Протокол межсетевого взаимодействия

Кафедра ИБ БГАРФ

Зензин Александр Степанович, к.т.н.

Обзор

1. Формат IP-пакета
2. Схема IP-маршрутизации
   - 2.1. Упрощенная таблица маршрутизации
   - 2.2. Таблицы маршрутизации конечных узлов
   - 2.3. Просмотр таблиц маршрутизации без масок
   - 2.4. Примеры таблиц маршрутизации разных форматов
   - 2.5. Источники и типы записей в таблице маршрутизации
   - 2.6. Пример IP-=маршрутизации без масок
3. Маршрутизация с использованием масок
   - 3.1. Структуризация сети масками одинаковой длины
   - 3.2. Просмотр таблиц маршрутизации с учетом масок
   - 3.3. Использование масок переменной длины
   - 3.4. Перекрытие адресных пространств
   - 3.5. CIDR
4. Фрагментация IP-пакетов
   - 4.1. Параметры фрагментации
   - 4.2. Механизм фрагментации

Протокол межсетевого взаимодействия

Протокол IP (Intrenet Protocol — межсетевой протокол), описан в документе RFC 751. В каждой очередной сети, лежащей на пути перемещения пакета, протокол IP обращается к средствам транспортировки этой сети, чтобы с их помощью передать пакет на маршрутизатор, ведущий к следующей сети, или непосредственно на узел-получатель. Таким образом, одной из важнейших функций IP является поддержание интерфейса с нижележащими технологиями сетей, образующих составную сеть. Кроме того, в функции протокола IP входит поддержание интерфейса с протоколами вышележащего транспортного уровня, в частности с протоколом TCP, который решает все вопросы обеспечения надежной доставки данных по составной сети в стеке TCP/IP.

Протокол IP относится к протоколам без установления соединений, он поддерживает обработку каждого IP-пакета как независимой единицы обмена, не связанной с другими пакетами. В протоколе IP *нет механизмов*, обычно применяемых для *обеспечения достоверности конечных данных*. Если во время продвижения пакета происходит какая-либо ошибка, то протокол IP по своей инициативе ничего не предпринимает для исправления этой ошибки.

Например, если на промежуточном маршрутизаторе пакет был отброшен из-за ошибки по контрольной сумме, то модуль IP не пытается заново послать потерянный пакет. Другими словами, протокол IP реализует **политику доставки «по возможности»** (с максимальными усилиями).

Формат IP-пакета

Имеется прямая связь между количеством полей заголовка пакета и функциональной сложностью протокола, который работает с этим заголовком. Чем проще заголовок — тем проще соответствующий протокол. Большая часть действий протокола связана с обработкой той служебной информации, которая переносится в полях заголовка пакета. Изучая назначение каждого поля заголовка IP-пакета, мы получаем не только формальные знания о структуре пакета, но и знакомимся с основными функциями протокола IP.

IP-пакет состоит из полей заголовка и данных. Структура заголовка IP-пакета.

Поле **номера версии** занимает 4 бита и идентифицирует версию протокола IP. Сейчас повсеместно используется версия 4 (IPv4), хотя все чаще встречается и новая версия (IPv6).

Значение **длины заголовка** IP-пакета также занимает 4 бита и измеряется в 32-битных словах. Обычно заголовок имеет длину в 20 байт (пять 32-битных слов), но при добавлении некоторой служебной информации это значение может быть увеличено за счет дополнительных байтов в поле параметров. Наибольшая длина заголовка составляет 60 байт.

Формат IP-пакета

Поле **типа сервиса** (Type of Service, ToS) имеет и другое, более современное название — **байт дифференцированного обслуживания**, или **DS-байт**. Этим двум названиям соответствуют два варианта интерпретации этого поля. В обоих случаях данное поле служит одной цели — хранению признаков, которые отражают требования к качеству обслуживания пакета. В прежнем варианте первые три бита содержат значение приоритета пакета: от самого низкого — 0 до самого высокого — 7. Маршрутизаторы и компьютеры могут принимать во внимание приоритет пакета и обрабатывать более важные пакеты в первую очередь. Следующие три бита поля ToS определяют **критерий выбора маршрута**. Если бит D (Delay - задержка) установлен в 1, то маршрут должен выбираться для минимизации задержки доставки данного пакета, установленный бит T (Throughput — пропускная способность) - для максимизации пропускной способности, а бит R (Reliability — надежность) — для максимизации надежности доставки. Оставшиеся два бита имеют нулевое значение. Стандарты дифференцированного обслуживания, принятые в конце 90-х годов, дали новое название этому полю и переопределили назначение его битов. В DS-байте также используются только старшие 6 бит, а два младших бита остаются в качестве резерва.

Поле **общей длины** занимает 2 байта и характеризует общую длину пакета с учетом заголовка и поля данных. Максимальная длина пакета ограничена разрядностью поля, определяющего эту величину, и составляет 65 535 байт, однако в большинстве компьютеров и сетей столь большие пакеты не используются. При передаче по сетям различного типа длина пакета выбирается с учетом максимальной длины пакета протокола нижнего уровня, несущего IP-пакеты. Если это кадры Ethernet, то выбираются пакеты с максимальной длиной 1500 байт, умещающиеся в поле данных кадра Ethernet. В стандартах TCP/IP предусматривается, что все хосты должны быть готовы принимать пакеты длиной вплоть до 576 байт (независимо от того, приходят ли они целиком или фрагментами).

Формат IP-пакета

Идентификатор пакета занимает 2 байта и используется для распознавания пакетов, образовавшихся путем деления на части (фрагментации) исходного пакета. Все части (фрагменты) одного пакета должны иметь одинаковое значение этого поля.

Флаги занимают 3 бита и содержат признаки, связанные с фрагментацией. Установленный в 1 бит DF (Do not Fragment — не фрагментировать) запрещает маршрутизатору фрагментировать данный пакет, а установленный в 1 бит MF (More Fragments — больше фрагментов) говорит о том, что данный пакет является промежуточным (не последним) фрагментом. Оставшийся бит зарезервирован.

Поле **смещения фрагмента** занимает 13 бит и задает смещение в байтах поля данных этого фрагмента относительно начала поля данных исходного (нефрагментированного) пакета. Используется при сборке/разборке фрагментов пакетов. Смещение должно быть кратно 8 байт.

Поле **времени жизни** (Time To Live, TTL) занимает один байт и используется для задания предельного срока, в течение которого пакет может перемещаться по сети. Время жизни пакета измеряется в секундах и задается источником. По истечении каждой секунды пребывания на каждом из маршрутизаторов, через которые проходит пакет во время своего «путешествия» по сети, из его текущего времени жизни вычитается единица; единица вычитается и в том случае, если время пребывания было меньше секунды. Поскольку современные маршрутизаторы редко обрабатывают пакет дольше, чем за одну секунду, то время жизни можно интерпретировать как максимальное число транзитных узлов, которые разрешено пройти пакету. Если значение поля времени жизни становится нулевым до того, как пакет достигает получателя, пакет уничтожается. Таким образом, время жизни является своего рода часовым механизмом самоуничтожения пакета.

Формат IP-пакета

Поле **протокола верхнего уровня** занимает один байт и содержит идентификатор, указывающий, какому протоколу верхнего уровня принадлежит информация, размещенная в поле данных пакета. Значения идентификаторов для разных протоколов приводятся в документе RFC 1700, доступном по адресу **http://www.iana.org**. Например, 6 означает, что в пакете находится сообщение протокола TCP, 17 — протокола UDP, 1 — протокола ICMP.

Контрольная сумма заголовка занимает 2 байта (16 бит) и рассчитывается только по заголовку. Поскольку некоторые поля заголовка меняют свое значение в процессе передачи пакета по сети (например, поле времени жизни), контрольная сумма проверяется и повторно рассчитывается на каждом маршрутизаторе и конечном узле как дополнение к сумме всех 16-битных слов заголовка. При вычислении контрольной суммы значение самого поля контрольной суммы устанавливается в нуль. Если контрольная сумма неверна, то пакет отбрасывается, как только обнаруживается ошибка.

Поля **IP-адресов источника и приемника** имеют одинаковую длину — 32 бита.

Поле **параметров** является необязательным и используется обычно только при отладке сети. Это поле состоит из нескольких подполей одного из восьми предопределенных типов. В этих подполях можно указывать точный маршрут, регистрировать проходимые пакетом маршрутизаторы, помещать данные системы безопасности или временные отметки.

Так как число подполей в поле параметров может быть произвольным, то в конце заголовка должно быть добавлено несколько нулевых байтов для **выравнивания** заголовка пакета по 32-битной границе.

Формат IP-пакета

Пример реальных IP-пакетов, захваченных в сети Ethernet средствами анализатора протоколов сетевого монитора (Network Monitor, NM) компании Microsoft. В данной распечатке NM в скобках дает шестнадцатеричные значения полей, кроме того, программа иногда представляет числовые коды полей в виде, более удобном для чтения. Например, дружественный программный интерфейс NM интерпретирует код 6 в поле протокола, помещая туда название соответствующего протокола — TCP (выделенная строка).

IP: Version = 4 (0x4)
IP: Header Length = 20 (0x14)
IP: Service Type = 0 (0x0)
IP: Precedence = Routine
IP: .. .0____ = Normal Delay
IP: ____0... = Normal Throughput
IP: .....0.. = Normal Reliability
IP: Total Length = 54 (0x36)
IP: Identification = 31746 (0x7C02)
IP: Flags Summary = 2 (0x2)
IP: ....0 = Last fragment in datagram
IP: ......1. = Cannot fragment datagram
IP: Fragment Offset = 0 (0x0) bytes
IP: Time to Live = 128 (0x80)
IP: Protocol - TCP - Transmission Control
IP: Checksum = 0xEB86
IP: Source Address = 194.85.135.75
IP: Destination Address = 194.85.135.66
IP: Data: Number of data bytes remaining = 34 (0x0022).

Схема IP-маршрутизации

Рассмотрим механизм IP-маршрутизации на примере составной сети, представленной на рисунке. В этой сети 20 маршрутизаторов (изображенных в виде пронуме- рованных квадрантных блоков) объединяют 18 сетей в общую сеть; N1, N2,N18 — это номера сетей. На каждом маршрутизаторе и конечных узлах A и B функционируют протоколы IP.

К нескольким интерфейсам (портам) маршрутизаторов присоединяются сети. Каждый интерфейс маршрутизатора можно рассматривать как отдельный узел сети: он имеет сетевой адрес и локальный адрес в той подсети, которая к нему подключена. Например, маршрутизатор под номером 1 имеет три интерфейса, к которым подключены сети N1, N2, N3. На рисунке сетевые адреса этих портов обозначены IP_{11} IP_{12} и IP_{13}. Интерфейс IP_{11} является узлом сети N1, и следовательно, в поле номера сети порта IP_{11} содержится номер N1. Аналогично интерфейс IP_{12} — это узел в сети N2, а порт IP_{13} — узел в сети N3. Таким образом, маршрутизатор можно рассматривать как совокупность нескольких узлов, каждый из которых входит в свою сеть. Как единое устройство маршрутизатор не имеет выделенного адреса, ни сетевого, ни локального.

В сложных составных сетях почти всегда существуют несколько альтернативных маршрутов для передачи пакетов между двумя конечными узлами. Так, пакет, отправленный из узла A в узел B, может пройти через маршрутизаторы 17,12,5,4 и 1 или маршрутизаторы 17,13,7,6 и 3. Нетрудно найти еще несколько маршрутов между узлами A и B.

Задачу выбора маршрута из нескольких возможных решают маршрутизаторы, а также конечные узлы. Маршрут выбирается на основании имеющейся у этих устройств *информации о текущей конфигурации сети*, а также на основании *критерия выбора маршрута* (задержка прохождения маршрута отдельным пакетом, средняя пропускная способность маршрута для последовательности пакетов или количество пройденных на маршруте промежуточных маршрутизаторов (ретрансляционных участков, или хопов)).

Схема IP-маршрутизации.

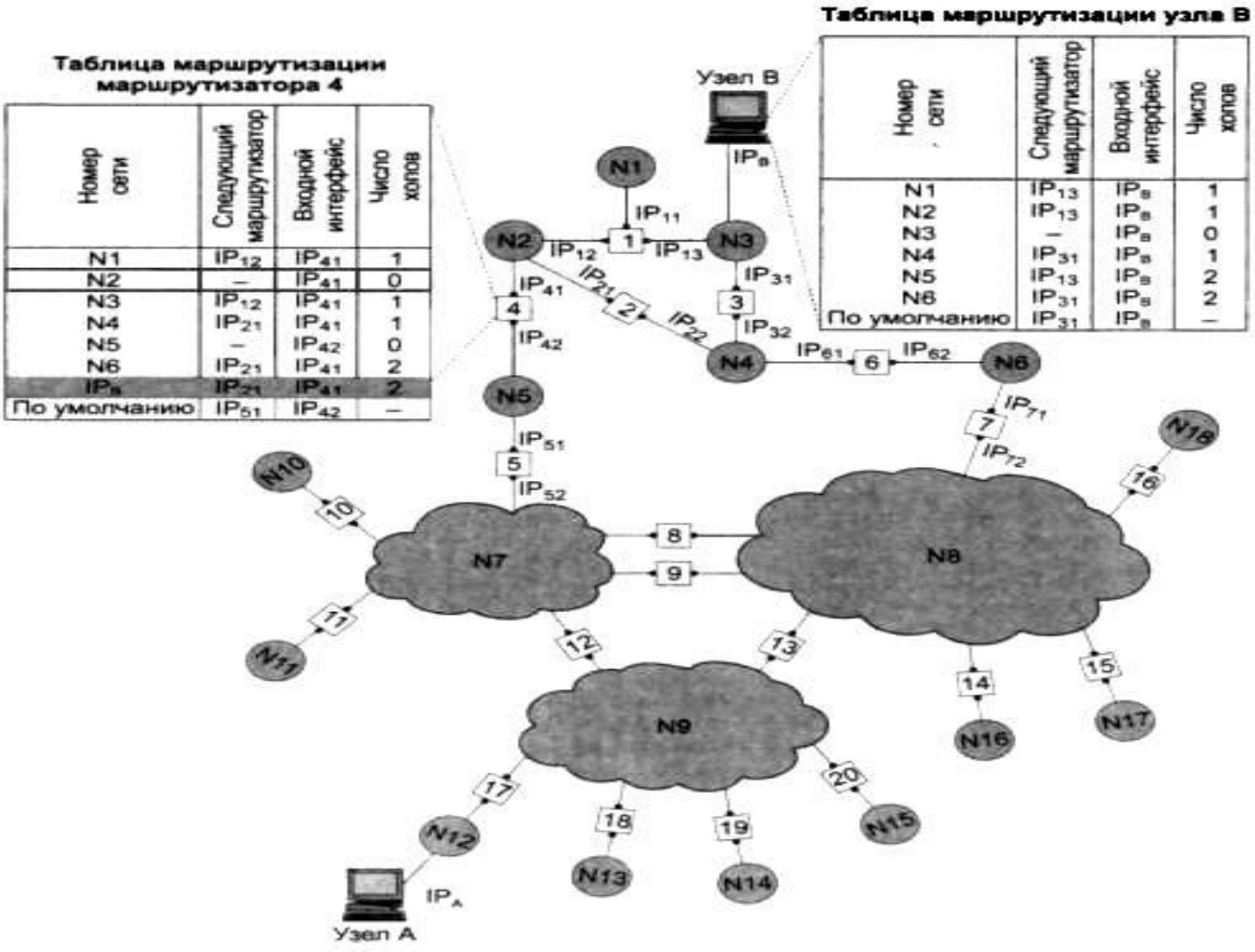
Полученная в результате анализа информация о маршрутах дальнейшего следования пакетов помещается в *таблицу маршрутизации*.

Таблица маршрутизации маршрутизатора 4

Номер сети	Следующий маршрутизатор	Входной интерфейс	Число хопов
N1	IP_{12}	IP_{41}	1
N2	–	IP_{41}	0
N3	IP_{12}	IP_{41}	1
N4	IP_{21}	IP_{41}	1
N5	–	IP_{42}	0
N6	IP_{21}	IP_{41}	2
IP_B	IP_{21}	IP_{41}	2
По умолчанию	IP_{51}	IP_{42}	–

Таблица маршрутизации узла B

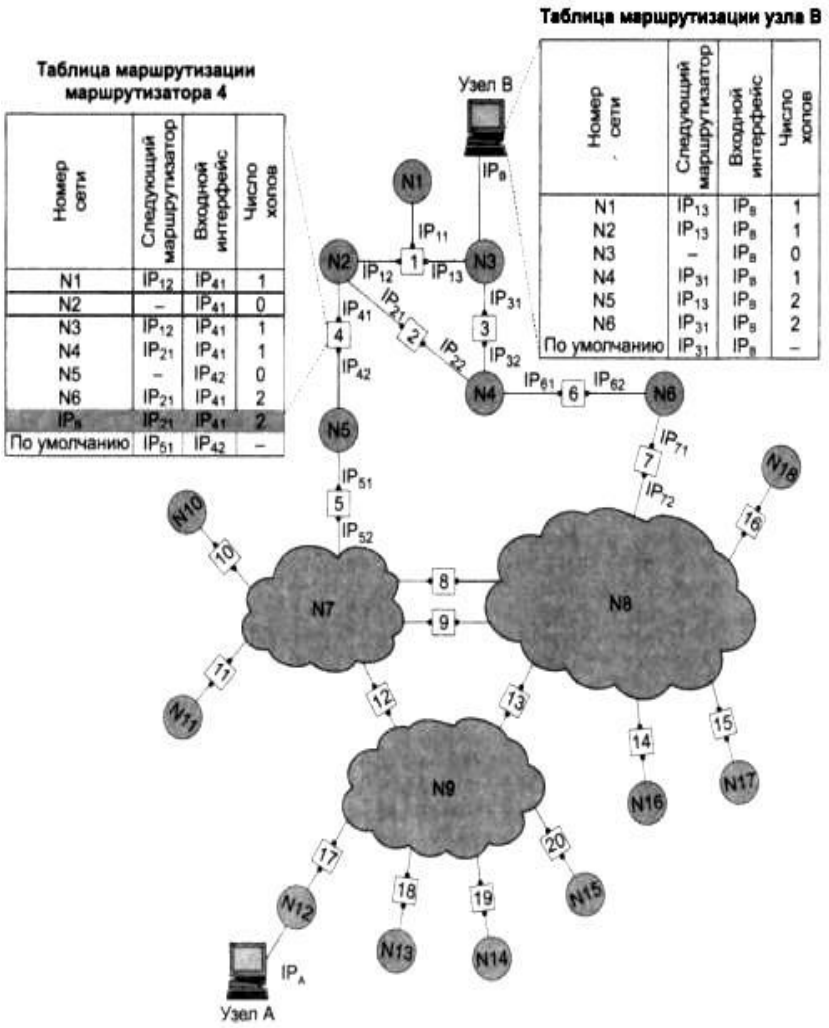
Номер сети	Следующий маршрутизатор	Входной интерфейс	Число хопов
N1	IP_{13}	IP_B	1
N2	IP_{13}	IP_B	1
N3	–	IP_B	0
N4	IP_{31}	IP_B	1
N5	IP_{13}	IP_B	2
N6	IP_{31}	IP_B	2
По умолчанию	IP_{31}	IP_B	–



Упрощенная таблица маршрутизации.

Используя условные обозначения для сетевых адресов маршрутизаторов и номеров сетей, показанные на предыдущем рисунке, посмотрим, как могла бы выглядеть таблица маршрутизации, например, в маршрутизаторе 4.

Адрес назначения	Сетевой адрес следующего маршрутизатора	Сетевой адрес выходного порта	Расстояние до сети назначения
N1	IP_{12}(R1)	IP41	1
N2	-	IP41	0(подсоединена)
N3	IP_{12} (R1)	IP41	1
N4	IP_{21} (R2)	IP41	1
N5	-	IP42	0(подсоединена)
N6	IP_{21} (R2)	IP21	2
IP_B	IP_{21} (R2)	IP41	2
Маршрут по умолчанию	IP_{51} (R5)	IP42	-



Первый столбец таблицы содержит адреса назначения пакетов. В каждой строке таблицы следом за адресом назначения указывается сетевой адрес следующего маршрутизатора (точнее , сетевой адрес интерфейса следующего маршрутизатора), на который надо направить пакет, чтобы тот передвигался по направлению к заданному адресу по рациональному маршруту.

Упрощенная таблица маршрутизации.

Перед тем как передать пакет следующему маршрутизатору, текущий маршрутизатор должен определить, на какой из нескольких собственных портов (IP41 или IP42) он должен поместить данный пакет. Для этого служит третий столбец таблицы маршрутизации, содержащий сетевые адреса выходных интерфейсов.

Некоторые реализации сетевых протоколов допускают наличие в таблице маршрутизации сразу нескольких строк, соответствующих одному и тому же адресу назначения. В этом случае при выборе маршрута принимается во внимание столбец, представляющий расстояние до сети назначения. При этом расстояние измеряется в любой метрике, используемой в соответствии с заданным в сетевом пакете критерием. Расстояние может измеряться временем прохождения пакета по линиям связи, различными характеристиками надежности линий связи на данном маршруте, пропускной способность той или другой величиной, отражающей качество данного маршрута по отношению к заданному критерию. В табл . 1 расстояние между сетями измеряется хопами. Расстояние для сетей, непосредственно подключенных к портам маршрутизатора, здесь принимается равным 0, однако в некоторых реализациях отсчет расстояний начинается с 1.

Когда пакет поступает на маршрутизатор, модуль IP извлекает и его заголовка номер сети назначения и последовательно сравнивает его с номерами сетей из каждой строки таблицы . Строка с совпавшим номером сети показывает ближайший маршрутизатор, на который следует направить пакет. Например, если на какой-либо порт маршрутизатора 4 поступает пакет, адресованный в сеть N6, то из таблицы маршрутизации следует, что адрес следующего маршрутизатора — IP21, то есть очередным этапом движения данного пакета будет движение к порту 1 маршрутизатора 2.

Упрощенная таблица маршрутизации.

Чаще всего в качестве адреса назначения в таблице указывается не весь IP-адрес, а только номер сети назначения. Таким образом, для всех пакетов, направляемых в одну и ту же сеть, протокол IP будет предлагать один и тот же маршрут (мы пока не принимаем во внимание возможные изменения состояния сети, такие как отказы маршрутизаторов или обрывы кабелей). Однако в некоторых случаях возникает необходимость для одного из узлов сети определить специфический маршрут, отличающийся от маршрута, заданного для всех остальных узлов сети. Для этого в таблицу маршрутизации помещают для данного узла отдельную строку, содержащую его полный IP-адрес и соответствующую маршрутную информацию. Такого рода запись имеется в табл. 1 для узла В. Пусть, например, администратор маршрутизатора 4 , руководствуясь соображениям и безопасности, решил, что пакеты, следующие в узел В (полный адрес IPв) , должны идти через маршрутизатор 2 (интерфейс IP21), а не маршрутизатор 1 (интерфейс IP12), через который передаются пакеты всем остальным узлам сети N3. Если в таблице имеются записи о маршрутах как к сети в целом, так и к её отдельному узлу, то при поступлении пакета, адресованного данному узлу, маршрутизатор отдаст предпочтение специфическому маршруту.

Упрощенная таблица маршрутизации.

Поскольку пакет может быть адресован в любую сеть составной сети, может показаться, что каждая таблица маршрутизации должна иметь записи обо всех сетях, входящих в составную сеть. Однако при таком подходе в случае крупной сети объем таблиц маршрутизации может оказаться очень большим, что повлияет на время её просмотра, потребует много места для хранения и т. п. Поэтому на практике широко известен прием уменьшения количества записей в таблице маршрутизации, основанный на введении **маршрута по умолчанию** (default route), учитывающего особенности топологии сети. Рассмотрим, например, маршрутизаторы, находящиеся на периферии составной сети. В их таблицах достаточно записать номера только тех сетей, которые непосредственно подсоединены к данному маршрутизатору или расположены поблизости на тупиковых маршрутах. Обо всех же остальных сетях можно сделать в таблице единственную запись, указывающую на маршрутизатор, через который пролегает путь ко всем этим сетям. Такой маршрутизатор называется маршрутизатором по умолчанию (default router). В нашем примере на маршрутизаторе 4 имеются специфические маршруты только для пакетов, следующих в сети N1-N6. Для всех остальных пакетов, адресованных в сети N7-N18, маршрутизатор предлагает продолжить путь через один и тот же порт IP51 маршрутизатора 5, который в данном случае и является маршрутизатором по умолчанию.

Таблицы маршрутизации конечных узлов

Задачу маршрутизации решают не только промежуточные узлы (маршрутизаторы), но и конечные узлы — компьютеры. Решение этой задачи начинается с того, что средствами протокола IP на конечном узле определяется, направлен ли пакет в другую сеть или адресован какому-нибудь узлу данной сети. Если номер сети назначения совпадает с номером данной сети, это означает, что пакет маршрутизировать не требуется. В противном случае маршрутизация нужна.

Структуры таблиц маршрутизации конечных узлов и транзитных маршрутизаторов аналогичны. Обратимся снова к сети, рассмотренной в предыдущем разделе "Схема IP-маршрутизации". Таблица маршрутизации конечного узла B, принадлежащего сети N3, могла бы выглядеть так, как табл. 1.

Номер сети назначения	Сетевой адрес следующего маршрутизатора	Сетевой адрес выходного порта	Расстояние до сети назначения
N1	IP_{13}(R1)	IP_B	1
N2	IP_{13}(R1)	IP_B	1
N3	-	IP_B	0
N4	IP_{31}(R3)	IP_B	1
N5	IP_{13}(R1)	IP_B	2
N6	IP_{31}(R3)	IP_B	2
Маршрут по умолчанию	IP_{31}(R3)	IP_B	-

Конечные узлы в еще большей степени, чем маршрутизаторы, пользуются приемом маршрутизации по умолчанию. Хотя они также в общем случае имеют в своем распоряжении таблицу маршрутизации, ее объем обычно незначителен, что объясняется периферийным расположением всех конечных узлов.

Таблицы маршрутизации конечных узлов

Конечный узел часто вообще работает без таблицы маршрутизации, имея только сведения об адресе маршрутизатора по умолчанию. При наличии одного маршрутизатора в локальной сети этот вариант — единственно возможный для всех конечных узлов. Но даже при наличии нескольких маршрутизаторов в локальной сети, когда перед конечным узлом стоит проблема их выбора, часто в компьютерах для повышения производительности прибегают к заданию маршрута по умолчанию.

Рассмотрим таблицу маршрутизации другого конечного узла составной сети — узла А (табл. 2). Компактный вид таблицы маршрутизации узла Л отражает тот факт, что все пакеты, направляемые из узла А, либо не выходят за пределы сети N12, либо непременно проходят через порт 1 маршрутизатора 17. Этот маршрутизатор и определен в таблице маршрутизации в качестве маршрутизатора по умолчанию.

Таблица 2. Таблица маршрутизации конечного узла А

Номер сети назначения	Сетевой адрес следующего маршрутизатора	Сетевой адрес выходного порта	Расстояние до сети назначения
N12	-	IP_A	0
Маршрут по умолчанию	IP17,1 (R17)	IP_A	-

Еще одним отличием работы маршрутизатора и конечного узла является способ построения таблицы маршрутизации. Если маршрутизаторы, как правило, автоматически создают таблицы маршрутизации, обмениваясь служебной информацией, то для конечных узлов таблицы маршрутизации часто создаются вручную администраторами и хранятся в виде постоянных файлов на дисках.

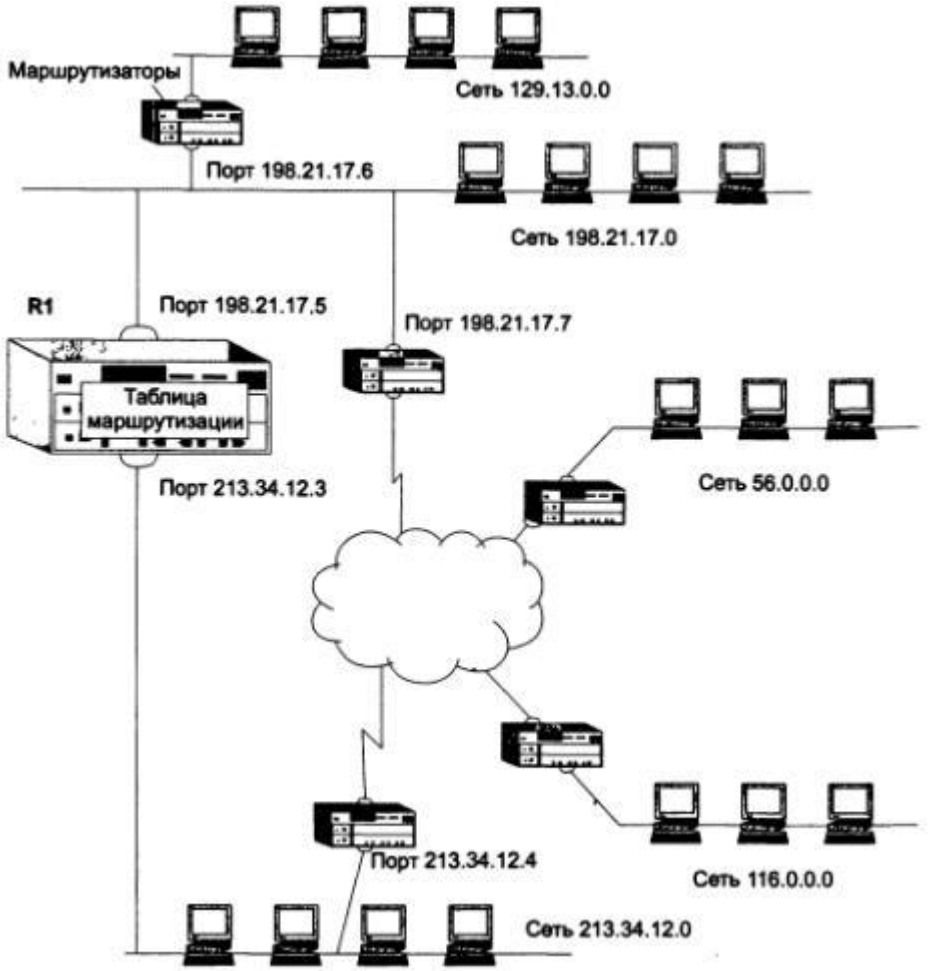
Просмотр таблиц маршрутизации без масок

Рассмотрим алгоритм просмотра таблицы маршрутизации, реализуемый на маршрутизаторе протоколом IP. При его описании мы будем использовать табл. 1 и рисунок сети предыдущего раздела.

1. Пусть на один из интерфейсов маршрутизатора поступает пакет. Протокол IP извлекает из пакета IP-адрес назначения (предположим, адрес назначения IPв).
2. Выполняется первая фаза просмотра таблицы — поиск конкретного маршрута к узлу. IP-адрес (целиком) последовательно строка за строкой сравнивается с содержимым поля адреса назначения таблицы маршрутизации. Если произошло совпадение (как в табл. 1), то из соответствующей строки извлекаются адрес следующего маршрутизатора (IP21) и идентификатор выходного интерфейса (IP41). На этом просмотр таблицы заканчивается.
3. Предположим теперь, что в таблице нет строки с адресом назначения IPв, а значит, совпадения не произошло. В этом случае протокол IP переходит ко второй фазе просмотра поиску маршрута к сети назначения. Из IP-адреса выделяется номер сети (в нашем примере из адреса IPв выделяется номер сети N3), и таблица снова просматривается на предмет совпадения номера сети в какой-либо строке с номером сети из пакета. При совпадении (в нашем примере оно произошло) из соответствующей строки таблицы извлекаются адрес следующего маршрутизатора (IP12) и идентификатор выходного интерфейса (IP41). Просмотр таблицы на этом завершается.
4. Наконец, предположим, что адрес назначения в пакете был таков, что совпадения не произошло ни в первой, ни во второй фазах просмотра. В таком случае средствами протокола IP либо выбирается маршрут по умолчанию (и пакет направляется по адресу IP51), либо, если маршрут по умолчанию отсутствует, пакет отбрасывается. Просмотр таблицы на этом заканчивается.

Примеры таблиц маршрутизации разных форматов

Структура реальных таблиц маршрутизации стека TCP/IP в целом соответствует упрощенной структуре рассмотренных ранее таблиц. Отметим, однако, что вид таблицы IP-маршрутизации зависит от конкретной реализации стека TCP/IP. Приведем пример нескольких вариантов таблицы маршрутизации, с которыми мог бы работать маршрутизатор R1 в сети, представленной на рисунке.



Примеры таблиц маршрутизации разных форматов

Начнем с «придуманного» предельно упрощенного варианта таблицы маршрутизации (табл. 1). Здесь имеются три маршрута к сетям (записи 56.0.0.0, 116.0.0.0 и 129.13.0.0), две записи о непосредственно подсоединенных сетях (198.21.17.0 и 213.34.12.0), а также запись о маршруте по умолчанию.

Таблица 1. Упрощенная таблица маршрутизации маршрутизатора R1.

Адрес сети назначения	Адрес следующего маршрутизатора	Адрес выходного интерфейса	Расстояние до сети назначения
56.0.0.0	213.34.12.4	213.34.12.3	15
116.0.0.0	213.34.12.4	213.34.12.3	13
129.13.0.0	198.21.17.6	198.21.17.5	2
198.21.17.0	198.21.17.5	198.21.17.5	1(подсоединена)
213.34.12.0	213.34.12.3	213.34.12.3	1(подсоединена)
Маршрут по умолчанию	198.21.17.7	198.21.17.5	-

Более сложный вид имеют таблицы, которые генерируются в промышленно выпускаемом сетевом оборудовании.

Примеры таблиц маршрутизации разных форматов

Если представить, что в качестве маршрутизатора R1 в данной сети работает штатный программный маршрутизатор операционной системы Microsoft Windows XP, то его таблица маршрутизации могла бы выглядеть так, как табл. 2.

Таблица 2. Таблица программного маршрутизатора ОС Windows XP.

Сетевой адрес	Маска	Адрес шлюза	Интерфейс	Метрика
127.0.0.0	255.0.0.0	127.0.0.1	127.0.0.1	1
0.0.0.0	0.0.0.0	198.21.17.7	198.21.17.5	1
56.0.0.0	255.0.0.0	213.34.12.4	213.34.12.3	15
116.0.0.0	255.0.0.0	213.34.12.4	213.34.12.3	13
129.13.0.0	255.255.0.0	198.21.17.6	198.21.17.5	2
198.21.17.0	255.255.255.0	198.21.17.5	198.21.17.5	1
198.21.17.5	255.255.255.255	127.0.0.1	127.0.0.1	1
198.21.17.255	255.255.255.255	198.21.17.5	198.21.17.5	1
213.34.12.0	255.255.255.0	213.34.12.3	213.34.12.3	1
213.34.12.3	255.255.255.255	127.0.0.1	127.0.0.1	1
213.34.12.255	255.255.255.255	213.34.12.3	213.34.12.3	1
224.0.0.0	224.0.0.0	198.21.17.6	198.21.17.6	1
224.0.0.0	224.0.0.0	213.34.12.3	213.34.12.3	1
255.255.255.255	255.255.255.255	198.21.17.6	198.21.17.6	1

Примеры таблиц маршрутизации разных форматов

Если на месте маршрутизатора R1 установить один из популярных аппаратных маршрутизаторов, то его таблица маршрутизации для этой же сети может выглядеть совсем иначе (табл. 3).

Таблица 3. Таблица маршрутизации аппаратного маршрутизатора.

Адрес назначения	Маска	Шлюз	Метрика	Статус	TTL	Источник
198.21.17.0	255.255.255.0	198.21.17.5	0	Up	-	Подключена
213.34.12.0	255.255.255.0	213.34.12.3	0	Up	-	Подключена
56.0.0.0	255.0.0.0	213.34.12.4	14	Up	-	Статическая
116.0.0.0	255.0.0.0	213.34.12.4	12	Up	-	Статическая
129.13.0.0	255.255.0.0	198.21.17.6	1	Up	160	RIP

Примеры таблиц маршрутизации разных форматов

И наконец табл. 4 представляет собой таблицу маршрутизации для того же маршрутизатора R1, реализованного в виде программного маршрутизатора одной из версий операционной системы Unix.

Таблица 4. Таблица маршрутизации Unix – маршрутизатора.

Адрес назначения	Шлюз	Флаги	Число ссылок	Загрузка	Интерфейс
127.0.0.0	127.0.0.1	UH	1	154	1o0
Маршрут по умолчанию	198.21.17.7	UG	5	43270	1e0
198.21.17.0	198.21.17.5	U	35	246876	1e0
213.34.12.0	213.34.12.3	U	44	132435	le1
129.13.0.0	198.21.1.7.6	UG	6	16450	1e0
56.0.0.0	213.34.12.4	UG	12	5764	le1
116.0.0.0	213.34.12.4	UG	21	23544	le1

ПРИМЕЧАНИЕ

Заметим, что поскольку между структурой сети и таблицей маршрутизации нет однозначного соответствия, для каждого из приведенных вариантов таблицы можно предложить свои «подварианты», отличающиеся выбранным маршрутом к той или иной сети. В данном случае внимание концентрируется на существенных различиях в форме представления маршрутной информации разными реализациями маршрутизаторов.

Примеры таблиц маршрутизации разных форматов

Несмотря на достаточно заметные внешние различия, во всех трех «реальных» таблицах присутствуют все ключевые данные из рассмотренной упрощенной таблицы, без которых невозможна маршрутизация пакетов.

К таким данным, во-первых, относятся адреса сети назначения (столбцы «Адрес назначения» в аппаратном маршрутизаторе и маршрутизаторе Unix или столбец «Сетевой адрес» в маршрутизаторе ОС Windows XP).

Вторым обязательным полем таблицы маршрутизации является адрес следующего маршрутизатора (столбцы «Шлюз» в аппаратном маршрутизаторе и маршрутизаторе Unix или столбец «Адрес шлюза» в маршрутизаторе ОС Windows XP).

Третий ключевой параметр — адрес порта, на который нужно направить пакет, в некоторых таблицах указывается прямо (столбец «Интерфейс» в таблице маршрутизатора ОС Windows XP), а в некоторых — косвенно. Так, в таблице маршрутизатора Unix вместо адреса порта задается его условное наименование — 1e0 для порта с адресом 198.21.17.5, le1 для порта с адресом 213.34.12.3 и 1o0 для внутреннего порта с адресом 127.0.0.1. В адпаратном маршрутизаторе поле, обозначающее выходной порт в какой-либо форме, вообще отсутствует. Это объясняется тем, что адрес выходного порта всегда можно косвенно определить по адресу следующего маршрутизатора. Например, определим по табл. 3 адрес выходного порта для сети 56.0.0.0. Из таблицы следует, что следующим маршрутизатором для этой сети будет маршрутизатор с адресом 213.34.12.4. Адрес следующего маршрутизатора должен принадлежать одной из непосредственно присоединенных к маршрутизатору сетей, и в данном случае это сеть 213.34.12.0. Маршрутизатор имеет порт, присоединенный к этой сети, и адрес этого порта 213.34.12.3 мы находим в столбце «Шлюз» второй строки таблицы маршрутизации, которая описывает непосредственно присоединенную сеть 213.34.12.0

Примеры таблиц маршрутизации разных форматов

Для непосредственно присоединенных сетей адресом следующего маршрутизатора всегда является адрес собственного порта маршрутизатора. Таким образом, для сети 56.0.0 адресом выходного порта является 213.34.12.3.

Стандартным решением сегодня является использование поля маски в каждой записи таблицы, как это сделано в таблицах маршрутизатора ОС Windows XP и аппаратного маршрутизатора (столбцы «Маска»). Механизм обработки масок при принятии решения маршрутизаторами рассматривается далее. Отсутствие поля маски говорит о том, что либо маршрутизатор рассчитан на работу только с тремя стандартными классами адресов, либо для всех записей используется одна и та же маска, что снижает гибкость маршрутизации. Поскольку в таблице маршрутизации маршрутизатора Unix каждая сеть назначения упомянута только один раз, а значит, возможность выбора маршрута отсутствует, то поле метрики является необязательным параметром. В остальных двух таблицах поле метрики используется только для указания на то, что сеть подключена непосредственно. Метрика 0 для аппаратного маршрутизатора или 1 для маршрутизатора ОС Windows XP говорит маршрутизатору, что эта сеть непосредственно подключена к его порту, а другое значение метрики соответствует удаленной сети. Выбор метрики для непосредственно подключенной сети (1 или 0) является произвольным, главное, чтобы метрика удаленной сети отсчитывалась с учетом этого выбранного начального значения. В маршрутизаторе Unix используется поле признаков, где флаг G (Gateway — шлюз) отмечает удаленную сеть, а его отсутствие — непосредственно подключенную.

Примеры таблиц маршрутизации разных форматов

Признак непосредственно подключенной *сети говорит маршрутизатору, что пакет уже достиг своей сети, поэтому протокол IP активизирует ARP-запрос относительно IP-адреса узла назначения, а не следующего маршрутизатора.*

Однако существуют ситуации, когда маршрутизатор должен обязательно хранить значение метрики для записи о каждой удаленной сети. Эти ситуации возникают, когда записи в таблице маршрутизации являются результатом работы некоторых протоколов маршрутизации, например протокола RIP. В таких протоколах новая информация о какой-либо удаленной сети сравнивается с информацией, содержащейся в таблице в данный момент, и если значение новой метрики лучше текущей, то новая запись вытесняет имеющуюся. В таблице маршрутизатора Unix поле метрики отсутствует, и это значит, что он не использует протокол RIP.

Флаги записей присутствуют только в таблице маршрутизатора Unix.

- U — маршрут активен и работоспособен. Аналогичный смысл имеет поле статуса в аппаратном маршрутизаторе.
- H — признак специфического маршрута к определенному хосту.
- G — означает, что маршрут пакета проходит через промежуточный маршрутизатор (шлюз). Отсутствие этого флага отмечает непосредственно подключенную сеть.
- D — означает, что маршрут получен из перенаправленного сообщения протокола ICMP. Этот признак может присутствовать только в таблице маршрутизации конечного узла. Признак означает, что конечный узел при какой-то предыдущей передаче пакета выбрал не самый рациональный следующий маршрутизатор на пути к данной сети, и этот маршрутизатор с помощью протокола ICMP сообщил конечному узлу, что все последующие пакеты к данной сети нужно отправлять через другой маршрутизатор.

Примеры таблиц маршрутизации разных форматов

В таблице маршрутизатора Unix используются еще два поля, имеющих справочное значение. Поле числа ссылок показывает, сколько раз на данный маршрут ссылались при продвижении пакетов. Поле загрузки отражает количество байтов, переданных по данному маршруту.

В записях таблиц аппаратного маршрутизатора также имеются два справочных поля. Поле времени жизни записи (TTL) в данном случае никак не связано со временем жизни пакета. Здесь оно показывает время, в течение которого значение данной записи еще действительно. Поле источника говорит об источнике появления записи в таблице маршрутизации.

Источники и типы записей в таблице маршрутизации

Практически для всех маршрутизаторов существуют три основных источника записей в таблице.

- Одним из источников записей в таблице маршрутизации является программное обеспечение стека TCP/IP, которое при инициализации маршрутизатора автоматически заносит в таблицу несколько записей, в результате чего создается так называемая минимальная таблица маршрутизации. Программное обеспечение формирует записи о непосредственно подключенных сетях и маршрутах по умолчанию, информация о которых появляется в стеке при ручном конфигурировании интерфейсов компьютера или маршрутизатора. К таким записям в приведенных примерах относятся записи о сетях 213.34.12.0 и 198.21.17.0, а также запись о маршруте по умолчанию в маршрутизаторе Unix и запись 0.0.0.0 в маршрутизаторе ОС Windows XP. Кроме того, программное обеспечение автоматически заносит в таблицу маршрутизации записи об адресах особого назначения. В приведенных примерах таблица маршрутизатора ОС Windows 2000 содержит наиболее полный набор записей такого рода. Несколько записей в этой таблице связано с особым адресом 127.0.0.0. Записи с адресом 224.0.0.0 требуются для обработки групповых адресов. Кроме того, в таблицу могут быть занесены адреса, предназначенные для обработки широковещательных рассылок (например, записи 8 и 11 содержат адрес отправки широковещательного сообщения в соответствующих подсетях, а последняя запись в таблице — адрес ограниченной широковещательной рассылки). Заметим, что в некоторых таблицах записи об особых адресах вообще отсутствуют.

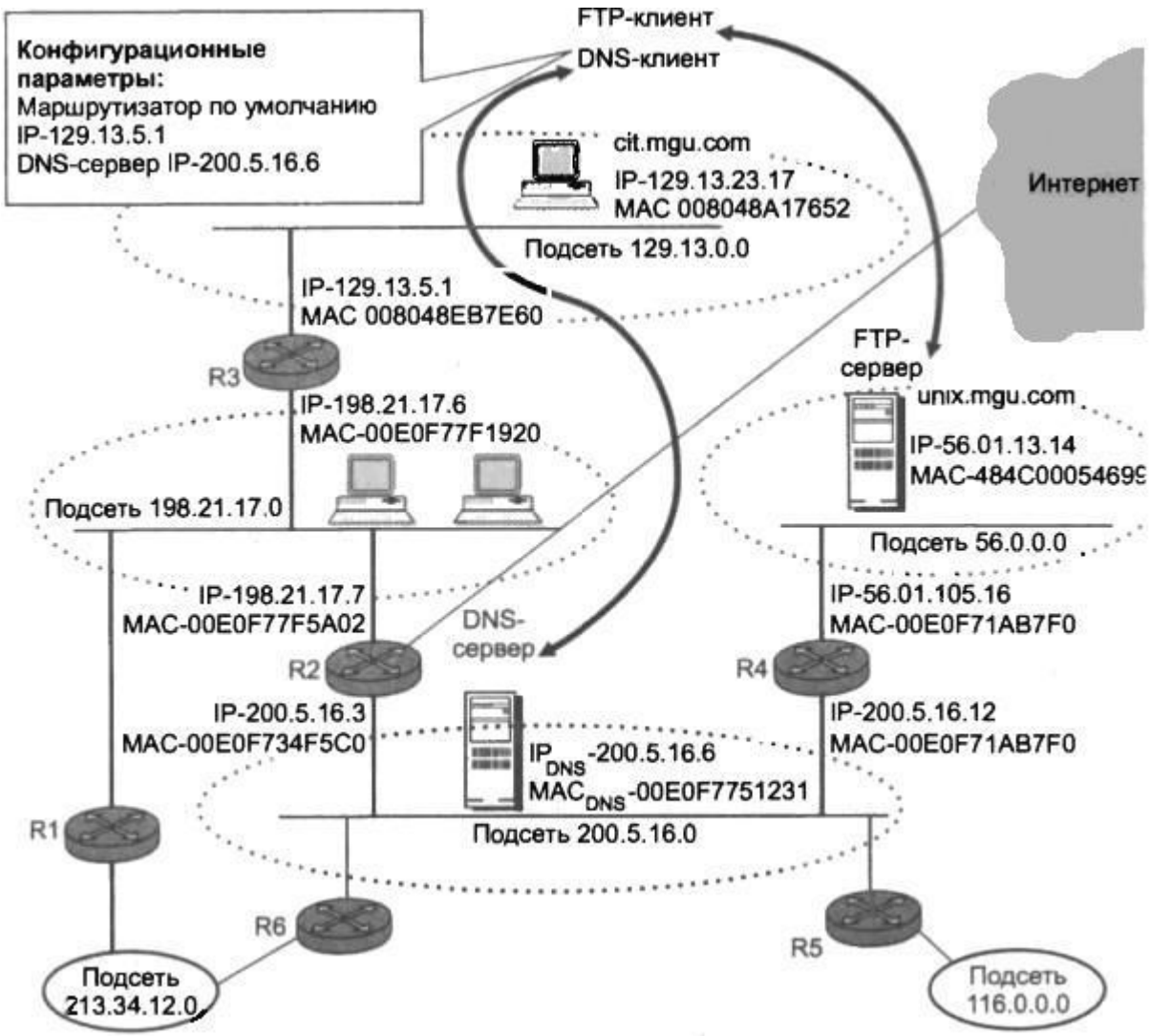
Источники и типы записей в таблице маршрутизации

- Еще одним источником записей в таблице является администратор, непосредственно формирующий записи с помощью некоторой системной утилиты, например программы route, имеющейся в операционных системах Unix и Windows XR В аппаратных маршрутизаторах также всегда имеется команда для ручного задания записей таблицы маршрутизации. Заданные вручную записи всегда являются статическими, то есть они не имеют срока жизни. Эти записи могут быть как постоянными, то есть сохраняющимися при перезагрузке маршрутизатора, так и временными, хранящимися в таблице только до выключения устройства. Часто администратор вручную заносит запись о маршруте по умолчанию. Таким же образом в таблицу маршрутизации может быть внесена запись о специфическом для узла маршруте.
- И наконец, третьим источником записей могут быть протоколы маршрутизации, такие как RIP или OSPF. Эти записи всегда являются динамическими, то есть имеют ограниченный срок жизни.

Программные маршрутизаторы Windows XP и Unix не показывают источник появления той или иной записи в таблице, а аппаратный маршрутизатор использует для этой цели поле источника. В приведенном в табл. 3 примере первые две записи созданы программным обеспечением стека на основании данных о конфигурации портов маршрутизатора — это показывает признак «Подключена». Следующие две записи обозначены как статические — это означает, что их ввел вручную администратор. Последняя запись является следствием работы протокола RIP, поэтому в ее поле «TTL» имеется значение 160.

Примеры IP- маршрутизации без масок

Рассмотрим процесс продвижения пакета в составной сети на примере IP-сети, показанной на рисунке.



Примеры IP- маршрутизации без масок

При этом будем считать, что все узлы сети, рассматриваемой в примере, имеют адреса, основанные на классах. Особое внимание будет уделено взаимодействию протокола IP с протоколами разрешения адресов ARP и DNS.

Итак, пусть пользователю компьютера cit.mgu.com, находящегося в сети 129.13.0.0, необходимо установить связь с FTP-сервером. Пользователю известно символьное имя сервера unix.mgu.com, поэтому он набирает на клавиатуре команду обращения к FTP-серверу по имени:

> ftp unix.mgu.com

Выполнение этой команды инициирует три последовательные операции:

1. DNS-клиент (работающий на компьютере cit.mgu.com) передает DNS-серверу сообщение, в котором содержится запрос об IP-адресе сервера unix.mgu.com, с которым он хочет связаться по протоколу FTP.
2. DNS-сервер, выполнив поиск, передает ответ DNS-клиенту о найденном IP-адресе сервера unix.mgu.com.
3. FTP-клиент (работающий на том же компьютере cit.mgu.com), используя найденный IP-адрес сервера unix.mgu.com, передает сообщение работающему на нем FTP-серверу.

Последовательно, по шагам, рассмотрим, как при решении этих задач взаимодействуют между собой протоколы DNS, IP, ARP и Ethernet и что происходит при этом с кадрами и пакетами.

Примеры IP- маршрутизации без масок

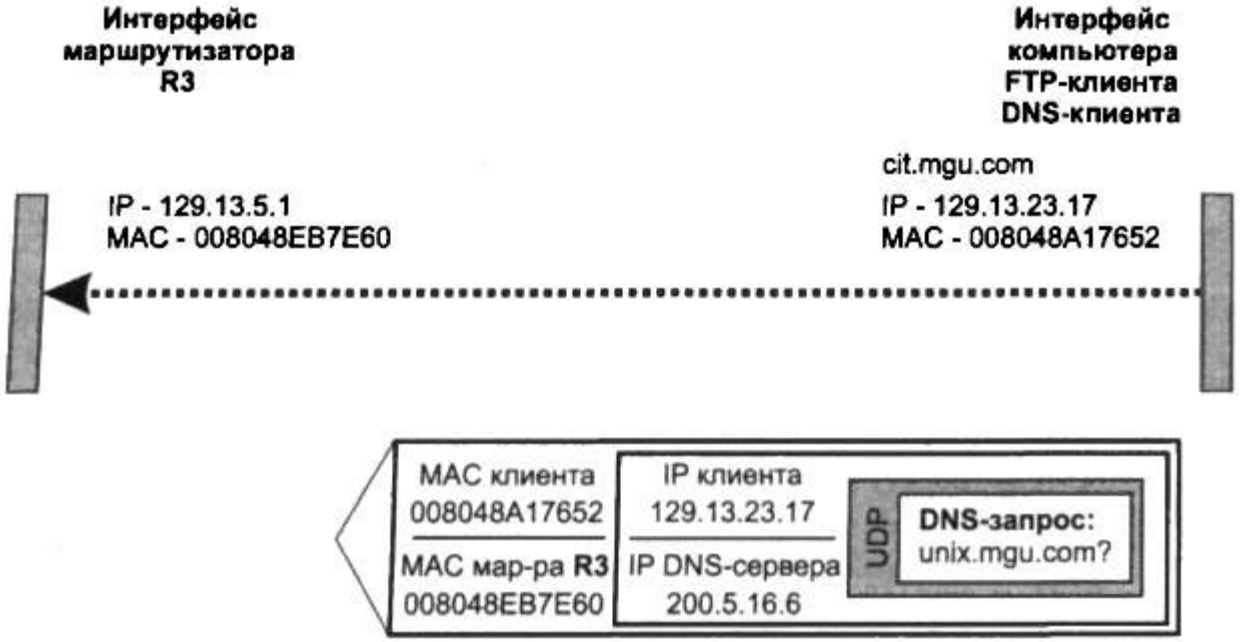
1. Формирование IP-пакета с инкапсулированным в него DNS-запросом. Программный модуль FTP-клиента, получив команду > ftp uni x .mgu. com, передает запрос к работающей на этом же компьютере клиентской части протокола DNS, которая, в свою очередь, формирует к DNS-серверу запрос, интерпретируемый примерно так: «Какой IP-адрес соответствует символьному имени unlx.mgu.com?» Запрос упаковывается в UDP-дейтаграмму, затем в IP-пакет. В заголовке пакета в качестве адреса назначения указывается IP-адрес 200.5.16.6 DNS-сервера. Этот адрес известен программному обеспечению клиентского компьютера, так как он входит в число его конфигурационных параметров. Сформированный IP-пакет будет перемещаться по сети в неизменном виде (как показано на рис. 2), пока не дойдет до адресата — DNS-сервера. (**поменять местами сервер и клиент !**)

Рис. 2. IP-пакет с инкапсулированным в него DNS-запросом.

2. Передача кадра Ethernet с IP-пакетом маршрутизатору R3. Для передачи этого IP-пакета необходимо его упаковать в кадр Ethernet, указав в заголовке MAC-адрес получателя. Технология Ethernet способна доставлять кадры только тем адресатам, которые находятся в пределах одной подсети с отправителем. Если же адресат расположен вне этой подсети, то кадр надо передать ближайшему маршрутизатору, чтобы тот взял на себя заботу о дальнейшем перемещении пакета.

Примеры IP- маршрутизации без масок

Для этого модуль IP, сравнив номера сетей в адресах отправителя и получателя, то есть 129.13.23.17 и 200.5.16.6, выясняет, что пакет направляется в другую сеть, следовательно, его необходимо передать маршрутизатору, в данном случае маршрутизатору по умолчанию. IP-адрес маршрутизатора по умолчанию также известен клиентскому узлу, поскольку он входит в число конфигурационных параметров. Однако для кадра Ethernet необходимо указать не IP-адрес, а MAC-адрес получателя. Эта проблема решается с помощью протокола ARP, который для ответа на вопрос: «Какой MAC-адрес соответствует IP-адресу 129.13.5.1?» — делает поиск в своей ARP-таблице. Поскольку обращения к маршрутизатору происходят часто, будем считать, что нужный MAC-адрес обнаруживается в таблице и имеет значение 008048EB7E60. После получения этой информации клиентский компьюте pcit.mgu.com отправляет маршрутизатору R3 пакет, упакованный в кадр Ethernet (рис . 3).



Примеры IP- маршрутизации без масок

3. Определение IP-адреса и MAC- адреса следующего маршрутизатора R2. Кадр принимается интерфейсом 129.13.5.1 маршрутизатора R3. Протокол Ethernet, работающий на этом интерфейсе, извлекает из этого кадра IP-пакет и передает его протоколу IP. Протокол IP находит в заголовке пакета адрес назначения 200.5.16.6 и просматривает записи своей таблицы маршрутизации. Пусть маршрутизатор R3 не обнаруживает специфического маршрута для адреса назначения 200.5.16.6, но находит в своей таблице следующую запись:

200.5.16.0 198.21.17.7 198.21.17.6

Эта запись говорит о том, что пакеты для сети 200.5.16.0 маршрутизатор R3 должен передавать на свой выходной интерфейс 198.21.17.6, с которого они поступят на интерфейс следующего маршрутизатора R2, имеющего IP-адрес 198.21.17.7. Однако знания IP-адреса недостаточно, чтобы передать пакет по сети Ethernet. Необходимо определить MAC-адрес маршрутизатора R3. Как известно, такой работой занимается протокол ARP. Пусть на этот раз в ARP-таблице нет записи об адресе маршрутизатора R3. Тогда в сеть отправляется широковещательный ARP-запрос, который поступает на все интерфейсы сети 198.21.17.0. Ответ приходит только от интерфейса маршрутизатора R3: "Я имею IP-адрес 198.21.17.7 и мой MAC-адрес 00E0F77F5A02" (рис. 4).

Примеры IP- маршрутизации без масок

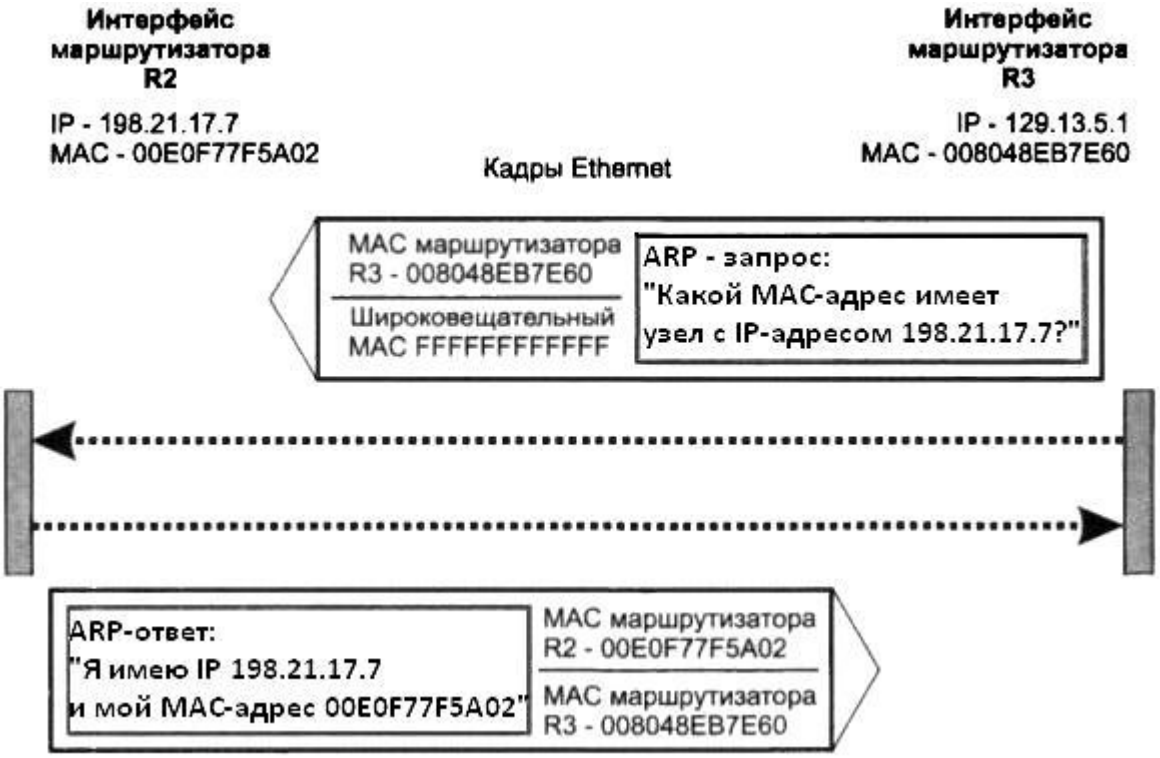


Рис 4. Кадры Ethernet с инкапсулированными ARP-запросом и ARP-ответом

Примеры IP- маршрутизации без масок

Теперь, зная MAC- адрес маршрутизатора R2 (00E0F77F5A02), маршрутизатор R3 отсылает ему IP-пакет с DNS-запросом (рис. 5).

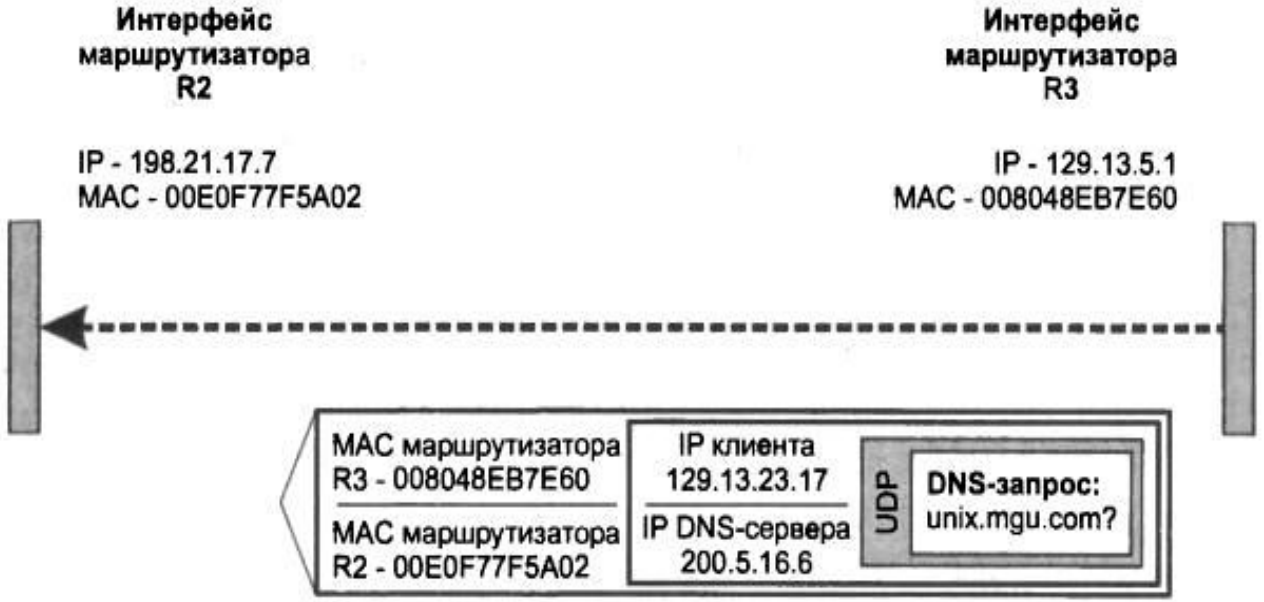


Рис 5. Кадры Ethernet с DNS-запросом, отправленный с маршрутизатора R3 маршрутизатору R2

Примеры IP- маршрутизации без масок

4. Маршрутизатор R2 доставляет пакет DNS-серверу. Модуль IP на маршрутизаторе R2 действует в соответствии с уже не раз описанной нами процедурой: отбросив заголовок кадра Ethernet, он извлекает из пакета IP-адрес назначения и просматривает свою таблицу маршрутизации. Там он обнаруживает, что сеть назначения 200.5.16.0 является непосредственно присоединенной к его второму интерфейсу. Следовательно, пакет не нужно маршрутизировать, однако требуется определить MAC-адрес узла назначения. Протокол ARP «по просьбе» протокола IP находит (либо из ARP-таблицы, либо по запросу) требуемый MAC-адрес 00E0F7751231 DNS-сервера. Получив ответ о MAC-адресе, маршрутизатор R2 отправляет в сеть назначения кадр Ethernet с DNS-запросом (рис. 6).

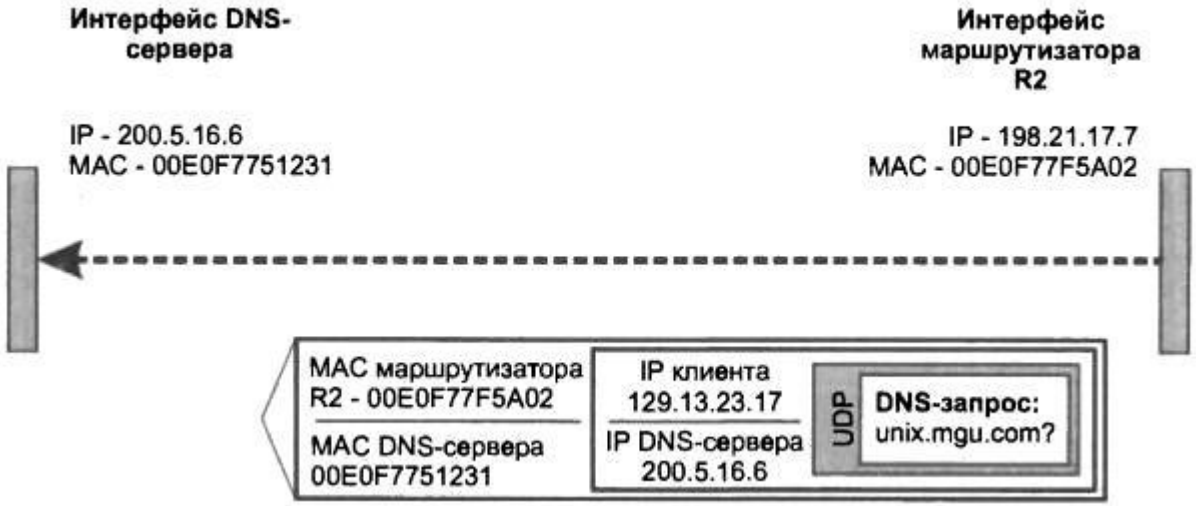


Рис. 6. Кадр Ethernet с DNS-запросом, отправленный с маршрутизатора R2

Примеры IP- маршрутизации без масок

5. Сетевой адаптер DNS-сервера захватывает кадр Ethernet, обнаруживает совпадение MAC- адреса назначения, содержащегося в заголовке, со своим собственным адресом и направляет его модулю IP. После анализа полей заголовка IP из пакета извлекаются данные вышележащих протоколов. DNS-запрос передается программному модулю DNS-сервера DNS-сервер просматривает свои таблицы, возможно, обращается к другим DNS-серверам и в результате формирует ответ, смысл которого состоит в следующем: «Символьному имени Unix.mgu.com соответствует IP-адрес 56.01.13.14».

Процесс доставки DNS-ответа клиенту cit.mgu.com совершенно аналогичен процессу передачи DNS-запроса, который мы только что так подробно описали. Работая в тесной кооперации, протоколы IP, ARP и Ethernet передают клиенту DNS-ответ через всю составную сеть (рис. 7).

Рис.7. Кадр Ethernet с DNS-ответом, отправленный с маршрутизатора R3 компьютеру-клиенту

FTP-клиент, получив IP-адрес FTP-сервера, посылает ему свое сообщение, используя те же описанные ранее механизмы доставки данных через составную сеть. Полезно мысленно воспроизвести этот процесс, обращая особое внимание на значения адресных полей заголовков кадров и заголовка вложенного IP-пакета.

Маршрутизация с использованием масок
Структуризация сети масками одинаковой длины

Основная причина отказа от хорошо себя зарекомендовавшего в течение многих лет метода адресации, основанного на классах — потребность в структуризации сетей в условиях дефицита нераспределенных номеров сетей.

Технологии масок позволяет разделить одну сеть на несколько, что обеспечивает администраторам сетей структурировать сеть, например, развести все слабо взаимодействующие компьютеры по разным сетям, в условиях недостатка централизованно выделенных им номеров сетей.

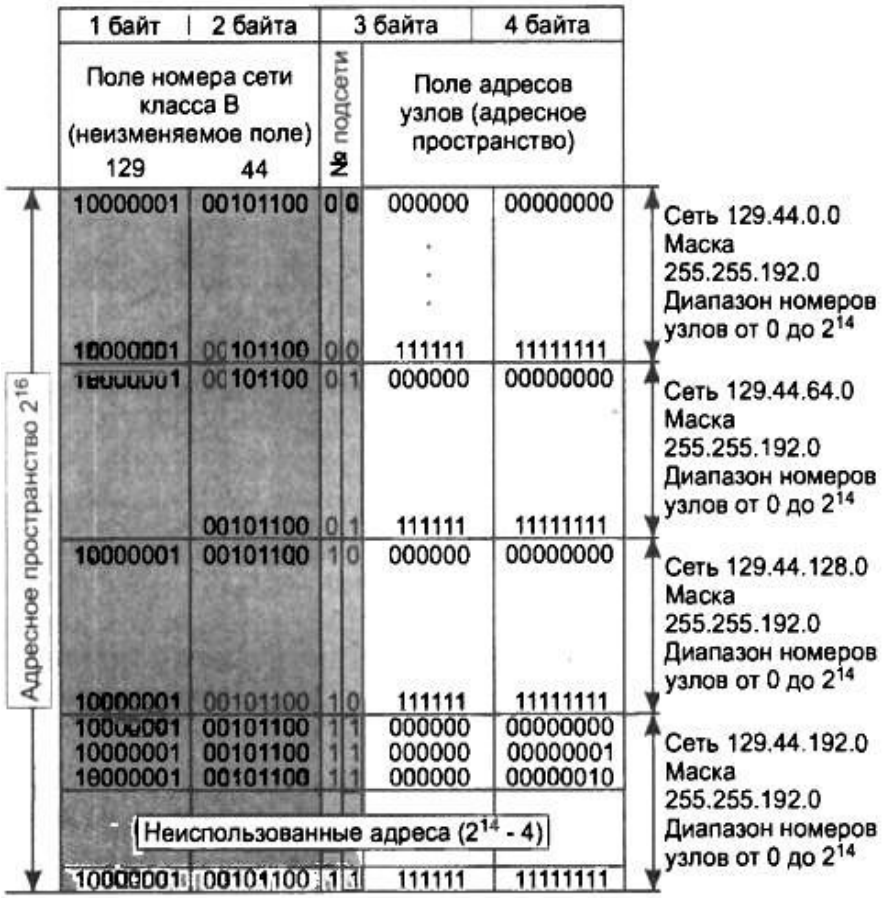
Допустим, администратор получил в свое распоряжение сеть класса B: 129.44.0.0. Он может организовать сеть с большим числом узлов, номера которых доступны ему из диапазона 0.0.0.1-0.0.255.254. Всего в его распоряжении имеется (2^{16} - 2) адреса. Вычитание двойки связано с учетом того, что адреса из одних нулей и одних единиц имеют специальное назначение и не годятся для адресации узлов. Однако ему не нужна одна большая неструктурированная сеть. Производственная необходимость диктует администратору другое решение, в соответствии с которым сеть должна быть разделена на три отдельных подсети, при этом трафик в каждой подсети должен быть надежно локализован. Это позволит легче диагностировать сеть и проводить в каждой из подсетей особую политику безопасности. (Заметим, что разделение большой сети с помощью масок имеет еще одно преимущество — оно позволяет скрыть внутреннюю структуру сети предприятия от внешнего наблюдения и тем самым повысить ее безопасность)

Маршрутизация с использованием масок
Структуризация сети масками одинаковой длины

На рисунке показано разделение всего полученного администратором адресного диапазона на 4 равные части — каждая по 2^{14} адресов. При этом число разрядов, доступное для нумерации узлов, уменьшилось на два бита, а префикс (номер) каждой из четырех сетей стал длиннее на два бита. Следовательно, каждый из четырех диапазонов можно записать в виде IP-адреса с маской, состоящей из 18 единиц, или в десятичной нотации - 255.255.192.0.)

129.44.0.0/18 (10000001 00101100 **00**000000 00000000)
129.44.64.0/18 (10000001 00101100 **01**000000 00000000)
129.44.128.0/18 (10000001 00101100 **10**000000 00000000)
129.44.192.0/18 (10000001 00101100 **11**000000 00000000)

Из приведенных записей видно, что администратор получает возможность использовать для нумерации подсетей два дополнительных бита (выделенных жирным шрифтом). Именно это позволяет ему сделать из одной централизованно выделенной сети четыре, в данном примере это 129.44.0.0/18, 129.44.64.0/18, 129.44.128.0/18, 129.44.192.0/18.

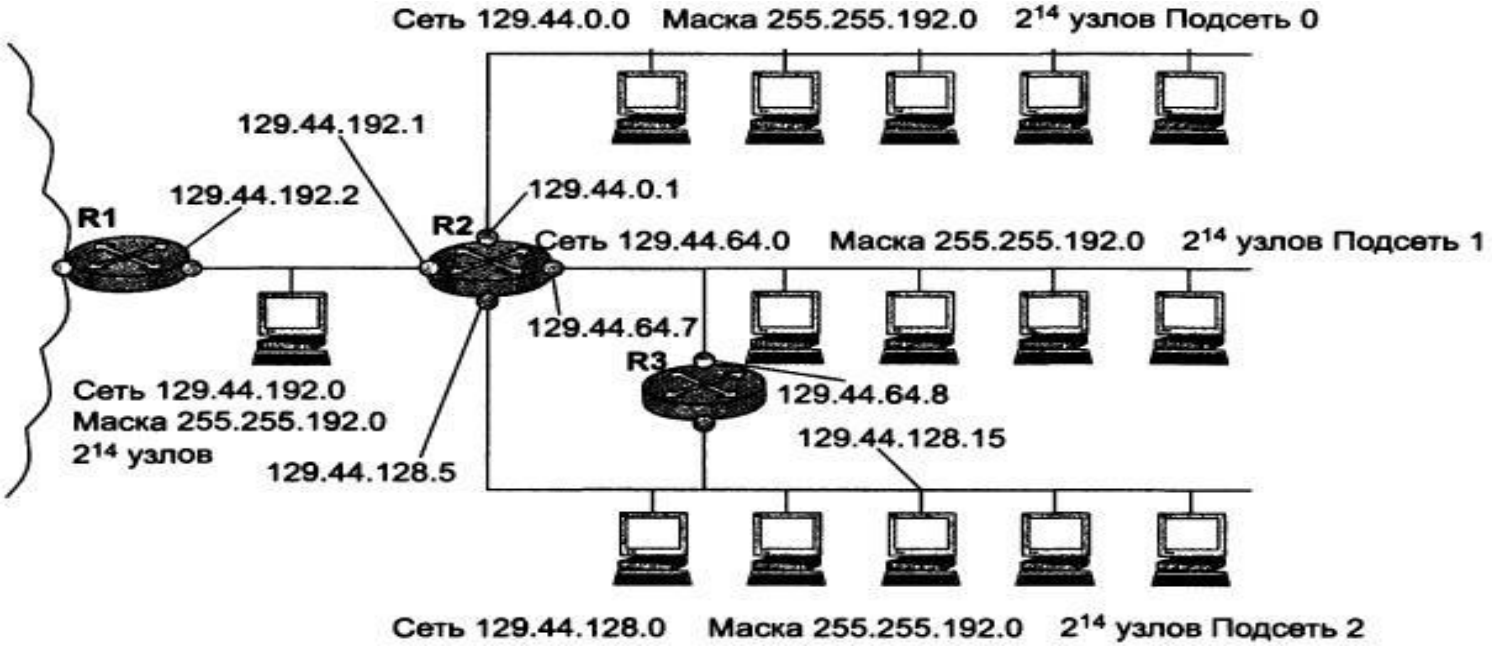


Разделение адресного пространства 129.44.0.0 сети класса B на четыре равные части

Маршрутизация с использованием масок
Структуризация сети масками одинаковой длины

Пример сети, построенной путем деления на 4 сети равного размера, показан на рисунке. Весь трафик во внутреннюю сеть 129.44.0.0, направляемый из внешней сети, поступает через маршрутизатор R1. В целях структуризации информационных потоков во внутренней сети установлен дополнительный маршрутизатор R2. Каждая из вновь образованных сетей 129.44.0.0/18, 129.44.64.0/18,129.44.128.0/18 и 129.44.192.0/18 подключена к соответственно сконфигурированным портам внутреннего маршрутизатора R2.

Извне сеть по-прежнему выглядит, как единая сеть класса B. Однако поступающий в сеть общий трафик разделяется локальным маршрутизатором R2 между четырьмя сетями. В условиях, когда механизм классов не действует, маршрутизатор должен иметь другое средство, которое позволило бы ему определять, какая часть 32-разрядного числа, помещенного в поле адреса назначения, является номером сети. Именно этой цели служит дополнительное поле маски, включенное в таблицу маршрутизации (табл. 1).



Маршрутизация с использованием масок
Структуризация сети масками одинаковой длины

Именно этой цели служит дополнительное поле маски, включенное в таблицу маршрутизации (табл. 1).

Адрес назначения	Маска	Адрес следующего маршрутизатора	Адрес порта	Расстояние
129.44.0.0	255.255.192.0	129.44.0.1	129.44.0.1	Подключена
129.44.64.0	255.255.192.0	129.44.64.7	129.44.64.7	Подключена
129.44.128.0	255.255.192.0	129.44.128.5	129.44.128.5	Подключена
129.44.192.0	255.255.192.0	129.44.192.1	129.44.192.1	Подключена
0.0.0.0	0.0.0.0	129.44.192.2	129.44.192.1	-
129.44.128.15	255.255.255.255	129.44.64.8	129.44.64.7	-

Первые четыре записи в таблице соответствуют внутренним подсетям, непосредственно подключенным к портам маршрутизатора R2.

Запись 0.0.0.0 с маской 0.0.0.0 соответствует маршруту по умолчанию.

Последняя запись определяет специфический маршрут к узлу 129.44.128.15. В тех строках таблицы, в которых в качестве адреса назначения указан полный IP-адрес узла, маска имеет значение 255.255.255.255. В отличие от всех других узлов сети 129.44.128.0, к которым пакеты поступают с интерфейса 129.44.128.5 маршрутизатора R2, к данному узлу они должны приходить через маршрутизатор R3.

Маршрутизация с использованием масок
Просмотр таблиц маршрутизации с учетом масок

Алгоритм просмотра таблиц маршрутизации, содержащих маски, имеет много общего с описанным алгоритмом просмотра таблиц, не содержащих маски. Однако в нем имеются и существенные изменения.

1. Поиск следующего маршрутизатора для вновь поступившего IP-пакета протокол начинает с того, что извлекает из пакета адрес назначения (обозначим его IP_D). Затем протокол IP приступает к процедуре просмотра таблицы маршрутизации, также состоящей из двух фаз, как и процедура просмотра таблицы, в которой столбец маски отсутствует.
2. Первая фаза состоит в поиске специфического маршрута для адреса IP_D. С этой целью из каждой записи таблицы, в которой маска имеет значение 255.255.255.255, извлекается адрес назначения и сравнивается с адресом из пакета IP_D . Если в какой-либо строке совпадение произошло, то адрес следующего маршрутизатора для данного пакета берется из данной строки.
3. Вторая фаза выполняется только в том случае, если во время первой фазы не произошло совпадения адресов. Она состоит в поиске неспецифического маршрута, общего для группы узлов, к которой относится и пакет с адресом IP_D. Для этого средствами IP заново просматривается таблица маршрутизации, причем с каждой записью производятся следующие действия:

1) маска (обозначим ее M), содержащаяся в данной записи, «накладывается» на IP-адрес узла назначения IP_D, извлеченный из пакета: IP_D AND M;
2) полученное в результате число сравнивается со значением, которое помещено в поле адреса назначения той же записи таблицы маршрутизации;
3) если происходит совпадение, протокол IP соответствующим образом отмечает эту строку.
4) если просмотрены не все строки, то протокол IP аналогичным образом просматривает следующую строку, если все (включая строку о маршруте по умолчанию), то просмотр записей заканчивается, и происходит переход к следующему шагу.

4. После просмотра всей таблицы маршрутизатор выполняет одно их трех действий:
1) если не произошло ни одного совпадения и маршрут по умолчанию отсутствует, то пакет отбрасывается;
2) если произошло одно совпадение, то пакет отправляется по маршруту, указанному в строке с совпавшим адресом;
3) если произошло несколько совпадений, то все помеченные строки сравниваются и выбирается маршрут из той строки, в которой количество совпавших двоичных разрядов наибольшее (другими словами, в ситуации, когда адрес назначения пакета принадлежит сразу нескольким подсетям, маршрутизатор использует наиболее специфический маршрут).

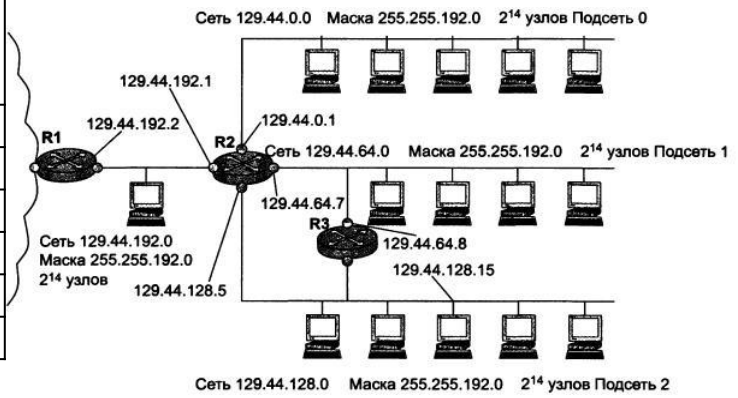
ПРИМЕЧАНИЕ

Во многих таблицах маршрутизации запись с адресом 0.0.0.0 и маской 0.0.0.0 соответствует маршруту по умолчанию. Действительно, любой адрес в пришедшем пакете после наложения на него маски 0.0.0.0 даст адрес сети 0.0.0.0, что совпадает с адресом, указанным в записи. Поскольку маска 0.0.0.0 имеет нулевую длину, то этот маршрут считается самым неспецифическим и используется только при отсутствии совпадений с остальными записями из таблицы маршрутизации.

Маршрутизация с использованием масок
Просмотр таблиц маршрутизации с учетом масок

Проиллюстрируем, как маршрутизатор использует описанный алгоритм для работы со своей таблицей маршрутизации.

Адрес назначения	Маска	Адрес следующего маршрутизатора	Адрес порта	Расстояние
129.44.0.0	255.255.192.0	129.44.0.1	129.44.0.1	Подключена
129.44.64.0	255.255.192.0	129.44.64.7	129.44.64.7	Подключена
129.44.128.0	255.255.192.0	129.44.128.5	129.44.128.5	Подключена
129.44.192.0	255.255.192.0	129.44.192.1	129.44.192.1	Подключена
0.0.0.0	0.0.0.0	129.44.192.2	129.44.192.1	-
129.44.128.15	255.255.255.255	129.44.64.8	129.44.64.7	-



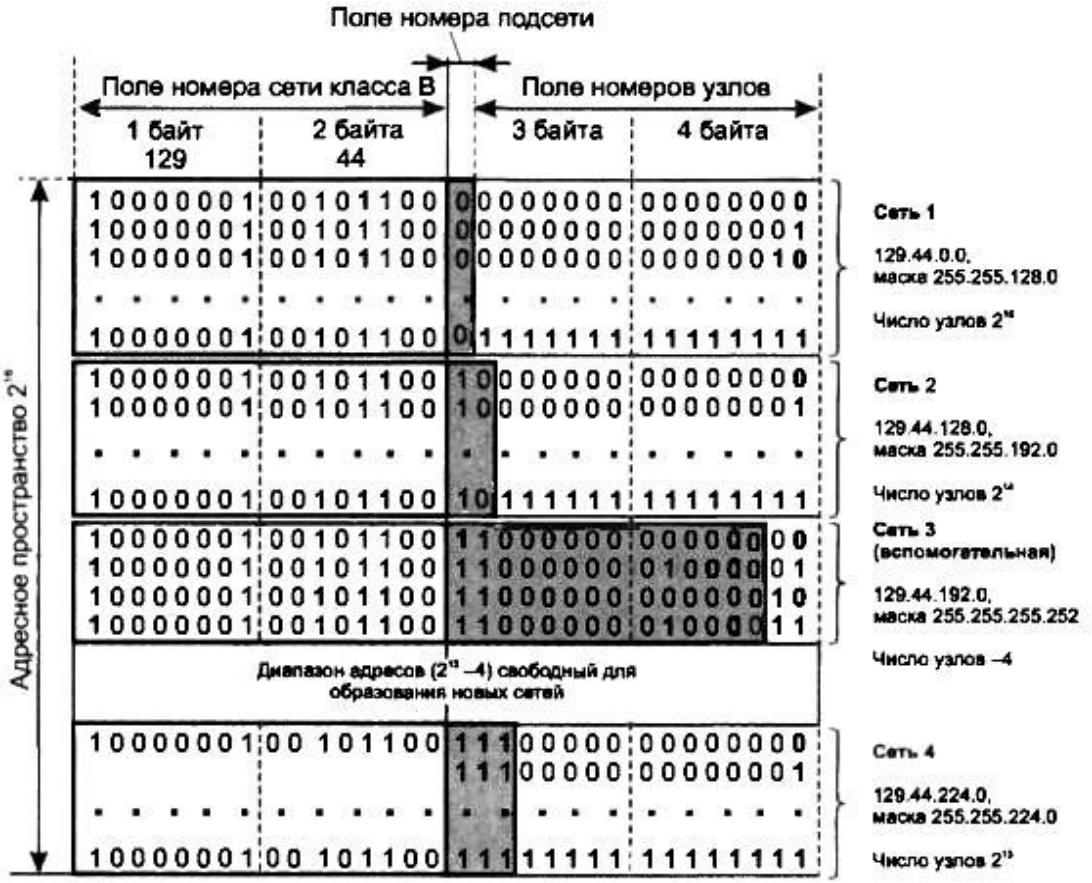
Пусть на маршрутизатор R2 поступает пакет с адресом назначения 129.44.78.200. Модуль IP, установленный на этом маршрутизаторе, прежде всего сравнит этот адрес с адресом 129.44.128.15, для которого определен специфический маршрут. Совпадения нет, поэтому модуль IP начинает последовательно обрабатывать все строки таблицы, накладывая маски и сравнивая результаты до тех пор, пока не найдет совпадения номера сети в адресе назначения и в строке таблицы. В результате определяется маршрут для пакета 129.44.78.200 — он должен быть отправлен на выходной порт маршрутизатора 129.44.64.7 в сеть 129.44.64.0, непосредственно подключенную к данному маршрутизатору.

Маршрутизация с использованием масок
Использование масок переменной длины

Во многих случаях более эффективным является разбиение сети на подсети разного размера. В частности, для подсети, которая связывает два маршрутизатора по двухточечной схеме, даже количество адресов сети класса С явно является избыточным.

На рисунке приведен другой пример распределения того же адресного пространства 129.44.0.0/16, что и в предыдущем примере. Здесь половина из имеющихся адресов (2^{15}) отведена для создания сети 1 имеющей адрес 129.44.0.0 и маску 255.255.128.0.

Следующая порция адресов, составляющая четверть всего адресного пространства (214), назначена для сети 2 129.44.128.0 с маской 255.255.192.0.
Далее в пространстве адресов был «вырезан» небольшой фрагмент для создания вспомогательной сети 3, предназначенной для связывания внутреннего маршрутизатора R2 с внешним маршрутизатором R1. Для нумерации узлов в такой вырожденной сети достаточно отвести два двоичных разряда. Из четырех возможных комбинаций номеров узлов: 00,01,10 и 11 два номера имеют специальное назначение и не могут быть присвоены узлам.



Поле номера узла в таком случае имеет два двоичных разряда, маска в десятичной нотации имеет вид 255.255.255.252, а номер сети, как видно из рисунка, равен 129.44.192.0.

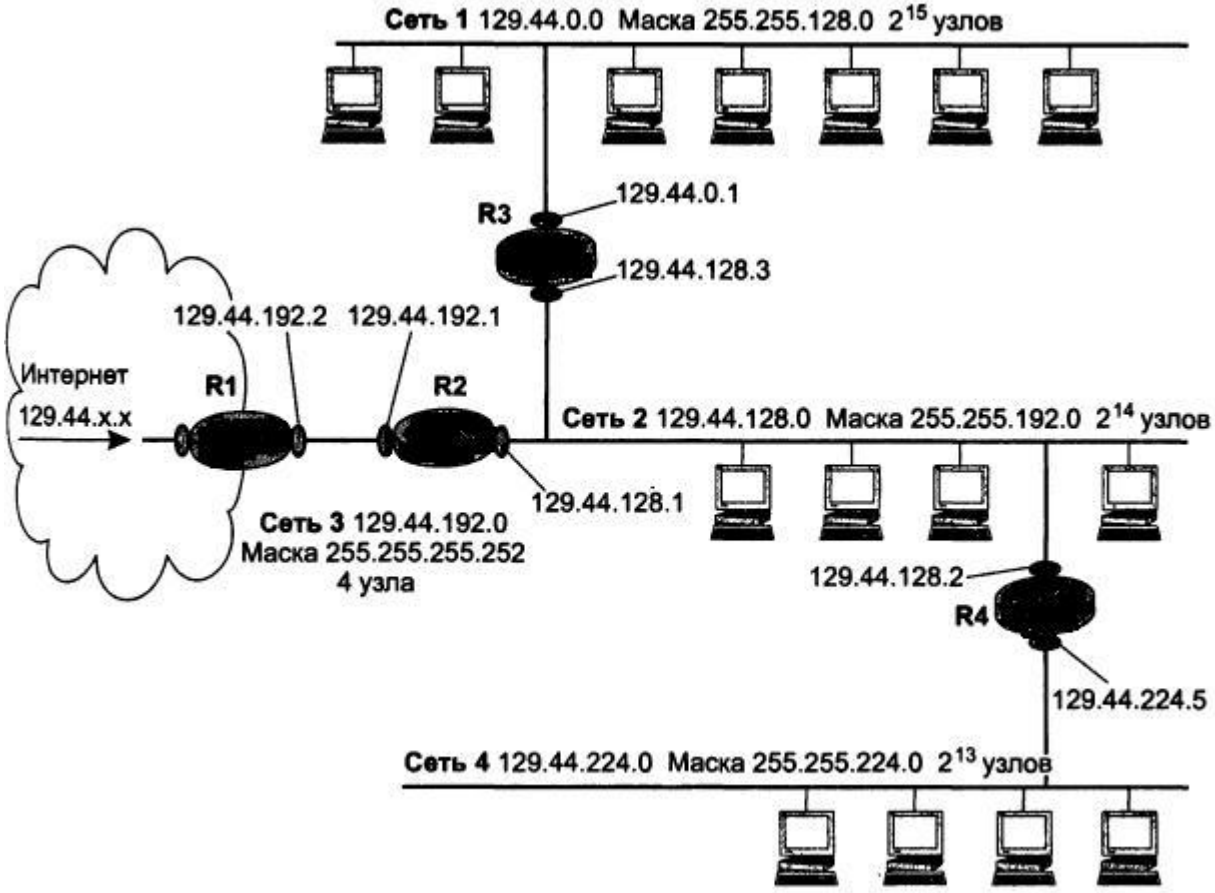
ПРИМЕЧАНИЕ

Глобальным связям между маршрутизаторами, соединенными по двухточечной схеме, не обязательно давать IP-адреса. Однако чаще всего такой вырожденной сети все же дают IP-адрес. Помимо прочего, это делается, например, для того, чтобы скрыть внутреннюю структуру сети и обращаться к ней по одному адресу входного порта маршрутизатора, в данном примере по адресу 129.44.192.1, применяя технику трансляции сетевых адресов (Network Address Translation, NAT1).

Оставшееся адресное пространство администратор может «нарезать» на разное количество сетей разного объема в зависимости от своих потребностей. Из оставшегося пула (2^{14} - 4) адресов администратор, например, может образовать еще одну достаточно большую сеть с числом узлов 2^{13}— на рисунке это сеть 4. При этом свободными останутся почти столько же адресов (2^{13}- 4), которые также могут быть использованы для создания новых сетей. К примеру, из этого «остатка» можно образовать 31 сеть, каждая из которых равна размеру сети класса С, и к тому же еще несколько сетей меньшего размера. Ясно, что разбиение может быть другим, но в любом случае с помощью масок переменного размера администратор имеет больше возможностей рационально использовать все имеющиеся у него адреса

Маршрутизация с использованием масок
Использование масок переменной длины

На рисунке показан пример сети, структурированной с помощью масок переменной длины.



Маршрутизация с использованием масок
Использование масок переменной длины

Процесс обработки R2 поступающих на его вход пакетов (см. табл. внизу).

Пусть поступивший на R2 пакет имеет адрес назначения 129.44.162.5. Поскольку специфические маршруты в таблице отсутствуют, маршрутизатор переходит ко второй фазе — фазе последовательного анализа строк на предмет поиска совпадения с адресом назначения:

- (129.44.162.5) AND (255.255.128.0) = 129.44.128.0 - нет совпадения;
- (129.44.162.5) AND (255.255.192.0) = ***129.44.128.0*** - совпадение;
- (129.44.162.5) AND (255.255.255.252) = 129.44.162.4 - нет совпадения;
- (129.44.162.5) AND (255.255.224.0) - 129.44.160.0 - нет совпадения.

Таким образом, совпадение имеет место в одной строке. Пакет будет отправлен в непосредственно подключенную к данному маршрутизатору сеть на выходной интерфейс 129.44.128.1.

Адрес назначения	Маска	Адрес следующего маршрутизатора	Адрес порта	Расстояние
129.44.0.0	255.255.128.0	129.44.128.3	129.44.128.1	1
129.44.128.0	255.255.192.0	129.44.128.1	129.44.128.1	Подключена
129.44.192.0	255.255.255.252	129.44.192.1	129.44.192.1	Подключена
129.44.224.0	255.255.224.0	129.44.128.2	129.44.128.1	1
0.0.0.0	0.0.0.0	129.44.192.2	129.44.192.1	-

Маршрутизация с использованием масок
Использование масок переменной длины

Если пакет с адресом 129.44.192.1 поступает из внешней сети и маршрутизатор R1 не использует маски, пакет передается маршрутизатору R2, а потом снова возвращается в соединительную сеть. Очевидно, что такие передачи пакета не выглядят рациональными.

Маршрутизация будет более эффективной, если в таблице маршрутизации маршрутизатора R1 задать маршруты масками переменной длины (табл. 2). Первая из приведенных двух записей говорит о том, что все пакеты, адреса которых начинаются с 129.44, должны быть переданы на маршрутизатор R2. Эта запись выполняет агрегирование адресов всех подсетей, созданных на базе одной сети 129.44.0.0. Вторая строка говорит о том, что среди всех возможных подсетей сети 129.44.0.0 есть одна (129.44.192.0/30), которой пакеты можно направлять непосредственно, а не через маршрутизатор R2.

Адрес назначения	Маска	Адрес следующего маршрутизатора	Адрес порта	Расстояние
129.44.0.0	255.255.0.0	129.44.192.1	129.44.191.2	2
129.44.192.0	255.255.255.252	129.44.192.2	129.44.191.2	Подключена

ПРИМЕЧАНИЕ

В IP-пакетах при использовании механизма масок по-прежнему передается только IP-адрес назначения, а маска сети назначения не передается. Поэтому из IP-адреса пришедшего пакета невозможно выяснить, какая часть адреса относится к номеру сети, а какая — к номеру узла. Если маски во всех подсетях имеют один размер, то это не создает проблем. Если же для образования подсетей применяют маски переменной длины, то маршрутизатор должен как-то узнавать, каким адресам сетей какие маски соответствуют. Для этого используются ***протоколы маршрутизации****, переносящие между маршрутизаторами не только служебную информацию об адресах сетей, но и о масках, соответствующих этим номерам. К таким протоколам относятся протоколы RIPv2 и OSPF, а вот, например, протокол RIP маски не переносит и для маршрутизации на основе масок переменной длины не подходит.*

Маршрутизация с использованием масок
Перекрытие адресных пространств

Со сложностями использования масок администратор впервые сталкивается не тогда, когда начинает конфигурировать сетевые интерфейсы и создавать таблицы маршрутизации, а гораздо раньше — на этапе планирования сети. Планирование включает определение количества сетей, из которых будет состоять корпоративная сеть, оценку требуемого количества адресов для каждой сети, получение пула адресов от поставщика услуг, распределение адресного пространства между сетями. Последняя задача часто оказывается нетривиальной, особенно когда решается в условиях дефицита адресов.

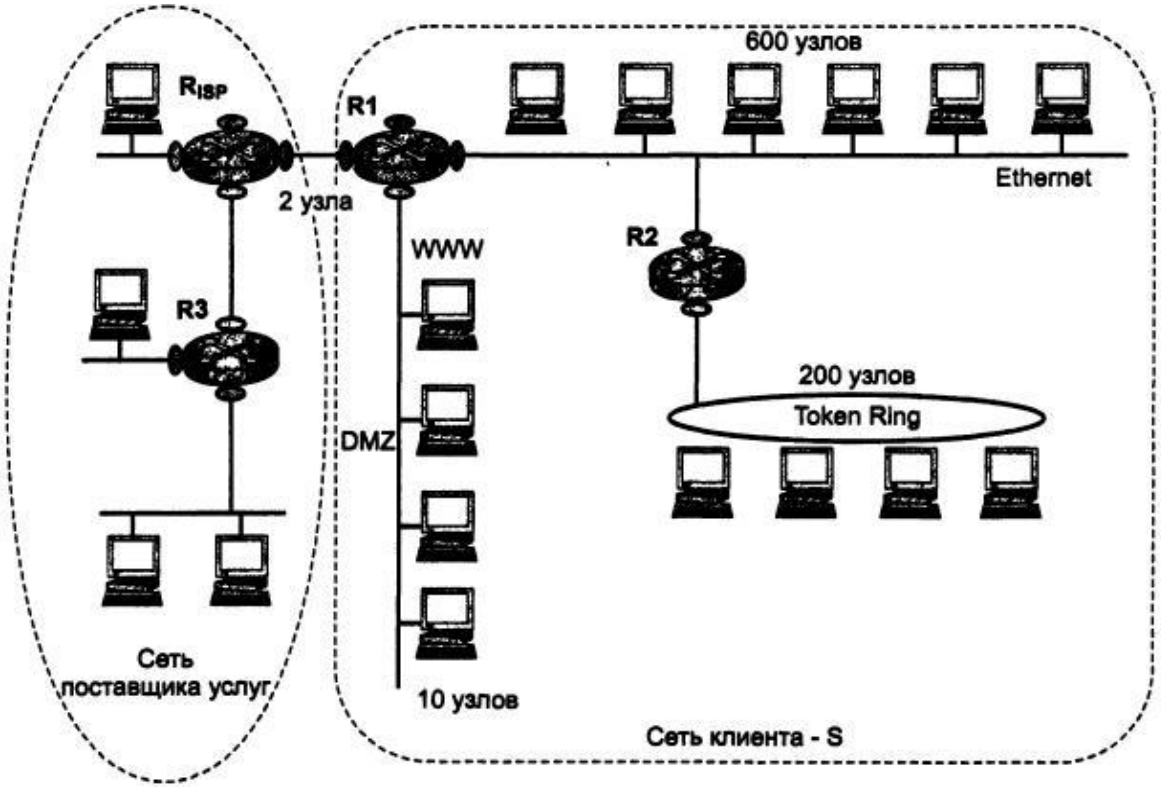
Рассмотрим пример использования масок для *организации перекрывающихся адресных пространств.*

Пусть на некотором предприятии было принято решение обратиться к поставщику услуг для получения пула адресов, достаточного для создания сети, структура, которой показана на рисунке ниже. Сеть клиента включает три подсети. Две из них — это надежно защищенные от внешних атак внутренние сети отделов: сеть Ethernet на 600 пользователей и сеть Token Ring на 200 пользователей. Предприятие также предусматривает отдельную, открытую для доступа извне сеть на 10 узлов, главное назначение которой — предоставление информации в режиме открытого доступа для потенциальных клиентов. Такого рода участки корпоративной сети, в которых располагаются веб-серверы, FTP-серверы и другие источники публичной информации, называют демилитаризованной зоной (Demilitarized Zone, DMZ). Еще одна сеть на два узла потребуется для связи с поставщиком услуг, то есть общее число адресов, требуемых для адресации сетевых интерфейсов, составляет 812. Кроме того, необходимо, чтобы пул доступных адресов включал для каждой из сетей широковещательные адреса, состоящие только из единиц, а также адреса, состоящие только из нулей.

Маршрутизация с использованием масок
Перекрытие адресных пространств

Учитывая также, что в любой сети адреса всех узлов должны иметь одинаковые префиксы, становится очевидным, что минимальное количество адресов, необходимое клиенту для построения задуманной сети, может значительно отличаться от значения 812, полученного простым суммированием.

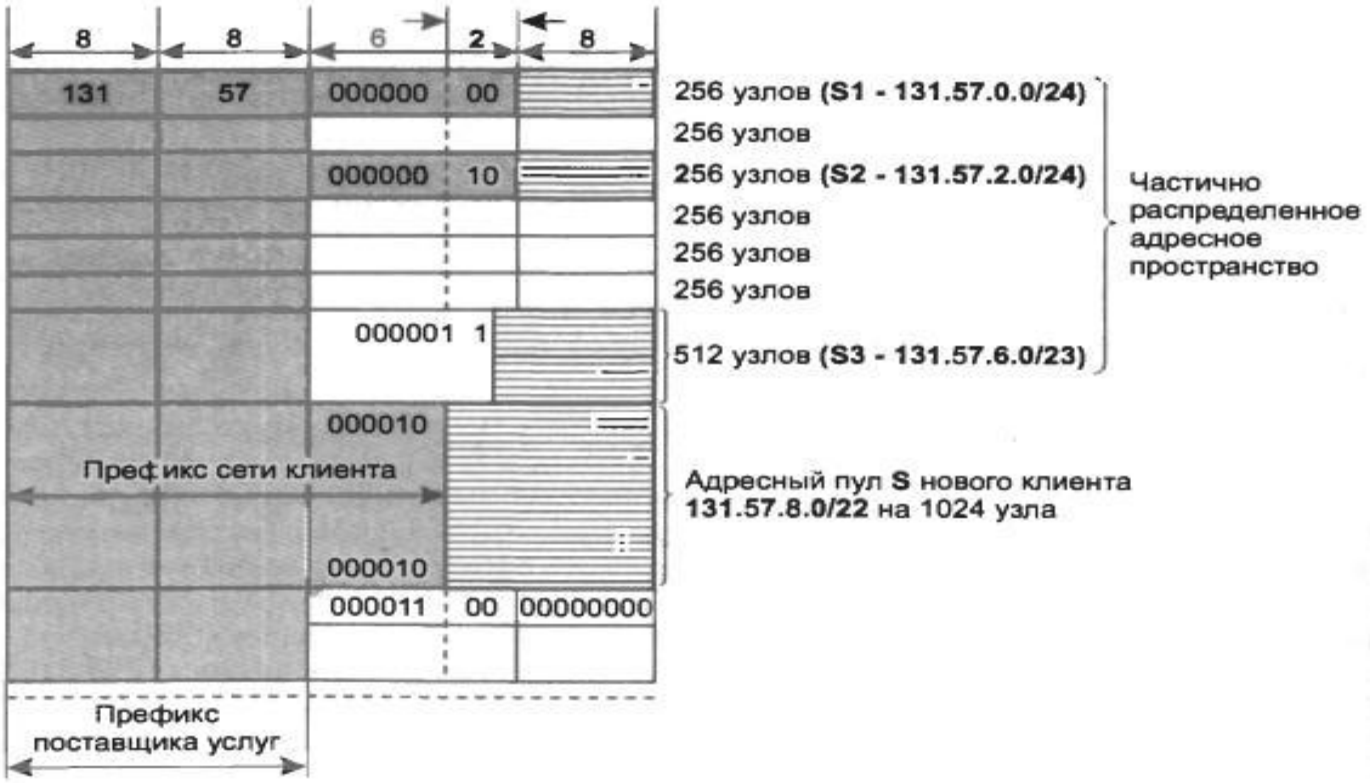
В данном примере поставщик услуг решает выделить клиенту непрерывный пул из 1024 адресов. Значение 1024 выбрано как наиболее близкое к требуемому количеству адресов, равному степени двойки (2^{10} = 1024).



Маршрутизация с использованием масок
Перекрытие адресных пространств

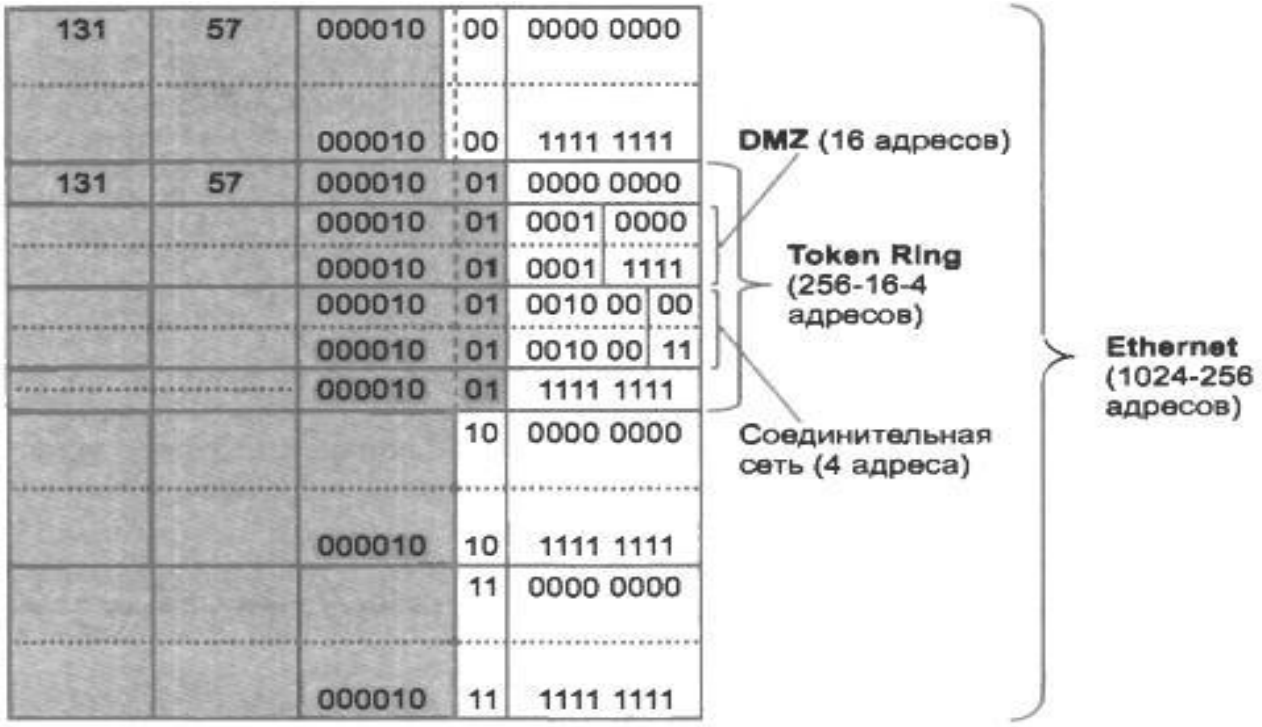
Поставщик услуг выполняет поиск области такого размера в имеющемся у него адресном пространстве — 131.57.0.0/16, часть которого, как показано на рисунке, уже распределена. Обозначим распределенные участки и владеющих ими клиентов через SI, S2 и S3. Поставщик услуг находит среди нераспределенных еще адресов непрерывный участок размером 1024 адреса, начальный адрес которого кратен размеру данного участка. Таким образом, наш клиент получает пул адресов 131.57.8.0/22, обозначенный на рисунке через S.

Далее начинается самый сложный этап — распределение полученного от поставщика услуг адресного пула S между четырьмя сетями клиента.



Маршрутизация с использованием масок
Перекрытие адресных пространств

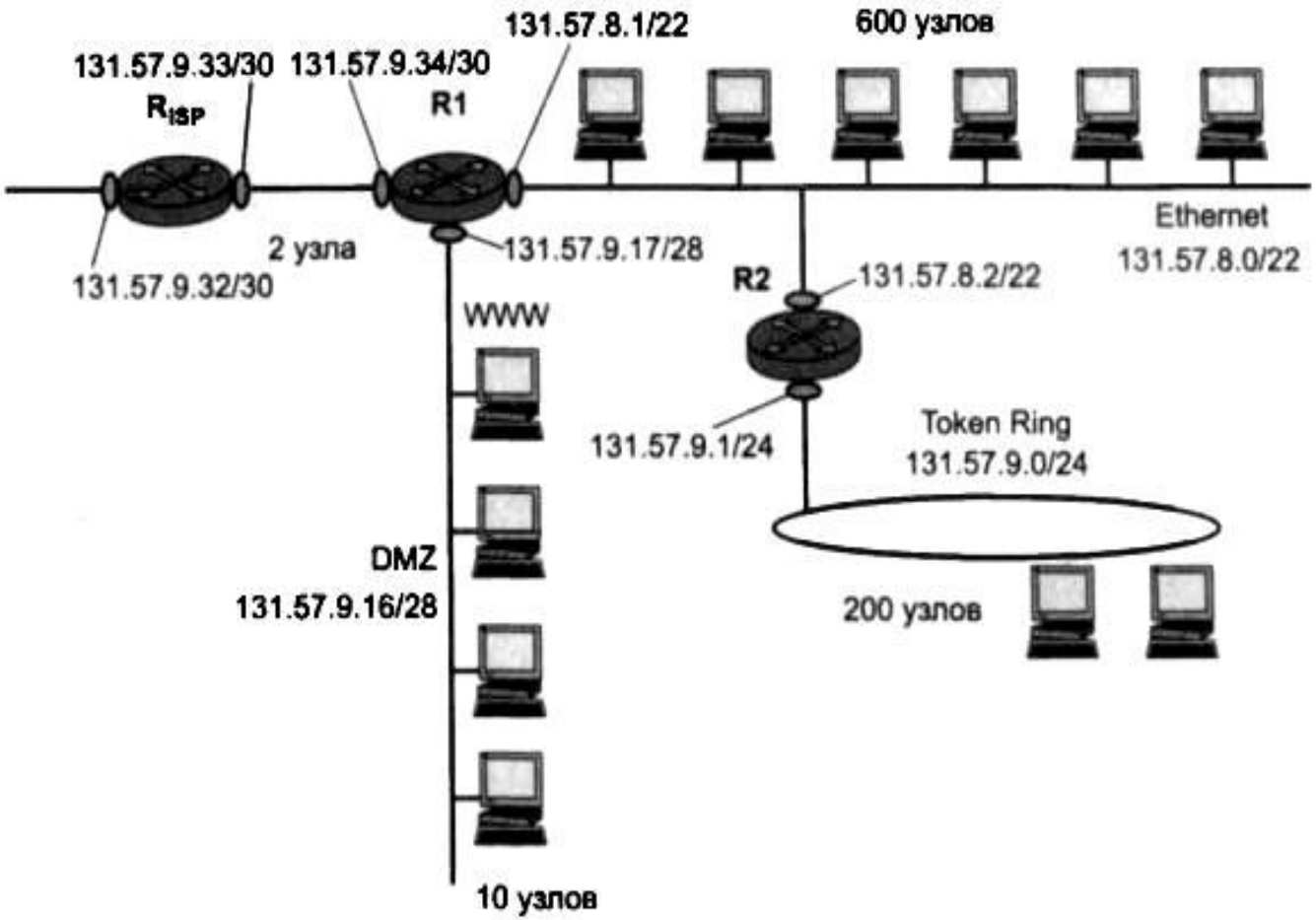
Прежде всего, администратор решил назначить для самой большой сети (Ethernet на 600 узлов) весь пул адресов 131.57.8.0/22, полученный от поставщика услуг (рис. ниже). Номер, назначенный для этой сети, совпадает с номером сети, полученным от поставщика услуг. А как же быть с оставшимися тремя сетями? Администратор учел, что для сети Ethernet требуется только 600 адресов, а из оставшихся 624 «выкроил» сеть Token Ring 131.57.9.0/24 на 250 адресов. Воспользовавшись тем, что для Token Ring требуется только 200 адресов, он «вырезал» из нее два участка: для сети DMZ 131.57.9.16/28 на 16 адресов и для связывающей сети 131.57.9.32/30 на 4 адреса. В результате все сети клиента получили достаточное (а иногда и с избытком) количество адресов.



Маршрутизация с использованием масок
Перекрытие адресных пространств

Следующий этап — это конфигурирование сетевых интерфейсов конечных узлов и маршрутизаторов. Каждому интерфейсу сообщается его IP-адрес и соответствующая маска.

На рисунке показана сконфигурированная сеть клиента.



Сеть клиента - S

Маршрутизация с использованием масок
Перекрытие адресных пространств

После конфигурирования сетевых интерфейсов должны быть созданы таблицы маршрутизации маршрутизаторов R1 и R2 клиента. Они могут быть сгенерированы автоматически или с участием администратора. Таблица маршрутизации маршрутизатора R2 соответствует табл. 1.

Таблица 1. Таблица маршрутизации маршрутизатора R2

Адрес назначения	Маска	Адре с следующего маршрутизатора	Адрес выходного интерфейса	Расстояние
131.57.8.0	255.255.252.0	131.57.8.2	131.57.8. 2	Подключена
131.57.9.0	255.255.255.0	131.57.9.1	131.57.9. 1	Подключена
131.57.9.16	255.255.255.240	131.57.8.1	131.57.8. 2	1
131.57.9.32	255.255.255.252	131.57.8.1	131.57.8. 2	1

В данной таблице нет маршрута по умолчанию, а значит, все пакеты, адресованные сетям, адреса которых явно не указаны в таблице, будут отбрасываться маршрутизатором.

Маршрутизация с использованием масок
Перекрытие адресных пространств

Пусть, например, на маршрутизатор R2 поступает пакет с адресом назначения 131.57.9.29. В результате просмотра таблицы получаем следующие результаты для каждой строки:

- (131.57.9.29) AND (255.255.252.0) - 131.57.8.0 - совпадение;
- (131.57.9.29) AND (255.255.255.0) - 131.57.9.0 - совпадение;
- (131.57.9.29) AND (255.255.255.240) - ***131.57.9.16*** - совпадение;
- (131.57.9.29) AND (255.255.255.252) - 131.57.9.28 - нет совпадения.

Поскольку при наличии нескольких совпадений выбирается маршрут из той строки, в которой совпадение адреса назначения с адресом из пакета имеет ***наибольшую длину***, определено, что пакет с адресом 131.57.9.29 направляется в сеть DMZ.

Адрес назначения	Маска	Адре с следующего маршрутизатора	Адрес выходного интерфейса	Расстояние
131.57.8.0	255.255.252.0	131.57.8.2	131.57.8. 2	Подключена
131.57.9.0	255.255.255.0	131.57.9.1	131.57.9. 1	Подключена
131.57.9.16	255.255.255.240	131.57.8.1	131.57.8. 2	1
131.57.9.32	255.255.255.252	131.57.8.1	131.57.8. 2	1

Маршрутизация с использованием масок
Бесклассовая междоменная маршрутизация

За последние несколько лет в Интернете многое изменилось: резко возросло число; и сетей, повысилась интенсивность трафика, изменился характер передаваемых данных. Из-за несовершенства протоколов маршрутизации обмен сообщениями об обновлении таблиц стал приводить к сбоям магистральных маршрутизаторов, происходящим из-за перегрузок при обработке большого объема служебной информации. Так, сегодня таблицы магистральных маршрутизаторов в Интернете могут содержать до нескольких сотен и даже тысяч маршрутов.

На решение этой проблемы направлена технология **бесклассовой междоменной маршрутизации** (Classless Inter-Domain Routing, CIDR).

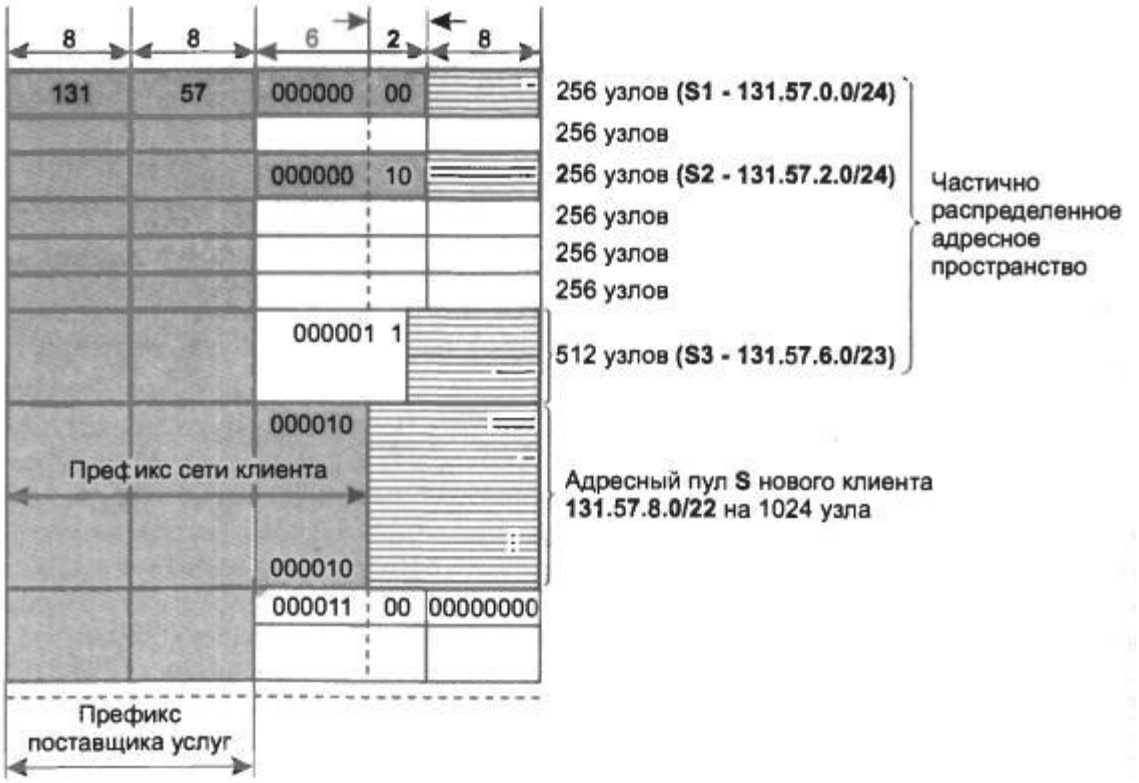
*Суть технологии CIDR заключается в следующем. Каждому поставщику услуг Интернета назначается непрерывный диапазон IP-адресов. При таком подходе все адреса каждого поставщика услуг имеют общую старшую часть — **префикс**, поэтому маршрутизация на магистралях Интернета может осуществляться на основе префиксов, а не полных адресов сетей. А это значит, что вместо множества записей по числу сетей будет достаточно поместить одну запись сразу для всех сетей, имеющих общий префикс. Такое агрегирование адресов позволит уменьшить объем таблиц в маршрутизаторах всех уровней, а следовательно, ускорить работу маршрутизаторов и повысить пропускную способность Интернета.*

Ранее мы рассматривали примеры, где администраторы корпоративных сетей с помощью масок делили на несколько частей непрерывный пул адресов, полученный от поставщика услуг, чтобы использовать эти части для структуризации своей сети. Такой вариант применения масок называется *разделением на подсети*.

Маршрутизация с использованием масок
Бесклассовая междоменная маршрутизация

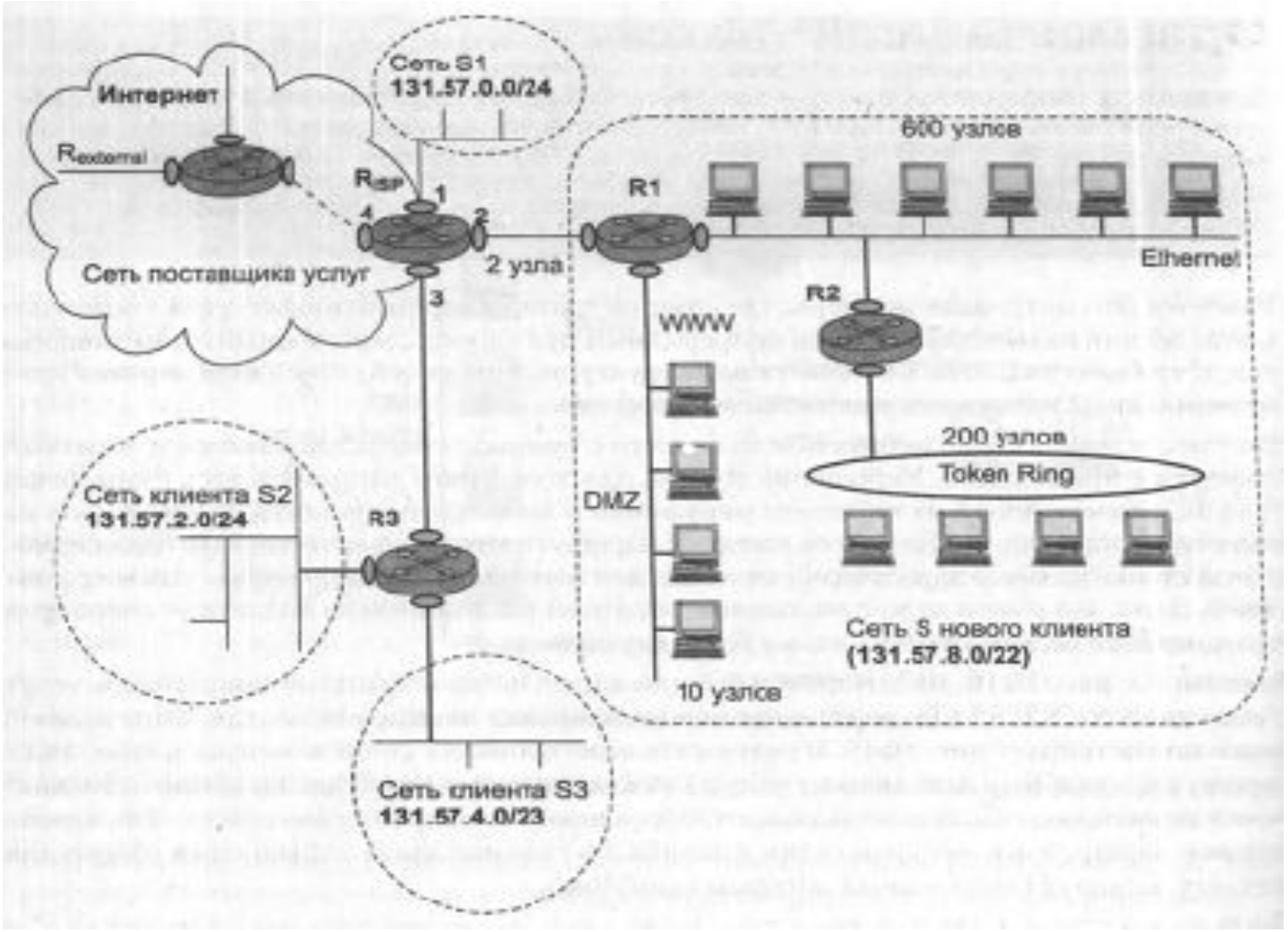
Вместе с тем в процессе разделения на подсети с помощью масок проявлялся и обратный эффект их применения. Упрощенно говоря, для того чтобы направить весь суммарный трафик, адресованный из внешнего окружения в корпоративную сеть, разделенную на подсети, достаточно, чтобы во всех внешних маршрутизаторах наличествовала одна строка. В этой строке на месте адреса назначения должен быть указан общий префикс для всех этих сетей. Здесь мы имеем дело с операцией, обратной разделению на подсети — операцией агрегирования несколько сетей в одну более крупную.

Вернемся к рисунку, на котором показано адресное пространство поставщика услуг с участками SI, S2, S3 и S, переданными в пользование четырем клиентам.



Маршрутизация с использованием масок
Бесклассовая междоменная маршрутизация

Этот пример также иллюстрирует рисунок ниже.



Маршрутизация с использованием масок
Бесклассовая междоменная маршрутизация

В результате агрегирования сетей клиентов в табл. 1 маршрутизатора R_{jsp} поставщика услуг для каждого клиента будет выделено по одной строке независимо от количества подсетей, организованных ими в своих сетях. Так, вместо четырех маршрутов к четырем сетям клиента S в таблице задан только один **общий** для всех них маршрут.

Таблица 1. Таблица маршрутизатора R_{isp} поставщика услуг

Адрес назначения	Маска	Следующий маршрутизатор	Номер выходного интерфейса	Расстояние Подключена
131.57 0.0 (S1)	255.255.255.0	R3	1	1
131.57 2.0 (S2)	255.255.255.0	R3	3	1
131.57 4. 0(S3)	255.255.254.0	R1	3	Подключена
131.57.8.0 (S)	**255.255.252.0**	**R1**	**2**	
Маршрут по умолчанию	0.0.0.0	$R_{external}$	4	-

Итак, внедрение технологии CIDR позволяет решить две основные задачи.

- Более экономное расходование адресного пространства. Благодаря технологии CIDR поставщики услуг получают возможность «нарезать» блоки из выделенного им адресного пространства в точном соответствии с требованиями каждого клиента, при этом у клиента остается пространство для маневра на случай будущего роста.
- Уменьшение числа записей в таблицах маршрутизации за счет объединения маршрутов — одна запись в таблице маршрутизации может представлять большое количество сетей. Если все поставщики услуг Интернета начнут придерживаться стратегии CIDR, то особенно заметный выигрыш будет достигаться в магистральных маршрутизаторах.

Маршрутизация с использованием масок
Бесклассовая междоменная маршрутизация

Необходимым условием эффективного использования технологии CIDR является локализация адресов, то есть назначение адресов, имеющих совпадающие префиксы, сетям, располагающимся территориально по соседству. Только в таком случае трафик может быть агрегирован.

К сожалению, сейчас распределение адресов носит во многом случайный характер. Кардинальный путь решения проблемы — перенумерование сетей. Однако эта процедура сопряжена с определенными временными и материальными затратами, и для ее проведения пользователей нужно каким-либо образом стимулировать. В качестве таких стимулов рассматривается, например, введение оплаты за строку в таблице маршрутизации или же количество узлов в сети. Первое требование подводит потребителя к мысли получить у поставщика услуг такой адрес, чтобы маршрутизация трафика в его сеть шла на основании префикса, и номер его сети не фигурировал больше в магистральных маршрутизаторах. Требование оплаты каждого адреса узла также может подтолкнуть пользователя решиться на перенумерование с тем, чтобы получить ровно столько адресов, сколько ему нужно.

Технология CIDR уже успешно используется в текущей версии протокола IP (IPv4) и поддерживается такими протоколами маршрутизации, как OSPF, RIP-2, BGP4 (в основном на магистральных маршрутизаторах Интернета).

Фрагментация IP-пакетов

Важной особенностью протокола IP, отличающей его от других сетевых протоколов (например, от сетевого протокола IPX, который какое-то время назад конкурировал с IP), является его способность выполнять динамическую фрагментацию пакетов при передаче их между сетями с различными максимально допустимыми значениями длины поля данных кадров (Maximum Transmission Unit, MTU). Значения MTU зависят как от протокола, так и от настройки сетевых интерфейсов.

Прежде всего отметим разницу между фрагментацией сообщений в узле-отправителе и динамической фрагментацией сообщений в транзитных узлах сети — маршрутизаторах.

В первом случае деление сообщения на несколько более мелких частей (фрагментация) происходит при передаче данных между протоколами одного и того же стека внутри компьютера. Протоколы, выполняющие фрагментацию в пределах узла, анализируют тип технологии нижнего уровня, определяют ее MTU и делят сообщения на такие части, которые умещаются в кадры канального уровня того же стека протоколов.

В стеке TCP/IP эту задачу решает протокол TCP, который разбивает поток байтов, передаваемый ему с прикладного уровня, на сегменты нужного размера, например, по 1460 байт, если на нижнем уровне данной сети работает протокол Ethernet. Протокол IP в узле-отправителе, как правило, не использует свои возможности по фрагментации пакетов.

А вот на транзитном узле — маршрутизаторе, когда пакет необходимо передать из сети с большим значением MTU в сеть с меньшим значением MTU, способности протокола IP выполнять фрагментацию становятся востребованными. Пакеты-фрагменты, путешествуя по сети, могут вторично подвергнуться фрагментации на каком-либо из промежуточных маршрутизаторов.

Фрагментация IP-пакетов
Параметры фрагментации

Каждый из фрагментов должен быть снабжен полноценным заголовком IP. Некоторые из полей заголовка (идентификатор, TTL, флаги DF и MF, смещение) непосредственно предназначены для последующей сборки фрагментов в исходное сообщение.

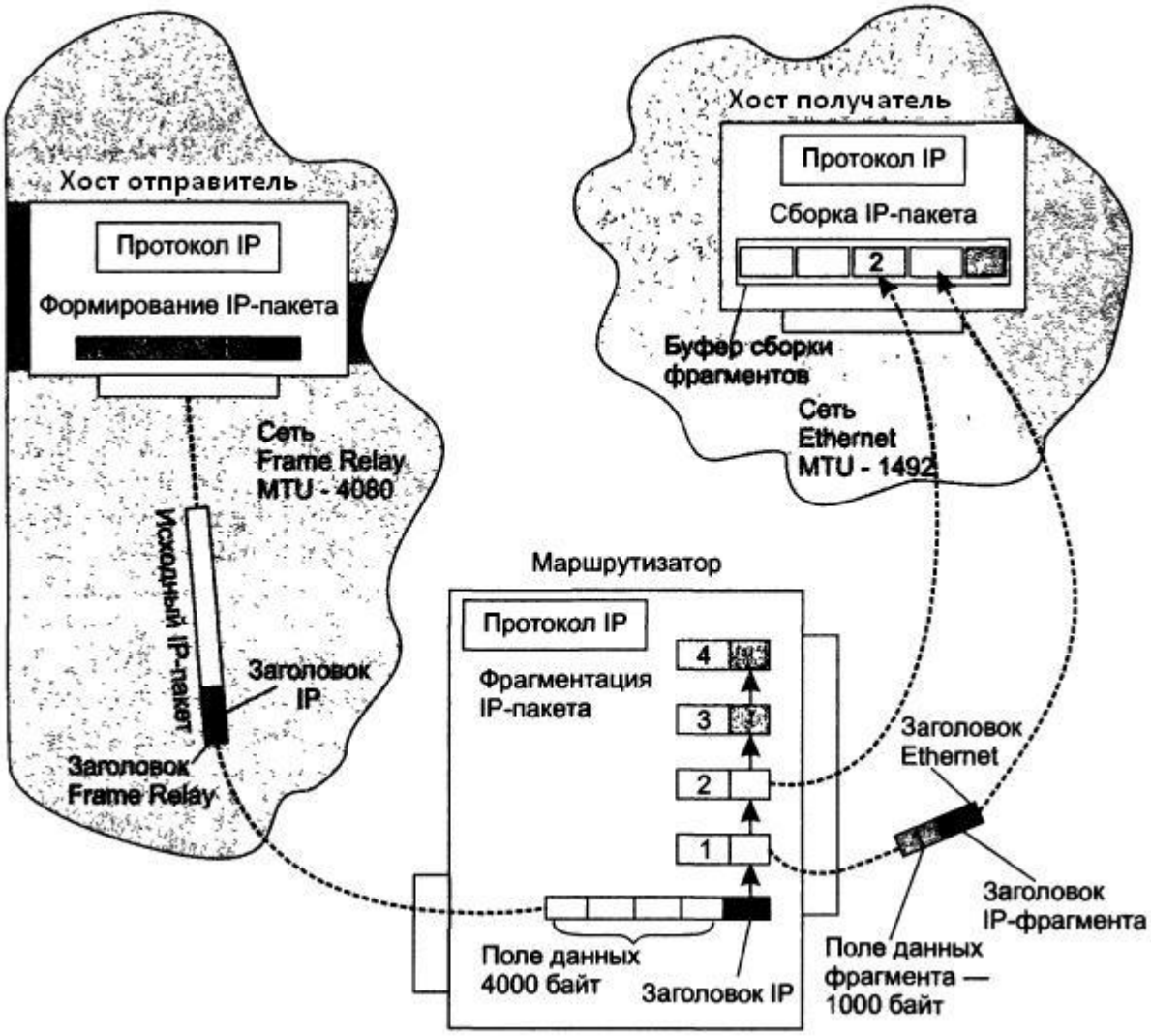
- Идентификатор пакета используется для распознавания пакетов, образовавшихся путем деления на части (фрагментации) исходного пакета. Все части (фрагменты) одного пакета должны иметь одинаковое значение этого поля. Модуль IP, отправляющий пакет, устанавливает в поле идентификатора значение, которое должно быть уникальным для данной пары отправителя и получателя в течение всего времени, пока данный пакет (или любой его фрагмент) может существовать в составной IP-сети.
- Поле времени жизни (Time To Live, TTL) занимает один байт и определяет предельный срок, в течение которого пакет может перемещаться по сети. Время жизни пакета измеряется в секундах и задается источником (отправителем). Как уже отмечалось в начале этой главы, по истечении каждой секунды пребывания на каждом из маршрутизаторов, через которые проходит пакет во время своего «путешествия» по сети, из его текущего времени жизни вычитается единица; единица вычитается и в том случае, если время пребывания было меньше секунды. Поскольку современные маршрутизаторы редко обрабатывают пакет дольше, чем за одну секунду, то время жизни можно интерпретировать как максимальное число транзитных узлов, которые разрешено пройти пакету. Если значение поля времени жизни становится нулевым до того, как пакет достигает получателя, пакет уничтожается. При сборке фрагментов хост-получатель использует значение TTL как крайний срок ожидания недостающих фрагментов.

Фрагментация IP-пакетов
Параметры фрагментации

- Поле смещения фрагмента предоставляет получателю информацию о положении фрагмента относительно начала поля данных исходного нефрагментированного пакета. Так, например, первый фрагмент будет иметь в поле смещения нулевое значение. В пакете, не разбитом на фрагменты, поле смещения также имеет нулевое значение. Смещение задается в байтах и должно быть кратно 8 байт.
- Установленный в единицу однобитный флаг MF (More Fragments — больше фрагментов) говорит о том, что данный пакет является промежуточным (не последним) фрагментом. Модуль IP, отправляющий нефрагментированный пакет, устанавливает бит MF в нуль.
- Флаг DF (Do not Fragment — не фрагментировать), установленный в единицу, запрещает маршрутизатору фрагментировать данный пакет. Если помеченный таким образом пакет не может достигнуть получателя без фрагментации, то модуль IP его уничтожает, а узлу-отправителю посылается диагностическое сообщение.

Фрагментация IP-пакетов
Механизм фрагментации

Рассмотрим механизм фрагментации на примере составной сети, показанной на рисунке.



Фрагментация IP-пакетов
Механизм фрагментации

В одной из подсетей (Frame Relay) значение MTU равно 4080, в другой (Ethernet) — 1492. Хост, принадлежащий сети Frame Relay, передает данные хосту в сети Ethernet. На обоих хостах, а также на маршрутизаторе, связывающем эти подсети, установлен стек протоколов TCP/IP.

Транспортному уровню хоста-отправителя известно значение MTU нижележащей технологии (4080). На основании этого модуль TCP и «нарезает» свои сегменты размером 4000 байт и передает вниз протоколу IP, который помещает сегменты в поле данных IP-пакетов и генерирует для них заголовки. Обратим особое внимание на заполнение тех полей заголовка, которые прямо связаны с фрагментацией:

- пакету присваивается уникальный идентификатор, например 12456;
- поскольку пакет пока еще не был фрагментирован, в поле смещения помещается значение 0;
- признак MF также обнуляется, это показывает, что пакет одновременно является и своим последним фрагментом;
- признак DF устанавливается в 1, это означает, что данный пакет можно фрагментировать.

Общая величина IP-пакета составляет 4000 плюс 20 (размер заголовка IP), то есть 4020 байт, что умещается в поле данных кадра Frame Relay, которое в данном примере равно 4080. Далее модуль IP хоста-отправителя передает этот кадр своему сетевому интерфейсу Frame Relay, который отправляет кадры следующему маршрутизатору.

Модуль IP маршрутизатора по сетевому адресу прибывшего IP-пакета определяет, что пакет нужно передать в сеть Ethernet. Однако она имеет значение MTU, равное 1492, что значительно меньше размера поступившего на входной интерфейс пакета. Следовательно, IP-пакет необходимо фрагментировать.

Фрагментация IP-пакетов
Механизм фрагментации

Модуль IP выбирает размер поля данных фрагмента равным 1000, так что из одного большого IP-пакета получается 4 маленьких пакета-фрагмента. Для каждого фрагмента и его заголовка IP в маршрутизаторе создается отдельный буфер (на рисунке фрагменты и соответствующие им буферы пронумерованы от 1 до 4). Протокол IP копирует в эти буферы содержимое некоторых полей заголовка IP исходного пакета, создавая тем самым «заготовки» заголовков IP всех новых пакетов-фрагментов. Одни параметры заголовка IP копируются в заголовки всех фрагментов, другие — лишь в заголовок, первого фрагмента.

В процессе фрагментации могут измениться значения некоторых полей заголовков IP в пакетах-фрагментах по сравнению с заголовком IP исходного пакета. Так, каждый фрагмент имеет собственные значения контрольной суммы заголовка, смещения фрагмента и общей длины пакета. Во всех пакетах, кроме последнего, флаг MF устанавливается в единицу, а в последнем фрагменте — в нуль. Полученные пакеты-фрагменты имеют длину 1020 байт (с учетом заголовка IP), поэтому они свободно помещаются в поле данных.

На рисунке показаны разные стадии перемещения фрагментов по сети. Фрагмент 2 уже достиг хоста-получателя и помещен в приемный буфер. Фрагмент 1 еще перемещается по сети Ethernet, остальные фрагменты находятся в буферах маршрутизатора.

А теперь обсудим, как происходит сборка фрагментированного пакета на хосте назначения.

На хосте назначения для каждого фрагментированного пакета отводится отдельный буфер. В этот буфер принимающий протокол IP помещает IP-фрагменты, у которых совпадают IP-адреса отправителя и получателя, а также значения в полях идентификатора (в нашем примере — 12456). Все эти признаки говорят модулю IP, что данные пакеты являются фрагментами одного исходного пакета. Сборка заключается в помещении данных из каждого фрагмента в позицию, определенную смещением, указанным в заголовке фрагмента.

Когда первый фрагмент исходного пакета приходит на хост-получатель, этот хост запускает таймер, который определяет максимальное время ожидания прибытия остальных фрагментов данного пакета. В различных реализациях IP применяются разные правила выбора максимального времени ожидания. В частности, таймер может быть установлен на фиксированный период времени (от 60 до 120 секунд), рекомендуемый RFC. Как правило, этот интервал достаточен для доставки пакета от отправителя получателю. В других реализациях максимальное время ожидания определяется с помощью адаптивных алгоритмов измерения и статистической обработки временных параметров сети, позволяющих оценивать ожидаемое время прибытия фрагментов. Наконец, тайм-аут может быть выбран на базе значений TTL прибывающих фрагментов. Последний подход основан на том, что нет смысла ожидать, пока прибудут другие фрагменты пакета, если время жизни одного из прибывших фрагментов уже истекло.