Урок 3

Сетевой уровень. Часть 1

Классовая IPv4 - адресация. Протокол ARP: связь IP-адреса и MAC-адреса. Формат IPv4-пакета. Статическая маршрутизация. Диагностика сетевого уровня.

Введение

На прошлом занятии мы рассмотрели канальный уровень и адресацию (физическую адресацию), которая применяется, чтобы сообщения могли передаваться по локальной сети через среду и быть полученными адресатами.

В отличие от адресации канального уровня, работающей только локально, задача адресации на сетевом уровне в общем случае (пока не берем частные адреса и широковещательные адреса) – глобальная адресация. Физическая адресация (несмотря на глобально-администрируемые, которые, теоретически, уникальные во всем Интернете) не решает задачи глобальной адресации (и не предназначена для этого). Задача MAC-адресов – уникальным образом идентифицировать сетевой интерфейс в рамках широковещательного домена. Т.е. необходимо различать только устройства непосредственно видящие друг друга. Для задачи глобальной адресации MAC-адреса не используются (хотя и могут, с определенного рода надстройкой, префиксом сети, что, кстати, и использовалось в IPX, и, частично, в IPv6).

Таким образом, на сетевом уровне у нас возникает задачи:

1) создать глобальную адресацию, позволяющую связать разные сети в единое адресное пространство;

2) обеспечить доставку (маршрутизацию) через разнородное оборудование.

Конечно, адресация сетевого уровня не всегда глобальна (существуют частные сети, изолированные сети, и соответствующие локальные адреса, которые не используются в сети Интернет; кроме того, даже в одной локальной сети требуются IP-адреса помимо MAC-адресов, точно также, как и взаимодействующие по радиосети люди также помимо позывных, используют всем привычные имена, и не бывает, чтобы человек был с позывным, но без имени), но принципы остаются теми же. Даже внутри крупной локальной сети может возникнуть задача выполнять маршрутизацию, если она состоит из нескольких подсетей, объединенных в единую сеть уже на сетевом уровне, с применением маршрутизаторов.

В отличие от канального уровня, сетевые протоколы не зависят от среды и не ориентируются на тот или иной ее тип. Фактически, сетевой уровень решает задачи объединения сетей, почему в модели OSI данный уровень называется межсетевым.

Данные на сетевом уровне вместе с заголовком как правило называют пакетом (реже используется термин дейтаграмма, также применяемый для сообщений транспортного уровня без гарантированной доставки).

Таким образом, протоколы сетевого уровня:

1) Инкапсулируют протоколы транспортного уровня (и некоторые протоколы сетевого уровня также);

2) Осуществляют адресацию отправителя и получателя, позволяющую доставить пакеты между разными сетями (логическая адресация);

3) Преобразуют логические (сетевые) адреса в физические (для доставки через локальную сеть);

4) Осуществляют маршрутизацию и пересылку пакетов;

5) Передают данные на нижеследующий, канальный уровень;

6) Осуществляют сообщение об ошибках в случае невозможности доставки;

На этом уровне не контролируется корректность доставки (это задачи канального уровня и транспортного). С другой стороны, генерируются сообщения об ошибках, если задача доставки не может быть выполнена.

Итак, имеем компьютер. Машина обладает несколькими сетевыми устройствами (сетевыми картами). Каждая сетевая карта обладает физическим – MAC-адресом. Теперь задача сделать нашу машину доступной по сети.

Для каждого сетевого устройства задаем сетевой интерфейс, для которого мы назначаем IP-адрес. Например, 192.168.0.1

IP-адрес может быть назначен статически, либо получен по протоколу DHCP (Dynamic Host Configuration Protocol — протокол динамической настройки хоста) динамически (используя широковещательную рассылку. При этом в качестве адреса отправителя используется 0.0.0.0, а в качестве адреса получателя broadcast-адрес 255.255.255.255. Таким образом действие протокола DHCP ограничено широковещательным доменом).

Этот адрес будет использоваться в качестве адреса отправителя (если мы отправляем сообщение), и в качестве получателя (когда получаем). В любом случае, будут указаны два адреса, адрес отправителя и получателя.

Среди наиболее примечательных сетевых протоколов, позволяющих инкапсулировать данные вышестоящих протоколов и идентифицировать отправителя и получателя с помощью логических адресов, можно назвать IPv4, IPX и IPv6. Часто под IP имеют в виду IPv4, но все большее распространение находит и IPv6. Каждый из этих протоколов имеет собственный формат адресов. В частности, IP-адреса (точнее IPv4-адреса) — это всем нам знакомые четыре октета, записываемые в десятичном виде и разделенные точками.

Например, 192.168.0.1, 127.0.0.1, 8.8.8.8, 5.255.255.55. Так как на самом деле каждое число – байт, оно может находиться в диапазоне от 0 до 255 (256 значений), то есть такие адреса как 192.168.0.256, 10.0.0.300, 1000.0.0.1 не возможны. На самом деле в памяти число хранится побайтово (побитово), и например «в сыром виде» адрес 5.255.255.55 выглядит как 05FFFF37.

Среди всего многообразия адресов имеются специальные адреса.

0.0.0.0 используется в качестве отправителя, пока еще IP-адрес не присвоен, в таблицах маршрутизации означает маршрут по умолчанию, в создании сокетов означает, что сокет прослушивает все сетевые интерфейсы.

255.255.255.255 – широковещательная рассылка, ограничена текущим широковещательным доменом (broadcast). Используется при поиске сервера, когда IP-адрес сервера не известен, в протоколах DHCP, PPPoE.

127.0.0.1 – текущий адрес данной машины. Используя этот адрес, можно связать два приложения на одной машине между собой (например, php и mysql, веб-сервер nginx и веб-сервер apache2).

192.168.0.1 – этот адрес часто имеют по умолчанию домашние роутеры, 192.168.1.1 – а этот DSL-модемы.

8.8.8.8 и 8.8.4.4 – адреса публичных DNS-серверов от Google.

Несмотря на то, что, как правило, пользователь использует доменные имена сайтов, с помощью системы DNS, они преобразуются в доменные имена, например, yandex.ru → 5.255.255.55, хотя в настоящее время наблюдается еще более сложная картина: пользователь пишет в поисковой строке поисковый запрос (одноклассники ру), который передается поисковой системе (например, яндексу), который находит адрес сайта (например, ok.ru), который, в свою очередь, с помощью системы DNS преобразуется в IP-адрес, например, 217.20.155.13. Спасибо поисковикам за это.

Поговорим о структуре IP-адресов. При создании IPv4-адресов было решено использовать часть адреса использовать для идентификации сети, и часть – для хоста. При этом уже тогда эти части оказались разной длины. Так была разработана классовая адресация. Первые от 1 до 4 бит служили для обозначения типа сети, далее в зависимости от этих бит, следовал адрес сети нужной длины. Фактически, эти биты также входили как и в сам адрес, так и в идентификатор сети.

Таким образом, мы имели 127 сетей (за вычетом 0.0.0.0) где для адреса сети использовался первый байт, и для хоста, остальные три октета. Это сети класса A.

Затем 16384 сетей с хостами по два октета. Сети класса B.

И еще чуть более чем 2 миллиона сетей по 254 машины (+ адрес сети и широковещательный адрес – т.е. на хост нам оставался один байт).

Также есть специальные адреса класса D, которые содержат ну не совсем сети, а мультикастные группы подписки. Мультикстная группа имеет только один адрес, но все, что отправляется на этот адрес, доставляется для всех узлов, которые на этот же адрес подписаны. Используется в задачах маршрутизации и для онлайн-стриминга. Например, IP-TV. С помощью протокола IGMP клиенты подписываются в группу, после чего начинают получать широковещательный трафик. В одну мультикстную группу могут входить клиенты из разных сетей. Фактический IP-адрес у подписчика не меняется, но он получает трафик, адресованный мульикстно. С помощью протокола IGMP происходит и отписка от вещания.

Сети класса E были зарезервированы для будущего использования, но в таковом статусе и остались. Особое значение имеет адрес 255.255.255.255. Сообщение с таким адресом отправителя будет разослано всем узлам широковещательного домена.

В классовой адресации, традиционно, в качестве адреса сети использовался адрес, в котором адрес хоста состоял из двоичных 0. Для широковещательного адреса – адрес хоста, состоящий из двоичных единиц (т.е. в октетах было 255).

Например, для сети 10.0.0.0 класса A:

10.0.0.0 – адрес сети,

10.255.255.255 – широковещательный адрес (Broadcast).

Некоторые из этих адресов были зарезервированы для специальных целей и не маршрутизируются в Интернет.

Например, сеть класса A – 127.0.0.0 – полностью отдавалась под локальный адрес. Пакеты, направляемые на этот адрес, не уйдут за пределы локальной машины. Тем не менее, такие адреса могут использоваться локальными службами для взаимодействия между собой, а также для отладки. Также часто используется адрес 127.0.0.1 для в качестве адреса веб-интерфейса локальных служб, которые должны открываться только на этой же машине.

Сеть класса A – 10.0.0.0 была выделена под локальные сети. Эти адреса не маршрутизируются в интернет, таким образом, любая организация может их использовать для локальной сети. В общем-то, благодаря механизму NAT (а в прошлом – Proxy), эти адреса часто и используются Интернет-провайдерами.

16 сетей класса B – от 172.16.0.0 до 172.31.0.0 была выделена также под локальные сети.

И еще все сети класса C от 192.168.0.0 до 192.168.255.0 также были выделены под локальное использование.

Для того чтобы выделить адрес сети из хоста, используется маска сети. Состоит она из двух частей. Та часть, что идентифицирует сеть, содержит двоичные единицы. А та часть, что выделена под хост, содержит двоичные нули. Например, 255.0.0.0 – маска для сетей класса A.

Произведя двоичное умножение побитово, маски сети на IP-адрес, мы получим адрес сети.

Для сетей класса B – маска 255.255.0.0.

Для сетей класса C – маска 255.255.255.0.

При отправке сообщения происходит маршрутизация – поиск маршрута по таблице маршрутизации. Именно благодаря маске мы определяем к какой сети принадлежит адрес, и в какой сетевой интерфейс или на какой шлюз его отправлять.

Предположим, имеется два маршрута.

default gw 192.168.1.1

192.168.1.0 255.255.255.0 eth0

Если мы попробуем сделать ping 192.168.1.20, то система попробует вычислить адрес сети.

Адрес: **192.168.1.**20

Маска: **255.255.255.**0

Сеть: **192.168.1**.0

Значит, пакет будет направлен в сетевой интерфейс eth0.

Предположим теперь, что мы хотим сделать ping 192.168.0.30. Точно также вычисляется адрес сети.

Адрес: **192.168.0.**30

Маска: **255.255.255.**0

Сеть: **192.168.0**.0

Это не та же сеть, которая доступна через eth0. Отдельный маршрут для этой сети не известен, значит, сообщение будет отправляться по маршруту по умолчанию.

Каким же образом будет осуществляться отправка? Отправка работает через канальный уровень. Чтобы узнать MAC-адрес, нужно выполнить запрос ARP.

Запрос ARP рассылается броадкастно на MAC-адрес FF:FF:FF:FF:FF:FF, и смысле его такой. Какой MAC-адрес у IP-адреса 192.168.0.20. Если в сети присутствует такая машина, она отвечает, MAC-адрес кешируется, и сообщение вкладывается в кадр, где указываются MAC-адреса получателя и отправителя.

А как же быть, если маршрут по умолчанию? Мы же не сможем получить MAC-адрес для адреса 192.168.100.1. Или тем более для 5.255.255.55. Не сможем. Но нам это и не нужно. ARP-запрос будет выяснять MAC-адрес шлюза. Поэтому в IP-заголовках будет IP-адрес получателя, а в заголовках кадра – MAC-адрес шлюза. Непосредственно IP-адрес шлюза на сетевом уровне никак не передается, он служит для преобразования в MAC-адрес.

Сетевой уровень

Маршрутизация

Маршрутизация (Routing) — процесс выбора маршрута для данных в компьютерных сетях.

Сетевые маршруты бывают статическими (задаются сетевым администратором) или динамические (вычисляются с помощью сетевых алгоритмов маршрутизации). Устройство может одновременно оперировать и статическими и динамическими маршрутами. Протоколы маршрутизации вычисляют динамические маршруты, используемые как наилучший путь опираясь на топологию, текущую загруженность, пропускную способность.

Маршрутизируемые протоколы (сообщения которых несут информацию):

- IP (Internet protocol);
- ICMP (Internet Control Message Protocol);
- IGMP (Internet Group Management Protocol).

Маршрутизирующие протоколы (служебные протоколы, обеспечивающие работу и передачу пакетов маршрутизируемых протоколов):

- Interior Routing Protocols (внутри AS):

- ❖ RIP, RIP2 (Routing Information Protocol);
- ❖ OSPF (Open Shortest Path First);
- ❖ (IS-IS, IGRP, EIGRP и д.р.).

- Exterior Routing Protocols (между AS):
  - ❖ EGP (Exterior Gateway Protocol);
  - ❖ BGP (Border Gateway Protocol).

Хотя протоколы маршрутизации относятся к сетевому уровню, технически реализация протокола маршрутизации может находиться на прикладном уровне (то есть протокол может использовать в качестве транспортного протокола TCP или UDP).

Internet Protocol (IP)

Internet Protocol (IP, Интернет-протокол или межсетевой протокол) — является маршрутизируемым протоколом сетевого уровня. На основе протокола IP работает большинство современных сетей. Основная задача протокола - это передача данных из одной сети в другую, поэтому одной из ключевых особенностей является адресация. В рамках решения проблем адресации необходимо назначить уникальное имя каждому устройству, подключенному к сети.

Стек протоколов TCP/IP применяет три типа адресов: локальные или mac адреса (иногда называются аппаратные), IP-адреса и доменные адреса, используемые в названиях сайтов.

Локальный адрес используется для передачи кадров внутри широковещательного домена (подсети). Локальные адреса в пределах подсети обязательно должны быть уникальны, в случае обнаружения нескольких компьютеров или сетевых устройств с одинаковым адресом это вызовет конфликт, и ОС сообщит пользователю об этом. ОС позволяет пользователю переназначить адрес, также обнаружение данного конфликта может свидетельствовать о попытке перехватить трафик (атака методом перехвата на канальном уровне). Локальный адрес используется средствами сетевой технологии для передачи информации внутри подсети, которая входит в интерсеть. Разные участки сети могут использовать разные сетевые технологии и протоколы, поэтому локальные адреса используются внутри сети, а сетевой IP-адрес для межсетевого взаимодействия. Для локальной сети Ethernet локальным адресом является MAC-адрес. Устройства в локальной сети могут иметь несколько локальных адресов для одного сетевого адаптера (интерфейса) и напротив, некоторые сетевые устройства могут не иметь локальных адресов, например, глобальные порты роутеров для соединений типа «точка-точка».

IP-адрес является основным типом адреса, используемым сетевым уровнем для передачи информации между сетями. IP-адрес версии 4 состоит из 4 байт, например, 19.16.18.23. IP-адрес может быть назначен пользователем или присвоен автоматически с использованием протокола DHCP. Протокол DHCP настраивается сетевым администратором в процессе конфигурации сервера или коммутационного оборудования (маршрутизаторы, коммутаторы 3 уровня). Сетевая адресация может быть задана сетевым администратором или использовано значение полученное от подразделения Internet (Internet Network Information Center, InterNIC). На данный момент IP-адреса версии IPv4 уже кончились и существует возможность только купить их. Это необходимо, если сетевые устройства должны работать как составная часть сети Интернет и быть доступны извне. Для случая просто доступа в Интернет без предоставления ресурсов сети внешним пользователям для

сети достаточно одного белого IP-адреса. Маршрутизаторы соединяют несколько сетей, поэтому каждый интерфейс обладает своим адресом. Интерфейсы, подключенные к виртуальным локальным сетям, могут иметь несколько IP-адресов, привязанных к разным виртуальным сетям. Такое подключение называется роутер на палочке (router on stick). Компьютер или сервер также может быть подключен к нескольким сетям через несколько сетевых карт, в этом случае каждый интерфейс будет обладать своим адресом, таким образом ip-адрес задается не для компьютера, но для сетевого соединения.

Символьные или доменные имена используются для облегчения пользователю работы в сети. Они строятся на основе иерархии. Доменные адреса привязываются к IP-адресам с помощью специальной сетевой службы размещенной на сервере. Служба доменных имен отвечает на запросы пользователей, позволяя им узнать IP-адрес сетевого узла по его доменному имени. Сетевая служба доменных имен Domain Name System (DNS) является распределенной и не имеет центрального сетевого узла. О работе доменной службы мы поговорим на следующих уроках.

Классификация сетевых адресов

IP-адрес 4 версии состоит из 4 байт (иногда говорят из 4 октетов). Адрес записывается в виде 4 чисел от 0-255 разделенных точками. Так же IP-адрес может быть записан в бинарном виде. Примеры приведены ниже:

121.11.21.32 - обычная десятичная форма записи IPv4 адреса;

01111001.00001011.00010101.00100000- бинарная форма записи IPv4 адреса.

Вы можете самостоятельно перевести адрес из одной формы в другой. Для этого вы можете найти информацию о том, как перевести десятичные числа в двоичные. Также можно воспользоваться инженерным калькулятором. В интернете есть специальные калькуляторы IP-адресов. Перевод адреса из одной системы в другую нужен для понимания системы работы масок и расчёта подсетей. Хорошее понимание этих понятий позволяет эффективно рассчитать адресное пространство и записать сетевой маршрут.

IP адрес включает 2 части - адрес сети и адрес узла в сети. Сетевую от узловой части можно отделить с помощью сетевой маски. Раньше использовалась классовая адресация, но этот подход с ростом производительности устройств и необходимостью построения более маленьких сетей быстро исчерпал себя. Но все же рассмотрим его обзорно. Сети по первым сетевым битам делились на несколько типов.

Ниже приведены маски для классов сетей:

Класс	Число возможных адресов сетей	Число возможных адресов хостов	Маска подсети	Начальный адрес	Конечный адрес
A	128	16 777 214	255.0.0.0	0.0.0.0	127.255.255.255
B	16 384	65 534	255.255.0.0	128.0.0.0	191.255.255.255
C	2 097 152	254	255.255.255.0	192.0.0.0	223.255.255.255
D	Групповой адрес			224.0.0.0	239.255.255.255
E	Зарезервировано			240.0.0.0	255.255.255.255

Адреса сетей, начинающиеся с 0, относят к классу A, при этом номер сети занимает один байт, оставшиеся 3 байта используются для номера сетевого узла. Сети этого класса легко отличить по первому байту, который лежит в диапазоне от 1 до 126. (0 адрес не используется, а 127 задействован для локальных адресов.)

Размер сети A класса вычисляется по формуле 2^24=16 777 216 узлов.

Адреса сетей, начинающиеся с 10, относят к классу B, при этом номер сети занимает два байта, оставшиеся 2 байта используются для номера сетевого узла. Сети класса B является средними и число узлов в них вычисляется по формуле 2^16= 65 536 узлов.

Адреса сетей, начинающиеся с 110, относят к классу C, при этом номер сети занимает уже три байта, оставшийся байт используются для номера сетевого узла. Сети класса C является малыми и число узлов в них вычисляется по формуле 2^8= 256 узлов.

Специальные IP-адреса

Стандартом определен ряд специальных адресов:

0.0.0.0 – текущий хост (сеть);

255.255.255.255 – все хосты в текущей сети (ограниченный широковещательный адрес);

127.0.0.0 – обратная петля (loopback).

- Сеть для тестирования;
- Данные не передаются в сеть, а приходят обратно;
- 127.0.0.1 – localhost (текущий компьютер).

169.254.0.0 – Link-local адреса:

- Назначаются ОС хоста автоматически, если недоступна другая IP-конфигурация;
- Могут использоваться в пределах локальной сети.

Пакет с адресом, в узловой части которого находятся единицы, называется широковещательным и рассылается всем абонентам сети. Такая рассылка называется broadcast.

В стандарте протокола не предусмотрена широковещательная рассылка, аналогичная канальному уровню на всех абонентов, рассылка производится только внутри локальной сети.

Есть определенные правила, которые необходимо учитывать при назначение адресов. Номер сети или узла не может состоять только из 1 или 0, таким образом число доступных адресов уменьшается на 2 адреса. Эти адреса выделены для обозначения сети (называется сетевой адрес первый в диапазоне) и широковещательной рассылки (последний адрес в сети). Для сети класса С из 256 номеров, адресация которых идет от 0 до 255 доступно 254 адреса для назначения хостам сети (0 и 255 недоступны).

Выделен специальный IP-адрес для локального тестирования, первый октет которого равен 127. Адрес применяется для проверки работы программ и сетевых сервисов на компьютере без обращения к сети. Когда вы обращаетесь по адресу 127.0.0.1, то ОС создают петлю, и пакеты возвращаются на сетевой интерфейс, таким образом можно обратиться к сервисам, развернутым локально без подключения к сети. Адрес называют loopback.

Выделенный в отдельный адресный диапазон групповые IP-адреса - multicast – позволяют передавать пакет группе устройств. Узлы сами рассылают сообщения о том, что они хотят получать групповую информацию, используя для этого специальный протокол IGMP. Участники мультикастовой группы могут находиться в разных локальных сетях. Групповые адреса не разделяются сетевую и узловую часть и обрабатываются маршрутизаторами по особому алгоритму.

Групповые рассылки предназначены для передачи в сети Интернет потоковых трансляций, которые необходимо передать на большое количество узлов. Таким образом они позволяют уменьшить количество передаваемых пакетов и снизить нагрузку на коммутационное оборудование.

Применение сетевых масок

Применение классовой адресации было оправдано для снижения нагрузки на коммутационное оборудование, а также пока количество адресов было достаточно для всех сетей и абонентов. С ростом количества узлов и исчерпанием адресного пространства появилась технология сетевых масок, позволившая использовать подсети. Подсеть - это сегмент классовой сети, не входящий в диапазоны других подсетей. Таким образом провайдер или организация может разбить выделенную адресную сеть на более маленькие подсети и изолировать их друг от друга. На практике подсеть соответствует сегменту канального уровня (сети Ethernet или Wi-Fi). Подсети позволяют обойти ограничения физического уровня на количество устройств или максимальную удаленность абонентов. Устройство, выполняющее разбиение на подсети и осуществляющее связь между подсетями, называют маршрутизатором. Маршрутизатор имеет несколько интерфейсов с адресами в каждой из подсетей.

Для создания подсетей используют узловую часть адреса, таким образом можно разбить сеть класса С на несколько сетей меньшего размера. Для этого используется маска сети, сетевая настройка, с которой мы уже знакомы. Стандартная сетевая маска, обычно предлагаемая компьютером 255.255.255.0, соответствует сети класса С и позволяет адресовать 254 узла. Применив данную маску к сети класса В, мы можем разбить ее на 254 подсети, в каждой из которых будет по 254 узла.

Маска 255.255.255.0 позволяет разбить сеть класса В на 254 подсети по 254 узла в каждой. Маску сети также могут называть префиксом. Префикс сети несет тот же самый смысл, но записывается в другом виде. Рассмотрим на примера стандартные маски и префиксы для классов сетей:

Сеть класс А – 11111111.00000000.00000000.00000000 (255.0.0.0) /8;

Сеть класс B – 11111111.11111111.00000000.00000000 (255.255.0.0) /16;

Сеть класс C –– 11111111.11111111.11111111.00000000 (255.255.255.0) /24.

Маски подсетей стандартных классов IP-адресов

Маска подсети – это число, записанное в десятичном или двоичном формате, которое позволяет отделить сетевую и узловую часть IP-адреса.

Префикс сети вычисляется по количеству 1 входящим в ее состав. Одна из особенностей масок в том, что маска может быть образована только 1, идущими подряд.

Сетевой префикс обозначается символом «/» после IP-адреса и записывает количество единиц в маске (например: 192.168.1.0/24)

128.0.0.0 = /1

255.0.0.0 = /8

255.255.0.0 = /16

255.255.255.0 = /24

255.255.255.0 = /24

255.255.255.192 = /26

По маске можно всегда можно вычислить префикс. Префикс удобнее использовать человеку для вычисления доступного количества адресов.

Дополняя сетевой адрес устройства маской, можно отказаться от классовой адресации, что позволит более гибко пользоваться существующей адресной системой.

Рассмотрим пример, адрес 190.23.44.205, он попадает в диапазон 128-191, то есть адрес относится к сетям класса B. Номер сети определяется по первым двум байтам, дополненным двумя нулевыми байтами - 190.23.0.0, а номер узла - 0.0.44.205. Если адрес использовать вместе с маской 255.255.255.0, тогда номер сети будет 190.23.44.0, а не 190.23.0.0, как было бы в случае применения классовой адресации.

Приватные или серые адреса

Зарезервированные немаршрутизируемые диапазоны адресов приведены в таблице ниже.

Диапазон	Маска	Кол-во узлов
10.0.0.0.-10.255.255.255.	255.0.0.0	≈16,5 млн
172.16.0.0.-172.32.255.255	255.255.0.0	≈ 65,5 тыс
192.168.0.0.-192.168.255.255.	255.255.255.0	254

Данные адреса могут быть использованы внутри локальной сети провайдера, организации или дома без обращения в ICANN. Доступ в сеть Интернет узлов обладающих приватными адресами осуществляется с применением технологии NAT (Network Address Translation), которая подменяет приватный адрес абонента на белый (маршрутизируемый) адрес маршрутизатора.

Типы IP-адресов и рассылок

- Сетевой адрес (network address);
- Широковещательный адрес / broadcast;
- Узловой адрес / unicast;
- Групповой адрес / multicast;
- Ближайшая группа / anycast.

Маршрутизация

Маршрутизация (routing) – поиск маршрута передачи пакетов из одной сети в другую с целью их доставки к адресу назначения. При маршрутизации необходимо учитывать изменения в топологии сети и загрузку каналов связи и маршрутизаторов.

Продвижение (forwarding) – передача пакета внутри маршрутизатора в соответствии с правилами маршрутизации.

Маршрутизатор может использовать следующую информацию:

- IP-адрес узла назначения;
- IP-адрес соседнего маршрутизатора, который он может использовать как путь по умолчанию;
- Маршруты к удаленным сетям (через доступные интерфейсы);
- Метрики маршрутов (определяющие какой путь лучше использовать для передачи);
- Способы обслуживания и обновления таблицы маршрутизации.

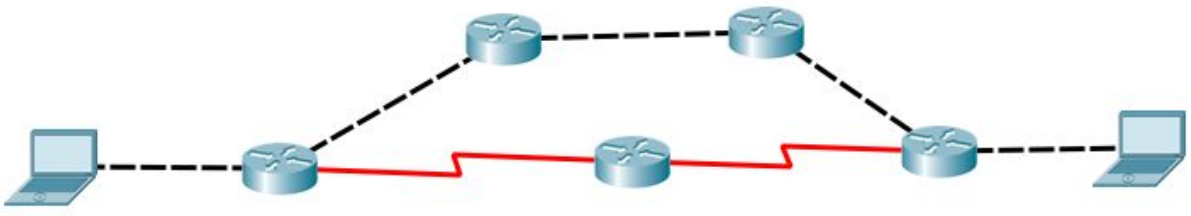
Статическая маршрутизация — используется в небольших сетях, где топология простая и фиксированная. Маршрутную информацию в таблицу маршрутизатора вносит сетевой администратор.

- Преимущества: снижена нагрузка на процессор и сеть, в связи отсутствием передачи служебной информации. Защищено от некорректного обновления таблицы извне.
- Недостатки: администратор должен хорошо знать структуру сети. Необходимо обновление настроек всех маршрутизаторов при изменение топологии или добавление новой сети. Не используется в крупных сетях.

Динамическая маршрутизация — используется в средних и крупных сетях. Маршрутная информация вычисляется на основе данных поступавших от соседних маршрутизаторов. Для обмена данными используется протокол динамической маршрутизации.

- Преимущества: быстрее настройка и проще в администрирование.
- Недостатки: использование процессора и передача служебной информации между маршрутизаторами для вычисления оптимальных маршрутов, что также нагружает сеть.

В многосвязных сетях при использовании различных протоколов маршрутизации могут использоваться различные маршруты для передачи информации между двумя узлами. Все протоколы динамической маршрутизации делят на 2 группы: протоколы вектора расстояния и протоколы состояния связи.



Протоколы вектора расстояния (Distance vector) — также называемые дистанционно векторными, используют алгоритм кратчайшего пути для поиска маршрута до удаленной сети. Каждый переход (перенаправление) пакета с помощью маршрутизатора называют хопом (HOP). Протоколы этого типа вычисляют маршрут согласно количеству переходов без учета производительности канала. Примерами таких протоколов являются: RIP, IGRP.

- К преимуществам можно отнести то, что они меньше нагружают процессоры маршрутизаторов и сеть, а недостаток - не эффективный учет пропускной способности и загруженности каналов.

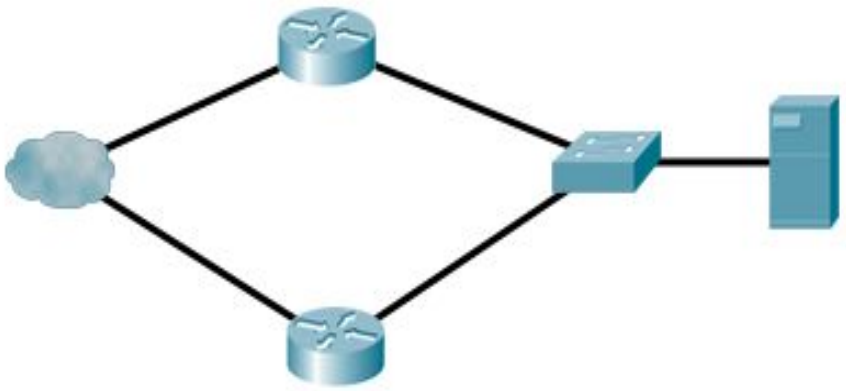
Протоколы состояния связи (Link state) — также называются «протоколами состояния канала». Все маршрутизаторы в сети, на которых запущен протокол, содержат и постоянно обновляют три таблицы. Первая отслеживает соседние устройства, вторая содержит топологию всей сети и третья используется для маршрутизации пакетов. Данные протоколы более эффективно учитывают текущее состояние сети, но сильнее утилизируют каналы связи и аппаратные мощности устройств в связи с

тем, что постоянно производят мониторинг состояния сети и обновления маршрутных таблицу. Устройства, использующие протокол состояния связи, обладает большей информацией о сети, чем протоколы вектора расстояния. Примерами протоколов состояния связи являются: OSPF, IS-IS.

- К недостаткам можно отнести то, что данная группа протоколов создает большую нагрузку на вычислительные ресурсы, и, в случае сбоя, тратится больше времени на конвергенцию в сети (конвергентная сеть – сеть, в которой все маршрутизаторы обладают актуальными данными о состояние сети).

Балансировка трафика

В терминологии компьютерных сетей балансировка нагрузки или выравнивание нагрузки (англ. load balancing) — метод распределения заданий между несколькими сетевыми устройствами (например, серверами) с целью оптимизации использования ресурсов, сокращения времени обслуживания запросов, горизонтального масштабирования кластера (динамическое добавление/удаление устройств), а также обеспечения отказоустойчивости (резервирования).



Качество обслуживания в сети QoS

Под термином Quality of Service в компьютерной сети называют вероятность того, что сеть передачи данных соответствует SLA (соглашение об уровне обслуживания).

Качество связи в сети можно оценить следующими основными параметрами:

- Полоса пропускания (Bandwidth) – характеризует пропускную способность канала передачи данных. Измеряется в бит/с, Кбит/с, Мбит/с. Скорость, в отличие от данных, всегда измеряется в битах, а не байтах.
- Время задержки при передаче данных, измеряется в миллисекундах (мс).
- Джиттер. Колебание времени задержки между доставкой пакетов.

- Количество потерянных пакетов.

Формат IPv4-пакета

- Версия (4 бита) – служит для определения IPv4, IPv5(ST) или IPv6. Для IPv4 содержит значение 4.
- Длина заголовка (4 бита) – в 32 битных словах. (на картинке эти слова пронумерованы в левой части. Например, если нет опций, то длина будет 5). Т.е. чтобы получить в битах длину заголовка, необходимо значение умножить на 32, в байтах – то на 4.

Обратите внимание, первый байт IPv4 пакета часто имеет значение 45 (проверьте в Wireshark).

Далее следует:

- тип обслуживания (Differentiated Services Code Point — DSCP);
- общая длина – длина пакета в байтах (всего, т.е. и заголовка и данных вместе).

Второе слово IPv4 пакета полностью управляет фрагментированием пакетов.

- Идентификатор пакета – служит для идентификации пакетов, главным образом при фрагментации (например, на маршрутизаторе при необходимости передать через канал с меньшим MTU). В случае фрагментации фрагменты пакета обладают одним и тем же идентификатором.
- Флаги – также служат для управления фрагментацией (или ее запретом). Если установлен флаг «не фрагментировать», в случае, если необходимо отправить пакет через канал с меньшим MTU, пакет будет отброшен, а в ответ направлено ICMP-сообщение: требуется фрагментация, но фрагментация запрещена. Если флаг сброшен, пакет будет фрагментирован.
- Смещение фрагмента – также служит для нужд фрагментации, для восстановления исходного (не фрагментированного) пакета. Если пакет фрагментирован, то каждый фрагмент обладает одним и тем же идентификатором, но разными смещениями, которые позволят собрать пакет обратно. Если фрагмент не последний, установлен соответствующий флаг.

Третье слово содержит два очень важных и один неиспользуемый на практике параметры.

- Время жизни – TTL (time to live) – максимальное число хопов (прыжков). При каждом прохождении маршрутизатора декрементируется. Если TTL=0, пакет отбрасывается. Важное

значение. Изначально планировалось, что должно изменяться в секундах. Но реализовать было сложно, потому сделали уменьшение TTL простым декрементам. Как оказалось, нет ничего более постоянного, чем временное. В IPv6 аналогичное поле называется числом прыжков. К слову сказать, TTL в секундах действительно применяется в DNS и служит для указания времени кеширования записи в секундах, а учитывать время прохождения пакета в сети умеет протокол NTP (Network Time Protocol).

- Контрольная сумма заголовка – только для заголовка, не для данных! На практике не используется, так как не несет полезной нагрузки, более того, заголовок частично меняется (например, TTL).
- Поле протокол – тип протокола верхнего уровня (ICMP, UDP, TCP и другие, которые инкапсулируются в IP).

Следующие два слова Адреса отправителя и получателя в формате Ipv4 (по 8 октетов).

Опции – служат для расширения и надстроек. Может содержать до 10 слов. (число ограничено максимальным значением длины заголовка в словах, так как 4 бита позволяют указать до 16 значений, учитывая слова самого заголовка).

Обратите внимание, что маска сети в протоколе не передается. Маска является принадлежностью непосредственно сетевого интерфейса. Поэтому, хосты с IP-адресами в разных сетях, но которые входят в диапазон обеих подсетей будут пинговать друг друга.

Например, 172.17.10.1/255.255.255.0 и 172.17.10.100/255.255.255.128

Проверьте!

Разрешение IPv4 в MAC-адреса

IP-адреса нужны при маршрутизации (3 уровень OSI). При каждом прохождении маршрутизатора, маршрутизатор анализирует IP-адреса и определяет сети, с которыми нужно иметь дело.

Но в рамках сегмента используется физическая адресация. Сетевыми картами чтобы отбрасывать кадры, которые в поле получателя имеют MAC-адрес, отличный от адреса карты. Коммутаторы, ведущие таблицы соответствия своих портов и MAC-адресов.

Поэтому каждый хост ведет таблицы ARP (address resolution protocol) – соответствие IP адресов и MAC-адресов.

Посмотрите таблицу на Вашем компьютере.

```
arp –a
```

При этом записи обозначаются, как статические и динамические.

Статические: вы можете принудительно сопоставить MAC-адрес сетевой карты вашего шлюза с его IP-адресом. (arp –s)

Динамические записи определяются с помощью протокола ARP.

Если MAC-адрес не известен, то отправляется ARP-запрос (вкладывается в кадр, занимает промежуточное положение между канальным и сетевым уровнем), с указанием IP-адресов отправителя и назначения, и MAC-адресов отправителя и назначения. В качестве MAC-адреса назначения выступает широковещательный адрес FF:FF:FF:FF:FF:FF

Для броадкаст-запросов ARP не используется, широковещательный адрес FF:FF:FF:FF:FF:FF используется сразу в качестве MAC-адреса назначения в заголовках кадра, несущего броадкаст-пакет. При этом узлы, получив сообщение, отвечать будут на юникаст-адрес. Они будут использовать ARP-запросы.

Соответственно, коммутатор направляет запрос во все порты, а нужная машина (искомый получатель, либо шлюз), направляет ответ с указанием MAC-адреса. Результат кешируется в arp таблицу, и пакеты на нужный IP-адрес уже инкапсулируются в кадры с MAC-адресом получателя, взятым из таблицы.

Протокол уязвим, потому как можно подделать ARP-ответ. Атака называется ARP-spoofing, одна из реализаций «человек посередине».

Для защиты от нее следует использовать либо статические ARP, либо использовать Access List, либо изолировать широковещательные домены, т.е. на канальном уровне использовать VLAN, либо на сетевом – тоннели.

Формат ARP-запроса

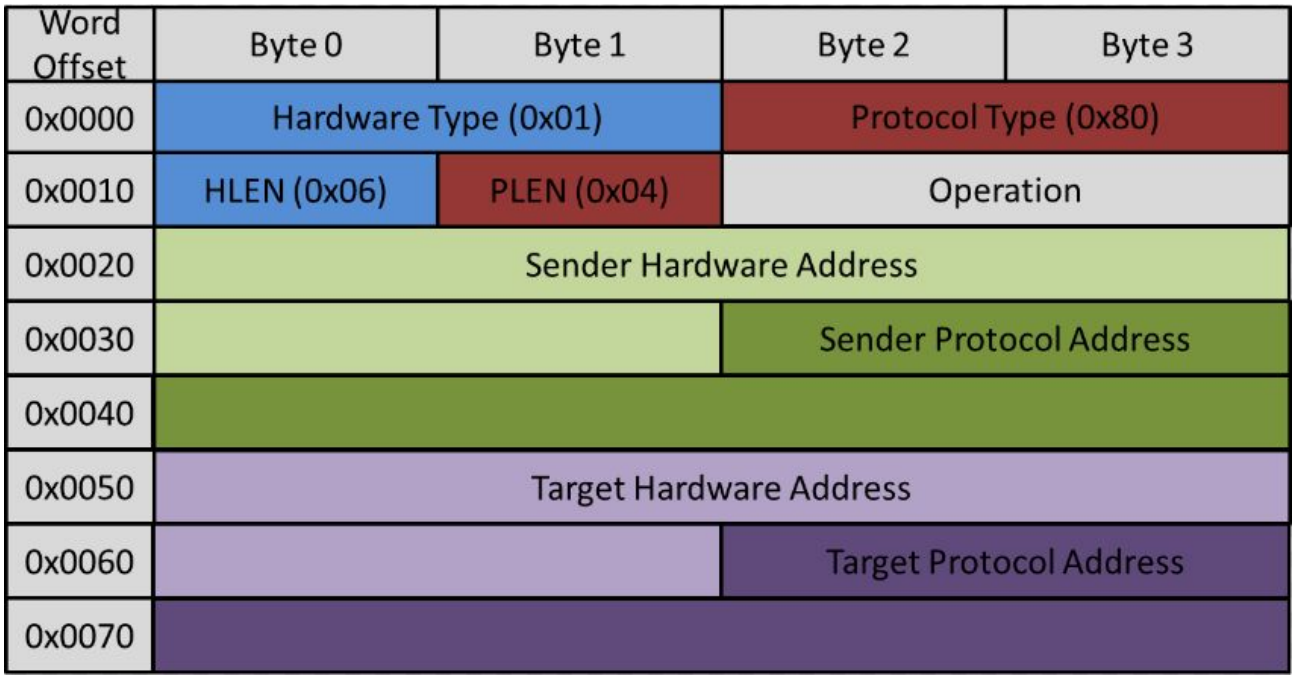
<table>
<tr><th>Октет</th><th colspan="32">0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31</th></tr>
<tr><td>0</td><td colspan="16">Hardware Type (HTYPE)</td><td colspan="16">Protocol Type (PTYPE)</td></tr>
<tr><td>4</td><td colspan="8">Hardware length (HLEN)</td><td colspan="8">Protocol length (PLEN)</td><td colspan="16">Operation (OPER)</td></tr>
<tr><td></td><td colspan="32">Sender hardware address (SHA)</td></tr>
<tr><td></td><td colspan="32">Sender protocol address (SPA)</td></tr>
<tr><td></td><td colspan="32">Target hardware address (THA)</td></tr>
<tr><td></td><td colspan="32">Target protocol address (TPA)</td></tr>
</table>

ARP-запросы и ответы сразу вкладываются в поле данные Ethernet-кадра.

- HTYPE – тип протокола канального уровня. Например, Ethernet — 0x0001.
- PTYPE – тип протокола сетевого уровня. Например, IPv4 — 0x0800.
- HLEN — длина физического адреса. Например, для Ethernet — 6 байт.
- PLEN — длина сетевого адреса. Например, для IPv4 — 4 байта.
- OPER — тип операции.
    - 1 — ARP-запрос.
    - 2 — ARP-ответ.
    - 3 — ReverseARP(RARP)-запрос.
    - 4 — ReverseARP(RARP)-ответ.
- SHA — физический адрес отправителя. Как правило, MAC-адрес отправителя.
- SPA — сетевой адрес отправителя. Как правило, IP-адрес отправителя.

- THA — физический адрес получателя. Если MAC-адрес не известен, устанавливается в 00:00:00:00:00:00 (в то время, как аналогичное поле Ethernet-кадра будет иметь значение FF:FF:FF:FF:FF:FF)
- TPA — сетевой адрес получателя. Как правило, IP-адрес получателя.

Так как длина MAC-адреса и IP-адреса не кратны, на практике формат ARP-пакета выглядит следующим образом:



RARP, BOOTP, DHCP

Стоит отметить, что существует целое семейство ARP-протоколов. ARP — наиболее известный и часто используемый протокол. Для решения обратной задачи (известен свой MAC-адрес, не известен свой IP-адрес) применялся протокол RARP (Reverse ARP), Формат пакета совпадает с ARP, но используются другие OPER-коды (3 и 4), но самое главное, в поле тип пакета канального уровня у RARP- другой код протокола. То есть несмотря на сходства, это разные протоколы, обрабатываемые разными механизмами. Тем не менее, если, например, сервис, отвечающий за RARP, получит ARP-запрос, он передаст его на обработку сервису, отвечающему за ARP. В случае использования RARP, в отличие от ARP, требовался выделенный сервер, содержащий таблицу соответствий MAC-адрес и IP-адресов. Для обращения к серверу использовался broadcast-адрес. Проблемой такого подхода оказалась необходимость использовать отдельного RARP-сервера для каждого широковещательного домена, потому на замену RARP был создан BOOTP — Bootstrap Protocol. Он обладал многими новыми возможностями, позволял не только назначить IP-адрес, но и другие известные нам параметры, такие как маску сети, адрес маршрутизатора по умолчанию, DNS-сервер и даже указать для бездисковой машины TFTP (trivial file transfer protocol) сервер с образом ОС для распаковки RAM. BOOTP протокол инкапсулировался в UDP-дейтаграммы и мог маршрутизироваться. Но и у BOOTP были проблемы. По прежнему при появлении каждой новой машины необходимо было вручную настраивать BOOTP-сервер. Поэтому следующим развитием BOOTP-протокола стала его новая версия, получившая и новое название Dynamic Host Configuration Protocol — протокол

динамической настройки хоста, который позволяет динамически настраивать подключающиеся машины и широко используется в настоящее время.

ICMP (Internet control message protocol)

Offsets	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Type								Code								Checksum															
4	32	Rest of Header																															

Несмотря на то, что ICMP-пакеты упаковываются в поле данных IP-пакетов, ICMP относится к сетевому уровню, и фактически является частью механизма IP.

В случае, если пакет не удается доставить, узел генерирует ICMP-сообщение в адрес отправителя (нет маршрута, требуется фрагментация пакета, но она запрещена флагом, хост недоступен и т.д. и т.п.).

Некоторые типы и коды ICMP-сообщений:

Тип=8.Код=0. Эхо-запрос.

Тип=0. Код=0. Эхо-ответ.

Поле данных содержит некие случайные тестовые данные (теоретически могут быть и осмысленные данные, что использовалось в свое время для организации так называемых ICMP-тоннелей).

Тип=3. Код=0. Сеть недостижима.

Тип=3. Код=1. Узел недостижим.

Тип=3. Код=3. Порт недостижим.

Тип=3. Код=4. Необходима фрагментация, но установлен флаг её запрета (DF).

Тип=11. Код 0. TTL истекло.

и т.д.

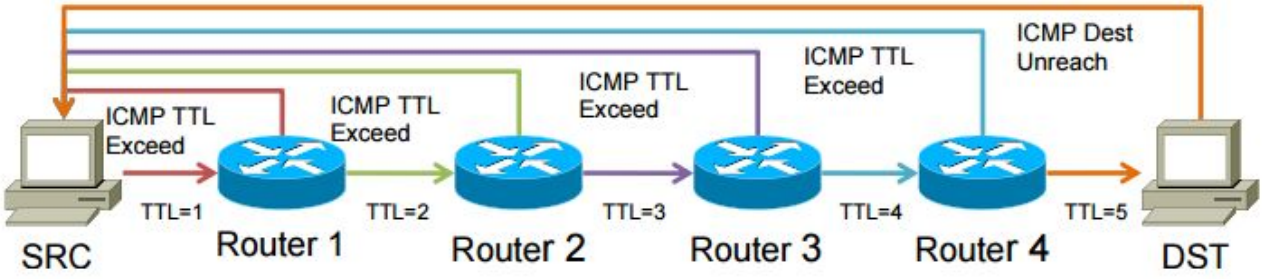
Для сообщений типа 3 и 11 в поле данных возвращается заголовок IP-пакета и начало дейтаграммы.

Ping и tracert/traceroute

Особый случай – echo-запрос. Используется для диагностики в утилитах ping и в tracert (traceroute -I). Команда ping формирует echo-запрос, содержащий некое случайное сообщение. Узел, получив echo-запрос, возвращает тоже сообщение обратно, отметив, как echo-ответ. Таким образом мы получаем сам факт того, что узел ответил, а, кроме того, информацию о времени получения ответа, о дублированиях и потерях.

Traceroute в Linux по умолчанию использует UDP-протокол (но с помощью ключа -I можно использовать и ICMP).

Рассмотрим, как работает tracert с помощью ICMP (на самом деле у UDP-трассировки лежит аналогичная идея).



Клиент посылает ICMP на узел назначения, точно также, как и при ping, но TTL выставляется в единичку. Первый маршрутизатор, получив сообщение, отбрасывает ICMP-сообщение, в ответ генерирует ICMP-сообщение об ошибке (TTL истекло), с указанием собственного IP-адреса в качестве адреса отправителя (его мы и видим в выдаче).

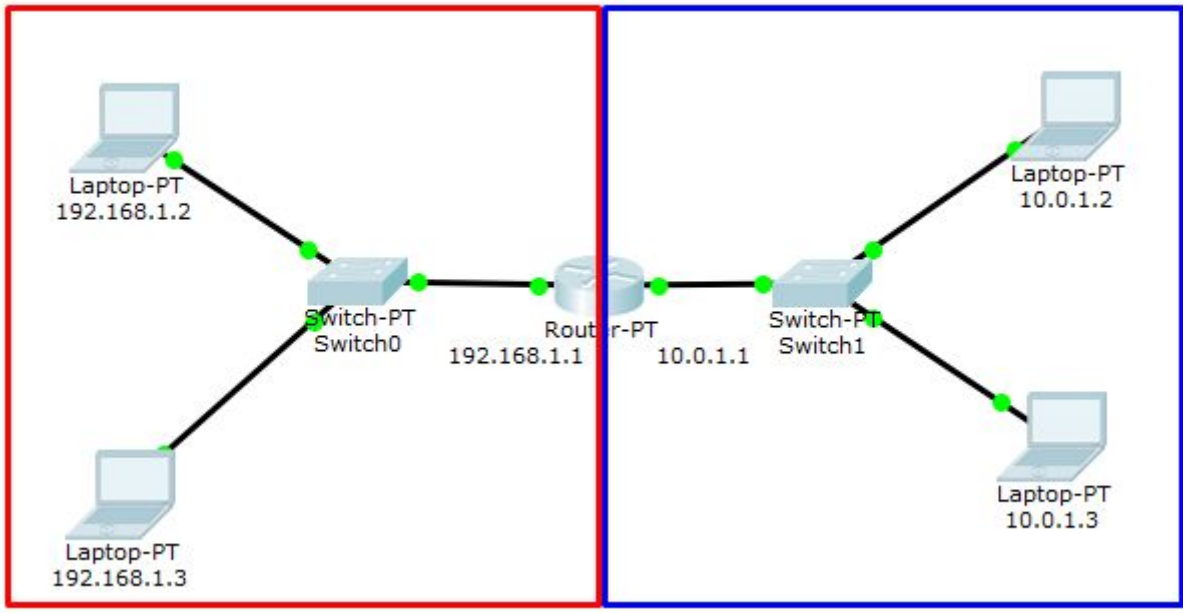
Сначала высылаются 3 сообщения с TTL=1. Затем с TTL=2 и мы получаем сообщение от второго прыжка. И так далее, пока не получим не сообщение об ошибке, а ответ от узла назначения (Echo replay), что означает, что трассировка завершена.

Статическая маршрутизация

Если несколько узлов объединены в сеть на канальном уровне, то дополнительная настройка маршрутизация не требуется (но маршрутизация уже используется, как минимум, существует маршрут, связывающий сеть, в которую входит в хост и сетевой интерфейс. Почему даже для работе в одной локальной сети требуются IP-адреса). По этой же причине, чтобы можно было с помощью ping проверить связность, необходимо, чтобы на сетевых интерфейсах IP-адреса входили в одну сеть.

Если мы объединим две сети с помощью узла, у которого в наличии два сетевых интерфейса (по одному на каждую сеть), пакеты могут попадать из одной сети в другую, а могут и не попадать. К примеру, в Linux необходимо включить IP Forwarding (/etc/sysctl.conf). Чтобы такая связка заработала, также надо, чтобы на хостах каждой из сетей IP-адрес соответствующий сети (соответствующего сетевого интерфейса) маршрутизатора был указан как маршрут по умолчанию (таким образом на каждом из хостов будет уже два маршрута — один для своей сети в сетевой интерфейс, и другой — маршрут по умолчанию на адрес шлюза). На маршрутизаторе (кроме, возможно, включения IP Forwarding на Linux-машинах) дополнительных маршрутов поднимать не надо, они уже подняты при включении соответствующих сетевых интерфейсов.

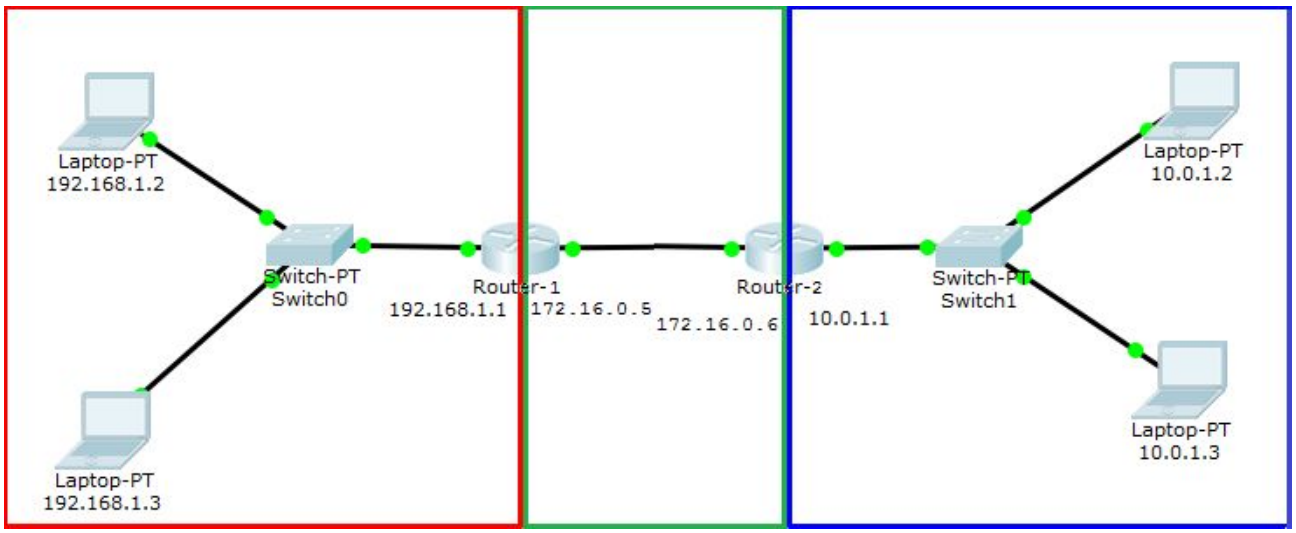
Настройка статической маршрутизации в маршрутизаторах Cisco



Для этого примера достаточно поднять сетевые интерфейсы на маршрутизаторе, а на клиентах прописать соответствующие IP-адреса маршрутизатора.

```
>ena
#conf term
#int fa0/0
#ip addr 192.168.1.1 255.255.255.0
#no shut
#int fa0/1
#ip addr 10.0.1.1 255.255.255.0
#no shut
#end
#wr
```

Настройка сетевых интерфейсов уже добавит маршруты в данные сети через необходимые сетевые интерфейсы. Сложнее настройка, когда имеется уже два маршрутизатора.



Router-1

```
>ena
#conf term
#! настраиваем интерфейсы (и маршруты для них будут созданы)
#int fa0/0
#ip addr 192.168.1.1 255.255.255.0
#no shut
#int fa0/1
#ip addr 172.15.0.5  255.255.255.0
#no shut
#! добавляем маршрут для 10.0.1.0 сети через Router-2
#ip route 10.0.1.0 255.255.255.0 172.16.0.6
#end
#wr
```

Router-2

```
>ena
#conf term
#! настраиваем интерфейсы (и маршруты для них будут созданы)
#int fa0/0
#ip addr 172.15.0.6  255.255.255.0
#no shut
#int fa0/1
#ip addr 10.0.1.1 255.255.255.0
#no shut
#! добавляем маршрут для 192.168.1.0 сети через Router-1
#ip route 192.168.1.0 255.255.255.0 172.16.0.5
#end
#wr
```

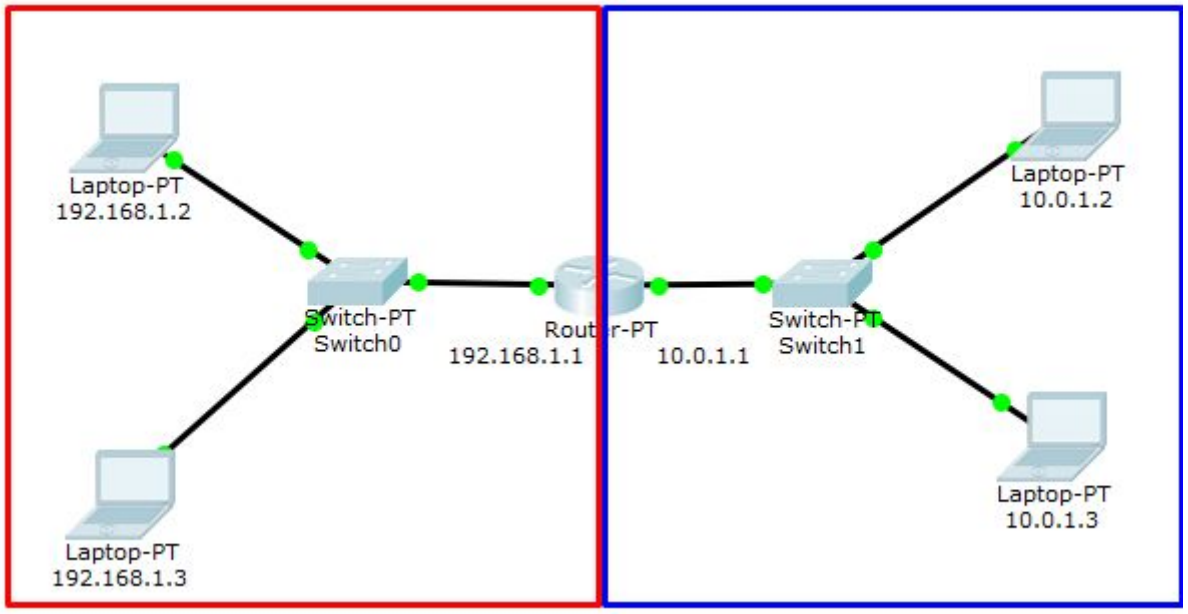
Обратите внимание, что при добавлении каждой новой сети, маршруты в нее нужно будет указывать на каждом маршрутизаторе, даже если она непосредственно не подключена к этому маршрутизатору.

При этом в качестве шлюза нужно будет указывать ближайший маршрутизатор, через который трафик пойдет к искомому узлу.

Статическая маршрутизация в Linux

Рассмотрим аналогичные примеры объединения сетей, где и компьютеры, и маршрутизаторы работают с операционной системой Linux

Пример объединения двух сетей с одним маршрутизатором.



Если на Linux-компьютере 192.168.1.2 посмотрим таблицу маршрутизации, мы увидим:

```
user@user-virtual-machine:~$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
default         192.168.1.1     0.0.0.0         UG    100    0        0 ens33
192.168.1.0     *               255.255.255.0   U     100    0        0 ens33
```

Но это если Router имеет и подключение к внешней сети. Может быть и такая картина:

```
user@user-virtual-machine:~$ route
Kernel IP routing table
Destination     Gateway         Genmask         Flags Metric Ref    Use Iface
10.0.1.0        192.168.1.1     255.255.255.0   UG    100    0        0 ens33
192.168.1.0     *               255.255.255.0   U     100    0        0 ens33
```

Таблица маршрутизации присутствует на каждой машине с поддержкой стека TCP/IP. Даже на Windows машинах. Когда операционная система получает к отправке пакет, она по IP-адресу и маске сравнивает, в какой сетевой интерфейс и какой шлюз его отправить. В первом случае у нас имелись

два маршрута. Один маршрут для устройств, подключенных к той же сети, 192.168.1.0 в сетевой интерфейс ens33. И другой маршрут по умолчанию — на шлюз 192.168.1.1 подключенный, обратите внимание, к той же сети. Пакеты на все другие IP-адресу будут отправляться туда.

Во втором случае у нас два маршрута.

Пакеты на адреса в сети 192.168.1.0 отправляются в сетевой интерфейс ens33, так как доступны через канальный уровень.

Пакеты на адреса в сети 10.0.1.0 отправляются на адрес шлюза с адресом 192.168.1.0 в этой же сети.

Остальные пакеты в данной конфигурации будут отброшены с сообщением No Route to host/

Для того, чтобы заработало на Linux-маршрутизаторе необходимо включить IP-forwarding:

```
user@router:~$ echo 1 >/etc/proc/sys/network/ipv4/ip_forward
```

Чтобы IP-forwarding работал постоянно, необходимо исправить соответствующую опцию в файле /etc/sysctl.conf

Сложнее настройка статической маршрутизации между двумя маршрутизаторами.

Предположим, что у нас имеются уже два маршрутизатора (в данном случае также в роли маршрутизаторов играют Linux-машины).

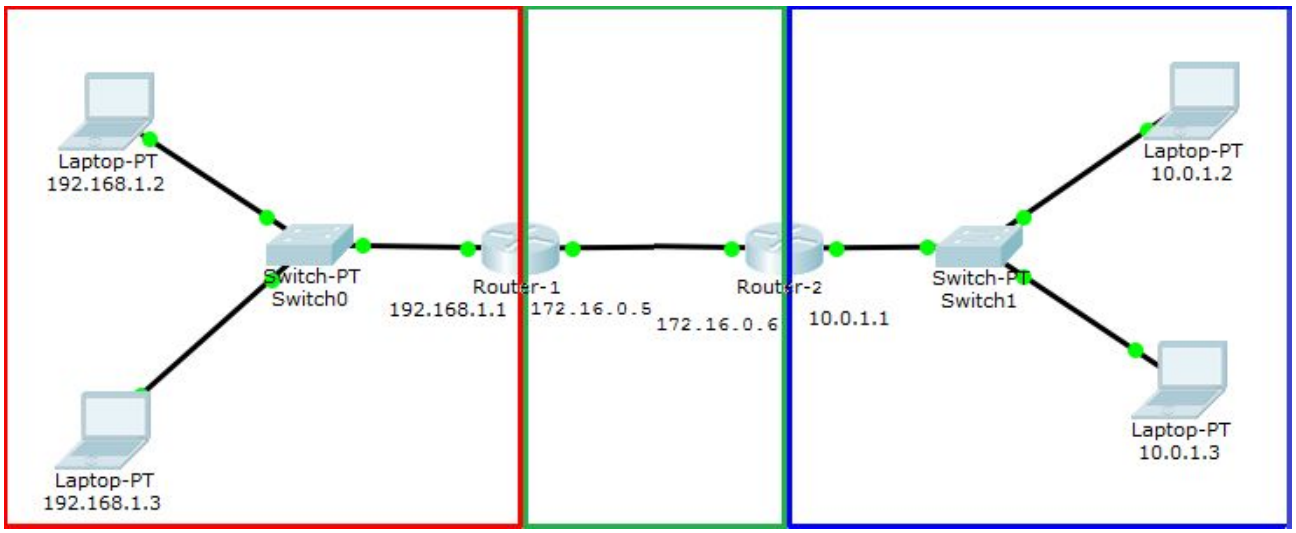
У каждой из машин имеется по два сетевых интерфейса, у клиентских машин ближняя машина-маршрутизатор указана в маршруте по умолчанию.

Но при этом машина 192.168.1.1, получая сообщение от, например, 192.168.1.2. ничего не знает о сети 10.0.1.0. В прошлом случае знала, так как подобный маршрут возникает уже при поднятии соответствующего сетевого интерфейса.

Вы в этом легко можете убедиться, подняв нужный сетевой интерфейс (настоящий, или как в примере, псевдоним) и посмотрев таблицу маршрутов:

```
# ifconfig ens33:dummy 172.30.1.1 netmask 255.255.255.0 up
# route
# ifconfig ens33:dummy down
```

Итак, оба маршрутизатора должны быть настроены на пересылку пакетов (IP Forwarding необходимо разрешить на обеих машинах).



Но также надо указать соответствующие маршруты и на маршрутизаторах.

В Linux такая настройка выполняется следующим образом:

На Router-1

```
# route add -net 10.0.1.0 netmask 255.255.255.0 gw 172.16.0.6
```

Мы указали, что маршрут в сеть 10.0.1.0 знает машина 172.16.0.6. Обратите внимание, мы указываем адрес в нашей сети.

На Router-2

```
# route add -net 172.16.0.0 netmask 255.255.255.0 gw 172.16.0.5
```

Аналогично.

После этого машины из сетей 192.168.1.0 и 10.0.1.3 смогут работать друг с другом. Если же между Router-1 и Router-2 нет прямой связи, а кроме того мы хотим адресовать частные адреса, нам понадобится уже тоннель.

Домашнее задание

1. В приложенном файле в Cisco Packet Tracer связать файлы с помощью статической маршрутизации.
2. Проследить в Cisco Packet Tracer, Wireshark работу протоколов arp, icmp (например, используя tracert или traceroute -I)

Дополнительные материалы

1. Таненбаум Э., Уэзеролл Д. Т18 Компьютерные сети. 5-е изд. — СПб.: Питер, 2012. — 960 с. (Глава 5)
2. http://just-networks.ru/seti-tcp-ip/protokol-ipv4
3. https://habrahabr.ru/company/cbs/blog/276863/
4. https://www.atraining.ru/arp-inarp-rarp-proxy-gratuitous-dai-sticky-and-more/
5. https://habrahabr.ru/post/108690/
6. http://xgu.ru/wiki/Маршрутизация_в_Linux

Используемая литература

Для подготовки данного методического пособия были использованы следующие ресурсы:

1. http://network-journal.mpei.ac.ru/cgi-bin/main.pl?l=ru&n=24&pa=11&ar=1
2. https://habrahabr.ru/post/281272/
3. https://rtfm.co.ua/unix-traceroute/
4. http://netwild.ru/tracert_and_tracerout/