



Николай Борисович Линеv

*Преподаватель подготовительных курсов
факультета вычислительной математики
и кибернетики МГУ им. М.В. Ломоносова.*

Этот таинственный IP-адрес

Каждый мало-мальски грамотный пользователь знает, что для подключения его компьютера к сети нужно, чтобы у компьютера был IP-адрес. Некие четыре числа, разделённые точками. Что же это такое, зачем нужно и что за этими числами прячется? Давайте разбираться.

Во-первых, вспомним, что любая компьютерная сеть нужна для того, чтобы компьютеры в ней могли обмениваться данными.

Во-вторых, для того чтобы компьютеры могли обмениваться данными, они должны иметь одинаковые наборы формальных правил, по которым осуществляется обмен. Эти правила носят название *сетевых протоколов*. Поскольку про-

цесс обмена сложен, то и протоколов компьютеры используют множество. Весь набор протоколов называют стеком протоколов. Наиболее распространённым на сегодня является стек TCP/IP. И если на вашем компьютере настроен IP-адрес, то значит, и ваш компьютер использует этот стек. Наш разговор пойдёт об одном представителе TCP/IP.

Internet Protocol

Среди всего множества протоколов стека есть протокол, название которого вошло в название стека. Это Internet Protocol, или IP. Протокол очень важный, нужный и полезный, особенно с точки зрения пользователей. Он занимается переносом наших с вами данных в сети с компьютера на компьютер.

Для того чтобы компьютеры сети могли обмениваться данными, необходимо, чтобы они были однозначно «узнаваемы». С этой целью в стандарте протокола описана система



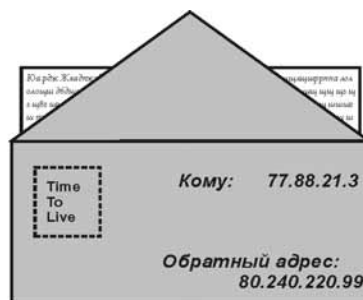
адресации. При отправке данных компьютер с помощью ряда протоколов формирует специальный блок данных, который называется IP-пакетом. Его можно представить как некую капсулу, в которую помещены данные. Ситуация очень похожа на подготовку к отправке обычного письма. Мы берём письмо (наши данные), кладем его в конверт (помещаем в капсулу). Далее требуется написать на конверте адрес получателя и опустить конверт в почтовый ящик. С IP-пакетом про-



исходит то же самое. Капсулу в нём представляют служебные данные, которые добавляются к данным пользователя. Официально они называются заголовком IP-пакета. В заголовок компьютер помещает адреса отправителя (свой собственный адрес) и получателя. Те самые четверки чисел, с которыми нам предстоит разобраться. После этого IP-пакет отправляется в сеть (опускаем письмо в почтовый ящик).

Процесс отправки IP-пакета в сеть непрост и содержит целый ряд операций, выполняемых другими

протоколами. Эти протоколы мы здесь не затрагиваем, как и протоколы, участвующие в подготовке пакета. Нам пока достаточно знать, что при отправке данных компьютер формирует и отправляет в сеть IP-пакет, который содержит адреса отправителя и получателя.



Компьютер-получатель принимает IP-пакет, с помощью такого же, как при отправке, ряда протоколов обрабатывает данные и представляет их в понятном и удобном для нас виде. Например, мы можем увидеть фотографию, которую нам прислали наши друзья, страницу сайта, письмо электронной почты и многое другое.

Именно адрес делает компьютер однозначно определяемым в сети. А поскольку правила адресации описаны в протоколе IP, то и адрес стали называть IP-адресом.

Ещё одно замечание. Существует несколько версий протокола IP, и сегодня мы, как правило, используем четвертую версию (IPv4). Сейчас в мире идёт постепенный переход на версию шесть (IPv6), который продиктован тем, что в ряде регионов мира адреса IPv4 закончились. Россию переход пока коснулся мало, и поэтому говорить мы будем о IPv4. Более ранние версии протокола уже не используются.

Части IP-адреса

IP-адрес есть некое двоичное значение, под которое отводится 32 бита памяти. Для удобства пред-

ставления адрес записывается как четыре десятичных числа, отделённых друг от друга точками. Каждое

десятичное число – это значение очередных восьми битов IP-адреса, или, по-другому, октета и может принимать значения от 0 до 255. И хотя такое представление более удобно для чтения, оно «спрятало» принципы адресации и доставки данных в сетях ТСП/IP, о которых мы и поговорим.

Любой IP-адрес состоит из двух частей. Первая часть (левая цепочка битов) задаёт адрес сети, в которой находится компьютер или сетевое устройство. Вторая часть (оставшаяся правая цепочка битов) определяет устройство внутри сети. Более коротко, все возможные устройства в сети называют хостами. Формально *хост* – компьютер в сети, сетевое устройство или некоторое периферийное устройство (например, принтер) с уникальным IP-адресом.

Такое разделение IP-адреса легло в основу механизма доставки данных, который называется *маршрутизацией*. По принципам работы протокола IP два компьютера могут спокойно общаться между собой, если они принадлежат одной сети (или IP-сети), то есть если у них совпадают первые части адреса. Если отправитель и получатель находятся в одной IP-сети, то для определения компьютера-получателя и доставки ему данных используется вторая часть адреса. Если же компьютер-получатель находится в другой IP-сети, то пакет доставляется с помощью специальных сетевых устройств – маршрутизаторов. Процесс маршрутизации и сами эти

устройства нужны, потому что IP «не знает», как доставлять данные. Он относится к маршрутизируемым (routed) протоколам. Протокол определяет точку назначения (адрес получателя) в своей системе адресации, но ни словом не упоминает о том, как туда доставить данные. «Я вам сказал, кто должен это получить, а дальше не моя забота!» – говорит IP. Вот одна из причин, из-за чего IP является протоколом, не гарантирующим доставку данных. Определением способа доставки данных получателю занимаются другие протоколы, которые называются маршрутизирующими (routing) протоколами, или протоколами маршрутизации. Другими словами, маршрутизируемые протоколы формируют пакет и определяют, кому его надо доставить, протоколы маршрутизации находят, как доставить пакет получателю, а реализуют доставку маршрутизаторы.



Адрес сети

Возникает вопрос: по какому принципу выделяются части IP-адреса и, как следствие, как определить адрес сети? Для этой цели были разработаны две схемы адресации. Согласно первой, для адресации устройств сети были введены

три класса адресов – А, В и С (всего в IPv4 пять классов адресов). Адреса классов предлагались для использования в больших, средних и малых сетях соответственно, причём в каждом классе чётко определялись части адреса. В классе А

больших сетей за адрес сети отвечал первый октет (первые восемь битов) адреса, три оставшихся октета отвечали за адрес устройства. В классе В средних сетей за адрес сети отвечали два первых октета, в классе С малых сетей – первые три. Такая схема адресации называется классовой. Здесь всё просто. Граница между адресом сети и адресом хоста проходит по границам октетов и чётко определена. Но схема классовой адресации имеет серьёзный недостаток – она очень неэффективно расходует IP-адреса. Простой пример. Организация имеет 300 хостов. Сети класса С ей мало, в ней может быть 254 хоста максимум, значит нужно брать сеть класса В. В такой сети может быть до 65534 хостов (почему эти значения не совпадают со степенями двойки, как казалось бы должны, – чуть позже). Из них организация берёт для себя 300. Но остальные 65234 адреса никто использовать уже не может. Точнее, может. Но тогда две организации будут иметь одну и ту же IP-сеть и все данные, приходящие каждой из организаций, будут сваливаться в одну кучу, что никого не устроит. А разделить на два диапазона одну сеть возможности нет.

В качестве решения возникшей проблемы была предложена схема бесклассовой адресации, которая применяется на адресах классов А, В и С. В ней граница между адресом сети и адресом хоста может проходить в любом месте IP-адреса, а не только по границам октетов. Это изменение позволило серьёзно сэкономить IP-адреса. Формальным механизмом, позволяющим разделить части адреса стала *сетевая маска* (Subnet Mask), которая обязательно сопровождает IP-адрес.

Сетевая маска по своей структуре совпадает с IP-адресом. Это такое же 32-битное двоичное значе-

ние, но в нём запрещается чередование 0 и 1. Биты левой цепочки имеют значение 1, следующие за этой цепочкой биты – значение 0. Так же как и IP-адрес, маска может записываться в виде четырёх десятичных чисел, разделённых точками или, в сокращённом виде, указанием количества единичных битов через символ «/» после IP-адреса. Например, 10.0.0.12 255.0.0.0, или, что то же самое, 10.0.0.12/8.

Операция выделения частей IP-адреса основана на логическом побитовом умножении IP-адреса на маску. Правила перемножения битов такие: $1 \times 1 = 1$, $1 \times 0 = 0$, $0 \times 1 = 0$ и $0 \times 0 = 0$. Отсюда следует, что при умножении IP-адреса на маску левые биты IP-адреса, которым соответствуют единичные биты маски, сохраняют свои значения. Действительно, как хорошо известно из арифметики, любое число при умножении его на единицу не изменится. А оставшиеся биты умножаются на 0 и поэтому в результате получают значение 0. Биты IP-адреса, соответствующие нулевым битам маски, называют хост-битами. Ещё один простой вывод: какие бы значения ни принимали хост-биты IP-адреса, при умножении их на маску произведение будет одинаковым. Результат произведения и есть адрес сети. Из-за того, что обработка проводится в двоичном виде, в бесклассовой адресации по десятичному представлению IP-адреса его части увидеть непросто, но если перевести в двоичный вид, всё сразу становится понятно. Рассмотрим такой пример: пусть есть IP-адрес 143.18.26.100 и этот адрес назначен устройству в некоторой сети с маской 255.255.252.0. Нам интересно, как в этом адресе распределены части. Чтобы ответить на поставленный вопрос, переведём данный нам IP-адрес и маску в двоичный вид. Двоичный вид IP-адреса

будет (для удобства чтения сохраним точки, разделяющие октеты):

10001111.00010010.00011010.01100100

Двоичный вид маски:

11111111.11111111.11111000.00000000

Выпишем их друг под другом и проведём операцию логического умножения:

10001111.00010010.00011010.01100100

11111111.11111111.11111000.00000000

10001111.00010010.00011000.00000000

В десятичном представлении результат произведения запишется 143.18.24.0. Значит, наш адрес принадлежит IP-сети 143.18.24.0 с маской 255.255.252.0, или 22 бита.

Внимательный читатель наверняка заметил, что переводить в двоичный вид все октеты IP-адреса и сетевой маски не обязательно. Достаточно посмотреть, в каком октете маски заканчиваются единичные биты и начинаются нулевые, и перевести этот октет маски и соответствующий ему по номеру октет IP-адреса. Октеты IP-адреса, расположенные левее, если они есть, при перемножении сохраняют свои значения, а октеты, расположенные правее, получают значение 0. В нашем примере достаточно ограничиться переводом третьего октета.

Зная сетевую маску, можно легко посчитать, сколько в ней нулевых битов. Из этого значения определяется диапазон IP-адресов сети и количество адресуемых устройств. Следует помнить, что количество адресов в сети всегда на два больше возможного количества хостов. Количество адресов определяется числом 2^n , где n – количество нулевых битов маски. Тогда возможное количество хостов $2^n - 2$. Разница объясняется тем, что два адреса диапазона, а именно первый и последний, запрещено назначать устройствам сети. Первый из этих адресов, как мы уже выяснили, является адресом сети, и в этом адресе все хост-биты имеют значение 0. Последний адрес

диапазона – адрес широковещательной рассылки (broadcast), в нём все хост-биты имеют значение 1. Этот вид рассылки используется в основном для служебных целей и означает, что такую отправку получат все сетевые устройства данной IP-сети. То есть один компьютер отправляет в сеть один пакет данных, а принимают этот пакет и его обрабатывают все компьютеры и сетевые устройства.

Все вышесказанное справедливо как для бесклассовой, так и для классовой систем адресации. В классовой адресации адреса сопровождаются умолчанными масками 8, 16 и 24 бита для сетей классов А, В и С со-



ответственно. Возможно, кто-то из читателей настраивал на своём компьютере IP-адрес и обратил внимание, что после ввода значения адреса и переключения на поле ввода маски там появляется какое-то значение. Значение берётся исходя из класса, к которому принадлежит введённый IP-адрес. А принадлежность классу, в свою очередь, определяется по первым битам первого октета адреса. Для адресов класса А первый бит всегда равен 0, для класса В первые два бита 10 и для

класса С – первые биты 110. А вот теперь должно стать понятным, почему каждый класс имеет свой диапазон изменения значений первого октета в десятичном виде. А именно:

Класс А – 1 ... 126
Класс В – 128 ... 191
Класс С – 192 ... 223.

Следует обратить внимание на то, что пропущено значение 127. Эта сеть, формально относящаяся к классу А, особая. Любой адрес этого диапазона по стандарту IPv4 является адресом обратной петли (*testing loopback*) и позволяет устройству протестировать самого себя. И, естественно, запрещён к назначению хосту.

Например, у пользователя возникли подозрения, что на его компьютере не работает сетевой адаптер (сетевая карта). Это можно проверить, воспользовавшись утилитой *ping* протокола ICMP (*Internet Con-*

trol Message Protocol), входящего в состав стека TCP/IP. Чаще всего используют адрес 127.0.0.1, но можно и любой другой, начинающийся на 127. И если сетевой адаптер «жив», то на *ping* 127.0.0.1 он ответит независимо от назначенного ему IP-адреса. Вообще утилита *ping* используется для проверки связи между двумя хостами. На некотором хосте, например, компьютере, даётся команда *ping*. Адрес назначения указывается в качестве параметра команды. Хост посылает IP-пакет, в котором содержится так называемый эхо-запрос на заданный адрес. И если связь между хостами есть, второй хост пришлёт IP-пакет с эхо-ответом. Как правило, при проверке связи посылается не один, а несколько эхо-запросов и определяется количество эхо-ответов. Также считается время, прошедшее с момента отправки запроса до прихода ответа.

Глобальные и частные адреса

Введение бесклассовой адресации частично решило проблему дефицита IP-адресов. Следующим шагом в решении этой проблемы стало разделение адресов на *частные* и *глобальные*. В каждом классе выделен диапазон адресов, которые нигде не регистрируются и которые не участвуют в маршрутизации пакетов в сети Интернет. Это частные адреса. В классе А под них отводится одна сеть 10.0.0.0/8. В классе В – 16 сетей с адресами с 172.16.0.0/16 по 172.31.0.0/16 и в классе С – 256 сетей с 192.168.0.0/24 по 192.168.255.0/24.

Как же, спросит читатель, его домашний компьютер, которому назначен частный адрес, связывается тогда с компьютерами в Интернете? Ведь для этого нужен глобальный адрес, который можно получить только у провайдера услуг связи. Иначе при запросе с компьютера с

частным адресом ресурса в Интернете, например портала MAIL.RU, IP-пакет с запросом до портала дойдёт, а вот ответ по частному адресу не вернётся никогда. Значит, нужны глобальные адреса.

Далее. В любой сети выход в Интернет хотят иметь все пользователи, следовательно, и глобальных адресов нужно много. Много адресов провайдер не даст, он может просто не иметь такого количества, да и стоит помнить, что за аренду каждого адреса нужно платить. Если в локальных сетях районов городов провайдер предоставляет услугу аренды глобального адреса пользователям, то каждый пользователь сам решает, будет ли он арендовать и оплачивать глобальный адрес самостоятельно или нет. А для организации с большим количеством сотрудников это может оказаться неоправданно дорого.

Для решения этой проблемы реализуется трансляция адресов. В зависимости от количества сотрудников организации, задач, которые должны решаться сотрудниками, и ряда других факторов у провайдера арендуется диапазон глобальных адресов. В худшем случае провайдер может выдать организации всего один адрес, но и этого достаточно, чтобы сотрудники могли выходить в Интернет.

Мы уже говорили, что в заголовке IP-пакета содержатся адреса отправителя и получателя. Трансляция заключается в том, что при выходе в Интернет в заголовке IP-пакета подменяется частный адрес компьютера (адрес отправителя) на

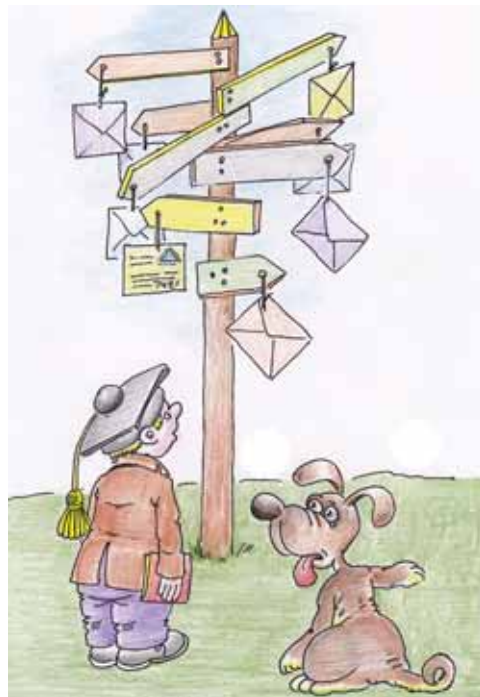
глобальный адрес, выданный провайдером. При ответе также формируется IP-пакет, в котором, по существу, адреса отправителя и получателя переставлены местами. То есть ответ отправляется также на глобальный адрес. При входе в локальную сеть выполняется обратная подмена – но на этот раз глобальный адрес получателя меняется обратно на частный. И хотя одной такой замены чаще всего недостаточно, пока на этом остановимся. Естественно, должно быть что-то, реализующее трансляцию. Подробный разговор о том, как реализуется трансляция, займёт много времени, и поэтому мы его опустим.

Немного о маршрутизации

Давайте попробуем проследить движение IP-пакета по сети. Перед отправкой IP-пакета в сеть компьютер проверяет принадлежность адреса получателя его IP-сети. Делается это с помощью той же операции умножения IP-адреса на маску. На маску, указанную в настройках сетевого адаптера, перемножаются два адреса: адрес получателя и свой собственный. Если результаты умножения совпали, получатель находится в той же сети и ему пересылаются данные. Если результаты произведения разные, пакет будет отправлен на адрес граничного маршрутизатора сети. Этот адрес должен быть задан в настройках сетевого адаптера и имеет названия «default gateway», «основной шлюз», или «шлюз по умолчанию».

Граничный маршрутизатор сети имеет соединения с другими маршрутизаторами, и к ним также могут быть подключены локальные сети. Каждый из маршрутизаторов хранит в памяти специальную таблицу, которая называется таблицей маршрутизации. Строки таблицы содержат ад-

реса известных маршрутизатору IP-сетей и их маски. В идеале (и надо сказать, достижимом) в таблице маршрутизации должна быть информация обо всех существующих сетях.



Коротко о том, как в таблице маршрутизации появляются записи. Адреса непосредственно подключённых сетей маршрутизатор определяет и заносит в свою таблицу автоматически. Удалённые сети в таблицу может внести администратор сети или они могут появиться как результат работы протоколов маршрутизации.

Итак, маршрутизатор получил IP-пакет, считал из заголовка принятого IP-пакета адрес назначения и пытается определить, куда дальше его следует переслать. В основе принятия решения о направлении пересылки лежит всё та же операция умножения IP-адреса на маску. Маршрутизатор начинает просматривать записи таблицы маршрутизации. Он берёт строку таблицы и перемножает адрес из заголовка IP-пакета на маску, записанную в этой строке. Адрес сети, полученный в результате произведения, сравнивается с адресом, записанным в этой же строке таблицы маршрутизации. Если адреса сетей не совпали, берётся другая строка таблицы и операция повторяется с маской из новой строки. Если совпадение нашлось, то из этой строки считывается информация, в какую сеть нужно дальше переслать пакет. Другими словами, маршрутизатор пытается найти в своей таблице информацию о том, где находится IP-сеть назначения и опирается в поиске на первую часть адреса, на адрес сети. Вариантов местонахождения сети на-

значения всего два: либо сеть назначения подключена непосредственно к маршрутизатору и он передаст пакет в эту сеть, либо сеть непосредственно к нему не подключена. Тогда маршрутизатор возьмёт из строки таблицы адрес соседнего маршрутизатора и передаст пакет ему. Следует отметить, что для пересылки пакета соседнему маршрутизатору пакет тоже отправляется в непосредственно подключённую сеть, соединяющую эти два маршрутизатора. Процесс передачи соседнему устройству продолжается до тех пор, пока пакет не получит маршрутизатор, к которому подключена сеть назначения. И только после того как пакет попал в сеть назначения, вступает в работу вторая часть IP-адреса.

А если после просмотра таблицы маршрутизации сеть назначения не нашлась, маршрутизатор удалит пакет, адресованный в эту сеть. Это ещё одна причина, по которой IP не гарантирует доставку данных.

Естественно, что о маршрутизации в сетях TCP/IP мы упомянули очень упрощённо. Подробное обсуждение этих вопросов – отдельная и большая тема разговора.

В заключение стоит сказать, что IP относится к протоколам открытого стандарта, то есть описание протокола находится в свободном доступе, и любой желающий может найти о нём полную информацию, изучив документ RFC 791.

Калейдоскоп**Калейдоскоп****Калейдоскоп**

Мудрые мысли

Невежество – ночь ума, ночь безлунная и беззвёздная.

Марк Туллий Цицерон

То, что мы знаем, ограничено, а то, чего не знаем, бесконечно.

Пьер Симон Лаплас