваться повторно
404	**Not Found.** Вызываемый пользователь не обнаружен. Сервер не обнаружил вызываемого пользователя в домене, указанном в поле Request-URI
405	**Method Not Allowed.** Не разрешается передавать запрос этого типа на адрес, указанный в поле Request-URI
406	**Not Acceptable.** Вызываемая сторона будет генерировать ответы, которые не будут поняты вызывающей стороной. Форматы передаваемых данных не соответствуют требованиям, предъявленным в заголовке Accept запроса
407	**Proxy Authentication Required.** Перед вызовом требуется аутентификация прокси-серверу
408	**Request Timeout.** Сервер не может передать ответ в течение промежутка времени, специфицированного вызывающим пользователем в заголовке Expires-запроса
410	**Gone.** Ответ передаётся, если вызываемый пользователь изменил свое местонахождение на длительный срок, а сервер не знает, куда переадресовать запрос
413	**Request Entity Too Large.** Размер запроса слишком велик для обработки на сервере
414	**Request-URI Too Long.** Адрес, указанный в поле Request-URI, слишком большой
415	**Unsupported Media Type.** Сервер не может принять запрос из-за того, что формат содержимого тела сообщения не поддерживается сервером для данного типа запроса
416	**Unsupported URI Scheme.** Схема URI в поле Request-URI не понятна серверу
420	**Bad Extension.** Сервер не понимает расширение протокола SIP, которое содержится в поле заголовка Proxy-Require или Require
421	**Extension Required.** Для правильной обработки запроса UAS вынужден применить определённое расширение, однако оно не указано в поле заголовка Supported-запроса
423	**Interval Too Brief.** Сервер-регистер отклоняет запрос на регистрацию, потому что время действия ресурса, обновлённого запросом, слишком короткое
480	**Temporarily Unavaailable.** Соединение с оконечной системой установлено успешно, но пользователь в данное время не доступен (например, находится вне системы или в системе, но в состоянии препятствующем установлению соединения с вызывающем абонентом или активировал опцию «Не беспокоить»)

Библиотека БГУИР

Окончание табл. 2.5

1	2
481	**Call/Transaction Does Not Exist.** Сервер получил запрос, не относящийся к текущему диалогу или транзакции. Запрос отбрасывается
482	**Loop Detected.** Обнаружен замкнутый маршрут передачи запроса
483	**Too Many Hops.** Запрос на своем пути к вызываемому пользователю прошел через большее количество прокси-серверов, чем разрешено
484	**Address Incomplete.** Принят запрос с неполным адресом в поле Request-URI
485	**Ambiguous.** Адрес вызываемого пользователя в поле Request-URI неоднозначен. Сервер может предложить вызывающему пользователю список адресов в поле заголовка Contact, по которым можно передать данный запрос
486	**Busy Here.** Вызываемый пользователь в данный момент либо не желает, либо не имеет возможности принять данный вызов в дополнение к существующим
487	**Request Terminated.** Запрос был сброшен сообщением BYE или CANCEL
488	**Not Acceptable Here.** Соединение с сервером было установлено, но отдельные элементы описания сеанса связи, такие как тип запрашиваемой информации, полоса пропускания, вид адресации, не допустимы
489	**Bad Event.** Данный ответ используется, чтобы указать, что сервер не понял типа **event package**, указанного в заголовке Event
491	**Request Pending.** Запрос поступил на сервер, который к этому времени не закончил обработку другого запроса, относящегося к тому же диалогу
493	**Undecipherable.** Запрос, полученный UAS, содержит зашифрованное MIME-тело сообщения, для которого получатель не в состоянии подобрать подходящий ключ дешифрирования
494	**Security Agreement Required.** Данный ответ передаётся сервером при выполнении процедуры выбора механизма обеспечения безопасности. Если запрос клиента, помимо заголовка Security-Client со списком механизмом, содержит option-tag «sec-agree» в заголовке Supported, сервер отсылает ответ с кодом 494. Ответ должен включать заголовок Security-Server со списком механизмов обеспечения безопасности, поддерживаемых сервером

Ответы «4xx» информируют о том, что в запросе обнаружена ошибка. После получения такого ответа пользователь не должен передавать тот же самый запрос, на который получен ответ «4xx», без его модификации.

Библиотека БГУИР

Ответы «5xx» информируют, что запрос не может быть обработан из-за ошибки сервера.

Таблица 2.6

Ответы отказа сервера: «5xx»

Код	Назначение ответов
500	**Server Internal Error.** Внутренняя ошибка сервера
501	**Not Implemented.** Сервер не может обслужить запрос, потому что в сервере не реализованы соответствующие функции
502	**Bad Gateway.** Сервер принял некорректный ответ от шлюза или прокси-сервера на пути к адресату вызова. Это типичный отказ для соединений, устанавливаемых через многочисленные сетевые сегменты и серверы
503	**Service Unavailable.** Обслуживание временно невозможно вследствие перегрузки или проведения мероприятий по техническому обслуживанию
504	**Server Time-out.** Прокси-сервер не получил ответа в течение установленного промежутка времени от сервера, к которому он обратился для завершения вызова, например от сервера определения местоположения пользователей
505	**Version Not Supported.** Сервер не поддерживает или отказывается поддерживать версию протокола SIP, используемую в запросе
513	**Message Too Large.** Сервер не в состоянии обработать запрос из-за большой длины сообщения
580	**Precondition Failure.** Когда UAS не может или не желает принимать параметры, предлагаемые в информации «offer», он должен отклонить запрос, передав ответ с кодом 580, – при этом информация «answer» не передаётся

Ответы «6xx» информируют пользователя о том, что передаваемый пользователем запрос не может обслужить ни один сервер и соединение с вызываемым пользователем установить невозможно.

Таблица 2.7

Ответы полной невозможности установления соединения: «6xx»

Код	Назначение ответов
600	**Busy Everywhere.** Вызываемый пользователь занят и не желает принимать вызов в данный момент

Библиотека БГУИР

Окончание табл. 2.7

1	2
603	**Decline.** Вызываемый пользователь не может или не желает принять входящий вызов без указания причины отказа
604	**Does Not Exist Anywhere.** Вызываемый пользователь не существует
606	**Not Acceptable.** Соединение с сервером было установлено, но отдельные элементы описания сеанса связи, такие как тип запрашиваемой информации, полоса пропускания, вид адресации, не допустимы

2.2.4. Структура и формат заголовков сообщений SIP

Поля заголовков SIP-сообщений соответствуют описаниям синтаксиса HTTP/1.1. В SIP определено четыре типа заголовков, содержание которых отображено в табл. 2.8 и включающей: общие заголовки, присутствующие в запросах и ответах, заголовки содержания, начинающиеся со слова «Content», которые переносят информацию о размере тела сообщения или об источнике запроса, заголовки запросов и ответов.

Таблица 2.8

Основные виды заголовков сообщений SIP

1	2	3	4
Общие заголовки	Заголовки содержания	Заголовки запросов	Заголовки ответов
Call-ID (идентификатор сеанса связи)	**Content-Encoding** (кодирование тела сообщения)	**Accept** (соглашение)	**Allow** (последовательность)
Contact (контакт)	**Content-Length** (размер тела сообщения)	**Accent-Encoding** (соглашение по типу кодирования)	**Proxy-Authenticate** (подтверждение подлинности прокси-сервера)
CSeq (последовательность)	**Content-Type** (тип содержимого)	**Accent-Language** (соглашение по языку)	**Retro-After** (повторить через некоторое время)
Date (дата)		**Authorization** (авторизация)	**Server** (сервер)
Encryption (шифрование)		**Hide** (скрыть)	**Un-/Supported** – (не/поддерживается)

Библиотека БГУИР

Окончание табл. 2.8

1	2	3	4
Expires (срабатывание таймера)		**Max-Forwards** (максимальное количество переадресаций)	**Warning** (предупреждение)
From (источник запроса)		**Organization** (организация)	**VWVW-Authenticate** (подтверждение подлинности WWW-сервера)
Record-Route (запись маршрута)		**Priority** (приоритет)	
Timestamp (метка времени)		**Proxy-Authorization** (авторизация прокси-сервера)	
To (адресат назначения)		**Proxy-Require** (требуется прокси-сервер)	
Via (через)		**Route** (маршрут)	
		Require	
		Response-Key (ключ кодирования ответа)	
		Subject (тема)	
		User-Agent (агент пользователя)	

Каждое поле заголовка состоит из имени поля, символа «двоеточие» и значения поля с параметрами: *Имя поля: значение поля.* Порядок следования заголовков не имеет значения. Однако рекомендуется размещать поля заголовков, которые требуются для обработки прокси-серверу (Via, Route, Record-Route, Proxy-Require, Max-Forwards, Proxy-Authorization и др.), в начале сообщения для ускорения анализа и обработки.

Библиотека БГУИР

Важным является порядок следования ряда заголовков с одинаковыми именами полей. Последовательности полей заголовков с одинаковыми именами могут содержаться в сообщении только в том случае, если содержимое поля представляет собой список значений, разделённых запятой. Возможно объединить такие заголовки в одну пару «имя поля: значение поля», не изменяя семантики сообщения, путём добавления каждого последующего значения к первому значению поля заголовка. При этом все значения должны быть отделены друг от друга запятой. Исключение составляют лишь заголовки WWW-Authenticate, Authorization, Proxy-Authenticate и Proxy-Authorization. Последовательности заголовков с такими именами тоже могут присутствовать в сообщении, но объединить их невозможно, поскольку грамматика этих заголовков не подчиняется общим для SIP-заголовков правилам.

Таблица 2.9

Формы имен заголовков

Сжатая форма	Полная форма
C	Content-Type
E	Content-Encoding
F	From
I	Call-ID
K	Supported
L	Content-Length
M	Contact (от «moved»)
S	Subject
O	Event
R	Refer-To
T	To
U	Allow-Events
V	Via

Несмотря на то, что заголовок может содержать неограниченное число параметров, одно и то же имя параметра не может использоваться более одного раза.

Для имен полей заголовков не имеет значения, в каком регистре они написаны. Значения полей, имена параметров и значения параметров также регистронезависимы, если это не определено по-иному в описании определенного

Библиотека БГУИР

заголовка. Если не определено иначе, значения, заключенные в кавычки, являются зависимыми от регистра. Например:

Contact: <sip:anton@niits.ru>;expires=3600

эквивалентно

CONTACT: <sip:anton@niits.ru>;ExPiReS=3600 и

Content-Disposition: session;handling=optional

эквивалентно

Content-disposition: Session;HANDLING=OPTIONAL

Два следующих поля заголовков не равнозначны:

Warning: 370 niits.ru "Требуется большая пропускная способность"

Warning: 370 niits.ru «ТРЕБУЕТСЯ БОЛЬШАЯ ПРОПУСКНАЯ СПОСОБНОСТЬ».

Некоторые заголовки имеют смысл только в запросах или ответах. Они называются заголовками запроса и заголовками ответа соответственно. Если заголовок появляется в сообщении не своей категории (например, заголовок запроса в ответе), то он игнорируется.

При передаче сообщений протокола SIP, упакованных в сигнальные сообщения протокола UDP, существует вероятность того, что размер запроса или ответа превысит максимально допустимый для данной сети размер, что приведет к фрагментации пакета. Для устранения этого используется сжатый формат имен основных заголовков подобно тому, как это делается в протоколе SDP. Ниже приведен список таких заголовков.

2.2.5. Типы заголовков

- **Accept:** Заголовок Accept сообщает тела, сообщений каких типов принимает клиент, например:

Accept:application/sdp; level=1, application/x- private, text/html.

Отсутствие значений в заголовке Accept говорит о том, что не поддерживаются никакие типы. Если в сообщении нет заголовка Accept, то сервер должен применить значение по умолчанию – application/sdp.

Библиотека БГУИР

- **Accept-Encoding:** Заголовок Accept-Encoding похож на Accept, он сообщает о поддерживаемых типах кодирования содержимого в ответе, например: Accept-Encoding: gzip Наличие пустого заголовка в сообщении также допускается. Это равнозначно Accept-Encoding: identity, что означает: кодирование запрещено.

- **Accept-Language:** Заголовок Accept-Language используется в запросах, чтобы указать предпочтительные языки для ключевых фраз, описаний сеансов связи, оповещений о текущем состоянии, содержащихся в ответах в качестве тел сообщения. Если заголовок отсутствует, сервер считает, что клиент поддерживает все языки. Правило упорядочения языков в списке по предпочтительности базируется на параметре «q», например:

Accept-Language: da, en-gb;q=0.8, en;q=0.7

- **Allow:** Заголовок Allow содержит список типов запросов, которых поддерживает агент пользователя, сформировавший сообщение. Все типы запросов, понимаемые UA, включая ACK и CANCEL, должны входить в этот список. Отсутствие заголовка Allow не означает, что передающий сообщение UA не поддерживает никаких типов запросов; это подразумевает, что агент пользователя отправителя не желает передавать информацию о том, какие типы запросов он поддерживает.

Применение заголовка Allow в ответах на запросы (за исключением ответов на запрос OPTION) приводит к уменьшению числа передаваемых сообщений, например: Allow: INVITE, ACK, OPTIONS, CANCEL, BYE.

- **Authorization:** Поле заголовка Authorization содержит отклик аутентификации агента пользователя, например:

Authorization: Digest username="Anton", realm="niits.ru",
nonce="84a4cc6f3082121f32b42a2187831a9e",
response="7587245234b3434cc3412213e5f113a5432"

- **Call-ID:** Заголовок Call-ID – уникальный идентификатор сеанса связи или всех регистраций отдельного клиента. Значение идентификатору присваи-

Библиотека БГУИР

вает сторона, которая инициирует вызов. Возможна ситуация, когда к одной мультимедийной конференции относятся несколько соединений – все они будут иметь разные идентификаторы Call-ID, например:

Call-ID:f81d4fae-7dec-11d0-a765-00a0c91e6bf6@loniis.ru

i:f81d4fae-7dec-11d0-a765-00a0c91e6bf6@192.0.2.4

Заголовок состоит из буквенно-числового значения и имени рабочей станции, которая присвоила этот идентификатор. Между ними стоит символ «@». Значения Call-ID регистрозависимы и могут сравниваться побайтно.

- **Contact:** Поле заголовка Contact несет в себе URI, значение которого зависит от типа передаваемого запроса или ответа. Как правило, в заголовке Contact находится текущий адрес пользователя, на который он может принимать входящие сообщения. Заголовок Contact может содержать отображаемое имя (display name), адрес с его параметрами и параметры заголовка.

Для заголовка Contact определены параметры «q» и «expires». Они используются только в случае, когда заголовок присутствует в запросе REGISTER, в ответе на него или в ответе класса «3xx».

Когда значение поля заголовка содержит отображаемое имя, URI со всеми его параметрами заключается в символы «**<**» и «**>**». В противном случае все параметры, следующие за URI, будут трактоваться как параметры заголовка. Отображаемым именем может быть комбинация символьных фраз или строка, заключенная в кавычки, если существует необходимость в более длинной характеристике, например:

Contact: "Alexander" <sip:alexander@loniis.ru>;q=0.7; expires=3600,

"Alexander" <mailto:alexander@loniis.ru> ;q=0.1

- **Content-Encoding:** Заголовок Content-Encoding используется в качестве модификатора типов тела сообщения. Когда такой заголовок присутствует, его значение указывает, какие дополнительные виды кодирования были применены к содержимому и, соответственно, какие следует применить механизмы деко-

Библиотека БГУИР

дирования для получения тела сообщения типа, обозначенного в поле заголовка Content-Type, например: Content-Encoding: gzip.

- **Content-Language:** Назначение заголовка Content-Language – определить и изменить содержимое тела сообщения в соответствии с предпочтительным языком пользователя, например: Content-Language:fr.

- **Content-Length:** Заголовок Content-Length указывает отображенный в десятичном виде размер (в байтах) тела сообщения, переданного получателю. Приложения должны помещать в это поле размер тела сообщения, подлежащего передаче, не взирая на тип тела сообщения. Если в качестве транспорта выступает потоко-ориентированный протокол (такой как TCP), заголовок Content-Length должен использоваться обязательно, например: Content-Length: 349.

- **Content-Type:** Заголовок Content-Type определяет тип тела сообщения, переданного получателю. Content-Type должен входить в сообщение, если тело сообщения не пустое, например:

Content-Type: application/sdp

c: text/html; charset=ISO-8859-4

Если же тело пустое, а Content-Type присутствует, он показывает, что тело определенного типа имеет нулевую длину (например, пустой аудиофайл).

- **CSeq:** Заголовок CSeq – уникальный идентификатор запроса, относящегося к одному соединению. Он служит для корреляции запроса с ответом на него, а также для того, чтобы первоначальные запросы отличались от переадресованных. Заголовок состоит из двух частей: натурального числа из диапазона от 1 до 2^{32} и типа запроса. Часть с типом запроса является регистрозависимой. Сервер должен проверять значение CSeq в каждом принимаемом запросе**,** и считает его новым, если значение больше предыдущего. Этот заголовок копируется из запроса в ответ, например: CSeq: 4711 INVITE

- **Date:** Заголовок Date содержит дату и время первой отправки сообщения, например: Date: Sat, 13 Nov 2010 23:29:00 GMT

Библиотека БГУИР

- **Expires:** Заголовок Expires устанавливает время, по истечении которого сообщение или его содержимое станет недействительным. Значение заголовка зависит от типа запроса. Присутствует в запросах REGISTER и INVITE. В запросе REGISTER заголовок указывает, сколько времени регистрация остается действительной. В запросах INVITE он ограничивает временя, в течение которого URI будет оставаться действительным в приемнике, оставаясь в кэш-памяти. Если этот заголовок отсутствует, буферизация на сервере производится не будет. Значение этого поля – выраженное в десятичном виде количество секунд, в интервале между 0 и (2^{32} –1), измеренное с момента получения запроса, например: Expires: 5

- **From:** Заголовок From содержит URI отправителя запроса. При этом адрес отправителя запроса может не совпадать с адресом инициатора диалога. Адрес из заголовка From запроса копируется в одноимённый заголовок ответа.

Отображаемое имя должно информировать пользователя об инициаторе запроса. В случае если не удается установить личность вызывающего пользователя, в качестве display name (отображаемого имени) фигурирует «Anonymous». В случае, если URI содержит запятую, точку с запятой или вопросительный знак, он заключается в угловые скобки, даже если отображаемое имя отсутствует.

Два заголовка From считаются эквивалентными, когда совпадают их URI и параметры. При сравнении параметры расширения, присутствующие лишь в одном из двух заголовков, во внимание не принимаются. Это означает, что отображаемое имя и наличие или отсутствие угловых скобок на результат сравнения не влияют. Примеры:

From: "Vladimir" <sip:vladimir@protei.ru> ;tag=a48s

From: sip:+79213434329@gateway.protei.ru;tag=887s

- **Max-Forwards:** Заголовок Max-Forwards используется в любом типе SIP запросов, чтобы ограничить число серверов или шлюзов, через которые проходит запрос. Значение заголовка должно быть целым числом в пределах от 0 до 255, отражающим оставшееся количество пересылок, которое разрешено

Библиотека БГУИР

для сообщения. Это число уменьшается каждым сервером, который пересылает запрос дальше. В качестве первоначального значения рекомендуется брать 70. Заголовок Max-Forwards должен вводиться теми элементами, которые иначе не могут гарантировать обнаружение петли, например: Max-Forwards: 6

- **MIME-Version:** Заголовок MIME-Version указывает версию стандарта MIME-информации, помещенной в тело сообщения, например: MIME-Version: 1.0
- **Organization:** Заголовок Organization содержит название организации, к которой относится SIP-элемент, передающий запросы или ответы. Это поле заголовка может использоваться клиентским программным обеспечением для фильтрации вызовов, например: Organization: Niits
- **Priority:** Заголовок Priority указывает на срочность запроса с точки зрения клиента. В нём содержится приоритет SIP-запроса для конечного пользователя или его UA, например:

Subject: У нас случился пожар!

Priority: emergency

или

Subject: Планы на выходные

Priority: non-urgent

- **Proxy-Authenticate:** Заголовок Proxy-Authenticate содержит запрос аутентификации, состоящий из поля, которое отображает схему аутентификации, и ряд параметров, необходимых для проведения процедуры аутентификации с данным прокси-сервером для указанного Request-URI, например:

Proxy-Authenticate: Digest realm="niits.ru",

qop="auth", nonce="f84f1cec41e6cbe5aea9c8e88d359",

opaque="", stale=FALSE, algorithm=MD5

Заголовок включается в состав ответа с кодом 407 (Proxy Authentication Required) или с кодом 401 (Unauthorized).

- **Proxy-Authorization:** Заголовок Proxy-Authorization идентифицирует пользователя прокси-серверу, который требует аутентификации. Значение поля

Библиотека БГУИР

заголовка состоит из отклика аутентификации, который содержит информацию аутентификации агента пользователя для прокси-сервера, обслуживающего область запрашиваемого ресурса, например:

Proxy-Authorization: Digest username="Anton", realm="niits.ru",
nonce="c60f3082ee1212b402a21831ae",
response="245f23415f11432b3434341c022"

- **Proxy-Require:** Заголовок Proxy-Require указывает функции прокси-сервера, которые должны им поддерживаться, например:

Proxy-Require: 100rel

- **Record-Route:** Заголовок Record-Route помещается прокси-серверами в запросы для того, чтобы следующие запросы в процессе диалога маршрутизировались через эти же прокси-серверы, например:

Record-Route: <sip:serv10.protei.ru;lr>,
<sip:site3.niits.ru;lr>

- **Require:** Поле заголовка Require используется UAC для того, чтобы сообщить UAS перечень опций, которые должен поддерживать сервер для обработки запроса. Содержимое поля заголовка представляет собой список идентификаторов новых функциональных возможностей (расширений) **option-tag**, каждый из которых определяет одно из SIP-расширений, необходимых для обработки сообщения, например: Require: 100rel.

- **Route:** Заголовок Route служит для принудительной маршрутизации запроса в соответствии со списком прокси-серверов, например:

Route: <sip:site5.niits.ru;lr>,
<sip:serv3.protei.ru;lr>

- **Server:** Поле заголовка Server содержит информацию о программном обеспечении, которое используется сервером для обработки запросов, например: Server: HomeServer v2

Библиотека БГУИР

- **Subject:** Заголовок Subject содержит дополнительную информацию о типе и характере сеанса связи, позволяя производить фильтрацию вызовов без анализа описания сеанса, например:

Subject: Требуется техническое описание оборудования

s: Техническая поддержка.

- **Supported:** Заголовок Supported содержит перечень всех расширений, поддерживаемых UAC или UAS. Содержимое поля заголовка представляет собой список идентификаторов **option-tag**, которые понимает UAC или UAS. Если поле заголовка пусто, значит, ни одно расширение не поддерживается. Рекомендуется, чтобы этот заголовок присутствовал во всех запросах и ответах, за исключением ACK, например:

Supported: sip-cc, sip-cc-02, timer

Приведенный заголовок указывает на поддержку возможностей управления сеансом связи по протоколу SIP и таймера сеанса.

- **To:** Заголовок To определяет логического адресата запроса. В случае присутствия отображаемого имени оно должно быть предоставлено вызываемому пользователю с помощью пользовательского интерфейса. Параметр «tag» является неотъемлемой частью механизма идентификации диалога. Процедура сравнения заголовков To аналогична процедуре с заголовками From, например:

To: The Operator <sip:operator@server10.protei.ru>;tag=287447

To: sip:+79213434329@gateway.protei.ru

- **Unsupported:** Заголовок Unsupported содержит перечень функциональных возможностей, не поддерживаемых UAS. Добавляется в ответ с кодом 420 (Bad Extension), например: Unsupported: 100rel

- **User-Agent:** Поле заголовка User-Agent несет информацию о клиенте агента пользователя, инициирующего запрос, например:

User-Agent: Softphone Beta1.5

- **Via:** Поле заголовка Via содержит список элементов сети SIP, через которые запрос на данный момент прошёл. Список нужен для того, чтобы избе-

Библиотека БГУИР

жать ситуаций, в которых запрос пойдёт по замкнутому пути, а также для тех случаев, когда необходимо, чтобы запросы и ответы обязательно проходили по одному и тому же пути. В конечном результате заголовок отображает весь путь, пройденный запросом: каждый прокси-сервер добавляет поле со своим адресом. Параметр «branch» в поле заголовка Via выполняет функцию идентификатора транзакции и используется прокси-серверами для обнаружения петель, например: Via:SIP/2.0/UDP serv1.niits.ru:5060;branch=z9hG4bK87asdks7

- **Warning:** Заголовок Warning содержит дополнительную информацию, как правило, связанную с проблемами обработки запроса сервером. Значения поля заголовка Warning отправляются вместе с ответами и содержат трёхзначный код, имя хоста и предупреждающий текст. Предупреждающий текст должен быть написан на национальном языке, наиболее лёгком для понимания конечным пользователем. Это решение может базироваться на любой доступной информации, такой как местоположение пользователя, поле заголовка Accept-Language запроса или поле заголовка Content-Language ответа.

- **WWW-Authenticate:** Заголовок WWW-Authenticate содержит запрос аутентификации, состоящий из поля, отображающего схему аутентификации, и ряда параметров, необходимых для проведения процедуры аутентификации с данным прокси-сервером для указанного Request-URI. Заголовок включается в состав ответа с кодом 407 (Proxy Authentication Required) или с кодом 401 (Unauthorized), например:

WWW-Authenticate: Digest realm="niits.ru",
qop="auth", nonce="f84f1cec41e6cbe5aea9c8e88d359",
opaque="", stale=FALSE, algorithm=MD5

2.3. Взаимодействие элементов SIP при установлении соединения

SIP-прокси-серверы – это элементы сети SIP, которые маршрутизируют SIP-запросы к серверам агента пользователя UAS и SIP-ответы – к клиентам агента пользователя UAC. На пути к UAS запрос может проходить несколько прокси-серверов. Каждый из них будет принимать решение о дальнейшей мар-

Библиотека БГУИР

шрутизации, внося изменения в запрос перед его пересылкой следующему элементу сети. Ответы будут маршрутизироваться через ту же группу прокси-серверов, которая была пройдена запросами, но в обратном порядке.

Прокси-сервер – это логический элемент сети SIP. Когда приходит запрос, элемент, выполняющий роль прокси-сервера, первоначально решает, существует ли необходимость отвечать на запрос самостоятельно. Например, запрос может содержать ошибки, или прокси-сервер может нуждаться в аутентификации клиента для выполнения своих функций. Элемент может передать ответ с подходящим кодом ошибки. Отвечая на запрос непосредственно, SIP-элемент выполняет роль UAS и должен действовать в соответствии с общими правилами для UAS вне диалога.

Прокси-сервер может функционировать с сохранением состояний (stateful) и без сохранения (stateless) состояний для каждого нового запроса. Сервер без сохранения состояний работает как ретранслирующий узел сети. Он пересылает каждый запрос следующему элементу, принимая решение о маршрутизации на основе информации, содержащейся в запросе. Полученные ответы он просто возвращает обратно. Прокси-серверы без сохранения состояний удаляют информацию о прошедшем сообщении, как только сообщение было ретранслировано.

Прокси-серверы с сохранением состояний хранят информацию (состояние транзакции) о каждом входящем запросе и о каждом переданном запросе, возникающем вследствие обработки входящего запроса. Stateful прокси-сервер может принять решение о размножении запроса, передавая его на несколько разных адресов, где может находиться вызываемый пользователь. Любой запрос, направляемый более чем на один SIP-элемент, должен обрабатываться с сохранением состояний.

Stateful прокси-сервер может перейти в stateless режим в любой момент обработки запроса при условии, что он ранее не сделал ничего, что могло бы предотвратить переход в stateless режим (например, размножение запросов или

Библиотека БГУИР

создание ответа с кодом 100). При выполнении такого перехода вся информация о состоянии транзакций удаляется. Большинство процедур обработки запроса в stateless и stateful режимах идентичны.

К наиболее важным функциям stateless UAS относятся:

- stateless UAS не должен передавать предварительных (1xx) ответов;
- stateless UAS не должен повторно передавать ответы;
- stateless UAS должен игнорировать запросы ACK;
- stateless UAS должен игнорировать запросы CANCEL;
- параметры «tag» заголовка To должны формироваться для ответов таким образом, что для одинаковых запросов должны генерироваться одинаковые «tag».

В остальном stateless UAS работает так же, как stateful UAS. Для каждого нового запроса UAS может выбрать, как работать с сохранением или без сохранения состояний.

Для некоторых сетевых архитектур может оказаться полезным снизить загрузку прокси-серверов, которые отвечают за маршрутизацию запросов, и производить доставку начального сообщения, используя сервер переадресации. Сервер переадресации помещает информацию маршрутизации для поступившего запроса в ответ, предназначенный клиенту. Клиент, получивший перенаправляющий ответ от этого сервера, снова передает запрос, используя новый, только что полученный адрес (адреса).

Сервер определения местоположения – это база данных, в которой каждому URI соответствует один или более адресов, где может находиться пользователь, идентифицируемый представленным URI. Сам сервер переадресации не может создавать своих запросов. После получения любого запроса, отличного от CANCEL, сервер переадресации либо отклоняет запрос, либо обращается к серверу определения местоположения, который передаёт ему список альтернативных адресов; затем он передает клиенту ответ класса «3xx». Для правильно составленных запросов CANCEL он передает ответ класса «2xx». Окончатель-

Библиотека БГУИР

ный ответ завершает эту SIP-транзакцию. На протяжении всей транзакции сервер переадресации сохраняет ее состояние.

В перенаправляющем ответе класса 3xx на запрос присутствует заголовок Contact, в котором содержится список контактных адресов. Значения заголовка могут содержать параметр «expires», указывающий время действия контактного адреса. Указанные в Contact адреса характеризуют места, где может находиться пользователь-адресат, или просто снабжают исходный адрес дополнительными транспортными параметрами. Например, ответ с кодом 301 (Moved Permanently) или 302 (Moved Temporarily) может сообщить адрес, совпадающий по местоположению с начальным, но доопределенный транспортными параметрами, указывающими другое имя сервера или мультикаст-адрес, которые должны быть использованы в новом запросе, или замену транспортного протокола UDP протоколом TCP (или наоборот).

Заголовок Contact содержит URI, указывающие на возможное текущее местоположение вызываемого пользователя, причем это могут быть не только SIP-адреса. Заголовок может включать в себя URI для телефона, факса, электронной почты. Значение заголовка Contact может направлять на ресурс, не имеющей связи с запрашиваемым. Например, для SIP-вызовов, связанных со шлюзом ТфОП, может оказаться необходимым доставить определенное информационное уведомление, такое как «Номер изменился».

Агент пользователя (UA) как логический объект может выполнять как функции клиента агента пользователя, так и функции сервера агента пользователя. В сети SIP он представляет оконечное оборудование абонента. Соответственно агент пользователя состоит из клиента агента пользователя (UAC), генерирующего запросы, и сервера агента пользователя (UAS), который формирует ответы.

UAC создает новые запросы, отправляет их и обрабатывает принятые ответы. Запросы генерируются в результате внешних воздействий (нажатия кнопок пользователем, сигнала из телефонной линии).

Библиотека БГУИР

Правильный запрос, составленный UAC, должен включать стартовую строку, содержащую тип запроса, поле Request-URI и версию SIP, и следующий базовый набор полей заголовков: To, From, CSeq, Call-ID, Max-Forwards и Via. Эти поля заголовков, как и Request-URI, обязательны для всех SIP-запросов.

Формирование поля Request-URI: поле Request-URI указывает на пользователя или сервис, к которому адресован запрос. Исходное значение поля Request-URI сообщения устанавливается таким же, как URI в поле To.

Формирование заголовка To: заголовок To устанавливает желаемого логического получателя запроса, или публичный адрес получателя, или адресуемый ресурс. Запросы вне диалога не должны содержать параметра “tag” в поле To.

Формирование заголовка From: заголовок From содержит логический идентификатор инициатора сообщения, возможно, например, сетевой адрес вызывающего пользователя. Поле From должно содержать параметр “tag”, выбранный клиентом UA.

Формирование заголовка Call-ID: заголовок Call-ID должен совпадать для всех запросов и ответов, отправляемых любым из двух UA в процессе диалога. При создании нового диалога заголовок Call-ID должен быть выбран UAC как уникальный идентификатор.

Формирование заголовка Cseq: для запросов вне диалога, кроме REGISTER, значение порядкового номера может быть произвольным. UAC может выбирать любой механизм для создания значений заголовка CSeq.

Формирование заголовка Max-Forwards: UAC должен вставлять заголовок Max-Forwards в каждый отправляемый запрос. Значение заголовка по умолчанию – 70.

Формирование заголовка Via: при создании запроса UAC должен вставить в него поле Via. При этом необходимо указать название протокола – SIP и его версию – 2.0. Поле заголовка Via должно содержать параметр “branch”.

Библиотека БГУИР

Формирование заголовка Contact: заголовок Contact содержит текущий адрес вызывающего пользователя.

Формирование заголовков Supported и Require: UAC должен включить в запрос заголовок Supported со списком идентификаторов option tag. Если UAC хочет потребовать, чтобы UAS понял расширение, которое UAC применит к запросу для его обработки, он должен вставить в запрос заголовок Require, указывающее option tag этого расширения.

Формирование необязательных заголовков: после того как создается новый запрос и заголовки, описанные выше, составлены, добавляются необязательные заголовки в соответствии с типом запроса: Content-Disposition, Content-Encoding, Content-Language, Content-Length, Content-Type.

При отправке запроса определяется место его назначения. Если местная политика безопасности не определена по-иному, адрес места назначения должен быть определен с применением DNS-процедур. Эти процедуры устанавливают упорядоченную последовательность, состоящую из адреса, порта и транспортных протоколов для запроса. Далее UAC посылает запрос по каждому из имеющихся адресов, пока не будет установлено соединение с сервером. Каждая попытка составляет новую транзакцию, и поэтому каждый новый запрос содержит заголовок Via с новым параметром «branch» в первом значении.

Ответы сначала обрабатываются транспортным уровнем SIP, а потом направляются на уровень транзакций. Уровень транзакций производит их обработку и затем передает вышестоящему уровню – уровню пользователя транзакций.

Обработка неизвестного ответа: UAC должен расценивать любой неизвестный окончательный ответ как эквивалентный «x00» ответу того же класса и должен быть готов обрабатывать ответы «x00» любого класса.

Обработка заголовка Via: если в ответе представлено более одного значения поля заголовка Via, UAC должен отбросить сообщение.

Библиотека БГУИР

Обработка ответов класса 3xx: при поступлении ответа переадресации UAC должны использовать адрес из поля Contact при составлении одного или нескольких новых запросов, основанных на перенаправленном запросе.

Обработка ответов класса 4xx: после обработки ответа запрос передается заново с соответствующими изменениями. Новый запрос составляет новую транзакцию и будет иметь такие же значения заголовков Call-ID, To и From, как и предыдущий запрос, но заголовок CSeq должен содержать новый порядковый номер, который на единицу больше предыдущего.

Сервер агента пользователя принимает запросы и генерирует ответы, основываясь на действиях пользователя, полученных сообщениях, результатах выполнения программ или каких-либо других механизмах.

При обработке сервером запроса вне диалога выполняется набор процедур обработки, независимых от типа запроса. Если запрос принимается, должны быть произведены любые связанные с ним изменения состояния соединения. Если он отклоняется, ни одно из изменений производится не должно.

Когда запрос прошел аутентификацию, UAS должен выяснить тип запроса. В случае поддержки сервером типа запроса обработка сообщения продолжается.

Если UAS не понимает заголовка, представленного в запросе, он должен игнорировать этот заголовок и продолжить обработку сообщения.

Одинаковые запросы могут придти на сервер более одного раза, следуя различными путями. Ядро UAS обрабатывает первый такой запрос и отправляет ответ с кодом 482 (Loop Detected) на последующие запросы.

Обработка заголовка Require: UAS определяет SIP-расширения, которые требуются для правильной обработки запроса.

Обработка тела сообщения: UAS изучает тело сообщения и поля заголовков, которые описывают его. Если в сообщении содержится непонятный тип, язык или кодек и эта часть тела является обязательной, UAS должен от-

Библиотека БГУИР

бросить запрос и отправить ответ с кодом 415 (Unsupported Media Type). После этих проверок тело обрабатывается в зависимости от типа запроса и типа тела.

Когда UAS создает ответ на запрос, он следует общим процедурам, описанным ниже. Также может потребоваться выполнение дополнительных действий, которые зависят от конкретного кода ответа. После завершения выполнения всех процедур, связанных с созданием ответа, UAS возвращает ответ серверной транзакции, от которой был получен запрос.

Запрос может проходить несколько прокси-серверов на пути к UAS. Каждый из них будет принимать решения о дальнейшей маршрутизации, внося изменения в запрос перед его пересылкой следующему элементу сети. Ответы будут маршрутизироваться через ту же группу прокси-серверов, которая была пройдена запросами, но в обратном порядке.

В режиме с сохранением состояний прокси-сервер действует как механизм обработки SIP-транзакций. При функционировании прокси-сервер задействует серверную транзакцию и одну или несколько клиентских транзакций, которые связаны друг с другом посредством ядра прокси-сервера – компонента верхнего уровня, выполняющего обработку сигнальной информации.

Ядро прокси-сервера определяет, куда маршрутизировать запрос, выбирая один или несколько мест назначения для следующей пересылки запроса. Исходящий запрос, адресованный одному пользователю, но отсылаемый по нескольким направлениям, для каждого направления обрабатывается средствами связанной с ним клиентской транзакции. Ядро прокси-сервера собирает ответы от клиентских транзакций и использует их для передачи ответов серверной транзакции.

Для каждого запроса элемент, выполняющий роль прокси-сервера, должен осуществлять следующие функции:

– проверка правильности составления запроса;

– предварительная обработка маршрутной информации;

– определение адреса (адресов) для пересылки;

Библиотека БГУИР

– пересылка запроса;

– обработка всех ответов.

Оценка правильности составления запроса включает следующие проверки:

1. Проверка корректности синтаксиса – запрос должен быть правильно составлен, чтобы быть поддержанным серверной транзакцией.

2. Проверка схемы URI – если поле Request-URI содержит URI, схема которого не понятна прокси-серверу, он должен отклонить запрос, отослав запрос с кодом 416 (Unsupported URI Scheme).

3. Проверка заголовка Max-Forwards – если запрос не содержит заголовка Max-Forwards, или запрос содержит его со значением большим нуля, то проверка успешно завершается. В противном случае отсылается ответ 483 (Too many hops).

4. Проверка на наличие замкнутого пути – если в заголовке Via запроса содержится значение, у которого имя хоста, номер порта и параметр «branch» совпадают со значением, которое прокси-сервер поместил в предыдущие запросы, то петля присутствует. Прокси-сервер может возвратить ответ с кодом 482 (Loop Detected).

5. Проверка заголовка Proxy-Require – если запрос содержит заголовок Proxy-Require с одним или несколькими идентификаторами новых функциональных возможностей option tag, непонятными прокси-серверу, он возвращает ответ 420 (Bad Extension).

6. Проверка заголовка Proxy-Authorization – при отсутствии отклика аутентификации в заголовке запрос отклоняется и посылается ответ 407 (Proxy Authentication Required).

Во время предварительной обработки маршрутной информации прокси-сервер должен изучить поле Request-URI-запроса. Если поле содержит значение, идентифицирующее данный прокси-сервер, он осуществляет замену значения Request-URI на последнее значение заголовка Route с последующим уда-

Библиотека БГУИР

лением последнего значения заголовка Route. Это возможно, если узлом, отсылающим запрос прокси-серверу, является strict-router. Если первое значение заголовка Route запроса является адресом данного прокси-сервера, он должен удалить это значение.

Перечень адресов для запроса будет назначен в содержимом запроса либо получен при обращении к серверу определения местоположения. Если домен в Request-URI является доменом, за который данный прокси-сервер не несет ответственности, Request-URI должен быть помещен в перечень как единственный адрес. Если перечень адресов не был назначен, то данный элемент ответственен за домен, указанный в Request-URI. Прокси-сервер обращается к базе данных, и полученные URI используются для формирования перечня адресов. Прокси-сервер может добавлять в перечень адресов контактные адреса, полученные в ответе класса «3xx». Если перечень адресов остается пуст после проведения вышеперечисленных операций, прокси-сервер должен отправить ответ с кодом ошибки 480 (Temporarily Unavailable).

Прокси-сервер с сохранением состояний обрабатывает перечень адресов. Адреса обрабатываются в очередности от большего значения параметра «q» заголовка Contact к меньшему. Для каждого адреса прокси-сервер передает запрос, последовательно выполняя следующие шаги:

1. Создаёт копию полученного запроса.
2. Обновляет поле Request-URI – адрес из target set записывается в Request-URI.
3. Обновляет заголовок Max-Forwards – значение заголовка уменьшается на единицу.
4. Добавляет своё значение Record-Route.
5. Добавляет дополнительные заголовки.
6. Выполняет заключительную обработку маршрутной информации путем анализа первого значения заголовка Route. Если оно не содержит параметра «lr», прокси-сервер помещает содержимое поля Request-URI в качестве послед-

Библиотека БГУИР

него значения в заголовок Route и затем записывает первое значение заголовка Route в поле Request-URI и удаляет его из заголовка Route.

7. Определяет адрес, порт и транспортный протокол. Для этого прокси-сервер выполняет процедуры DNS-поиска SIP-сервера, которому следует отослать запрос.

8. Добавляет значение заголовка Via. Для этого прокси-сервер помещает свое значение заголовка Via; значение содержит уникальный параметр «branch».

9. Добавляет заголовок Content-Length при надёжном транспортном протоколе.

10. Пересылает новый запрос. При этом прокси-сервер создает новую клиентскую транзакцию и передаёт ей запрос.

11. Устанавливает таймер «C» при создании каждой клиентской транзакции для ограничения времени ожидания окончательного ответа.

При получении ответа прокси-сервер определяет клиентскую транзакцию, соответствующую этому ответу. Как только клиентская транзакция начинает передачу ответов ядру прокси-сервера, вступают в силу следующие процедуры обработки:

1. Обнаружение буфера ответов – прокси-сервер находит буфер ответов, созданный перед пересылкой запроса.

2. Перезапуск таймера «C» при получении предварительных ответов с кодом 101-199.

3. Удаление верхнего значения заголовка Via из ответа.

4. Добавление полученного окончательного ответа в буфер ответов.

5. Проверка на необходимость немедленной отправки. Если при этом, приходит ответ с кодом «101-199» или класса «2xx», то ответ должен быть сразу отослан серверной транзакции.

6. Выбор «наилучшего» окончательного ответа из буфера ответов. В этом случае прокси-сервер выбирает из буфера ответов ответы, класс которых

Библиотека БГУИР

имеет наименьшее значение, если нет ответов класса «6xx». В пределах выбранного класса должно быть отдано преимущество тем ответам, которые обеспечивают информацию, влияющую на очередную передачу запроса, таким как 401, 407, 415, 420 и 484, если выбран «4xx класс».

7. Объединение значений в заголовке Authorization. Прокси-сервер должен собрать все значения из заголовков WWW-Authenticate и Proxy-Authenticate ответов с кодом 401 и 407, полученных до настоящего времени в буфер ответов, и перед пересылкой добавить их без изменений в «наилучший» ответ.

8. Перезапись значения заголовка Record-Route. Если выбранный ответ содержит значение заголовка Record-Route, помещённое этим прокси-сервером при прохождении запроса, он может изменить это значение перед пересылкой ответа.

9. Пересылка ответа. Прокси-сервер передает ответ серверной транзакции, связанной с буфером ответов. Это приведёт к отсылке ответа в соответствии с местонахождением, указанным в верхнем значении заголовка Via.

10. Создание необходимых запросов CANCEL. Если пересланный ответ был окончательным, прокси-сервер должен создать запрос CANCEL для всех незавершенных клиентских транзакций, связанных с этим буфером.

При срабатывании таймера «C» прокси-сервер должен его либо перезапустить с любым выбранным значением, либо завершить клиентскую транзакцию, а при извещении об ошибке при пересылке ответа – он отбрасывает ответ.

Если запрос CANCEL обрабатывается прокси-сервером с сохранением состояний в рамках собственной серверной транзакции, то буфер ответов для него не создаётся. Вместо этого ядро прокси-сервера ищет существующий буфер ответов для серверной транзакции запроса, отменяемого данным CANCEL.

В режиме без сохранения состояний прокси-сервер работает как простой ретранслятор сообщений. Большая часть процедур обработки, выполняемая

Библиотека БГУИР

прокси-сервером с сохранением состояний, характерна для работы и прокси-сервера без сохранения состояний.

Однако у прокси-сервера без сохранения состояний нет никакого представления о механизме транзакций или о буфере ответов, который используется для описания поведения прокси-сервера с сохранением состояний. Вместо этого прокси-серверы без сохранения состояний принимают сообщения – запросы и ответы – напрямую от транспортного уровня протокола SIP. В результате они не могут передавать повторные сообщения, созданные самостоятельно. Они только пересылают все повторные сообщения, которые получают; повторные передачи пересылаются также как оригинальные сообщения, поскольку прокси-серверы без сохранения состояний не различают их.

2.4. Протокол SIP для телефонии

Протокол SIP-T (SIP для телефонии) предназначен для прозрачной передачи сигнализации CCS №7 по сети SIP и использует методы и процедуры, описанные в RFC 3261.

SIP-T использует два механизма переноса сигнализации, известных как «инкапсуляция» и «трансляция». В шлюзах SIP–ISUP сообщение ISUP протокола CCS №7 инкапсулируется в сообщение протокола SIP, при этом сохраняется только информация, необходимая для обслуживания. Однако некоторые посредники при передаче, например, прокси-серверы, принимающие решения о продвижении запроса SIP дальше по сети, могут неправильно распознать ISUP. Поэтому наряду с инкапсуляцией важная информация транслируется из сообщения ISUP в заголовки сообщений SIP в порядке, определяемом тем, как дальше будет маршрутизироваться SIP-запрос.

Процедуры, необходимые при взаимодействии сети SIP и ISUP, приведены в табл. 2.10, а на рис. 2.7 приведен фрагмент сети, обеспечивающей обслуживание вызова двух абонентов ISDN через транзитную IP-сеть. Такой сценарий называют «SIP bridging».

Библиотека БГУИР

Для сообщения ISUP сеть SIP является прозрачной, т. е. сообщение проходит через сеть SIP, не изменяясь. Это достигается путем инкапсуляции сообщения ISUP в тело SIP-запроса.

Когда вызов, предназначенный для сети SIP, идет от абонента ТфОП, то сообщение ISUP будет получено шлюзом сигнализации в MGC (Media Gateway Controller), который является точкой взаимодействия ТфОП и сети SIP. MGC стандартными средствами протокола SIP посылает запрос в сеть.

Таблица 2.10

Основные процедуры конвертирования сообщений SIP и ISUP

Требования к интерфейсу при взаимодействии ISUP-SIP	Функции протокола SIP-T
Прозрачность сети SIP для сигнализации ISUP	Инкапсуляция сообщений ISUP в тело запросов SIP
Маршрутизация запросов SIP по информации, содержащейся в сообщениях ISUP	Трансляция параметров сообщений ISUP в заголовки запросов SIP
Передача сигнальной информации ISUP во время мультимедийной сессии	Использование запроса INFO

Протокол SIP осуществляет стандартную маршрутизацию внутри сети, чтобы определить нужную точку выхода для вызова. Найдя ее, он начинает диалог установления соединения между начальной и конечной точками. Оконечный MGC, который является точкой выхода из сети SIP в ТфОП, передает сообщение ISUP в сеть ТфОП, используя любое вложенное сообщение ISUP, пришедшее в запросе SIP без анализа его содержимого.

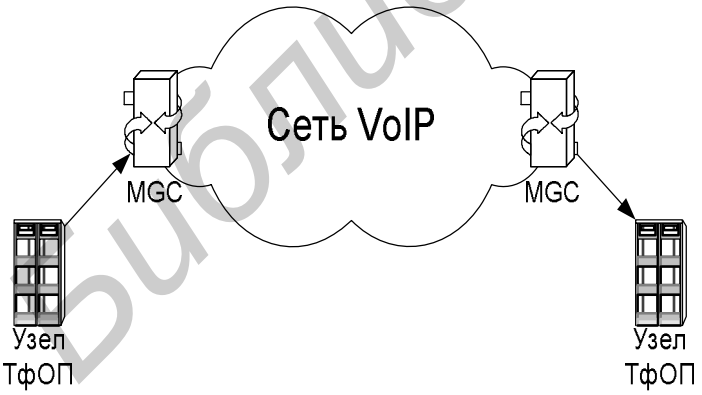


Рис. 2.7. Сеть в случае транзита трафика ТфОП через сеть VoIP

Библиотека БГУИР

Протокол SIP обычно работает поверх протокола IP, а разговор пользователей рассматривается как мультимедийный сеанс связи, включающий в себя передачу аудио данных.

Преобразование сообщений между протоколами ISUP и SIP выполняется в модуле Media Gateway Controller (MGC). MGC имеет логический интерфейс для работы с сетями обоих типов ISUP и SIP. Преобразование аудиоинформации из формата, принятого в сети SIP, в формат ТфОП осуществляется в медиашлюзе Media Gateway (MG) с магистральным интерфейсом E1/T1 (со стороны ТфОП) и интерфейсом IP (со стороны сети IP). MGC и MG могут быть объединены физически в одно устройство или находиться отдельно.

Для сообщений ISUP, проходящих через сеть SIP, инкапсуляция позволяет таким элементам в сети SIP, как прокси-серверы, которые не умеют работать непосредственно с сообщениями ISUP, правильно передавать сообщение, основываясь на данных, полученных из сообщения ISUP, в заголовок запроса SIP (например номер вызываемого абонента).

Возможность инкапсуляции сигнализации ТфОП является одним из основных требований к SIP-T. Для этого используется разделяемое на необходимое число частей тело сообщения в кодировке MIME, что позволяет включать в сообщения SIP различную информацию (данные протоколов SDP, ISUP и т. д.). Для спецификации ISUP введен специальный MIME-тип – ISUP Media Type, позволяющий удобно получать информацию об используемом варианте ISUP.

Таблица 2.11

ISUP Media Type

Media type name:	application
Media subtype name:	ISUP
Required parameters:	version
Optional parameters:	base
Encoding scheme:	binary
Security considerations:	SIP

Библиотека БГУИР

ISUP Media Type содержит информацию, представленную в табл. 2.11. Использование параметра «version» позволяет системным администраторам узнать тип ISUP. Это дает возможность каждому SoftSwitch/MGC корректно обработать сообщение или передать. Однако у прокси-сервера без сохранения состояний нет никакого представления о механизме транзакций или о буфере ответов, который используется пользователю сообщение о том, что данный тип ISUP не поддерживается. Спецификация не ограничивает значения, которые могут быть использованы в «version»; это оставлено на усмотрение системных администраторов.

Параметр «base» может включатся опционально в некоторые сообщения, если требуется, чтобы получатель правильно распознал используемый тип ISUP, так как параметр «version» может быть не понят.

Заголовок Content-Disposition может служить для описания процесса обработки вложенного сообщения ISUP, в частности, какие действия необходимо предпринять, если приемником не было понято содержимое заголовка Content-Type. По умолчанию значение заголовка Content-Disposition для ISUP сообщений – «signal». Это показывает, что данная часть тела сообщения содержит сигнальную информацию, но не содержит описания соединения. Типичный заголовок (параметр «base» может отсутствовать) имеет вид:

Content-Type: application/ISUP; version=nxv3; base=etsi121

Content-Disposition: signal; handling=optional

Пример сообщения INVITE, которое содержит информацию SDP и инкапсулированное сообщение ISUP IAM имеет следующий формат:

INVITE sip:78123877658@max.loniis.ru SIP/2.0
Via: SIP/2.0/UDP anton.loniis.ru
From: sip:78124513355@anton.loniis.ru
To: sip:78123877658@max.loniis.ru
Call-ID: MAX123199902171209550099@max.loniis.ru
CSeq: 8348 INVITE
Contact: <sip:anton@loniis.ru>
Content-Length: 436
Content-Type: multipart/mixed; boundary=unique-boundary-1

Библиотека БГУИР

```
MIME-Version: 1.0
--unique-boundary-1
Content-Type: application/SDP; charset=ISO-10646

v=0
o=jpeterson 2890844526 2890842807 IN IP4 126.16.64.4
s=SDP seminar
c=IN IP4 MG122.loniis.ru
t= 2873397496  2873404696
m=audio 9092 RTP/AVP 0 3 4
--unique-boundary-1
Content-Type: application/ISUP; version=nxv3;
base=etsi121
Content-Disposition: signal; handling=optional

01 00 49 00 00 03 02 00 07 04 10 00 33 63 21
43 00 00 03 06 0d 03 80 90 a2 07 03 10 03 63
53 00 10 0a 07 03 10 27 80 88 03 00 00 89 8b
0e 95 1e 1e 1e 06 26 05 0d f5 01 06 10 04 00
--unique-boundary-1
```

Части сообщения разделяются специальной строкой, задаваемой параметром «boundary». В приведенном примере для разделения используется пустая строка «unique-boundary-1».

Конвертирование сигнальной информации между протоколами ISUP и SIP включает в себя два компонента:

1. Преобразование сигнализации ISUP в SIP на уровне сообщений. В SIP-T предполагается использование MGC, который создает сообщения ISUP из поступающих сообщений SIP и наоборот. Для этого необходимо точное определение правил преобразования между сообщениями ISUP и SIP, каждое сообщение ISUP должно быть транслировано в конкретное сообщение SIP. Например, IAM в INVITE, REL в BYE и т. д.

2. Преобразование параметров сообщения ISUP в заголовок сообщения SIP. Запрос SIP, который используется для установки соединения, должен содержать необходимую для маршрутизации прокси-серверами информацию, например, это может быть телефонный номер, набранный вызывающим абонентом.

Библиотека БГУИР

На практике очень важно стандартизировать процедуры трансляции информации из ISUP в SIP (например, Called Party Number в ISUP IAM должен быть записан в заголовок To и поле Request-URI и т. д.).

Одной из проблем трансляции при транзите трафика через сеть SIP является то, что параметры ISUP, переведенные в заголовках сообщения SIP, могут изменяться промежуточными узлами сети. В этом случае оконечный MGC (точка выхода из сети SIP) получит сообщение, в котором параметры заголовка сообщения SIP не соответствуют параметрам вложенного сообщения ISUP. Например, параметр заголовка To и поля Request-URI запроса SIP отличается от параметра Called Party Number (номер вызываемого абонента) во вложенном сообщении ISUP.

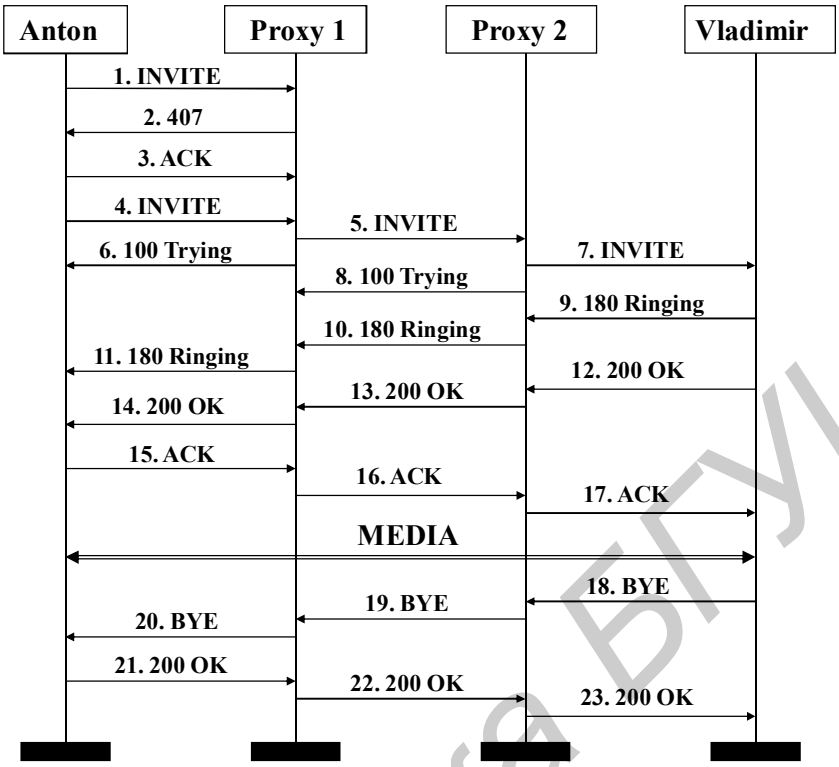
В этом случае значения имеют заголовки и при создании нового сообщения параметры будут заполняться значениями из заголовков запроса SIP, а недостающаяся информация будет взята из вложенного сообщения ISUP, если оно присутствует.

Базовые сообщения протокола SIP не могут обеспечить передачу сигнальных сообщений во время сеанса связи. Для этих целей необходимо использовать дополнительное сообщение INFO. Однако этот запрос не подходит для передачи сигналов, в случае если телефонный номер из ТфОП передается по частям (overlap dialing), но его рекомендуется использовать для передачи сигналов DTMF.

Библиотека БГУИР

Приложение 1

Установление соединения с участием прокси-сервера



1. **INVITE Anton -> Proxy 1**

   INVITE sip:vladimir@protei.ru SIP/2.0
   Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74b43
   Max-Forwards: 70
   Route: <sip:ss1.niits.ru;lr>
   From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
   To: Vladimir <sip:vladimir@protei.ru>
   Call-ID: 3848276298220188511@niits.ru
   CSeq: 1 INVITE
   Contact: <sip:anton@serv1.niits.ru;transport=tcp>
   Content-Type: application/sdp
   Content-Length: 151

   v=0
   o=anton 2890844526 2890844526 IN IP4 serv1.niits.ru
   s=-
   c=IN IP4 192.0.2.101
   t=0 0
   m=audio 49172 RTP/AVP 0
   a=rtpmap:0 PCMU/8000

2. **407 (Proxy Authorization Required) Proxy 1 -> Anton**

   SIP/2.0 407 Proxy Authorization Required
   Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74b43
   received=192.0.2.101
   From: Anton <sip:anton@niits.ru>;tag=9fxced76sl

Библиотека БГУИР

To: Vladimir <sip:vladimir@protei.ru>;tag=3flal12sf
Call-ID: 3848276298220188511@niits.ru
CSeq: 1 INVITE
Proxy-Authenticate: Digest realm="niits.ru", qop="auth",
nonce="f84f1cec41e6cbe5aea9c8e88d359",
opaque="", stale=FALSE, algorithm=MD5
Content-Length: 0

3. **ACK Anton -> Proxy 1**

ACK sip:vladimir@protei.ru SIP/2.0
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74b43
Max-Forwards: 70
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=3flal12sf
Call-ID: 3848276298220188511@niits.ru
CSeq: 1 ACK
Content-Length: 0

4. **INVITE Anton -> Proxy 1**

INVITE sip:vladimir@protei.ru SIP/2.0
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
Route: <sip:ss1.niits.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>
Call-ID: 3848276298220188511@niits.ru
CSeq: 2 INVITE
Contact: <sip:anton@serv1.niits.ru;transport=tcp>
Proxy-Authorization: Digest username="anton",
realm="niits.ru",
nonce="wf84f1ceczx41ae6cbe5aea9c8e88d359", opaque="",
uri="sip:vladimir@protei.ru",
response="42ce3cef44b22f50c6a6071bc8"
Content-Type: application/sdp
Content-Length: 151

v=0
o=anton 2890844526 2890844526 IN IP4 serv1.niits.ru
s=-
c=IN IP4 192.0.2.101
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

5. **INVITE Proxy 1 -> Proxy 2**

INVITE sip:vladimir@protei.ru SIP/2.0
Via: SIP/2.0/TCP ss1.niits.ru:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9
received=192.0.2.101
Max-Forwards: 69

Библиотека БГУИР

Record-Route: <sip:ss1.niits.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>
Call-ID: 3848276298220188511@niits.ru
CSeq: 2 INVITE
Contact: <sip:anton@serv1.niits.ru;transport=tcp>
Content-Type: application/sdp
Content-Length: 151

v=0
o=anton 2890844526 2890844526 IN IP4 serv1.niits.ru
s=-
c=IN IP4 192.0.2.101
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

6. **100 (Trying) Proxy 1 -> Anton**

SIP/2.0 100 Trying
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9
received=192.0.2.101
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>
Call-ID: 3848276298220188511@niits.ru
CSeq: 2 INVITE
Content-Length: 0

7. **INVITE Proxy 2 -> Vladimir**

INVITE sip:vladimir@serv3.protei.ru SIP/2.0
Via: SIP/2.0/TCP ss2.protei.ru:5060;branch=z9hG4bK721e4.1
Via: SIP/2.0/TCP ss1.niits.ru:5060;branch=z9hG4bK2d4790.1
received=192.0.2.111
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9
received=192.0.2.101
Max-Forwards: 68
Record-Route: <sip:ss2.protei.ru;lr>,
<sip:ss1.niits.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>
Call-ID: 3848276298220188511@niits.ru
CSeq: 2 INVITE
Contact: <sip:anton@serv1.niits.ru;transport=tcp>
Content-Type: application/sdp
Content-Length: 151

v=0
o=anton 2890844526 2890844526 IN IP4 serv1.niits.ru
s=-
c=IN IP4 192.0.2.101
t=0 0
m=audio 49172 RTP/AVP 0

Библиотека БГУИР

a=rtpmap:0 PCMU/8000

8. 100 (Trying) Proxy 2 -> Proxy 1

SIP/2.0 100 Trying
Via: SIP/2.0/TCP ss1.niits.ru:5060;branch=z9hG4bK2d4790.1
received=192.0.2.111
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9
received=192.0.2.101
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>
Call-ID: 3848276298220188511@niits.ru
CSeq: 2 INVITE
Content-Length: 0

9. 180 (Ringing) Vladimir -> Proxy 2

SIP/2.0 180 Ringing
Via: SIP/2.0/TCP ss2.protei.ru:5060;branch=z9hG4bK721e4.1
received=192.0.2.222
Via: SIP/2.0/TCP ss1.niits.ru:5060;branch=z9hG4bK2d4790.1
received=192.0.2.111
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9
received=192.0.2.101
Record-Route: <sip:ss2.protei.ru;lr>,
<sip:ss1.niits.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 3848276298220188511@niits.ru
Contact: <sip:vladimir@serv3.protei.ru;transport=tcp>
CSeq: 2 INVITE
Content-Length: 0

10. 180 (Ringing) Proxy 2 -> Proxy 1

SIP/2.0 180 Ringing
Via: SIP/2.0/TCP ss1.niits.ru:5060;branch=z9hG4bK2d4790.1
received=192.0.2.111
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9
received=192.0.2.101
Record-Route: <sip:ss2.protei.ru;lr>,
<sip:ss1.niits.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 3848276298220188511@niits.ru
Contact: <sip:vladimir@serv3.protei.ru;transport=tcp>
CSeq: 2 INVITE
Content-Length: 0

11. 180 (Ringing) Proxy 1 -> Anton

SIP/2.0 180 Ringing
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9
received=192.0.2.101
Record-Route: <sip:ss2.protei.ru;lr>,

Библиотека БГУИР

<sip:ss1.niits.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 3848276298220188511@niits.ru
Contact: <sip:vladimir@serv3.protei.ru;transport=tcp>
CSeq: 2 INVITE
Content-Length: 0

12. 200 (OK) Vladimir -> Proxy 2

SIP/2.0 200 OK
Via: SIP/2.0/TCP ss2.protei.ru:5060;branch=z9hG4bK721e4.1
received=192.0.2.222
Via: SIP/2.0/TCP ss1.niits.ru:5060;branch=z9hG4bK2d4790.1
received=192.0.2.111
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9
received=192.0.2.101
Record-Route: <sip:ss2.protei.ru;lr>,
<sip:ss1.niits.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 3848276298220188511@niits.ru
CSeq: 2 INVITE
Contact: <sip:vladimir@serv3.protei.ru;transport=tcp>
Content-Type: application/sdp
Content-Length: 147

v=0
o=vladimir 2890844527 2890844527 IN IP4 serv3.protei.ru
s=-
c=IN IP4 192.0.2.201
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

13. 200 (OK) Proxy 2 -> Proxy 1

SIP/2.0 200 OK
Via: SIP/2.0/TCP ss1.niits.ru:5060;branch=z9hG4bK2d4790.1
received=192.0.2.111
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9
received=192.0.2.101
Record-Route: <sip:ss2.protei.ru;lr>,
<sip:ss1.niits.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 3848276298220188511@niits.ru
CSeq: 2 INVITE
Contact: <sip:vladimir@serv3.protei.ru;transport=tcp>
Content-Type: application/sdp
Content-Length: 147

Библиотека БГУИР

v=0
o=vladimir 2890844527 2890844527 IN IP4 serv3.protei.ru
s=-
c=IN IP4 192.0.2.201
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

14. **200 (OK) Proxy 1 -> Anton**

SIP/2.0 200 OK
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9
received=192.0.2.101
Record-Route: <sip:ss2.protei.ru;lr>,
<sip:ss1.niits.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 3848276298220188511@niits.ru
CSeq: 2 INVITE
Contact: <sip:vladimir@serv3.protei.ru;transport=tcp>
Content-Type: application/sdp
Content-Length: 147

v=0
o=vladimir 2890844527 2890844527 IN IP4 serv3.protei.ru
s=-
c=IN IP4 192.0.2.201
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

15. **ACK Anton -> Proxy 1**

ACK sip:vladimir@serv3.protei.ru SIP/2.0
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74b76
Max-Forwards: 70
Route: <sip:ss1.niits.ru;lr>,
<sip:ss2.protei.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 3848276298220188511@niits.ru
CSeq: 2 ACK
Content-Length: 0

16. **ACK Proxy 1 -> Proxy 2**

ACK sip:vladimir@serv3.protei.ru SIP/2.0
Via: SIP/2.0/TCP ss1.niits.ru:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74b76
received=192.0.2.101
Max-Forwards: 69
Route: <sip:ss2.protei.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 3848276298220188511@niits.ru

Библиотека БГУИР

CSeq: 2 ACK
Content-Length: 0

17. ACK Proxy 2 -> Vladimir

ACK sip:vladimir@serv3.protei.ru SIP/2.0
Via: SIP/2.0/TCP ss2.protei.ru:5060;branch=z9hG4bK721e4.1
Via: SIP/2.0/TCP ss1.niits.ru:5060;branch=z9hG4bK2d4790.1
received=192.0.2.111
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74b76
received=192.0.2.101
Max-Forwards: 68
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 3848276298220188511@niits.ru
CSeq: 2 ACK
Content-Length: 0

Между терминалами пользователей Anton и Vladimir созданы RTP-потоки.

18. BYE Vladimir -> Proxy 2

BYE sip:anton@serv1.niits.ru SIP/2.0
Via: SIP/2.0/TCP serv3.protei.ru:5060;branch=z9hG4bKnashds7
Max-Forwards: 70
Route: <sip:ss2.protei.ru;lr>,
<sip:ss1.niits.ru;lr>
From: Vladimir <sip:vladimir@protei.ru>;tag=314159
To: Anton <sip:anton@niits.ru>;tag=9fxced76sl
Call-ID: 3848276298220188511@niits.ru
CSeq: 1 BYE
Content-Length: 0

19. BYE Proxy 2 -> Proxy 1

BYE sip:anton@serv1.niits.ru SIP/2.0
Via: SIP/2.0/TCP ss2.protei.ru:5060;branch=z9hG4bK721e4.1
Via: SIP/2.0/TCP serv3.protei.ru:5060;branch=z9hG4bKnashds7
received=192.0.2.201
Max-Forwards: 69
Route: <sip:ss1.niits.ru;lr>
From: Vladimir <sip:vladimir@protei.ru>;tag=314159
To: Anton <sip:anton@niits.ru>;tag=9fxced76sl
Call-ID: 3848276298220188511@niits.ru
CSeq: 1 BYE
Content-Length: 0

20. BYE Proxy 1 -> Anton

BYE sip:anton@serv1.niits.ru SIP/2.0
Via: SIP/2.0/TCP ss1.niits.ru:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/TCP ss2.protei.ru:5060;branch=z9hG4bK721e4.1
received=192.0.2.222
Via: SIP/2.0/TCP serv3.protei.ru:5060;branch=z9hG4bKnashds7
received=192.0.2.201
Max-Forwards: 68

Библиотека БГУИР

From: Vladimir <sip:vladimir@protei.ru>;tag=314159
To: Anton <sip:anton@niits.ru>;tag=9fxced76sl
Call-ID: 3848276298220188511@niits.ru
CSeq: 1 BYE
Content-Length: 0

21. **200 (OK) Anton -> Proxy 1**

SIP/2.0 200 OK
Via: SIP/2.0/TCP ss1.niits.ru:5060;branch=z9hG4bK2d4790.1
received=192.0.2.111
Via: SIP/2.0/TCP ss2.protei.ru:5060;branch=z9hG4bK721e4.1
received=192.0.2.222
Via: SIP/2.0/TCP serv3.protei.ru:5060;branch=z9hG4bKnashds7
received=192.0.2.201
From: Vladimir <sip:vladimir@protei.ru>;tag=314159
To: Anton <sip:anton@niits.ru>;tag=9fxced76sl
Call-ID: 3848276298220188511@niits.ru
CSeq: 1 BYE
Content-Length: 0

22. **200 (OK) Proxy 1 -> Proxy 2**

SIP/2.0 200 OK
Via: SIP/2.0/TCP ss2.protei.ru:5060;branch=z9hG4bK721e4.1
received=192.0.2.222
Via: SIP/2.0/TCP serv3.protei.ru:5060;branch=z9hG4bKnashds7
received=192.0.2.101
From: Vladimir <sip:vladimir@protei.ru>;tag=314159
To: Anton <sip:anton@niits.ru>;tag=9fxced76sl
Call-ID: 3848276298220188511@niits.ru
CSeq: 1 BYE
Content-Length: 0

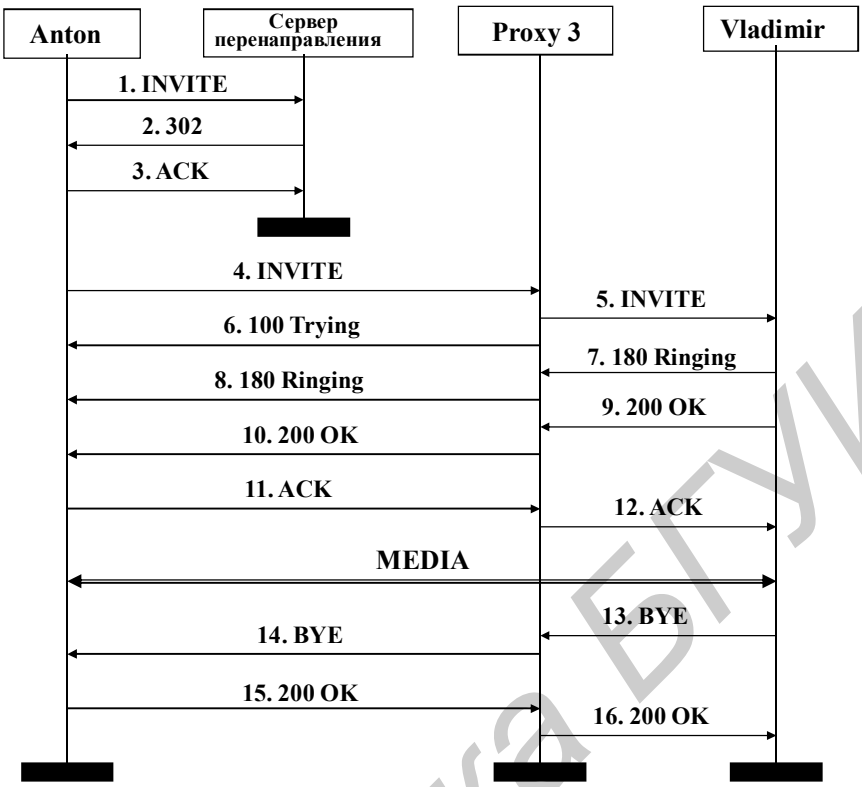
23. **200 (OK) Proxy 2 -> Vladimir**

SIP/2.0 200 OK
Via: SIP/2.0/TCP serv3.protei.ru:5060;branch=z9hG4bKnashds7
received=192.0.2.201
From: Vladimir <sip:vladimir@protei.ru>;tag=314159
To: Anton <sip:anton@niits.ru>;tag=9fxced76sl
Call-ID: 3848276298220188511@niits.ru
CSeq: 1 BYE
Content-Length: 0

Библиотека БГУИР

Приложение 2

Установление соединения с участием сервера перенаправления



1. INVITE Anton -> Сервер перенаправления

INVITE sip:vladimir@protei.ru SIP/2.0
Via: SIP/2.0/UDP serv1.niits.ru:5060;branch=z9hG4bKbf9f44
Max-Forwards: 70
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>
Call-ID: 2xTb9vxSit55XU7p8@niits.ru CSeq: 1 INVITE
Contact: <sip:anton@serv1.niits.ru>
Content-Length: 0

2. 302 (Moved Temporarily) Сервер перенаправления -> Anton

SIP/2.0 302 Moved Temporarily
Via: SIP/2.0/UDP serv1.niits.ru:5060;branch=z9hG4bKbf9f44 ;received=192.0.2.101
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=53fHlqlQ2
Call-ID: 2xTb9vxSit55XU7p8@niits.ru
CSeq: 1 INVITE
Contact: <sip:vladimir@loniis.ru;transport=tcp>
Content-Length: 0

3. ACK Anton -> Сервер перенаправления

ACK sip:vladimir@protei.ru SIP/2.0
Via: SIP/2.0/UDP serv1.niits.ru:5060;branch=z9hG4bKbf9f44
Max-Forwards: 70
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=53fHlqlQ2

Библиотека БГУИР

Call-ID: 2xTb9vxSit55XU7p8@niits.ru
CSeq: 1 ACK
Content-Length: 0

4. INVITE Anton -> Proxy 3

INVITE sip:vladimir@loniis.ru SIP/2.0
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9
Max-Forwards: 70
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>
Call-ID: 2xTb9vxSit55XU7p8@niits.ru CSeq: 2 INVITE
Contact: <sip:anton@serv1.niits.ru;transport=tcp>
Content-Length: 0

5. INVITE Proxy 3 -> Vladimir

INVITE sip:vladimir@serv5.loniis.ru SIP/2.0
Via: SIP/2.0/TCP ss3.loniis.ru:5060;branch=z9hG4bK721e.1
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9;received=192.0.2.101
Max-Forwards: 69
Record-Route:
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>
Call-ID: 2xTb9vxSit55XU7p8@niits.ru CSeq: 2 INVITE
Contact: <sip:anton@serv1.niits.ru;transport=tcp>
Content-Length: 0

6. 100 (Trying) Proxy 3 -> Anton

SIP/2.0 100 Trying
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9;received=192.0.2.101
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>
Call-ID: 2xTb9vxSit55XU7p8@niits.ru
CSeq: 2 INVITE
Content-Length: 0

7. 180 (Ringing) Vladimir -> Proxy 3

SIP/2.0 180 Ringing
Via: SIP/2.0/TCP ss3.loniis.ru:5060;branch=z9hG4bK721e.1;received=192.0.2.233
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9;received=192.0.2.101
Record-Route: <sip:ss3.loniis.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@niits.ru
CSeq: 2 INVITE
Contact: <sip:vladimir@serv5.loniis.ru;transport=tcp>
Content-Length: 0

8. 180 (Ringing) Proxy 3 -> Anton

SIP/2.0 180 Ringing
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9;received=192.0.2.101
Record-Route: <sip:ss3.loniis.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl

Библиотека БГУИР

To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@niits.ru
CSeq: 2 INVITE
Contact: <sip:vladimir@serv5.loniis.ru;transport=tcp>
Content-Length: 0

9. **200 (OK) Vladimir -> Proxy 3**

SIP/2.0 200 OK
Via: SIP/2.0/TCP ss3.loniis.ru:5060;branch=z9hG4bK721e.1;received=192.0.2.233
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9;received=192.0.2.101
Record-Route: <sip:ss3.loniis.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@niits.ru
CSeq: 2 INVITE
Contact: <sip:vladimir@serv5.loniis.ru;transport=tcp>
Content-Type: application/sdp
Content-Length: 148

v=0
o=vladimir 2890844527 2890844527 IN IP4 serv5.loniis.ru
s=-
c=IN IP4 192.0.2.100
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

10. **200 (OK) Proxy -> Anton**

SIP/2.0 200 OK
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bf9;received=192.0.2.101
Record-Route: <sip:ss3.loniis.ru;lr>
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@niits.ru
CSeq: 2 INVITE
Contact: <sip:vladimir@serv5.loniis.ru;transport=tcp>
Content-Type: application/sdp
Content-Length: 148

v=0
o=vladimir 2890844527 2890844527 IN IP4 serv5.loniis.ru
s=-
c=IN IP4 192.0.2.100
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

11. **ACK Anton -> Proxy 3**

ACK sip:vladimir@serv5.loniis.ru SIP/2.0
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bq9
Max-Forwards: 70
Route: <sip:ss3.loniis.ru;lr>

Библиотека БГУИР

From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@niits.ru
CSeq: 2 ACK
Content-Type: application/sdp
Content-Length: 151

v=0
o=anton 2890844526 2890844526 IN IP4 serv1.niits.ru
s=-
c=IN IP4 192.0.2.101
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

12. ACK Proxy 3 -> Vladimir

ACK sip:vladimir@serv5.loniis.ru SIP/2.0
Via: SIP/2.0/TCP ss3.loniis.ru:5060;branch=z9hG4bK721e.1
Via: SIP/2.0/TCP serv1.niits.ru:5060;branch=z9hG4bK74bq9;received=192.0.2.101
Max-Forwards: 69
From: Anton <sip:anton@niits.ru>;tag=9fxced76sl
To: Vladimir <sip:vladimir@protei.ru>;tag=314159
Call-ID: 2xTb9vxSit55XU7p8@niits.ru
CSeq: 2 ACK
Content-Type: application/sdp
Content-Length: 151

v=0
o=anton 2890844526 2890844526 IN IP4 serv1.niits.ru
s=-
c=IN IP4 192.0.2.101
t=0 0
m=audio 49172 RTP/AVP 0
a=rtpmap:0 PCMU/8000

Между пользователями Anton и Vladimir создаются RTP-потоки.

13. BYE Vladimir -> Proxy 3

BYE sip:anton@serv1.niits.ru SIP/2.0
Via: SIP/2.0/TCP serv5.loniis.ru:5060;branch=z9hG4bKfgaw2
Max-Forwards: 70
Route: <sip:ss3.loniis.ru;lr>
From: Vladimir <sip:vladimir@protei.ru>;tag=314159
To: Anton <sip:anton@niits.ru>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@niits.ru
CSeq: 1 BYE
Content-Length: 0

14. BYE Proxy 3 -> Anton

BYE sip:anton@serv1.niits.ru SIP/2.0
Via: SIP/2.0/TCP ss3.loniis.ru:5060;branch=z9hG4bK721e.1;received=192.0.2.100
Via: SIP/2.0/TCP serv5.loniis.ru:5060;branch=z9hG4bKfgaw2
Max-Forwards: 69

Библиотека БГУИР

From: Vladimir <sip:vladimir@protei.ru>;tag=314159
To: Anton <sip:anton@niits.ru>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@niits.ru
CSeq: 1 BYE
Content-Length: 0

15. **200 (OK) Anton -> Proxy 3**

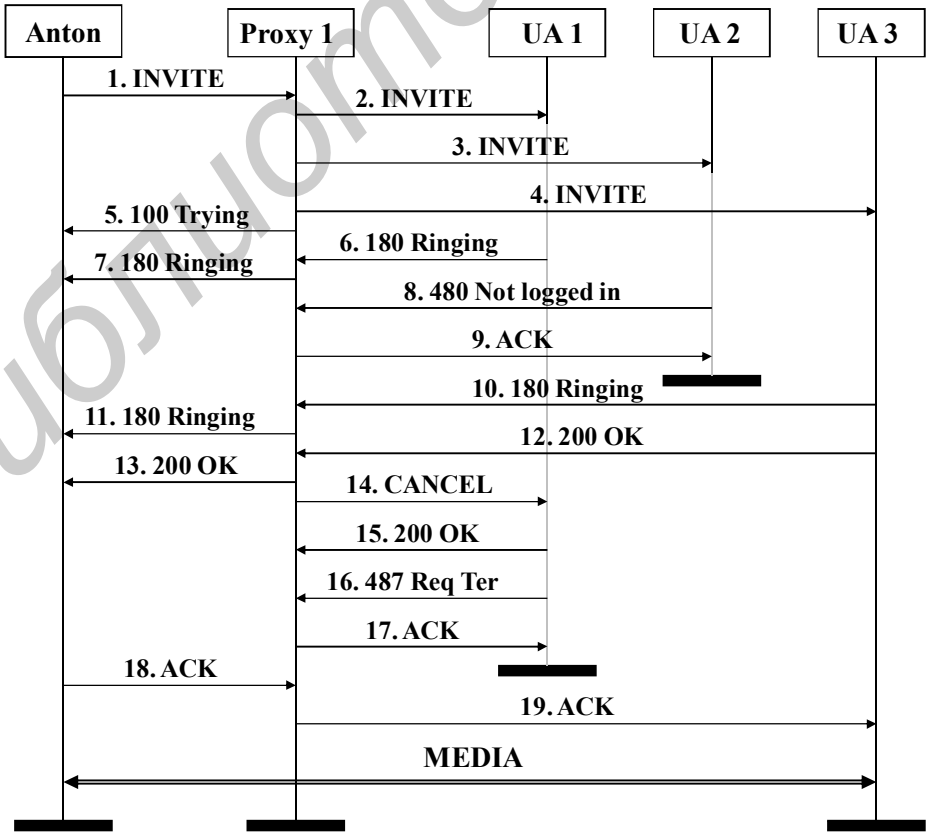
SIP/2.0 200 OK
Via: SIP/2.0/TCP ss3.loniis.ru:5060;branch=z9hG4bK721e.1;received=192.0.2.233
Via: SIP/2.0/TCP serv5.loniis.ru:5060;branch=z9hG4bKfgaw2;received=192.0.2.100
From: Vladimir <sip:vladimir@protei.ru>;tag=314159
To: Anton <sip:anton@niits.ru>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@niits.ru
CSeq: 1 BYE
Content-Length: 0

16. **200 (OK) Proxy 3 -> Vladimir**

SIP/2.0 200 OK
Via: SIP/2.0/TCP serv5.loniis.ru:5060;branch=z9hG4bKfgaw2;received=192.0.2.100
From: Vladimir <sip:vladimir@protei.ru>;tag=314159
To: Anton <sip:anton@niits.ru>;tag=9fxced76sl
Call-ID: 2xTb9vxSit55XU7p8@niits.ru
CSeq: 1 BYE
Content-Length: 0

Приложение 3

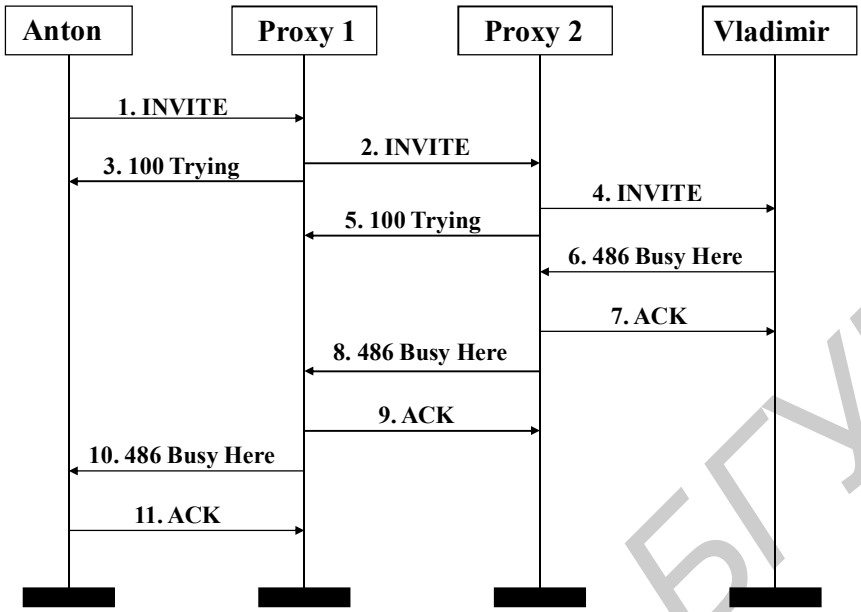
Установление соединения с участием прокси-сервера с параллельным поиском по нескольким направлениям



Библиотека БГУИР

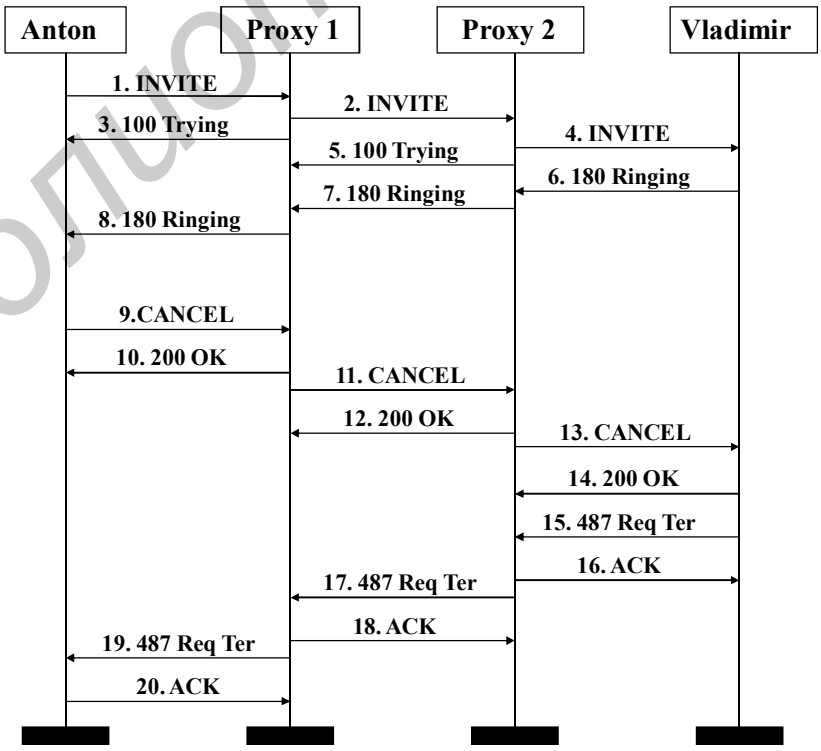
Приложение 4

Неудачная попытка установления соединения:
вызываемый абонент занят



Приложение 5

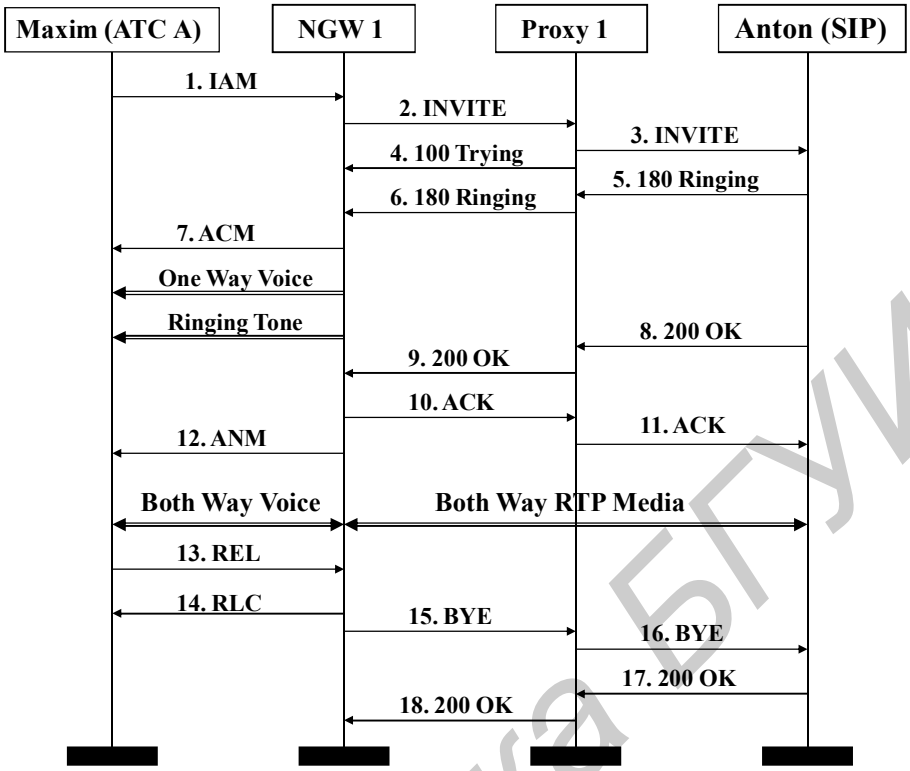
Неудачная попытка установления соединения: вызываемый абонент
не отвечает, вызывающий абонент прерывает вызов



Библиотека БГУИР

Приложение 6

Успешное установление соединения от абонента ISDN к пользователю сети SIP



1. IAM Maxim -> NGW 1

Получение сообщения IAM
CgPN=095-386-4515,NPI=E.164,NOA=National
CdPN=812-262-5326,NPI=E.164,NOA=National

2. INVITE NGW 1 -> Proxy 1

INVITE sip:+78122625326@ss1.a.loniis.ru;user=phone SIP/2.0
Via: SIP/2.0/UDP ngw1.a.loniis.ru:5060;branch=z9hG4bKlueha2
Max-Forwards: 70
From: <sip:+70953864515@ngw1.a.loniis.ru;user=phone>;tag=7643kals
To: <sip:+78122625326@ss1.a.loniis.ru;user=phone>
Call-ID: 4Fde34wkd11wsGFDs3@ngw1.a.loniis.ru
CSeq: 1 INVITE
Contact: <sip:ngw1@a.loniis.ru>
Content-Type: application/sdp
Content-Length: 146

v=0
o=GW 2890844527 2890844527 IN IP4 ngw1.a.loniis.ru
s=-
c=IN IP4 ngw1.a.loniis.ru
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

Библиотека БГУИР

3. INVITE Proxy 1 -> Anton

```
INVITE sip:anton@client.b.loniis.ru SIP/2.0
Via: SIP/2.0/UDP ss1.a.loniis.ru:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP ngw1.a.loniis.ru:5060;branch=z9hG4bKlueha2
 received=192.0.2.103
Max-Forwards: 69
Record-Route: <sip:ss1.a.loniis.ru;lr>
From: <sip:+70953864515@ngw1.a.loniis.ru;user=phone>;tag=7643kals
To: <sip:+78122625326@ss1.a.loniis.ru;user=phone>
Call-ID: 4Fde34wkd11wsGFDs3@ngw1.a.loniis.ru
CSeq: 1 INVITE
Contact: <sip:ngw1@a.loniis.ru>
Content-Type: application/sdp
Content-Length: 146

v=0
o=GW 2890844527 2890844527 IN IP4 ngw1.a.loniis.ru
s=-
c=IN IP4 ngw1.a.loniis.ru
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

4. 100 Trying Proxy 1 -> NGW 1

```
SIP/2.0 100 Trying
Via: SIP/2.0/UDP ss1.a.loniis.ru:5060;branch=z9hG4bK2d4790.1
 received=192.0.2.111
Via: SIP/2.0/UDP ngw1.a.loniis.ru:5060;branch=z9hG4bKlueha2
 received=192.0.2.103
From: <sip:+70953864515@ngw1.a.loniis.ru;user=phone>;tag=7643kals
To: <sip:+78122625326@ss1.a.loniis.ru;user=phone>
Call-ID: 4Fde34wkd11wsGFDs3@ngw1.a.loniis.ru
CSeq: 1 INVITE
Content-Length: 0
```

5. 180 Ringing Anton -> Proxy 1

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ss1.a.loniis.ru:5060;branch=z9hG4bK2d4790.1
 received=192.0.2.111
Via: SIP/2.0/UDP ngw1.a.loniis.ru:5060;branch=z9hG4bKlueha2
 received=192.0.2.103
Record-Route: <sip:ss1.a.loniis.ru;lr>
From: <sip:+70953864515@ngw1.a.loniis.ru;user=phone>;tag=7643kals
To: <sip:+78122625326@ss1.a.loniis.ru;user=phone>;tag=314159
Call-ID: 4Fde34wkd11wsGFDs3@ngw1.a.loniis.ru
CSeq: 1 INVITE
Contact: <sip:anton@client.b.loniis.ru>
Content-Length: 0
```

Библиотека БГУИР

6. 180 Ringing Proxy 1 -> NGW 1

```
SIP/2.0 180 Ringing
Via: SIP/2.0/UDP ngw1.a.loniis.ru:5060;branch=z9hG4bKlueha2
 received=192.0.2.103
Record-Route: <sip:ss1.a.loniis.ru;lr>
From: <sip:+70953864515@ngw1.a.loniis.ru;user=phone>;tag=7643kals
To: <sip:+78122625326@ss1.a.loniis.ru;user=phone>;tag=314159
Call-ID: 4Fde34wkd11wsGFDs3@ngw1.a.loniis.ru
CSeq: 1 INVITE
Contact: <sip:anton@client.b.loniis.ru>
Content-Length: 0
```

7. ACM NGW 1 -> Maxim

Получение сообщения ACM. NGW1 проключает односторонний тракт и передает абоненту Maxim акустический сигнал "контроль посылки вызова".

8. 200 OK Anton -> Proxy 1

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP ss1.a.loniis.ru:5060;branch=z9hG4bK2d4790.1
 received=192.0.2.111
Via: SIP/2.0/UDP ngw1.a.loniis.ru:5060;branch=z9hG4bKlueha2
 received=192.0.2.103
Record-Route: <sip:ss1.a.loniis.ru;lr>
From: <sip:+70953864515@ngw1.a.loniis.ru;user=phone>;tag=7643kals
To: <sip:+78122625326@ss1.a.loniis.ru;user=phone>;tag=314159
Call-ID: 4Fde34wkd11wsGFDs3@ngw1.a.loniis.ru
Contact: <sip:anton@client.b.loniis.ru>
CSeq: 1 INVITE
Content-Type: application/sdp
Content-Length: 151

v=0
o=Anton 2890844527 2890844527 IN IP4 client.b.loniis.ru
s=-
c=IN IP4 client.b.loniis.ru
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

9. 200 OK Proxy 1 -> NGW 1

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP ngw1.a.loniis.ru:5060;branch=z9hG4bKlueha2
 received=192.0.2.103
Record-Route: <sip:ss1.a.loniis.ru;lr>
From: <sip:+70953864515@ngw1.a.loniis.ru;user=phone>;tag=7643kals
To: <sip:+78122625326@ss1.a.loniis.ru;user=phone>;tag=314159
Call-ID: 4Fde34wkd11wsGFDs3@ngw1.a.loniis.ru
CSeq: 1 INVITE
Contact: <sip:anton@client.b.loniis.ru>
```

Библиотека БГУИР

Content-Type: application/sdp
Content-Length: 151

v=0
o=Anton 2890844527 2890844527 IN IP4 client.b.loniis.ru
s=-
c=IN IP4 client.b.loniis.ru
t=0 0
m=audio 3456 RTP/AVP 0
a=rtpmap:0 PCMU/8000

10. **ACK NGW 1 -> Proxy 1**

ACK sip:anton@client.b.loniis.ru SIP/2.0
Via: SIP/2.0/UDP ngw1.a.loniis.ru:5060;branch=z9hG4bKlueha2
Max-Forwards: 70
Route: <sip:ss1.a.loniis.ru;lr>
From: <sip:+70953864515@ngw1.a.loniis.ru;user=phone>;tag=7643kals
To: <sip:+78122625326@ss1.a.loniis.ru;user=phone>;tag=314159
Call-ID: 4Fde34wkd11wsGFDs3@ngw1.a.loniis.ru
CSeq: 1 ACK
Content-Length: 0

11. **ACK Proxy 1 -> Anton**

ACK sip:anton@client.b.loniis.ru SIP/2.0
Via: SIP/2.0/UDP ss1.a.loniis.ru:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP ngw1.a.loniis.ru:5060;branch=z9hG4bKlueha2
received=192.0.2.103
Max-Forwards: 69
From: <sip:+70953864515@ngw1.a.loniis.ru;user=phone>;tag=7643kals
To: <sip:+78122625326@ss1.a.loniis.ru;user=phone>;tag=314159
Call-ID: 4Fde34wkd11wsGFDs3@ngw1.a.loniis.ru
CSeq: 1 ACK
Content-Length: 0

12. **ANM NGW 1 -> Maxim**

Получение сообщения ANM. Устанавливается прямое двухстороннее мультимедийное соединение между абонентом ТфОП и пользователем сети SIP.

13. **REL Maxim -> NGW 1**

Получение сообщения REL с CauseCode=16 Normal

14. **RLC NGW 1 -> Maxim**

Получение сообщения RLC

15. **BYE NGW 1-> Proxy 1**

BYE sip:anton@client.b.loniis.ru SIP/2.0
Via: SIP/2.0/UDP ngw1.a.loniis.ru:5060;branch=z9hG4bKlueha2
Max-Forwards: 70
Route: <sip:ss1.a.loniis.ru;lr>
From: <sip:+70953864515@ngw1.a.loniis.ru;user=phone>;tag=7643kals
To: <sip:+78122625326@ss1.a.loniis.ru;user=phone>;tag=314159
Call-ID: 4Fde34wkd11wsGFDs3@ngw1.a.loniis.ru
CSeq: 2 BYE
Content-Length: 0

Библиотека БГУИР

16. BYE Proxy 1 -> Anton

```
BYE sip:anton@client.b.loniis.ru SIP/2.0
Via: SIP/2.0/UDP ss1.a.loniis.ru:5060;branch=z9hG4bK2d4790.1
Via: SIP/2.0/UDP ngw1.a.loniis.ru:5060;branch=z9hG4bKlueha2
   received=192.0.2.103
Max-Forwards: 69
From: <sip:+70953864515@ngw1.a.loniis.ru;user=phone>;tag=7643kals
To: <sip:+78122625326@ss1.a.loniis.ru;user=phone>;tag=314159
Call-ID: 4Fde34wkd11wsGFDs3@ngw1.a.loniis.ru
CSeq: 2 BYE
Content-Length: 0
```

17. 200 OK Anton -> Proxy 1

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP ss1.a.loniis.ru:5060;branch=z9hG4bK2d4790.1
  received=192.0.2.111
Via: SIP/2.0/UDP ngw1.a.loniis.ru:5060;branch=z9hG4bKlueha2
  received=192.0.2.103
From: <sip:+70953864515@ngw1.a.loniis.ru;user=phone>;tag=7643kals
To: <sip:+78122625326@ss1.a.loniis.ru;user=phone>;tag=314159
Call-ID: 4Fde34wkd11wsGFDs3@ngw1.a.loniis.ru
CSeq: 2 BYE
Content-Length: 0
```

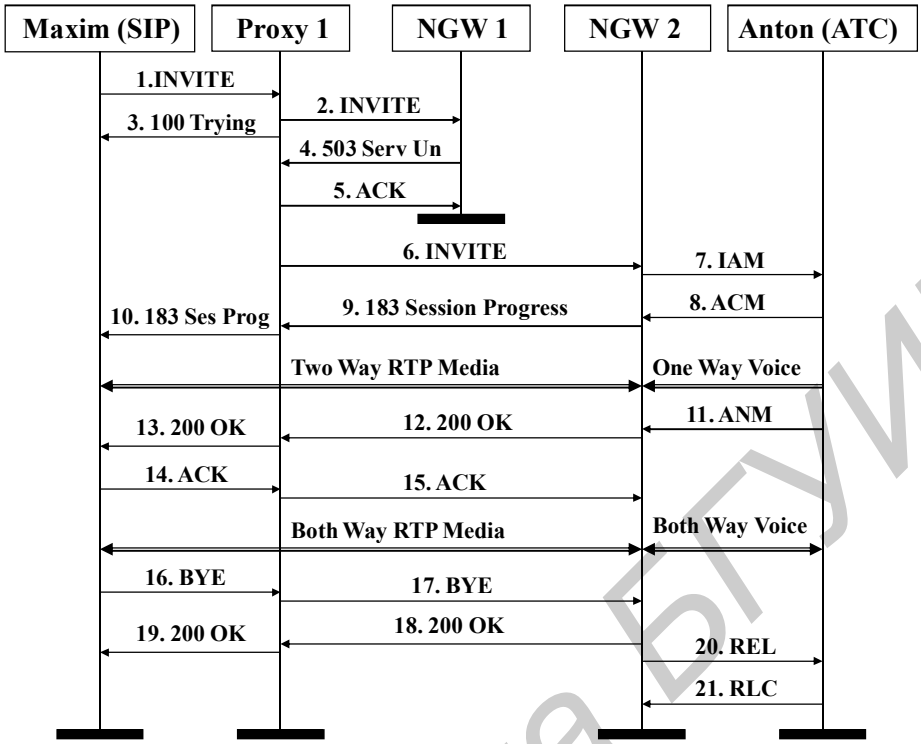
18. 200 OK Proxy 1 -> NGW 1

```
SIP/2.0 200 OK
Via: SIP/2.0/UDP ngw1.a.loniis.ru:5060;branch=z9hG4bKlueha2
  received=192.0.2.103
From: <sip:+70953864515@ngw1.a.loniis.ru;user=phone>;tag=7643kals
To: <sip:+78122625326@ss1.a.loniis.ru;user=phone>;tag=314159
Call-ID: 4Fde34wkd11wsGFDs3@ngw1.a.loniis.ru
CSeq: 2 BYE
Content-Length: 0
```

Библиотека БГУИР

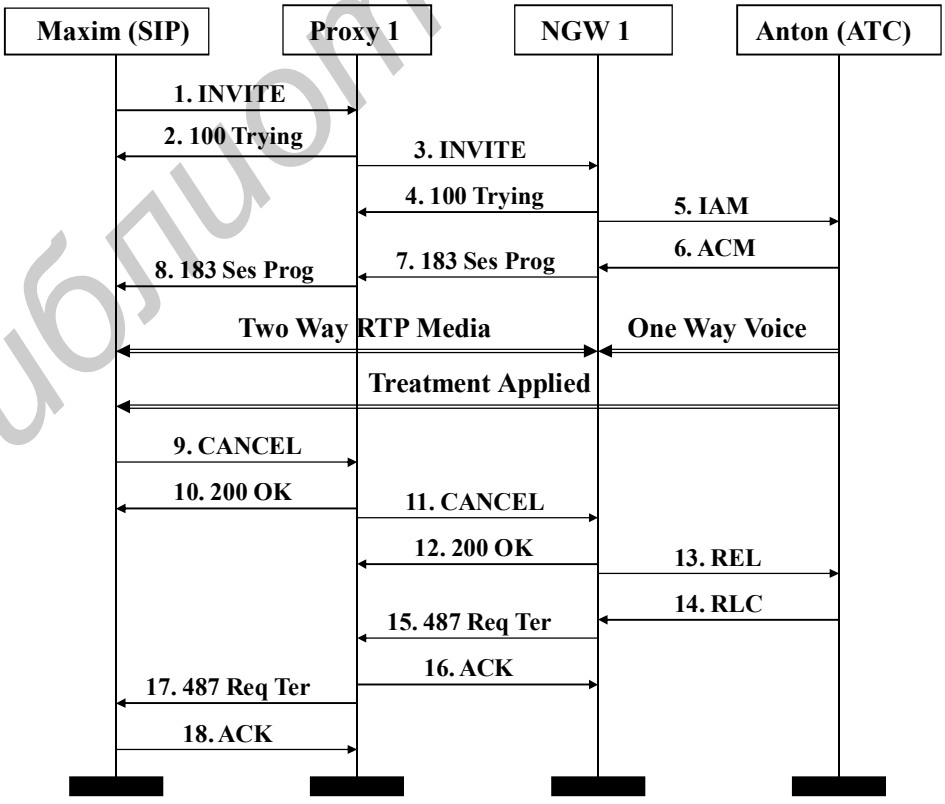
Приложение 7

Успешное установление соединения из сети SIP в сеть ТфОП
в условиях перегрузки шлюза



Приложение 8

Неуспешное установление соединения из сети SIP в сеть ТфОП:
сообщение об ошибке из ТфОП



Библиотека БГУИР

Контрольные вопросы

1. Перечислить основные элементы SIP-сети и их функции.
2. Тип адресации, используемый в протоколе SIP.
3. Основные типы SIP-адресов и назначение их элементов.
4. Описать процесс установления соединения между терминалами на примере SIP-сети.
5. Формат и структура сообщений протокола SIP.
6. Назначение основных заголовков сообщений SIP.
7. Назначение запросов и ответов протокола SIP.
8. Описать процесс установления соединения через сервер переадресации.
9. Описать процесс установления соединения через прокси-сервер.
10. В чем разница сценариев согласно п. 8 и п. 9?
11. В какие моменты времени терминалы пользователей посылают информацию о своих функциональных возможностях? В каких сообщениях эта информация располагается?
12. Какое минимальное число сообщений необходимо для установления соединения?
13. Как выглядел бы сценарий, если бы сервер определения местоположения не нашел пользователя?
14. Составить сценарий установления успешного соединения между терминалами пользователей A и B согласно заданию:

Терминал A→ прокси-сервер→ сервер переадресации→ прокси-сервер→ терминал B	Терминал A→ сервер переадресации→ прокси-сервер→ сервер переадресации→ терминал B	Терминал A→ прокси-сервер→ прокси-сервер→ сервер переадресации→ терминал B	Терминал A→ сервер переадресации→ прокси-сервер→ прокси-сервер→ терминал B

15. Для заданного сценария установления соединения определить синтаксическую структуру сообщений SIP.

Библиотека БГУИР

Литература

1. Гольдштейн Б. С. Протокол SIP : Справочник / Б. С. Гольдштейн. – СПб. : БХВ – Санкт-Петербург, 2005.

2. Гольдштейн, Б. С. IP-телефония / Б. С. Гольдштейн. – М. : Радио и связь, 2001.

3. Johnston, Alan B. SIP–Understanding the Session Initiation Protocol / Second Edition. – Artech House : Boston – London, 2006.

4. International Telecommunication Union. – «Packet based multimedia communication systems», Recommendation H.323 / Telecommunication Standardization Sector of ITU, Geneva, Switzerland, Feb., 1998.

5. Internet Engineering Task Force. «SIP: Session Initiation Protocol. Internet-Draft» / MMUSIC WG, Columbia University, May 2000 Expires: November 2000.

Библиотека БГУИР

Содержание

Библиотека БГУИР

Св. план 2010, поз. 91

Учебное издание

Хоменок Михаил Юлианович

СИГНАЛИЗАЦИЯ НА СЕТЯХ ПЕРЕДАЧИ ДАННЫХ С ПАКЕТНОЙ КОММУТАЦИЕЙ. ПРОТОКОЛ SIP

Методическое пособие
по курсу
«Сетевые технологии и сигнализация в телекоммуникациях»
для студентов специальностей
1-45 01 03 «Сети телекоммуникаций»,
1-45 01 05 «Системы распределения мультимедийной информации»,
1-98 01 02 «Защита информации в телекоммуникациях»
всех форм обучения

Библиотека БГУИР

Редактор Т. П. Андрейченко
Корректор А. В. Тюхай
Компьютерная верстка М. В. Гуртатовская

Подписано в печать	Формат 60×84 1/16.	Бумага офсетная.
Гарнитура «Таймс».	Отпечатано на ризографе.	Усл. печ. л.
Уч. изд. л. 3,4.	Тираж 50 экз.	Заказ 501.

Издатель и полиграфическое исполнение: учреждение образования
«Белорусский государственный университет информатики и радиоэлектроники»
ЛИ №02330/0494371 от 16.03.2009. ЛП №02330/0494175 от 03.04.2009.
220013, Минск, П. Бровки, 6