**МИНОБРНАУКИ РОССИИ**

Федеральное государственное бюджетное образовательное учреждение

высшего образования

**«Гжельский государственный университет»** (ГГУ)

Колледж ГГУ

Специальность 09.02.07. Информационные системы и программирование.

**Реферат**

по предмету « КС»

на тему «Типы адресов стека TCP/IP».

ВЫПОЛНИЛ:

Студент группы ИСП-О-17

Белкин И.В.

ПРОВЕРИЛА:

Прокуронова А.Ю.

Оценка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

п. Электроизолятор

2019 г.

**Введение**

Стек протоколов TCP/IP тесно связан с сетью Internet, ее историей и современностью. Создан он был в 1969 году, когда для сети ARPANET понадобился ряд стандартов для объединения в единую сеть компьютеров с различными архитектурами и операционными системами. На базе этих стандартов и был разработан набор протоколов, получивших название TCP/IP. Вместе с ростом Internet протокол TCP/IP завоевывал позиции и в других сетях. На сегодняшний день этот сетевой протокол используется как для связи компьютеров всемирной сети, так и в подавляющем большинстве корпоративных сетей. В наши дни используется версия протокола IP, известная как IPv4. В статье мы рассмотрим стандартную схему адресации и более новые методы рационального использования адресного пространства, введенные в результате обнаруженных недостатков в реализации протокола IP.

**1. Адресация протокола IP**

Согласно спецификации протокола, каждому узлу, подсоединенному к IP-сети, присваивается уникальный номер. Узел может представлять собой компьютер, маршрутизатор, межсетевой экран и др. Если один узел имеет несколько физических подключений к сети, то каждому подключению должен быть присвоен свой уникальный номер. Этот номер, или по-другому IP-адрес, имеет длину в четыре октета, и состоит из двух частей. Первая часть определяет сеть, к которой принадлежит узел, а вторая -- уникальный адрес самого узла внутри сети. В классической реализации протокола первую часть адреса называли "сетевым префиксом", поскольку она однозначно определяла сеть. Однако в современной реализации это уже не так и сеть идентифицируют другим образом, ниже речь пойдет о классической адресной схеме протокола ip.

Изначально все адресное пространство разделили на пять классов: A, B, C, D и Е. Такая схема получила название "классовой". Каждый класс однозначно идентифицировался первыми битами левого байта адреса. Сами же классы отличались размерами сетевой и узловой частей. Зная класс адреса, вы могли определить границу между его сетевой и узловой частями. Кроме того, такая схема позволяла при маршрутизации не передавать вместе с пакетом информацию о длине сетевой части IP-адреса.

Таблица 1-Иеархическая схема протоколов IP

|  |
| --- |
| Класс А |
| Номер бита | 0 | 8 | 16 | 24 31 |
| Адрес | 0....... | ........ | ........ | ........ |
| Сетевая часть |  |  |  |  |
| Класс В |  |  |  |  |
| Номер бита | 0 | 8 | 16 | 24 31 |
| Адрес | 10...... | ........ | ........ | ........ |
| Сетевая часть |  |  |  |  |
| Класс С |  |  |  |  |
| Номер бита | 0 | 8 | 16 | 24 31 |
| Адрес | 110..... | ........ | ........ | ........ |
| Сетевая часть |  |  |  |  |
| Класс D |  |  |  |  |
| Номер бита | 0 | 8 | 16 | 24 31 |
| Адрес | 1110.... | ........ | ........ | ........ |
| Класс E |  |  |  |  |
| Номер бита | 0 | 8 | 16 | 24 31 |
| Адрес | 1111.... | ........ | ........ | ........ |

Класс А ориентирован на очень большие сети. Все адреса, принадлежащие этому классу, имеют 8-битный сетевой префикс, на что указывает первый бит левого байта адреса установленный в нуль. Соответственно, на идентификацию узла отведено 24 бита и каждая сеть "восьмерка" может содержать до 224-2 узлов. Два адреса необходимо отнять, поскольку адреса, содержащие в правом октете все нули (идентифицирует указанную сеть) и все единицы (широковещательный адрес) используются в служебных целях и не могут быть присвоены узлам. Самих же сетей "восьмерок" может быть 27-2. Снова мы вычитаем двойку, но это уже две служебных сети: 127/8 и 0/8 (по-старому: 127.0.0.0 и 0.0.0.0). Наконец, можно заметить, что класс А содержит всего 27 \* 224 = 231 адресов, или половину всех возможных IP-адресов. Класс В предназначен для сетей большого и среднего размеров. Адреса этого класса идентифицируются двумя старшими битами, равными соответственно 1 и 0. Сетевой префикс класса состоит из шестнадцати бит или первых двух октетов адреса. Поскольку два первых бита сетевого префикса заняты определяющим класс ключом, то можно задать лишь 214 различных сетей. Узлов же в каждой сети можно определить до 216-2. В некоторых источниках, для определения количества возможных сетей используется формула 2х-2 для всех классов, а не только для А. Это связано с определенными причинами, которые более детально будут изложены ниже. На сегодняшний день нет никакой необходимости уменьшать количество возможных сетей на две. Проведя вычисления, аналогичные приведенным для класса А, мы увидим, что класс В занимает четверть адресного пространства протокола IPНаконец, самый употребляемый класс сетей – класс С – имеет 24 битный сетевой префикс, определяется старшими битами, установленными в 110, и может идентифицировать до 221 сетей. Соответственно, класс позволяет адресовать до 28-2 узлов. Занимает восьмую часть адресного пространства протокола TCP/IP. Последние два класса занимают оставшуюся восьмую часть в адресном пространстве и предназначены для служебного (класс D) и экспериментального (класс Е) использования. Для класса D старшие четыре бита адреса установлены в 1110, для класса Е -- 1111. Сегодня класс D используется для групповой передачи информации. Поскольку длинные последовательности из единиц и нулей трудно запомнить, IP адреса обычно записывают в десятичной форме. Для этого каждый октет адреса представляется в виде десятичного числа. Между собой октеты отделяются точкой. Иногда октеты обозначаются как w.x.y.z и называются "z-октет", "y-октет", "x-октет" и "w-октет". Представление IP-адреса в виде четырех десятичных чисел разделенных точками и называется "точечно-десятичная нотация".

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Октет | W | X | Y | Z |
| Номер бита | 0 | 8 | 16 | 24 31 |
| Адрес | 11011100 | 11010111 | 00001110 | 00010110 |
| 220 | 215 | 14 | 22 |  |
| Точечно- десятичный формат | 220.215.14.22 | | | |

Рисунок 1 - IP в точечно-десятичной нотации

На рис. 1 показано, как IP-адрес представляется в точечно-десятичной нотации.

Подытожим информацию о классах сетей в таблице:

Таблица 2- Классовая сеть

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Класс | Количество сетей | Количество узлов | Десятичный диапазон | |
| A | 27 – 2 (126) | 224 – 2 (2 147 483 648) | 1.ххх.ххх.ххх | 126.ххх.ххх.ххх |
| B | 214 (16 384) | 216 – 2 (65 534) | 128.0.ххх.ххх | 191.255.ххх.ххх |
| C | 221 (2 097 152) | 28 – 2 (254) | 192.0.0.ххх | 223.255.255.ххх |
| D | - | - | 224.0.0.ххх | 239.255.255.ххх |
| E | - | - | 240.0.0.ххх | 254.255.255.ххх |

**1.1 Соглашение о специальных: : broadcast, multicast, loopback**

В протоколе IP существует несколько соглашений об особой интерпретации IP-адресов:

если IР-адрес состоит только из двоичных нулей,

|  |
| --- |
| 0 0 0 0 ................................... 0 0 0 0 |

то он обозначает адрес того узла, который сгенерировал этот пакет;

если в поле номера сети стоят 0,

|  |  |
| --- | --- |
| 0 0 0 0 .......0 | Номер узла |

то по умолчанию считается, что этот узел принадлежит той же самой сети, что и узел, который отправил пакет;

если все двоичные разряды IP-адреса равны 1, то пакет с таким адресом назначения должен рассылаться всем узлам, находящимся в той же сети, что и источник этого пакета.

|  |
| --- |
| 1 1 1 1 .........................................1 1 |

Такая рассылка называется ограниченным широковещательным сообщением (limited broadcast);

если в поле адреса назначения стоят сплошные 1,

|  |  |
| --- | --- |
| Номер сети | 1111................11 |

то пакет, имеющий такой адрес рассылается всем узлам сети с заданным номером. Такая рассылка называется широковещательным сообщением (broadcast); адрес 127.0.0.1 зарезервирован для организации обратной связи при тестировании работы программного обеспечения узла без реальной отправки пакета по сети. Этот адрес имеет название loopback.

Уже упоминавшаяся форма группового IP-адреса - multicast - означает, что данный пакет должен быть доставлен сразу нескольким узлам, которые образуют группу с номером, указанным в поле адреса. Узлы сами идентифицируют себя, то есть определяют, к какой из групп они относятся. Один и тот же узел может входить в несколько групп. Такие сообщения в отличие от широковещательных называются мультивещательными. Групповой адрес не делится на поля номера сети и узла и обрабатывается маршрутизатором особым образом. В протоколе IP нет понятия широковещательности в том смысле, в котором оно используется в протоколах канального уровня локальных сетей, когда данные должны быть доставлены абсолютно всем узлам. Как ограниченный широковещательный IP-адрес, так и широковещательный IP-адрес имеют пределы распространения в интерсети - они ограничены либо сетью, к которой принадлежит узел - источник пакета, либо сетью, номер которой указан в адресе назначения. Поэтому деление сети с помощью маршрутизаторов на части локализует широковещательный шторм пределами одной из составляющих общую сеть частей просто потому, что нет способа адресовать пакет одновременно всем узлам всех сетей составной сети.

**1.2****Отображение физических адресов на IP-адреса: протоколы ARP и RARP**

В протоколе IP-адрес узла, то есть адрес компьютера или порта маршрутизатора, назначается произвольно администратором сети и прямо не связан с его локальным адресом, как это сделано, например, в протоколе IPX. Подход, используемый в IP, удобно использовать в крупных сетях и по причине его независимости от формата локального адреса, и по причине стабильности, так как в противном случае, при смене на компьютере сетевого адаптера это изменение должны бы были учитывать все адресаты всемирной сети Internet (в том случае, конечно, если сеть подключена к Internet'у). Локальный адрес используется в протоколе IP только в пределах локальной сети при обмене данными между маршрутизатором и узлом этой сети. Маршрутизатор, получив пакет для узла одной из сетей, непосредственно подключенных к его портам, должен для передачи пакета сформировать кадр в соответствии с требованиями принятой в этой сети технологии и указать в нем локальный адрес узла, например его МАС-адрес. В пришедшем пакете этот адрес не указан, поэтому перед маршрутизатором встает задача поиска его по известному IP-адресу, который указан в пакете в качестве адреса назначения. С аналогичной задачей сталкивается и конечный узел, когда он хочет отправить пакет в удаленную сеть через маршрутизатор, подключенный к той же локальной сети, что и данный узел. Для определения локального адреса по IP-адресу используется протокол разрешения адреса Address Resolution Protocol, ARP. Протокол ARP работает различным образом в зависимости от того, какой протокол канального уровня работает в данной сети - протокол локальной сети (Ethernet, Token Ring, FDDI) с возможностью широковещательного доступа одновременно ко всем узлам сети, или же протокол глобальной сети (X.25, frame relay), как правило не поддерживающий широковещательный доступ. Существует также протокол, решающий обратную задачу - нахождение IP-адреса по известному локальному адресу. Он называется реверсивный ARP - RARP (Reverse Address Resolution Protocol) и используется при старте бездисковых станций, не знающих в начальный момент своего IP-адреса, но знающих адрес своего сетевого адаптера. В локальных сетях протокол ARP использует широковещательные кадры протокола канального уровня для поиска в сети узла с заданным IP-адресом. Узел, которому нужно выполнить отображение IP-адреса на локальный адрес, формирует ARP запрос, вкладывает его в кадр протокола канального уровня, указывая в нем известный IP-адрес, и рассылает запрос широковещательно. Все узлы локальной сети получают ARP запрос и сравнивают указанный там IP-адрес с собственным. В случае их совпадения узел формирует ARP-ответ, в котором указывает свой IP-адрес и свой локальный адрес и отправляет его уже направленно, так как в ARP запросе отправитель указывает свой локальный адрес. ARP-запросы и ответы используют один и тот же формат пакета. Так как локальные адреса могут в различных типах сетей иметь различную длину, то формат пакета протокола ARP зависит от типа сети. На рисунке 2 показан формат пакета протокола ARP для передачи по сети Ethernet.

|  |  |  |
| --- | --- | --- |
| Тип сети | | Тип протокола |
| Длина локального адреса | Длина сетевого адреса | Операция |
| Локальный адрес отправителя (байты 0 - 3) | |  |
| Локальный адрес отправителя (байты 4 - 5) | | IP-адрес отправителя (байты 0-1) |
| IP-адрес отправителя (байты 2-3) | | Искомый локальный адрес (байты 0 - 1) |
| Искомый локальный адрес (байты 2-5) | |  |
| Искомый IP-адрес (байты 0 - 3) | |  |

Рисунок 2- Формат пакета протокола ARP

В поле типа сети для сетей Ethernet указывается значение 1. Поле типа протокола позволяет использовать пакеты ARP не только для протокола IP, но и для других сетевых протоколов. Для IP значение этого поля равно 080016. Длина локального адреса для протокола Ethernet равна 6 байтам, а длина IP-адреса - 4 байтам. В поле операции для ARP запросов указывается значение 1 для протокола ARP и 2 для протокола RARP. Узел, отправляющий ARP-запрос, заполняет в пакете все поля, кроме поля искомого локального адреса (для RARP-запроса не указывается искомый IP-адрес). Значение этого поля заполняется узлом, опознавшим свой IP-адрес. В глобальных сетях администратору сети чаще всего приходится вручную формировать ARP-таблицы, в которых он задает, например, соответствие IP-адреса адресу узла сети X.25, который имеет смысл локального адреса. В последнее время наметилась тенденция автоматизации работы протокола ARP и в глобальных сетях. Для этой цели среди всех маршрутизаторов, подключенных к какой-либо глобальной сети, выделяется специальный маршрутизатор, который ведет ARP-таблицу для всех остальных узлов и маршрутизаторов этой сети. При таком централизованном подходе для всех узлов и маршрутизаторов вручную нужно задать только IP-адрес и локальный адрес выделенного маршрутизатора. Затем каждый узел и маршрутизатор регистрирует свои адреса в выделенном маршрутизаторе, а при необходимости установления соответствия между IP-адресом и локальным адресом узел обращается к выделенному маршрутизатору с запросом и автоматически получает ответ без участия администратора.

**1.3 Организация доменов и доменных имен**

Для идентификации компьютеров аппаратное и программное обеспечение в сетях TCP/IP полагается на IP-адреса, поэтому для доступа к сетевому ресурсу в параметрах программы вполне достаточно указать IP-адрес, чтобы программа правильно поняла, к какому хосту ей нужно обратиться. Например, команда ftp://192.45.66.17 будет устанавливать сеанс связи с нужным ftp-сервером, а команда http://203.23.106.33 откроет начальную страницу на корпоративном web-сервере. Однако пользователи обычно предпочитают работать с символьными именами компьютеров, и операционные системы локальных сетей приучили их к этому удобному способу. Следовательно, в сетях TCP/IP должны существовать символьные имена хостов и механизм для установления соответствия между символьными именами и IP-адресами. В операционных системах, которые первоначально разрабатывались для работы в локальных сетях, таких как Novell NetWare, Microsoft Windows или IBM OS/2, пользователи всегда работали с символьными именами компьютеров. Так как локальные сети состояли из небольшого числа компьютеров, то использовались так называемые плоские имена, состоящие из последовательности символов, не разделенных на части. Примерами таких имен являются: NW1\_1, mail2, MOSCOW\_ SALES\_2. Для установления соответствия между символьными именами и МАС-адресами в этих операционных системах применялся механизм широковещательных запросов, подобный механизму запросов протокола ARP. Так, широковещательный способ разрешения Имен реализован в протоколе NetBIOS, на котором были построены многие локальные ОС. Так называемые NetBIOS-имена стали на долгие годы одним из основных типов плоских имен в локальных сетях. Для стека TCP/IP, рассчитанного в общем случае на работу в больших территориально распределенных сетях, подобный подход оказывается неэффективным по нескольким причинам. Плоские имена не дают возможности разработать единый алгоритм обеспечения уникальности имен в пределах большой сети. В небольших сетях уникальность имен компьютеров обеспечивает администратор сети, записывая несколько десятков имен в журнале или файле. При росте сети задачу решают уже несколько администраторов, согласовывая имена между собой неформальным способом. Однако если сеть расположена в разных городах или странах, то администраторам каждой части сети нужно придумать способ именования, который позволил бы им давать имена новым компьютерам независимо от других администраторов, обеспечивая в то же время уникальность имен для всей сети. Самый надежный способ решения этой задачи — отказ от плоских имен в принципе. Широковещательный способ установления соответствия между символьными именами и локальными адресами хорошо работает только в небольшой локальной сети, не разделенной на подсети. В крупных сетях, где общая широковещательность не поддерживается, нужен другой способ разрешения символьных имен. Обычно хорошей альтернативой широковещательности является применение централизованной службы, поддерживающей соответствие между различными типами адресов всех компьютеров сети. Компания Microsoft для своей корпоративной операционной системы Windows NT разработала централизованную службу WINS, которая поддерживает базу данных NetBIOS-имен и соответствующих им IP-адресов. Для эффективной организации именования компьютеров в больших сетях естественным является применение иерархических составных имен. В стеке TCP/IP применяется доменная система имен, которая имеет иерархическую древовидную структуру, допускающую использование в имени произвольного количества составных частей (рис. 3).

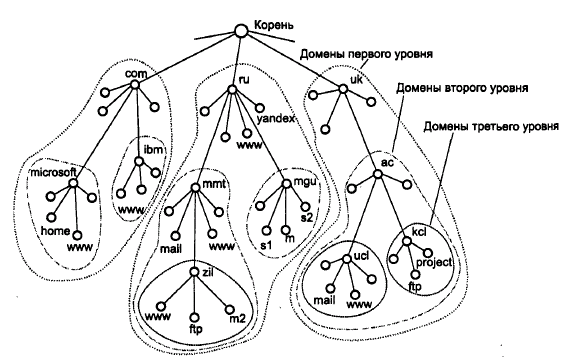


Рисунок 3 - Пространство доменных имен

Иерархия доменных имен аналогична иерархии имен файлов, принятой во многих популярных файловых системах. Дерево имен начинается с корня, обозначаемого здесь точкой (.). Затем следует старшая символьная часть имени, вторая по старшинству символьная часть имени и т. д. Младшая часть имени соответствует конечному узлу сети. В отличие от имен файлов, при записи которых сначала указывается самая старшая составляющая, затем составляющая более низкого уровня и т. д., запись доменного имени начинается с самой младшей составляющей, а заканчивается самой старшей. Составные части доменного имени отделяется друг от друга точкой. Например, в имени partnering.microsoft.com составляющая partnering является именем одного из компьютеров в домене microsoft.com.Разделение имени на части позволяет разделить административную ответственность за назначение уникальных имен между различными людьми или организациями в пределах своего уровня иерархии. Так, для примера, приведенного на рис.1.3, один человек может нести ответственность за то, чтобы все имена, которые имеют окончание «ru», имели уникальную следующую вниз по иерархии часть. Если этот человек справляется со своими обязанностями, то все имена типа www.ru, mail.mmt.ru или m2.zil.mmt.ru будут отличаться второй по старшинству частью.

Разделение административной ответственности позволяет решить проблему образования уникальных имен без взаимных консультаций между организациями, отвечающими за имена одного уровня иерархии. Очевидно, что должна существовать одна организация, отвечающая за назначение имен верхнего уровня иерархии. Совокупность имен, у которых несколько старших составных частей совпадают, образуют домен {domain) имен. Например, имена www1.zil.mmt.ru, ftp.zil.mmt.ru, yauidex.ru и s1 .mgu.ru входят в домен ru, так как все эти имена имеют одну общую старшую часть — имя ru. Другим примером является домен mgu.ru. Из представленных на рис. 3 имен в него входят имена s1.mgu.ru, s2.mgu.ru и rn.mgu.ru. Этот домен образуют имена, у которых две старшие части всегда равны mgu.ru. Имя www.mmt.ru в домен mgu.ru не входит, так как имеет отличающуюся составляющую mmt.Если один домен входит в другой домен как его составная часть, то такой домен могут называть поддоменом (subdomain), хотя название домен за ним также остается. Обычно поддомен называют по имени той его старшей составляющей, которая отличает его от других поддоменов. Например, поддомен mmt.ru обычно называют поддоменом (или доменом) mmt. Имя поддомену назначает администратор вышестоящего домена. Хорошей аналогией домена является каталог файловой системы. Если в каждом домене и поддомене обеспечивается уникальность имен следующего уровня иерархии, то и вся система имен будет состоять из уникальных имен.

По аналогии с файловой системой в доменной системе имен различают краткие имена, относительные имена и полные доменные имена. Краткое имя — это имя конечного узла сети: хоста или порта маршрутизатора. Краткое имя — это лист дерева имен. Относительное имя — это составное имя, начинающееся с некоторого уровня иерархии, но не самого верхнего. Например, wwwl.zil — это относительное имя. Полное доменное имя (fully qualified domain name, FQDN) включает составляющие всех уровней иерархии, начиная от краткого имени и кончая корневой точкой: wwwl .zil.mmt.ru.Корневой домен управляется центральными органами Интернета: IANA и InterNIC. Домены верхнего уровня назначаются для каждой страны, а также на организационной основе. Имена этих доменов должны следовать международному стандарту ISO 3166. Для обозначения стран используются трехбуквенные и двухбуквенные аббревиатуры, например, ru (Россия), uk (Великобритания), fin (Финляндия), us (Соединенные Штаты), а для различных типов организаций — следующие обозначения:

com — коммерческие организации (например, microsoft.com);

edu — образовательные организации (например, mit.edu);

gov — правительственные организации (например, nsf.gov);

org — некоммерческие организации (например, fidonet.org);

net — организации поддержки сетей (например, nsf.net).

Каждый домен администрируется отдельной организацией, которая обычно разбивает свой домен на поддомены и передает функции администрирования этих поддоменов другим организациям. Чтобы получить доменное имя, необходимо зарегистрироваться в какой-либо организации, которой организация InterNIC делегировала свои полномочия по распределению имен доменов. В России такой организацией является РосНИИРОС, которая отвечает за делегирование имен поддоменов в домене ru. Необходимо подчеркнуть, что компьютеры входят в домен в соответствии со своими составными именами, при этом они могут иметь абсолютно независимые друг от друга IP-адреса, принадлежащие к различным сетям и подсетям. Например, в домен mgu.ru могут входить хосты с адресами 132.13.34.15, 201.22.100.33 и 14.0.0.6.

Доменная система имен реализована в Интернете, но она может работать и как автономная система имен в любой крупной корпоративной сети, которая также использует стек TCP/IP, но никак не связана с Интернетом.

**1.4 Автоматизация процесса порядка назначения IP-адресов узлами сети протокол DHCP**

У каждой подсети в пределах составной сети должен быть собственный уникальный номер, следовательно, процедура распределения номеров должна быть централизованной. Аналогично, узлы должны быть однозначно пронумерованы в пределах каждой из подсетей, отсюда следует, что централизованный характер должна иметь и процедура распределения номеров узлов в пределах каждой подсети. Если сеть небольшая, то уникальность адресов может быть обеспечена вручную администратором.

В больших сетях, подобных Интернету, уникальность сетевых адресов гарантируется централизованной, иерархически организованной системой их распределения. Главным органом регистрации глобальных адресов в Интернете с 1998 года является ICANN (Internet Corporation for Assigned Names and Numbers) — неправительственная некоммерческая организация, управляемая советом директоров. Эта организация координирует работу региональных отделов, деятельность которых охватывает большие географические площади: ARIN (Америка), RIPE (Европа), APNIC (Азия и Тихоокеанский регион). Региональные отделы выделяют блоки адресов сетей крупным поставщикам услуг, те, в свою очередь присваивают их своим клиентам, среди которых могут быть ц более мелкие поставщики услуг.

Понятно, что в любой автономной сети могут быть использованы произвольные IP-адреса, лишь бы они были синтаксически правильными и уникальными в пределах этой сети. Совпадение адресов в не связанных между собой сетях не вызовет никаких отрицательных последствий. Однако во всех сетях, входящих в единую составную сеть (например, Интернет), должны использоваться глобально уникальные IP-адреса, то есть адреса, однозначно определяющие сетевые интерфейсы в пределах всей составной сети. Уже сравнительно давно наблюдается дефицит IP-адресов. Очень трудно получить адрес класса В и практически невозможно стать обладателем адреса класса А. При этом надо отметить, что дефицит обусловлен не только ростом сетей, но и тем, что имеющееся адресное пространство используется нерационально. Очень часто владельцы сетей класса С расходуют лишь небольшую часть из имеющихся у них 254 адресов. Рассмотрим пример, когда две сети необходимо соединить глобальной связью. В таких случаях в качестве канала связи используют два маршрутизатора, соединенных по схеме «точка-точка» (рис. 4). Для вырожденной сети, образованной каналом, связывающим порты двух смежных маршрутизаторов, приходится выделять отдельный номер сети, хотя в этой сети имеются всего два узла.



Рисунок 4 - Нерациональное использование пространства IP-адресов

Если же некоторая IP-сеть создана для работы в «автономном режиме», без связи с Интернетом, тогда администратор этой сети волен назначить ей произвольно выбранный номер. Но и в этой ситуации для того, чтобы избежать каких-либо коллизий, в стандартах Интернета определено несколько адресов, рекомендуемых для автономного использования: в классе А — это сеть 10.0.0.0, в классе В — это диапазон из 16 номеров сетей 172.16.0.0-172.31.0.0, в классе С — это диапазон из 255 сетей 192.168.0,0-192.168.255.0. Подробнее о том, как использование данных зарезервированных адресов позволяет справляться с дефицитом сетевых адресов, читайте в разделе «Трансляция сетевых адресов» главы .

Для смягчения проблемы дефицита адресов разработчики стека TCP/IP предлагают разные подходы. Принципиальным решением является переход на новую версию IPv6, в которой резко расширяется адресное пространство за счет использования 16-байтных адресов. Однако и текущая версия IPv4 поддерживает некоторые технологии, направленные на более экономное расходование IP-адресов. Одной из таких технологий является технология бесклассовой междоменной маршрутизации (Classless Inter-Domain Routing, CIDR). Технология CIDR основана на масках, она отказывается от традиционной концепции разделения адресов протокола IP на классы, что позволяет выдавать в пользование столько адресов, сколько реально необходимо потребителю. Благодаря CIDR поставщик услуг получает возможность «нарезать» блоки из выделенного ему адресного пространства в точном соответствии с требованиями каждого клиента. Как уже было сказано, IP-адреса могут назначаться администратором сети вручную. Это представляет для администратора утомительную процедуру. Ситуация усложняется еще тем, что многие пользователи не обладают достаточными знаниями для того, чтобы конфигурировать свои компьютеры для работы в интерсети и должны поэтому полагаться на администраторов. Протокол Dynamic Host Configuration Protocol (DHCP) был разработан для того, чтобы освободить администратора от этих проблем. Основным назначением DHCP является динамическое назначение IP-адресов. Однако, кроме динамического, DHCP может поддерживать и более простые способы ручного и автоматического статического назначения адресов. В ручной процедуре назначения адресов активное участие принимает администратор, который предоставляет DHCP-серверу информацию о соответствии IP-адресов физическим адресам или другим идентификаторам клиентов. Эти адреса сообщаются клиентам в ответ на их запросы к DHCP-серверу. При автоматическом статическом способе DHCP-сервер присваивает IP-адрес (и, возможно, другие параметры конфигурации клиента) из пула наличных IP-адресов без вмешательства оператора. Границы пула назначаемых адресов задает администратор при конфигурировании DHCP-сервера. Между идентификатором клиента и его IP-адресом по-прежнему, как и при ручном назначении, существует постоянное соответствие. Оно устанавливается в момент первичного назначения сервером DHCP IP-адреса клиенту. При всех последующих запросах сервер возвращает тот же самый IP-адрес. При динамическом распределении адресов DHCP-сервер выдает адрес клиенту на ограниченное время, что дает возможность впоследствии повторно использовать IP-адреса другими компьютерами. Динамическое разделение адресов позволяет строить IP-сеть, количество узлов в которой намного превышает количество имеющихся в распоряжении администратора IP-адресов. DHCP обеспечивает надежный и простой способ конфигурации сети TCP/IP, гарантируя отсутствие конфликтов адресов за счет централизованного управления их распределением. Администратор управляет процессом назначения адресов с помощью параметра "продолжительности аренды" (lease duration), которая определяет, как долго компьютер может использовать назначенный IP-адрес, перед тем как снова запросить его от сервера DHCP в аренду. Примером работы протокола DHCP может служить ситуация, когда компьютер, являющийся клиентом DHCP, удаляется из подсети. При этом назначенный ему IP-адрес автоматически освобождается. Когда компьютер подключается к другой подсети, то ему автоматически назначается новый адрес. Ни пользователь, ни сетевой администратор не вмешиваются в этот процесс. Это свойство очень важно для мобильных пользователей. Протокол DHCP использует модель клиент-сервер. Во время старта системы компьютер-клиент DHCP, находящийся в состоянии "инициализация", посылает сообщение discover (исследовать), которое широковещательно распространяется по локальной сети и передается всем DHCP-серверам частной интерсети. Каждый DHCP-сервер, получивший это сообщение, отвечает на него сообщением offer (предложение), которое содержит IP-адрес и конфигурационную информацию. Компьютер-клиент DHCP переходит в состояние "выбор" и собирает конфигурационные предложения от DHCP-серверов. Затем он выбирает одно из этих предложений, переходит в состояние "запрос" и отправляет сообщение request (запрос) тому DHCP-серверу, чье предложение было выбрано. Выбранный DHCP-сервер посылает сообщение DHCP-acknowledgment (подтверждение), содержащее тот же IP-адрес, который уже был послан ранее на стадии исследования, а также параметр аренды для этого адреса. Кроме того, DHCP-сервер посылает параметры сетевой конфигурации. После того, как клиент получит это подтверждение, он переходит в состояние "связь", находясь в котором он может принимать участие в работе сети TCP/IP. Компьютеры-клиенты, которые имеют локальные диски, сохраняют полученный адрес для использования при последующих стартах системы. При приближении момента истечения срока аренды адреса компьютер пытается обновить параметры аренды у DHCP-сервера, а если этот IP-адрес не может быть выделен снова, то ему возвращается другой IP-адрес. В протоколе DHCP описывается несколько типов сообщений, которые используются для обнаружения и выбора DHCP-серверов, для запросов информации о конфигурации, для продления и досрочного прекращения лицензии на IP-адрес. Все эти операции направлены на то, чтобы освободить администратора сети от утомительных рутинных операций по конфигурированию сети. Однако использование DHCP несет в себе и некоторые проблемы. Во-первых, это проблема согласования информационной адресной базы в службах DHCP и DNS. Как известно, DNS служит для преобразования символьных имен в IP-адреса. Если IP-адреса будут динамически изменятся сервером DHCP, то эти изменения необходимо также динамически вносить в базу данных сервера DNS. Хотя протокол динамического взаимодействия между службами DNS и DHCP уже реализован некоторыми фирмами (так называемая служба Dynamic DNS), стандарт на него пока не принят. Во-вторых, нестабильность IP-адресов усложняет процесс управления сетью. Системы управления, основанные на протоколе SNMP, разработаны с расчетом на статичность IP-адресов. Аналогичные проблемы возникают и при конфигурировании фильтров маршрутизаторов, которые оперируют с IP-адресами. Наконец, централизация процедуры назначения адресов снижает надежность системы: при отказе DHCP-сервера все его клиенты оказываются не в состоянии получить IP-адрес и другую информацию о конфигурации. Последствия такого отказа могут быть уменьшены путем использовании в сети нескольких серверов DHCP, каждый из которых имеет свой пул IP-адресов. Как уже отмечалось, в адресной схеме протокола выделяют особые IP-адреса. Если биты всех октетов адреса равны нулю, то он обозначает адрес того узла, который сгенерировал данный пакет. Это используется в ограниченных случаях, например в некоторых сообщениях протокола IP. Если биты сетевого префикса равны нулю, полагается, что узел назначения принадлежит той же сети, что и источник пакета. Когда биты всех октетов адреса назначения равны двоичной единице, пакет доставляется всем узлам, принадлежащим той же сети, что и отправитель пакета. Такая рассылка называется ограниченным широковещанием. Наконец, если в битах адреса, соответствующих узлу назначения, стоят единицы, то такой пакет рассылается всем узлам указанной сети. Это называется широковещанием. Специальное значение имеет, так же, адреса сети 127/8. Они используются для тестирования программ и взаимодействия процессов в пределах одной машины. Пакеты, отправленные на этот интерфейс, обрабатываются локально, как входящие. Потому адреса из этой сети нельзя присваивать физическим сетевым интерфейсам.

**2. Организация подсетей**

Очень редко в локальную вычислительную сеть входит более 100-200 узлов: даже если взять сеть с большим количеством узлов, многие сетевые среды накладывают ограничения, например, в 1024 узла. Исходя из этого, целесообразность использования сетей класса А и В весьма сомнительна. Да и использование класса С для сетей, состоящих из 20-30 узлов, тоже является расточительством. Для решения этих проблем в двухуровневую иерархию IP-адресов (сеть -- узел) была введена новая составляющая -- подсеть. Идея заключается в "заимствовании" нескольких битов из узловой части адреса для определения подсети. Полный префикс сети, состоящий из сетевого префикса и номера подсети, получил название расширенного сетевого префикса. Двоичное число, и его десятичный эквивалент, содержащее единицы в разрядах, относящихся к расширенному сетевому префиксу, а в остальных разрядах -- нули, назвали маской подсети. Но маску в десятичном представлении удобно использовать лишь тогда, когда расширенный сетевой префикс заканчивается на границе октетов, в других случаях ее расшифровать сложнее. Допустим, что мы хотели бы для подсети использовать не 8 бит, а десять. Тогда в последнем (z-ом) октете мы имели бы не нули, а число 11000000. В десятичном представлении получаем 255.255.255.192. Очевидно, что такое представление не очень удобно. В наше время чаще используют обозначение вида "/xx", где хх -- количество бит в расширенном сетевом префиксе. Таким образом, вместо указания: "144.144.19.22 с маской 255.255.255.192", мы можем записать: 144.144.19.22/26. Как видно, такое представление более компактно и понятно.

**2.1 Маска под сети переменной длины (Variable Length Subnet Mask)**

Однако вскоре стало ясно, что подсети, несмотря на все их достоинства, обладают и недостатками. Так, определив однажды маску подсети, приходится использовать подсети фиксированных размеров. Скажем, у нас есть сеть 144.144.0.0/16 с расширенным префиксом /23.

Таблица 3 - С расширенный префикс

|  |  |  |  |
| --- | --- | --- | --- |
| Сетевой префикс | | Подсеть | Узел |
| 144.144.0.0/23 | <--> | 10010000 | 10010000 | 0000000 | 0 00000000 |
| Расширенный сетевой префикс | | |  |  |  |

Такая схема позволяет создать 27 подсетей размером в 29 узлов каждая. Это подходит к случаю, когда есть много подсетей с большим количеством узлов. Но если среди этих сетей есть такие, количество узлов в которых находится в пределах ста, то в каждой их них будет пропадать около 400 адресов. Решение состоит в том, что бы для одной сети указывать более одного расширенного сетевого префикса. О такой сети говорят, что это сеть с маской подсети переменной длины (VLSM). Действительно, если для сети 144.144.0.0/16 использовать расширенный сетевой префикс /25, то это больше бы подходило сетям размерами около ста узлов. Если допустить использование обеих масок, то это бы значительно увеличило гибкость применения подсетей. Общая схема разбиения сети на подсети с масками переменной длины такова: сеть делится на подсети максимально необходимого размера. Затем некоторые подсети делятся на более мелкие, и рекурсивно далее, до тех пор, пока это необходимо. Кроме того, технология VLSM, путем скрытия части подсетей, позволяет уменьшить объем данных, передаваемых маршрутизаторами. Так, если сеть 12/8 конфигурируется с расширенным сетевым префиксом /16, после чего сети 12.1/16 и 12.2/16 разбиваются на подсети /20, то маршрутизатору в сети 12.1 незачем знать о подсетях 12.2 с префиксом /20, ему достаточно знать маршрут на сеть 12.1/16.

**2.2 Протокол межсетевого взаимодействия IP. Формат IP дейтограмм**

Перенос между сетями различных типов адресной информации в унифицированной форме, сборка и разборка пакетов при передаче их между сетями с различным максимальным значением длины пакета.

Таблица 4 - Формат IP дейтаграммы.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| версия | длина | тип сервиса | общая длина пакета в байтах | | |
| Идентификация  (для всех фрагментов одинаковое) | | | флаги (3бита) | | Смещение фрагмента |
| время жизни | | протокол | FCS заголовка | | |
| IP-адрес отправителя | | | | | |
| IP-адрес получателя | | | | | |
| Опции IP (если есть) | | | | заполнение (до 32 бит) | |
| Данные | | | | | |

Версия (IPv4), длина заголовка в 32 бит. словах, тип сервиса (для интеллектуальных маршрутизаторов, PPPDTRхх, P - приоритет (для будущего), D,T,R - запрашиваются мин. задержки, макс. пропускная способность, макс.надежность).Флаги Do not Fragment - DF, More Fragments - MF - еще фрагменты.Time to live - в секундах сколько жить пакету(перегрузки и кольца, снятие 1 при любом переходе). Опции IP (если есть) - для тестирования или отладки сети (напр. запись маршрута или обязательное прохождение по маршруту).

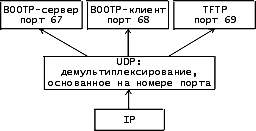


Рисунок 5 - Дейтаграмма UDP

Протокол доставки пользовательских дейтаграмм UDP. Формат сообщений UDP. Протокол надежной доставки сообщений TCP (Transmission Control Protocol). Порты и установление TCP-соединений.Протокол доставки пользовательских дейтаграмм UDP. Без гарантий доставки, поэтому его пакеты могут быть потеряны, продублированы или прийти не в том порядке, главное - быстрота. Мультиплексирование и демультиплексирование прикладных протоколов с помощью протокола UDP.

Формат сообщений UDP.

UDP source port - номер порта процесса-отправителя,

UDP destination port - номер порта процесса-получателя,

UDP message length - длина UDP-пакета в байтах,

UDP checksum - контрольная сумма UDP-пакета.

(!) Можно не заполнять поля 1 и 4.

Протокол надежной доставки сообщений TCP (Transmission Control Protocol).

Сверху - неструктурированный поток байт, вниз - сегменты (осн. единица TCP). Договор о макс. длине сегмента (не должен превышать поле данных IP дейтаграммы).

Порты и установление TCP-соединений.

Установление логического соединения. Адрес каждой оконечной точки включает IP адрес и номер порта TCP. Одна оконечная точка может участвовать в нескольких соединениях.

**2.3 Проблемы классической схемы**

В середине 80-х годов Internet впервые столкнулся с проблемой переполнения таблиц магистральных маршрутизаторов. Решение, однако, было быстро найдено -- подсети устранили проблему на несколько лет. Но уже в начале 90-х к проблеме большого количества маршрутов прибавилась нехватка адресного пространства. Ограничение в 4 миллиарда адресов, заложенное в протокол и казавшееся недосягаемой величиной, стало весьма ощутимым. В качестве решения проблемы были одновременно предложены два подхода -- один на ближайшее будущее, другой комплексный и долгосрочный. Первое решение -- это внедрение протокола бесклассовой маршрутизации (CIDR), к которому позже присоединилась система NAT. Долгосрочное решение -- это протокол IP следующей версии. Он обозначается, как IPv6, или IPng (Internet Protocol next generation). В этой реализации протокола длина адреса увеличена до 16-ти байтов (128 бит!), исключены некоторые элементы действующего протокола, которые оказались неиспользуемыми. Новая версия обеспечит, как любят указывать, плотность в 3 911 873 538 269 506 102 IP адресов на квадратный метр поверхности Земли. Однако то, что и в 2000-м году протокол все еще проходил стандартизацию, и то, что протокол CIDR вместе с системой NAT оказались эффективным решением, заставляет думать, что переход с IPv4 на IPng потребует очень много времени. Появление этой технологии было вызвано резким увеличением объема трафика в Internet и, как следствие, увеличением количества маршрутов на магистральных маршрутизаторах. Так, если в 1994 году, до развертывания CIDR, таблицы маршрутизаторов содержали до 70 000 маршрутов, то после внедрения их количество сократилось до 30 000. На сентябрь 2002, количество маршрутов перевалило за отметку 110 000! Можете себе представить, сколько маршрутов нужно было бы держать в таблицах сегодня, если бы не было CIDR! Что же представляет собой эта технология? Она позволяет уйти от классовой схемы адресации, эффективней использовать адресное пространство протокола IP. Кроме того, CIDR позволяет агрегировать маршрутные записи. Одной записью в таблице маршрутизатора описываются пути ко многим сетям. Суть технологии CIDR состоит в том, что каждому поставщику услуг Internet (или, для корпоративных сетей, какому-либо структурно-территориальному подразделению) должен быть назначен неразрывный диапазон IP-адресов. При этом вводится понятие обобщенного сетевого префикса, определяющего общую часть всех назначенных адресов. Соответственно, маршрутизация на магистральных каналах может реализовываться на основе обобщенного сетевого префикса. Результатом является агрегирование маршрутных записей, уменьшение размера таблиц маршрутных записей и увеличение скорости обработки пакетов. Допустим, центральный офис компании выделяет одному своему региональному подразделению сети 172.16.0.0/16 и 172.17.0.0/16, а другому -- 172.18.0.0/16 и 172.19.0.0/16. У каждого регионального подразделения есть свои областные филиалы и из полученного адресного блока им выделяются подсети разных размеров. Использование технологии бесклассовой маршрутизации позволяет при помощи всего одной записи на маршрутизаторе второго подразделения адресовать все сети и подсети первого подразделения. Для этого указывается маршрут к сети 172.16.0.0 с обобщенным сетевым префиксом 15. Он должен указывать на маршрутизатор первого регионального подразделения. По своей сути технология CIDR родственна VLSM. Только если в случае с VLSM есть возможность рекурсивного деления на подсети, невидимые извне, то CIDR позволяет рекурсивно адресовать целые адресные блоки. Использование CIDR позволило разделить Internet на адресные домены, внутри которых передается информация исключительно о внутренних сетях. Вне домена используется только общий префикс сетей. В результате многим сетям соответствует одна маршрутная запись.

**2.4 Примеры организации адресации в IP сетях**

В конце статьи хотелось бы привести практические примеры по затронутым в статье темам. Проектирование адресной схемы требует от специалиста тщательной проработки многих факторов, учета возможного роста и развития сети. Начнем с примера разбиения сети на подсети. При любом планировании нужно знать, сколько подсетей необходимо сегодня и может понадобиться завтра, сколько узлов находится в самой большой подсети сегодня и сколько может быть в будущем. Кроме того, следует разработать хотя бы схематическую топологию сети с указанием всех маршрутизаторов и шлюзов. Хорошей практикой является резервирование ресурсов на будущее. Так, если в самой большой подсети находится 60 узлов, не следует выделять подсеть размерностью в 26 - 2 (=62) узла! Не скупитесь, стоимость решения возможной проблемы будет больше, нежели стоимость выделения в два раза большего блока адресов. Однако не нужно впадать и в другую крайность.

Пример 1

Организации выделен блок адресов 220.215.14.0/24. Разбить блок на 4 подсети, наибольшая из которых насчитывает 50 узлов. Учесть возможный рост в 10%. На первом этапе необходимое число подсетей мы округляем в большую сторону к ближайшей степени числа 2. Поскольку в данном примере число необходимых подсетей равно 4, округлять не нужно. Определим количество бит, нужных для организации 4 подсетей. Для этого представим 4 в виде степени двойки: 4 = 22 . Степень -- это и есть количество бит отводимых для номера подсети. Так как сетевой префикс блока равен 24, то расширенный сетевой префикс будет равен 24 + 2 = 26.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Сетевой префикс | | | Подсеть | Узел |
| 0 | 8 | 16 | 24 25 | 31 |
| 220.215.14.0/26 | | 10010000 | 10010000 | 00001110 | 0 0 | 000000 |
| Расширенный сетевой префикс | | | |  |  |  |

Оставшиеся 32 - 26 = 6 бит будут использоваться для номера узла. Проверим, сколько узлов можно адресовать 6-ю битами: 26 - 2 = 62 узла. Достаточно ли это для 10% роста? 10% от 50 узлов -- это 5 узлов, а 55 узлов меньше возможных 62-х. Следовательно, два бита для номера подсети нас устраивают. Следующим этапом будет нахождение подсетей. Для этого двоичное представление номера подсети, начиная с нуля, подставляется в биты, отведенные для номера подсети.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Основная сеть | 11011100 | 11010111 | 00001110 | 00 | 000000 | 220.215.14.0/24 | |
| Подсеть 0(00) | 11011100 | 11010111 | 00001110 | 00 | 000000 | 220.215.14.0/26 | |
| Подсеть 1(01) | 11011100 | 11010111 | 00001110 | 01 | 000000 | 220.215.14.64/26 | |
| Подсеть 2(10) | 11011100 | 11010111 | 00001110 | 10 | 000000 | 220.215.14.128/26 | |
| Подсеть 3(11) | 11011100 | 11010111 | 00001110 | 11 | 000000 | 220.215.14.192/26 | |
| Расширенный сетевой префикс | | | | |  |  |  |

Для проверки правильности наших вычислений, следует помнить простое правило: десятичные номера подсетей должны быть кратными номеру первой подсети. Из этого правила можно вывести и другое, упрощающее расчет подсетей: достаточно вычислить адрес первой подсети, а адреса последующих определяются произведением адреса первой на соответствующий номер подсети. В нашем примере мы легко могли установить адрес третьей подсети, просто умножив 64 \* 3 = 192. Как уже упоминалось, кроме адреса подсети, в котором все биты узловой части равны нулю, есть еще один служебный адрес – широковещательный. Особенность широковещательного адреса состоит в том, что все биты узловой части равны единице. Рассчитаем широковещательные адреса наших подсетей:

подсеть |

ШВА подсети 0 (00) | 11011100.11011100.00001110.00 111111 | 220.215.14.63/26

ШВА подсети 0 (01) | 11011100.11011100.00001110.01 111111 | 220.215.14.127/26

ШВА подсети 0 (10) | 11011100.11011100.00001110.10 111111 | 220.215.14.191/26

ШВА подсети 0 (11) | 11011100.11011100.00001110.11 111111 | 220.215.14.255/26

Расширенный сетевой префикс. Узловая часть = все 1

Легко заметить, что широковещательным адресом является наибольший адрес подсети. Теперь, получив адреса подсетей и их широковещательные адреса, мы можем построить таблицу используемых адресов:

|  |  |  |
| --- | --- | --- |
| № подсети | Наименьший адрес подсети | Наибольший адрес подсети |
| 0 | 220.215.14.1 | - 220.215.14.62 |
| 1 | 220.215.14.65 | - 220.215.14.126 |
| 2 | 220.215.14.129 | - 220.215.14.190 |
| 3 | 220.215.14.193 | - 220.215.14.254 |

Это и есть разбиение, удовлетворяющее условию.

Пример 2

В первом примере все подсети были одинакового размера -- по 6 разрядов. Часто удобнее иметь подсети разного размера. Допустим, одна подсеть нужна для задания адресов двух маршрутизаторов, связанных по схеме "точка-точка". В этом случае используется всего лишь два адреса. Рассмотрим теперь случай, когда компании выделен блок адресов 144.144.0.0/16. Нужно разбить адресное пространство на три части, выделить адреса для двух пар маршрутизаторов и оставить некоторый резерв. Разделим сеть 144.144.0.0/16 на четыре равных части, выделив два бита для номера подсети:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Октет | W | X | Y | | Z |
| Подсеть 0(00) | 10010000 | 10010000 | 0 | 000000 | 00000000 | 144.144.0.0/18 |
| Подсеть 1(01) | 10010000 | 10010000 | 1 | 000000 | 00000000 | 144.144.64.0/18 |
| Подсеть 2(10) | 10010000 | 10010000 | 0 | 000000 | 00000000 | 144.144.128.0/18 |
| Подсеть 3(11) | 10010000 | 10010000 | 1 | 000000 | 00000000 | 144.144.192.0/18 |

Внутри третьей подсети выделим две подсети размером в четыре адреса:

|  |  |  |  |
| --- | --- | --- | --- |
| Подсеть № 3 | | № узла | |
| Подсеть 0(0) | 10010000 | 10010000 | 1 | 00000 | 00000 | 0 | 144.144.192.0/30 |
| Подсеть 1(1) | 10010000 | 10010000 | 1 | 000000 | 00001 | 0 | 144.144.192.4/30 |
| Номер подсети | |  |  |  |  |  |  |

Полученные две сети будем использовать для адресации интерфейсов маршрутизаторов. Оставшееся адресное пространство будет резервом, из которого можно будет выделять адресные блоки по потребности. Из оставшихся адресов можно, например, образовать 62 сети размерности класса С и еще несколько, размером поменьше.

**Заключение**

Установление соответствия между IP-адресом и аппаратным адресом осуществляется протоколом разрешения адресов.

Существует два принципиально отличных подхода к разрешению адресов: в сетях, поддерживающих широковещание, и в сетях, его не поддерживающих.

Протокол АКР, работающий в сетях Ethernet, Token Ring, FDDI, для трансляции IP-адреса в МАС-адрес выполняет широковещательный ARP-запрос. Для ускорения процедуры преобразования адресов протокол ARP кэширует полученные ответы в ARP-таблицах.

В сетях, в которых не поддерживаются широковещательные сообщения, ARP-таблицы хранятся централизовано на выделенном ARP-сервере. Таблицы составляются либо вручную администратором, либо автоматически при включении каждый узел регистрирует в них свои адреса. При необходимости установления соответствия между IP-адресом и локальным адресом узел обращается к ARP-серверу с запросом и автоматически получает ответ без участия администратора.

В стеке TCP/IP применяется доменная система символьных имен, которая имеет иерархическую древовидную структуру, допускающую использование в имени произвольного количества составных частей. Совокупность имен, у которых несколько старших составных частей совпадают, образуют домен имен. Доменные имена назначаются централизованно, если сеть является частью Интернета, в противном случае — локально.

Соответствие между доменными именами и IP-адресами может устанавливаться как средствами локального хоста с использованием файла hosts, так и с помощью централизованной службы DNS, основанной на распределенной базе отображений «доменное имя — IP-адрес».