Моим женщинам:
маме, жене и дочери

Андрей Робачевский

ИНТЕРНЕТ ИЗНУТРИ
Экосистема глобальной Сети

Москва
2015

УДК 004.738.5
ББК 32.973.202
Р12

Издано при содействии компании MSK-IX

Редактор В. Иванченко

Робачевский А.

Р12 Интернет изнутри: Экосистема глобальной Сети / Андрей Робачевский. — М.: Альпина Паблишер, 2015. — 223 с.

ISBN 978-5-9614-4803-0

Книга рассказывает об архитектуре и технологиях Интернета, фокусируясь на его основных компонентах: глобальной адресации и протоколе IP, системе доменных имен и глобальной межсетевой маршрутизации. Рассматриваются аспекты и принципы работы Всемирной сети, вопросы стандартизации, развития и безопасности основных систем Интернета. Обсуждается архитектурная эволюция Интернета в целом, а также связанные с ней вопросы внедрения новых протоколов и технологий.

Особое внимание уделено экосистеме Интернета, ее истории, а также основным организациям, включенным в систему принятия решений в Интернете.

Книга рассчитана на техническую аудиторию: сетевых операторов (администраторов), разработчиков программного обеспечения. Она также будет полезна тем, кто интересуется архитектурными аспектами Сети, вопросами «управления» Интернетом, и всем желающим расширить свой кругозор в области интернет-технологий.

УДК 004.738.5
ББК 32.973.202

ISBN 978-5-9614-4803-0

Содержание

Предисловие

Интернет — это просто и легко?

Казалось бы, чего проще — нажал кнопку и получил письмо (зашел на сайт, прочитал афишу, ленту новостей, обменялся комментариями, заказал билет). Мы делаем это ежечасно, ежеминутно. Интернет давно стал неотъемлемой частью нашей жизни уже настолько, что если где-то нет интернет-связи — это вызывает искреннее возмущение.

Миллиарды человек во всем мире используют Интернет, не задумываясь над тем, что за внешней простотой скрывается сложная инфраструктура.

Что происходит после нажатия на Enter? Куда и как идут данные? Почему возникают сбои в Сети? Как защититься в Сети? Почему все так сложно, если все так просто?

А что внутри?

Миллионы сетей в разных странах, находящихся под автономным управлением, умудряются работать четко и слаженно, как единый организм. Телекоммуникационная инфраструктура, по которой передаются потоки данных, технические стандарты и услуги, благодаря которым Интернет работает, стандарты контента и прило-

жений, безопасность и стабильность — вот основополагающие вопросы, связанные с функционированием Интернета.

Андрей Робачевский написал замечательную книгу «Интернет изнутри», альтернатив которой вы не найдете сегодня. В ней автор дает доходчивые ответы на многие вопросы об устройстве Интернета. Профессионализм автора, который с начала 1990-х гг. работает в интернет-отрасли, и глубокое понимание основополагающих вопросов, связанных с функционированием Интернета, лежат в основе каждой статьи, составляющих эту книгу.

MSK-IX, начиная с 1995 г., поддерживает и развивает инфраструктурные проекты в сети Интернет. Мы выступили инициатором создания этой замечательной книги и подготовили ее издание специально к юбилейному, X Пиринговому форуму MSK-IX.

Я буду рада, если эта книга поможет всем нам еще на шаг приблизиться к пониманию устройства Интернета, к пониманию того, что за внешней простотой лежат сложные технологии и процессы, работа многих профессионалов и уникальных специалистов.

Елена Воронина,
исполнительный директор MSK-IX

Универсальный коннектор

В моем смартфоне 15 новых событий: у приятеля сегодня день рождения — не забыть поздравить его в «Фейсбуке»; знакомый пытался связаться по «Скайпу» и оставил видеосообщение; судя по фотографиям в «Инстаграме», которые опубликовал мой брат, в Санкт-Петербурге отличная погода. В почтовом ящике — новые письма, новостной сайт предлагает свежую подборку на интересующие меня темы, авиакомпания обнадеживает, что мой рейс задержится всего на 30 минут…

Типичный момент из жизни сотен миллионов людей во всем мире.

Интернет меняет наше представление о расстоянии и времени. Информация, личные данные и даже сами человеческие отношения приобретают новое измерение благодаря Сети. Интернет сегодня — нечто гораздо большее, чем технологии и протоколы, глобально взаимосвязанные сети и разнообразные устройства, онлайн-услуги, приложения и колоссальные объемы информации. Это — экосистема, живущая и развивающаяся по своим законам. Это — универсальный коннектор, поглощающий любой подключенный к нему элемент, который, в свою очередь, сам становится частью Сети.

Но чтобы лучше понять законы, по которым живет Интернет, нам придется разобраться, как работают его основные подсистемы и как они взаимодействуют между собой. Мы заглянем за облачный фасад

приложений и услуг, чтобы увидеть Интернет изнутри. Мы узнаем, как зародились и развились протоколы, технологии и взаимоотношения, составляющие основу Всемирной паутины.

К счастью, нам не придется совершать археологическое исследование, наш прыжок в прошлое будет длиной лишь одно поколение. Ведь уникальность интернет-революции еще и в том, что ее непосредственными участниками являемся мы сами.

А. Робачевский,
2014 г.

Глава 1

Интернет-протокол IP и глобальная система адресации

> Следует определить различия между именами, адресами и маршрутами. Имя определяет то, что мы пытаемся найти. Адрес указывает, где это находится. Маршрут показывает, как туда попасть.
>
> RFC 760, первая спецификация Интернет-протокола IPv4, 1980 г.

Три дня рождения Интернета

Ранним утром 29 октября 1969 г. произошло историческое событие — рождение Интернета. В тот момент мало кто осознавал значимость этого события. Чарли Кляйн (Charley Kline) на своем терминале в университете Калифорнии (UCLA) набрал слово LOGIN, чтобы отправить эту команду компьютеру в Стэнфордском исследовательском институте (SRI), за которым ожидал коллега Чарли, Билл Дювал (Bill Duvall). Первый символ 'L' проделал путь 500 км, был принят компьютером Билла и послан обратно, появившись на терминале Чарли. За ним последовал символ 'O'. На символе 'G' система сломалась, но была полностью восстановлена часом позже. Так был рожден Интернет.

Чарли и Билл были молодыми программистами, сотрудниками двух крупнейших американских научных центров. Рождению Интернета предшествовало десятилетие научных исследований, а свой вклад внесли десятки, если не сотни, людей, разработавших базовые концепции архитектуры Интернета.

Еще в начале 1960-х гг. ряд исследователей, многие из которых в дальнейшем участвовали в проекте ARPANET, увидел огромные перспективы в способности компьютеров обмениваться друг с другом данными. В 1965 г. было установлено тестовое соединение между компьютерами Массачусетского института технологии и Университета Южной Калифорнии — использовалась традиционная телефонная технология синхронной коммутации каналов. Стало очевидно, что такая коммутация не позволяет эффективно использовать канал связи, но именно в ходе этого эксперимента начал обретать очертания «эмбрион» будущего Интернета.

Слово «Интернет» вошло в обиход в середине 1970-х, а до того Сеть называлась ARPANET. По сравнению с телефонными сетями, основанными на коммутации каналов, в ARPANET было решено использовать технологию коммутации пакетов, или дейтаграмм, — данных ограниченного объема, заключенных в «конверты» с указанием источника и получателя. Поскольку каждый пакет обрабатывался независимо, сети не требовалось хранить информацию о соединениях между оконечными компьютерами и потоках данных между ними. Этот подход позволил существенно упростить архитектуру сети и повысить ее надежность. Узел сети мог выйти из строя — но его функцию немедленно брал на себя другой, рабочий узел. Кроме того, асинхронная пакетная передача больше соответствовала характеру работы многозадачных операционных систем. Так, ОС Unix позволяла разделять ресурсы между несколькими задачами одновременно — процессор занимался и обработкой команд с многочисленных терминалов, и вычислением крупных массивов данных.

Telnet (удаленный доступ в режиме терминала) и электронная почта (e-mail) появились в ARPANET в 1972 г., а ftp (обмен файлами) — годом позже. Первое время для обмена данными между компьютерами, или хостами, использовался протокол NCP (Network Control Protocol), предтеча сегодняшнего TCP/IP.

Функциональность протокола NCP ограничивалась тем, что это по существу был транспортный протокол. Он не был хорошо приспособлен для работы с разнообразными технологиями — например, цифровой радио- и спутниковой связью. Более того, он предназначался для работы только с одной сетью — ARPANET, а значит, не был способен осуществлять адресацию в других сетях и среди подключенных к ним компьютеров.

В это же время Роберт Кан (Robert Kahn), сотрудник агентства передовых исследовательских проектов DARPA, работал над концепцией открытой сетевой архитектуры. В рамках этой концепции независимые сети, различные по своей архитектуре и использу-

емым технологиям, должны были свободно обмениваться данными. Требовалась лишь единая межсетевая модель для «прозрачного» обмена данными между компьютерами в различных сетях. Особенностью концепции Канна было то, что он рассматривал и функциональность беспроводных сетей пакетной коммутации. Поскольку радиосигнал может подвергаться искажениям до полной потери (например, при перемещении в туннеле), протокол должен был обеспечить надежную передачу данных независимо от качества сети.

В 1973 г. Кан начал разработку протокола, который позволил бы передавать данные между хостами, используя любую коммуникационную технологию. Кан пригласил в свой проект Винтона Серфа (Vinton Cerf), в то время сотрудника Стэнфордского университета. Серф обладал необходимым опытом: ранее он участвовал в создании протокола NCP и разрабатывал сетевые интерфейсы к различным операционным системам. Благодаря совместным усилиям Кана и Серфа концепция нового протокола была представлена уже в сентябре 1973 г., а годом позже, в декабре 1974-го, Серф вместе со своими аспирантами Йогеном Далалем (Yogen Dalal) и Карлом Саншайном (Carl Sunshine) опубликовал первую полную спецификацию протокола TCP. Аббревиатура означала Transmission Control Program, а сам протокол объединял в себе функции сегодняшних протоколов TCP и IP. Новейшая спецификация была зафиксирована в серии документов Request for Comments (RFC) под номером RFC 675 *(https://datatracker.ietf.org/doc/rfc675)*

Интересно, что изначально архитектура протокола TCP предполагала использование 4 бит для адресации сети (допуская тем самым существование 16 сетей, из которых 6 были уже назначены: ARPANET, UCL, CYCLADES, NPL, CADC, EPSS), а также 16 бит для адресации хостов в сети (или «процессов TCP»). При этом заголовок пакета также содержал поле длины сетевого адреса, тем самым обеспечивая расширение адресного пространства при необходимости до 64 бит. Однако в следующей версии протокола TCP v2, опубликованной в 1977 г., уже использовались только адреса фиксированной длины. А ведь изначальная структура TCP предлагала более элегантный способ борьбы с нехваткой адресного пространства, нежели создание протокола IPv6, несовместимого с IPv4!

В том же 1977 г. была проведена первая серьезная демонстрация работы «Интер-Нета»: три сети, использующие различные сетевые технологии — ARPANET, SATNET и сеть пакетного радио, — успешно обменивались данными по протоколу TCP. Так Интернет был рожден во второй раз.

Двумя месяцами позже в то время аспирант калифорнийского университета Лос-Анджелеса (UCLA) Джон Постел (Jon Postel) опубликовал статью, где предложил новый архитектурный взгляд на TCP — протокол, состоящий из двух компонентов. Первый компонент, который в последующей спецификации TCP v3 получит название IP (Internet Protocol), отвечал только за передачу пакетов между узлами сети и маршрутизацию. Второй же компонент, TCP, обеспечивал сквозной поток данных между оконечными устройствами, контроль ошибок и повторную передачу потерянных данных.

В 1978 г. Постел публикует четвертую версию протоколов IP и TCP. И наконец в 1980 г. публикуется документ RFC 760, содержащий спецификацию IPv4 и принципы архитектуры Интернета, какими мы их знаем сегодня.

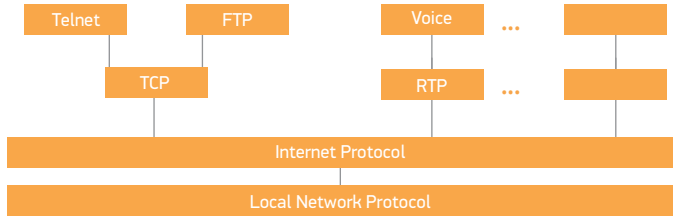

Рис. 1. Взаимодействие протоколов стека TCP/IP, определенное в спецификации RFC 760 (https://datatracker.ietf.org/doc/rfc760/)

В рамках этой архитектуры IP отвечает за адресацию и фрагментацию пакетов при передаче от одного узла сети к другому. Адреса позволяют узлу принять решение, какому следующему узлу направить данные, а с помощью фрагментации данные можно передавать между сетями с различными допустимыми размерами пакета.

Этим, собственно, функции протокола IP и ограничиваются. В своей работе IP опирается на протоколы нижнего уровня, используемые в локальной сети, и транспортные протоколы, например, TCP. Сам же интернет-протокол не обеспечивает надежную передачу. В спецификации RFC 760 указано, что в протоколе IP «отсутствуют подтверждения, как сквозные, так и межузловые. Отсутствует контроль ошибок, за исключением контрольной суммы заголовка. Отсутствует функция повторной передачи. Отсутствует управление потоком данных».

Переход к семейству протоколов TCP/IP

Если вы думаете, что с переходом к протоколу IPv6 Интернет впервые переживает столь фундаментальное изменение базового про-

токола, то это не так. Сеть ARPANET конца 1970-х по-прежнему использовала протокол NCP, ограничивая возможности прозрачного обмена данными с другими сетями, например с сетями пакетного радио или спутниковыми сетями. А в этом и заключалась основа концепции Интернета.

В ноябре 1981 г. Джон Постел опубликовал план перехода ARPANET от протокола NCP к протоколам TCP/IP (https://datatracker.ietf.org/doc/rfc801/). Учитывая, что новый протокол уже прошел успешное тестирование в различных конфигурациях, на переход отводился один год.

Тогдашнюю ARPANET невозможно сравнить с сегодняшней сетью Интернет — и по размеру, и по зависимости общества и экономики от ее функционирования, и по степени контроля и координации. Тем не менее переход занял целый год и потребовал определенного количества напоминаний и увещеваний со стороны Джона Постела. Кроме того, на сутки был отключен протокол NCP по всему ARPANET'у, так что только узлы, поддерживавшие протокол TCP/IP, могли обмениваться данными.

Окончательный переход на TCP/IP произошел, как и было запланировано, 1 января 1983 г. Так Интернет был рожден в третий раз, теперь с протоколом IPv4.

Эволюция системы адресации: от протокола IPv4 к протоколу IPv6

В 1981 г. трудно было представить, что 32 бита адреса IPv4, позволяющие присвоить уникальный номер 4 миллиардам систем (компьютеров, маршрутизаторов и т.п.), когда-либо станут реальным ограничением. Однако уже к 1992 г. масштабируемость и ограниченность адресного пространства IPv4 встала на повестку дня.

Для поиска решения проблемы в ноябре 1991 г. организация по стандартизации IETF сформировала специальную группу для «мозгового штурма» в области маршрутизации и адресации — ROAD (Routing and Addressing). Учеными было найдено краткосрочное решение проблемы: они предложили супернеты — концепцию, впоследствии переработанную в архитектуру CIDR (Classless Inter-Domain Routing, бесклассовая междоменная маршрутизация). Этот подход, который был стандартизован в 1993 г. (RFC 1518, RFC 1519), позволил существенно замедлить расходование запаса доступных адресов.

В чем заключалась суть концепции CIDR? Граница подсетей становилась подвижной в зависимости от фактического разме-

ра адресуемой сети. Вместо распределения сетей класса C (/24) фиксированного размера (254 устройства) стало возможным создавать сети /23, /22 и так далее. Внутри же сервис-провайдер мог создать структуру, более соответствующую реальной топологии, распределяя сети меньшего размера, например /25. CIDR предполагал изменения как в системе распределения адресного пространства (об этом мы поговорим позже, в разделе «Глобальная система администрирования адресного пространства»), так и в системе маршрутизации.

Последнее было связано с тем, что фактически произошел отказ от концепции классов сетей (A, B, C и D), в которой деление между сетевым адресом и адресом устройства в сети было предопределено. Для маршрутизации CIDR стало необходимым явно указывать, сколько битов IP-адреса относятся к адресу сети (это данные, которые носят название «сетевая маска»).

Вот что отметили в то время члены руководящего комитета IETF — IESG (https://datatracker.ietf.org/doc/rfc1380): «CIDR потребует изменения в политике [распределения адресных ресурсов], спецификации протоколов, разработке и внедрении ПО для маршрутизаторов, но не требуется изменение программного обеспечения оконечных устройств». И действительно, новая архитектура была внедрена достаточно быстро. Этому содействовали и относительно небольшой размер Интернета, и его научно-исследовательский характер, и то, что опорная инфраструктура и протоколы находились в стадии разработки.

CIDR позволил избавиться от острых симптомов надвигающейся проблемы, но глобальное решение еще требовалось найти. Поэтому в начале 1994 г. IETF начал работу над созданием новой версии протокола IP, позднее получившей название IPv6. Базовая спецификация была опубликована в 1998 г. (RFC 2460), а окончательная версия структуры адресации IPv6 — в 2006-м (RFC 4291).

Основные отличия IPv6 от протокола предыдущего поколения — IPv4

Размер адресного пространства

Размер адреса IPv6 составляет 128 байт. Чтобы лучше представить, насколько у IPv6 больше адресного пространства, возьмем следующую аналогию: если бы все адреса IPv4 уместились в iPod'e, то для IPv6 потребовался бы весь земной шар (http://blog.icann.org/2007/06/ipv6-the-ipod-and-the-earth)!

IPv6 — это колоссальное количество доступных адресов, это возможность адресации любого мыслимого и немыслимого устройства. Эффективное использование возможностей нового протокола способно породить новый виток информационной революции. Достаточно посмотреть на текущий уровень распределения адресного пространства РИРами (региональными интернет-регистраторами): ясно видно, что их IPv6-пул далек от опустошения (рис. 2).

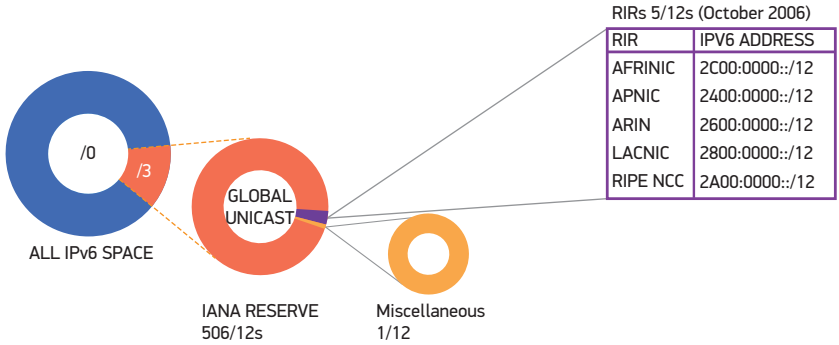

Рис. 2. Адресное пространство IPv6, предоставленное для распределения через региональные интернет-регистратуры. На настоящий момент IPv6-пул РИРов составляет чуть больше 5 блоков /12, и он далек от опустошения; Эти блоки составляют ничтожный процент всего доступного в будущем адресного пространства

Источник: статистика NRO http://www.nro.net/statistics

Помимо значительного увеличения адресного пространства изначально предполагалось, что IPv6 сможет поддерживать большее число уровней сетевой иерархии и обеспечит оптимальное распределение адресного пространства с точки зрения маршрутизации и конфигурации. Но в этом отношении ожидания создателей не оправдались: довольно жесткая иерархическая структура адресации была отвергнута операторами в пользу гибкой архитектуры CIDR. Также на сегодняшний момент IPv6 унаследовал многие «болячки» IPv4 (например, независимое от провайдера адресное пространство), которые не способствуют сдерживанию роста таблицы маршрутизации.

Расширяемость и дополнительные функции

При разработке протокола IPv6 особое внимание было уделено возможности добавления новых функций без потери эффективности обработки пакетов на сетевом уровне. IPv6 предполагает наличие

дополнительных заголовков для различных расширений (extension header, EH) — например, для криптографической защиты данных (Authentication EH и Encapsulating Security Payload EH). В то же время базовый заголовок IPv6 содержит минимальное число полей и имеет фиксированный размер. В частности, в IPv6 маршрутизаторы не производят фрагментацию, поэтому поля, относящиеся к этой функции, перенесены в соответствующий заголовок расширений (Fragmentation EH).

Как видно из рис. 3, заголовки расширений связаны цепочкой указателей Next Header («Следующий заголовок»).

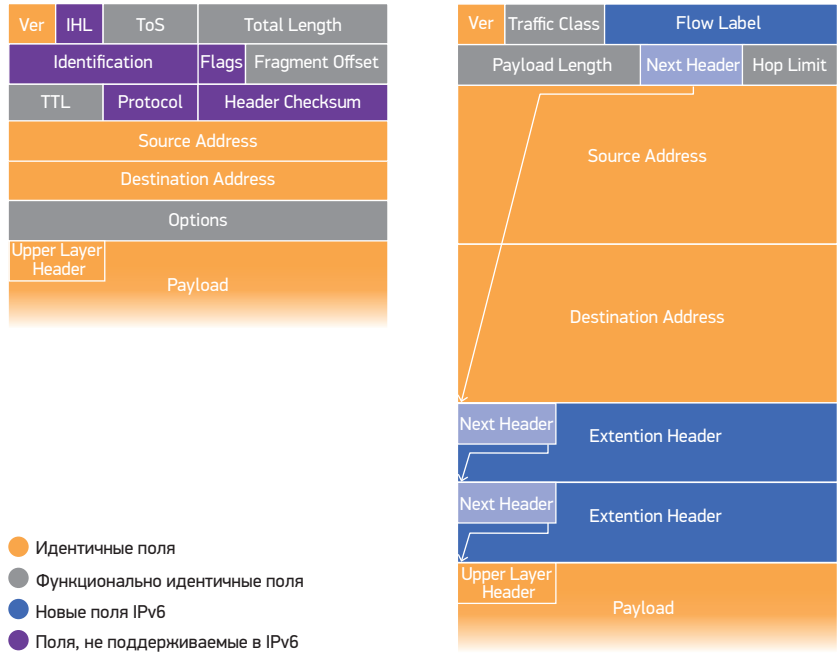

Рис. 3. Форматы пакетов IPv4 и IPv6

Фрагментация

Как было упомянуто выше, протокол IPv6 иначе обрабатывает фрагментацию пакетов. В случае IPv4, когда маршрутизатор получает пакет, размер которого превышает предел передачи через интерфейс, маршрутизатор производит фрагментацию — дробление пакета на более мелкие части. В дальнейшем они консолидируются получателем в исходный пакет. Заголовок пакета IPv4 имеет соответствующее поле (Fragment Offset), поддерживающее эту функцию.

В IPv6 фрагментация промежуточными устройствами запрещена. Если пакет IPv6 превышает допустимый размер для последующей передачи, маршрутизатор генерирует сообщение ICMP «packet too big» («слишком большой пакет») и посылает его обратно отправителю. В зависимости от приложения отправитель либо выбирает размер пакета, который позволит ему на всем пути следовать без фрагментации, либо дробит пакет самостоятельно. Как и в случае IPv4, консолидация фрагментированных пакетов входит в задачу получателя. Как следствие, передача пакетов IPv6 требует меньших затрат от промежуточного сетевого оборудования.

Автоконфигурация

Для протокола IPv6 была разработана так называемая система автоконфигурации без сохранения состояния (Stateless Autoconfiguration). Данный протокол позволяет различным устройствам, присоединенным к сети IPv6, получить необходимые установки для доступа в Интернет без дополнительных средств — например, без сервиса DHCP (Dynamic Host Configuration Protocol). Суть подхода заключается в том, что устройство получает адрес, состоящий из префикса сети и идентификатора устройства, автоматически сгенерированного с использованием MAC-адреса.

Защита данных

В протокол IPv6 изначально включена система безопасности, основанная на технологии IPsec. Предусмотрено два режима работы: транспортный и туннельный. В транспортном режиме производится защита (шифрование) данных пакета, но не заголовка. С точки зрения маршрутизации такой IP-пакет выглядит вполне обычно, а в задачу получателя входит декодирование содержимого пакета. При использовании туннельного режима данные всего пакета, включая заголовок, шифруются и инкапсулируются в новый пакет. Получатель, указанный в этом новом пакете, является окончанием защищенного канала, или туннеля, и в его задачу входит извлечение изначального пакета и последующая обработка. Дополнительно пакет IPv6 содержит заголовок аутентификации (Authentication EH) для определения подлинности и отсутствия модификации данных пакета.

Мобильность

Поддержка мобильности в протоколах IP означает, что оконечное устройство может изменить свое местоположение в сети и IP-адрес

без потери существующих связей, которые соответствуют потокам передачи данных. Для этого мобильные устройства используют отдельные IP-адреса, по которым устройства всегда доступны при передаче данных. За авторизацию мобильного устройства в сети и обеспечение соответствия между реальным и мобильным IP-адресами отвечает «Домашний агент» — устройство, расположенное в «домашней» сети мобильного пользователя. Реализация мобильности в протоколах IPv4 и IPv6 различается. В случае IPv4 передача данных также производится (туннелируется) через «Домашнего агента», в то время как в IPv6 «Домашний агент» обеспечивает только контролирующие функции (авторизацию и обеспечение соответствия между реальным и мобильным адресами). При этом передача данных производится между отправителем и получателем напрямую. Такой подход оптимизирует маршрутизацию данных и, как следствие, повышает качество передачи.

Приведенные особенности протокола IPv6 призваны улучшить производительность, качество и защиту передачи данных. Однако опыт практического внедрения протокола IPv6 показывает, что указанные улучшения весьма незначительны и во многих случаях не используются. Напротив, операторы зачастую прибегают к проверенным методам, разработанным для сетей IPv4. Так, для конфигурации подключенных устройств используется система DHCP, а в области защиты данных технология IPsec может быть использована в IPv4 почти так же эффективно, как и в IPv6. Эффективная поддержка multihoming (подключения клиента к нескольким сервис-провайдерам для повышения надежности) в IPv6 потребовала отдельного решения и существенно усложнила элегантную структуру маршрутизации, считающейся одним из преимуществ IPv6. В результате на практике multihoming реализуется аналогично IPv4, что приводит к неоправданному росту таблиц маршрутизации.

Неудивительно, что в среде сетевых операторов существует мнение, что основное преимущество IPv6 — только лишь расширение доступного адресного пространства.

Практика и проблемы внедрения протокола IPv6

Стратегия развития: сосуществование IPv4 и IPv6

Основная проблема перехода от IPv4 к IPv6 — несовместимость двух протоколов. Клиент IPv6 не может напрямую общаться с клиентом, поддерживающим только IPv4.

Изначально представлялось, что эту проблему решит внедрение «двойного стека» — когда компьютеры сети поддерживают оба протокола и подключены как к сети IPv4, так и к сети IPv6. Данное разделение является логическим, а физически используется одна и та же сетевая инфраструктура. Для доступа к ресурсам IPv4 используется протокол IPv4, а к ресурсам IPv6 — протокол IPv6. Все достаточно просто, но…

Темпы внедрения IPv6 оказались незначительными. План «двойного стека» сработает, если в ближайшем будущем подавляющее большинство компьютеров Интернета будут иметь доступ как к IPv4, так и к IPv6. В таком случае можно будет просто отключить поддержку IPv4 и — чудо! — Интернет перейдет на новый протокол. Однако реальных предпосылок для этого нет.

Сложность внедрения протокола IPv6 во многом связана с так называемым «сетевым эффектом». Этот экономический термин описывает явление, когда ценность технологии зависит от числа игроков, ее использующих. Действительно, возможность обмениваться трафиком IPv6 с парой других энтузиастов, как это было в начале 2000-х, с практической точки зрения не представляет особого интереса. Этот эффект усугубляется тем, что большая часть Интернета по-прежнему доступна только через протокол IPv4. Размер этой части Интернета определяет значимость протокола IPv4 и, в обратной пропорции, протокола IPv6 для сервис-провайдеров.

Каждый новый подключенный клиент должен иметь возможность обмениваться данными с Интернетом по протоколу IPv4, что требует предоставления ему адреса IPv4. Скажем прямо, для растущих сервис-провайдеров, возможно, более приоритетным станет решение проблемы нехватки адресов IPv4, чем внедрение IPv6. В то же время важно отметить, что обсуждаемая стратегия и динамика сосуществования двух протоколов основана на предположении, что инфраструктура сервис-провайдера обеспечивает полноценную поддержку IPv6.

Динамика потребности в адресном пространстве IPv4 по мере глобального внедрения IPv6 показана на рис. 4. На нем светло-зеленой линией обозначен рост глобального Интернета. По мере внедрения протокола IPv6 доля Интернета, доступного только по IPv4, будет неуклонно уменьшаться (темно-зеленая кривая). Синяя линия отображает размер сервис-провайдера, характеризуемый, например, числом подключенных пользователей. В данном случае рассматривается растущий провайдер. Наконец, потребность в адресах IPv4 показана кривой красного цвета.

По мере расширения клиентской базы провайдера пропорционально увеличивается потребность в дополнительных адресах IPv4. В то же время все большая и большая часть Интернета становится доступной по протоколу IPv6, что выражается в обратной тенденции, когда все меньшее число пользовательских соединений основано на протоколе IPv4. Соответственно, потребность в адресах IPv4 снижается. Наконец, когда подавляющее большинство ресурсов Интернета станет доступным по IPv6, потребность в IPv4 станет ничтожной. Таким образом, завершится фаза перехода Интернета на протокол IPv6. Продолжительность этой фазы может занять несколько лет. Не исключена, правда, вероятность, что данная фаза не закончится никогда, но об этом — чуть позже.

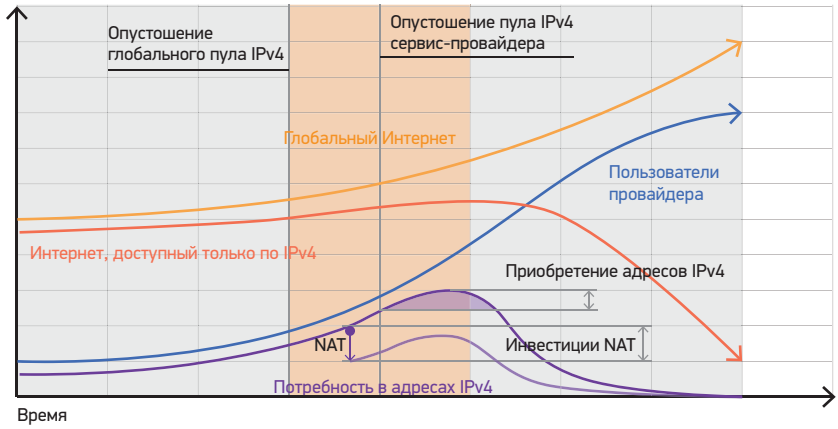

Рис. 4. Динамика сосуществования IPv4 и IPv6

Как видно из графика, наиболее критичной фазой для сервис-провайдера является промежуток времени с момента опустошения глобального свободного пула IPv4 до момента, когда потребность в дополнительных адресах IPv4 начнет уменьшаться. Эта фаза отмечена на графике розовым цветом.

Надо заметить, что высота порога, образуемого красной кривой, для разных провайдеров отличается. Также различен момент завершения свободных адресов в собственном пуле провайдера (вторая вертикальная красная линия). Другими словами, умеренно растущий провайдер с достаточным запасом свободных адресов имеет шансы «перезимовать» переходный период без особых ухищрений. Важно отметить, что и в этом случае необходимой является полноценная поддержка IPv6 в инфраструктуре провайдера и неуклонное массовое распространение IPv6 в глобальном Интернете.

Однако многим сервис-провайдерам придется столкнуться с проблемой нехватки IPv4 и задуматься над ее решением.

Существует два способа решения этой проблемы. Первый — это получение дополнительных адресов IPv4. Однако в соответствии с текущей политикой распределения оставшегося адресного пространства IPv4 (IPv4 Address Allocation and Assignment Policies for the RIPE NCC Service Region, секция 5.1, http://www.ripe.net/ripe/docs/ripe-606) максимум, на что может рассчитывать провайдер, — это одноразовый блок размером /22. Известно, что уже некоторое время происходит перераспределение адресов между игроками путем купли-продажи, дарения, объединения и поглощения компаний и т.п. Трудно сказать, как будет развиваться этот сценарий и насколько объемным и ликвидным окажется рынок адресного пространства. В любом случае второй способ — повышение эффективности использования адресного пространства с помощью технологии NAT (Network Address Translation) — является более реальной альтернативой или дополнительным решением. Этот сценарий показан на графике кривой розового цвета.

Поскольку мы заговорили о технологии NAT, пожалуй, стоит остановиться на ней поподробнее. Ведь эта технология является ключевой в моделях сосуществования IPv4 и IPv6.

Техническое отступление: как происходит передача данных в Интернете

Прежде чем перейти непосредственно к разговору о будущем Интернета и перспективах IPv6, давайте совершим краткий экскурс в техническую область и в общих чертах рассмотрим, как же происходит передача данных в Интернете и какую роль играют адреса.

Работа Интернета основана на технологии пакетной коммутации без установления соединения. Структура пакета определена протоколом IP, при этом каждый пакет содержит IP-адрес отправителя и получателя. В задачу каждого узла сети (также называемого маршрутизатором) входит передача пакета, полученного от соседнего узла, к последующему узлу. Выбор каждого следующего узла происходит с помощью системы маршрутизации. Благодаря этой системе маршрутизатор знает, какому из своих соседей следует передать пакет с конкретным IP-адресом получателя.

Однако, с точки зрения пользователя, передача данных происходит между его приложением и приложением получателя. Например, между веб-браузером и веб-сайтом. Поэтому можно представить, что существует виртуальное соединение между этими приложениями: по нему и происходит передача данных. Помимо IP-адреса

отправителя (в данном случае — компьютера пользователя) и IP-адреса получателя (веб-сервера) это соединение характеризуется дополнительными параметрами — так называемыми портами получателя и отправителя. Их можно рассматривать как локальные идентификаторы конкретных приложений на компьютере. Наконец, транспортный протокол (например, TCP или UDP) является пятым параметром, однозначно определяющим поток данных в Интернете в пределах ограниченного времени.

Таким образом, отправитель и получатель данных в действительности каждый адресуются парой {IP-адрес, порт}. Именно эта особенность используется в технологии NAT (Network Address Translation), или более точно — NAPT (Network Address & Port Translation). С помощью одного IP-адреса можно теоретически адресовать 65535 «соединений» — число, значительно превышающее потребности единичного пользователя. В этом случае устройство NAT для внешней сети будет выглядеть как компьютер с очень большим числом одновременно работающих приложений. Хотя на самом деле устройство NAT при передаче пакетов подставляет вместо порта и собственного IP-адреса (как адреса получателя с точки зрения внешних приложений) порт и локальный IP-адрес реального получателя. Обычно для адресации конечных устройств локальной сети, расположенной за устройством NAT, используется специальное зарезервированное адресное пространство. Схема работы NAT показана на рис. 5.

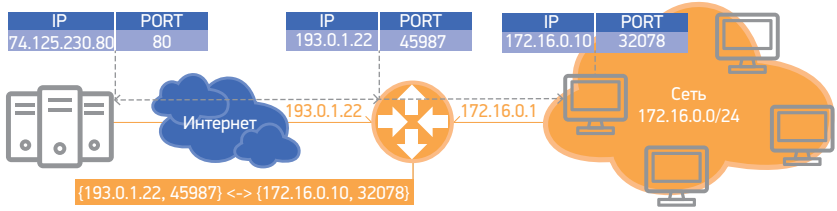

Рис. 5. Схема работы NAT

Насколько эффективен NAT? Это зависит от характера приложений, работающих на конечных устройствах, и интенсивности их взаимодействия с глобальным Интернетом. На сегодня компьютер обычного пользователя во время работы в сети создает от 60 до 100 соединений с различными ресурсами глобального Интернета. Цифра может показаться большой, но ведь многие приложения открывают более одного соединения — так функционирует большинство веб-приложений. Например, Google Maps одновременно использует несколько десятков соединений. Но даже если эта цифра на поря-

док крупнее, трудностей не возникает: технология NAT позволяет совместно использовать один и тот же IP-адрес более чем 60 пользователям.

Звучит очень привлекательно — но, к сожалению, в реальности все не так радужно. Технология NAT содержит ряд серьезных недостатков, о которых мы поговорим позже. Здесь же отметим, что NAT нарушает принцип «прозрачности» соединений между любыми конечными устройствами в Интернете. Помимо усложнения архитектуры сети, для полноценной работы некоторых приложений требуются дополнительные средства, такие как STUN (RFC 5389, https://datatracker.ietf.org/doc/rfc5389/), ICE (RFC 5245, https://datatracker.ietf.org/doc/rfc5245/), TURN (http://ru.wikipedia.org/wiki/Traversal_Using_Relay_NAT, RFC 5766, https://datatracker.ietf.org/doc/rfc5766/). Использование каскадов NAT, когда в сети за устройством NAT расположены еще и NAT со «вложенными» сетями, только усугубляет эти проблемы.

Переходные технологии сосуществования

Итак, технология NAT-мультиплексирования — еще один метод решения проблемы сосуществования двух протоколов, позволяющий бороться с острой нехваткой адресов IPv4. Однако по-прежнему одним из основных препятствий перехода к IPv6 является его несовместимость со своим предшественником — протоколом IPv4. Устройство, поддерживающее только IPv6, не может непосредственно обмениваться данными с устройством IPv4. Виной этому является, скорее, протокол IPv4, который был разработан для адресации нескольких десятков, может быть — сотен или тысяч устройств Сети, и не предусматривал способа расширения.

Переходный план «двойного стека» предполагал отсутствие устройств, «говорящих» только на одном из протоколов, другими словами — глобальное двуязычие. Но этому плану было не суждено воплотиться в жизнь. Вот почему для обмена данными между устройствами и сетями разных протоколов необходимо применение дополнительных технологий — так же как мы прибегаем к услугам переводчика для преодоления языкового барьера.

Давайте посмотрим, что же имеется в арсенале сервис-провайдеров.

Технологии туннелирования

Технологии туннелирования приходят на помощь, когда инфраструктура сервис-провайдера не поддерживает один из протоколов.

6to4

Технология 6to4 была стандартизована еще в 2001 г. (RFC 3056, https://datatracker.ietf.org/doc/rfc3056/) и с тех пор является наиболее распространенным методом для соединения изолированных островков IPv6 с другими такими же островами, а также с глобальным Интернетом IPv6 через океан IPv4. Для этого используются автоматически создаваемые туннели.

Шлюз, или маршрутизатор 6to4, обеспечивает создание динамических туннелей путем инкапсуляции пакетов IPv6 в IPv4 для передачи через Интернет IPv4 к другому острову. Для определения принимающего конца туннеля шлюз извлекает из IPv6-адреса получателя адрес IPv4, который является адресом принимающего шлюза 6to4.

Особенность этого метода в том, что все островки 6to4 совместно используют адресное пространство, определяемое префиксом 2002::/16. Адресное пространство каждого островка определяется путем «присоединения» 32 бит IPv4-адреса шлюза 6to4 к 16 битам префикса 2002. Например, если IPv4-адрес шлюза 193.0.7.5, то адресное пространство сети 6to4 определено префиксом 2002: c100:705::/48 (адрес IPv4 записывается в шестнадцатеричном виде).

Для обеспечения связности с глобальным Интернетом IPv6 используются так называемые релеи 6to4. Это также шлюзы 6to4, однако они являются интерфейсом между сетями 6to4 и остальным Интернетом IPv6. Чтобы не приходилось задавать адреса релеев вручную, все они имеют один и тот же адрес — 192.88.99.1, который анонсируется с использованием технологии аникаст (anycast, http://ru.wikipedia.org/wiki/Anycast).

Схема работы системы 6to4 представлена на рис. 6.

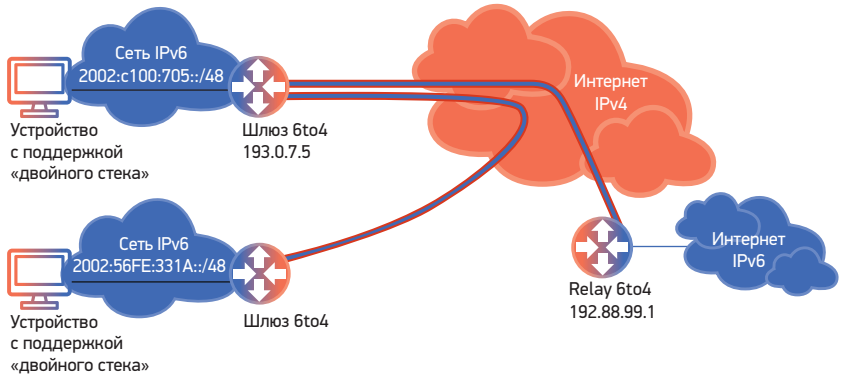

Рис. 6. Схема работы технологии 6to4

Данная технология является наиболее популярной, она намного опережает подобные туннельные технологии — такие как ISATAP и Teredo, — но ее применимость в основном ограничена пользователями-энтузиастами и небольшими корпоративными сетями. Согласно измерениям Google (https://www.google.com/intl/en/ipv6/statistics.html), начиная с середины 2009 г. использование технологий 6to4 и Teredo идет на убыль и составляет лишь 0,5% от общего объема IPv6-трафика. Для серьезных игроков больший интерес представляют технологии 6rd и DS-lite.

6rd

Одним из недостатков 6to4 является отсутствие контроля над релеем, обеспечивающим выход в глобальный IPv6. По этой причине невозможно гарантировать какие-либо параметры качества и связность, не говоря уже о том, что и выбор конкретного релея происходит автоматически, с использованием технологии аникаст.

На помощь здесь приходит технология 6rd (RFC 5969, https://datatracker.ietf.org/doc/rfc5969/). Она делает доступным Интернет IPv6 пользователям провайдера широкополосного доступа, не требуя при этом поддержки IPv6 в сети самого провайдера.

Во-первых, сеть 6rd использует собственное адресное пространство IPv6, полученное от региональной интернет-регистратуры. Это позволяет сервис-провайдеру анонсировать реальные IPv6-префиксы и, таким образом, более точно определять собственную политику маршрутизации.

Во-вторых, вся зона функционирования 6rd ограничена сетью сервис-провайдера. Используя терминологию 6to4, шлюзы 6rd встроены в оконечное оборудование пользователя, а релеи также являются частью инфраструктуры сервис-провайдера.

Эти отличия показаны на рис. 7.

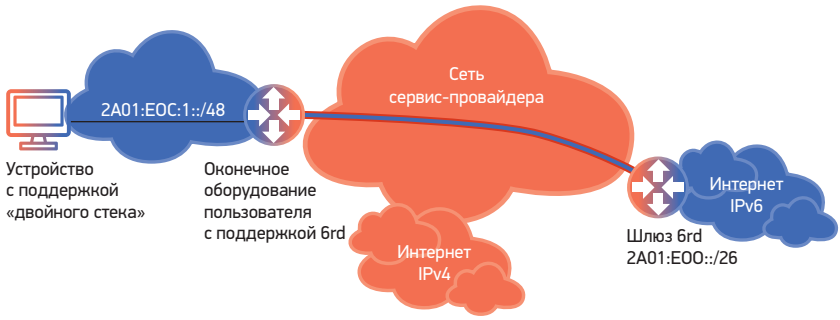

Рис. 7. Схема работы технологии 6rd

DS-lite

DS-lite (https://datatracker.ietf.org/doc/rfc6333/) в некотором смысле является зеркальной технологией по отношению к 6rd. DS-lite предполагает, что сеть провайдера полностью поддерживает IPv6, а туннели используются для передачи трафика IPv4 от сети пользователя к устройствам NAT сервис-провайдера. Также подразумевается, что устройства сети пользователя поддерживают «двойной стек», а именно оба протокола IPv4 и IPv6.

Суть метода заключается в одновременном применении технологий туннелирования (инкапсуляция трафика IPv4 в пакеты IPv6) и централизованного NAT, или CGN (Carrier Grade NAT, также называемого LSN, Large Scale NAT). Благодаря этому ограниченный пул адресов IPv4 совместно используется всеми пользователями сервис-провайдера. Обмен трафиком с ресурсами Интернет IPv4 происходит с использованием протокола IPv4, а с ресурсами IPv6 — с использованием IPv6. Эта схема не предусматривает трансляции протоколов.

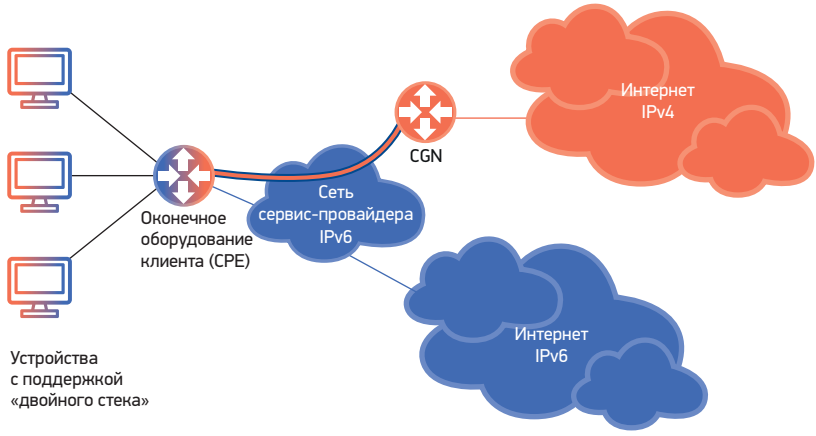

Рис. 8. Схема работы DS-lite

На рис. 8 представлена схема работы DS-lite. Как можно заметить, обмен трафиком с ресурсами IPv6 происходит непосредственно, без использования каких-либо промежуточных технологий, например туннелей.

В отношении IPv4 ситуация гораздо сложнее. Нехватка адресного пространства IPv4 — это уже сегодняшняя реальность. Поэтому схемы, предусматривающие назначение каждому абоненту публичного адреса IPv4, используемого устройством NAT пользователя для построения домашней локальной сети, имеют все более ограниченное применение.

Возможным решением этой проблемы (кстати, уже применяемым некоторыми сервис-провайдерами) является создание еще одного уровня NAT в сети сервис-провайдера. Такая схема работает в общем случае, но результат ее применения — существенные ограничения для многих сегодняшних и будущих приложений, а также сложность обслуживания.

Задача DS-lite — исключить каскадирование устройств NAT, когда все устройства пользователей непосредственно взаимодействуют с центральным устройством NAT сервис-провайдера. В этом случае оконечное устройство пользователя не выполняет никаких функций NAT, а вместо этого обеспечивает создание туннелей к центральному NAT для каждого нового соединения между приложениями пользователя и сервисами Интернета.

Таким образом, все пользовательские соединения, так же как и в схеме каскадирования NAT, отображаются центральным CGN. Однако значительно повышается прозрачность архитектуры, растет эффективность использования адресного пространства IPv4.

Кстати, о прозрачности. Одна из основных проблем, связанных с применением NAT, — это контроль приложений за значениями порта и IP-адреса соединений, поскольку устройство NAT заменяет их на динамически присваиваемые. От этого зависит нормальное функционирование некоторых приложений, например большинства мультимедийных интерактивных программ. На сегодняшний день разработано несколько механизмов решения этой проблемы — такие как STUN, ICE и TURN. Но очевидно, что каскадирование устройств NAT усложняет ситуацию.

Отметим, что технология DS-lite не предусматривает поддержку устройств, работающих только по протоколу IPv6. Для этого используются технологии трансляции.

Технология трансляции: NAT64 + DNS64

Логично предположить, что в недалеком будущем появятся устройства, поддерживающие только IPv6. Если мы говорим о масштабных мобильных, сенсорных или RFID-сетях, необходимость поддержки двух протоколов усложнит и удорожит такие устройства.

Для взаимодействия таких сетей с Интернетом IPv4 необходимо применение трансляции адресов IPv6 в адреса IPv4 и обратно. Ввиду недостатка ресурсов IPv4 здесь, как и в случаях, рассмотренных выше, необходимо применение мультиплексирования потоков. По существу, нужно использовать технологию централизованного NAT с внедрением дополнительной функции трансляции протоколов. Этот компонент еще называют NAT64 (https://datatracker.ietf.

org/doc/rfc6146/). Взаимодействие с другими сетями IPv6 происходит прозрачно: эта архитектура показана на рис. 9.

Однако в данной схеме есть одна особенность, а именно необходимость дополнительной поддержки одного из наиболее критических приложений Интернета — системы доменных имен DNS.

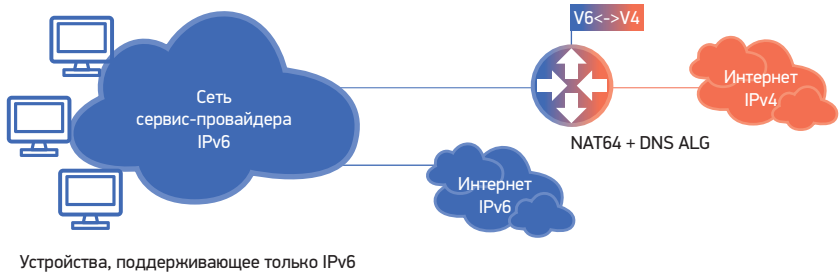

Рис. 9. Архитектура системы трансляции протоколов NAT64

Дело в том, что для большей части ресурсов Интернета запрос DNS вернет адрес IPv4. Поскольку сети, о которых идет речь, поддерживают только IPv6, такой ответ DNS вряд ли окажется полезным. Для решения этой проблемы используется дополнительный компонент — шлюз приложений (Application Layer Gateway, ALG). Суть его заключается в замещении адреса IPv4 в ответе DNS на синтезированный адрес IPv6, который понятен и клиенту, и транслятору протоколов NAT64.

Работа DNS ALG происходит следующим образом. Как обычно, перед началом связи клиент посылает запрос локальному DNS-серверу. В нашем случае его роль выполняет ALG. Он производит разрешение запроса и, допустим, получает IPv4-адрес искомого ресурса. Но в ответ клиенту ALG подставляет синтезированный адрес IPv6. По существу, этот адрес состоит из предустановленного префикса (извест-

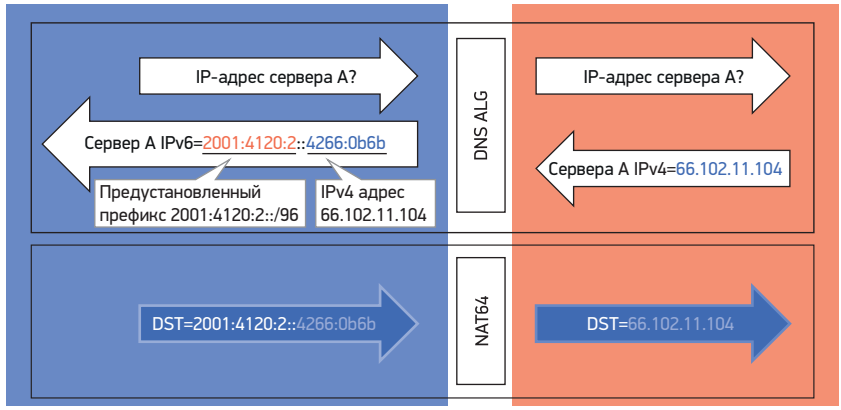

Рис. 10. Схема работы DNS ALG

ного и ALG, и NAT64), а также из IPv4-адреса ресурса. Теперь, когда клиент попытается установить связь с ресурсом, NAT64 поймет, что клиент использует синтезированный адрес, и преобразует его в исходный IPv4-адрес получателя. Схематически это показано на рис. 10.

Вопросы внедрения IPv6 в мобильных сетях

Сегодня разговор об эволюции Интернета немыслим без взгляда на мобильные сети. Здесь мы наблюдаем наиболее стремительный рост как по количеству абонентов, так и по возможностям, которые они открывают для пользователей. Архитектурные решения, принимаемые при разработке или модернизации мобильных сетей, определяют архитектуру будущего Интернета. Поставим вопрос более остро: останется ли Интернет уникальной коммуникационной средой с колоссальным инновационным потенциалом или наше информационное пространство будут определять закрытые и ограниченные платформы мобильных приложений, такие как Apple Store или Google Play?

Развитие мобильных сетей началось с сетей мобильной телефонии, основанных на телефонных стандартах и технологии коммутации каналов. Передача данных была внедрена позже как отдельная подсистема, существенно отличающаяся как по архитектуре, так и по используемым технологиям. Так, для обеспечения услуг на основе пакетной передачи — в первую очередь для доступа к Интернету — в сетях 2G и 3G была разработана система GPRS (General Packet Radio Service). Для возможности предоставления услуг голосовой связи на основе протокола IP в 2002 г. была разработана система IMS (IP Multimedia System). Сегодняшние 3G-сети предоставляют услуги передачи данных и доступа в Интернет в качестве стандартного пакета, однако для осуществления голосовой связи, как правило, по-прежнему используются сети коммутации каналов.

Появление сетей следующего поколения LTE/4G существенно изменило ситуацию. Действительно, эти сети используют исключительно технологию пакетной передачи на основе протокола IP. Для оператора это означает возможность унификации передачи голоса и данных. И хотя для связи с традиционными телефонными сетями необходимы шлюзы, связь между абонентами собственной сети и сетями партнеров, также использующих эти технологии, а также предоставление доступа в Интернет осуществляется унифицированной инфраструктурой на основе пакетной коммутации IP.

В архитектуре LTE опорная сеть, так называемая EPC (Evolved Packet Core), представляет собой нормальную сеть пакетной коммутации на основе IP. Дополнительные устройства и шлюзы необ-