GeekBrains

Компьютерные сети

Ethernet. Канальный уровень

Ethernet. MAC-адрес. MTU. CSMA/CD. Домен коллизий

Введение

OSI/ISO	TCP/IP (DOD)
7. Прикладной уровень	4. Уровень приложений
6. Уровень представления	
5. Сеансовый уровень	
4. Транспортный уровень	3. Транспортный уровень
3. Сетевой уровень	2. Сетевой уровень
2. Канальный уровень	1. Уровень сетевых интерфейсов
1. Физический уровень	

Канальный уровень служит для передачи структурированного набора битов между устройствами, объединенными общей средой передачи данных и находящимися в одной локальной сети (точнее в широковещательном домене, о котором мы сегодня поговорим). Как правило, в одной физической сети (витая пара, оптоволокно, радиоканал на одной и той же частоте) возможен логический, «виртуальный вариант», например в случае туннелирования. В этом случае вместо кодирования через параметры физической среды происходит дальнейшая упаковка с помощью протоколов того же (PPPoE) или вышестоящего (PPTP) уровня.

В модели OSI/ISO канальный уровень по счету второй, следует после физического. В модели TCP/IP он объединен с физическим и называется уровнем сетевых интерфейсов. Смысл в этом есть: провести четкую грань между физическим и канальным уровнем невозможно. Например, в структуре кадра протокола Ethernet 802.3 поле преамбулы (самое первое поле, которое фактически даже не входит в сам кадр) служит для синхронизации приемника и передатчика. Этим и определяется структура преамбулы, которая состоит из семи байт, содержащих одну и ту же битовую последовательность: 10101010. Фактически она нужна, чтобы сформировать определенную форму сигнала, т. е. идет речь напрямую о физическом уровне. С другой стороны, устройства физического (L1) и канального (L2) уровня значительно отличаются: если на уровне L1, как правило, есть только физические соединения (на уровне электрической схемы или порядка подключения патч-кордов в патч-панели), то L2 — это «умные устройства», умеющие анализировать заголовки кадров (фреймов) канального уровня, имеющие реализацию определенного алгоритма, процессор или реализуемую на программируемой логической микросхеме логику и оперативную память. Если все устройства в сети соединены общей шиной или сетевым концентратором (хабом, устройством L1), то выполнять задачи канального уровня будут сами сетевые карты. В дальнейшем появились сетевые мосты, а потом и

сетевые коммутаторы, умеющие анализировать заголовки проходящих через них кадров и принимать решения о направлении в тот или иной порт коммутатора.

Сначала определим, как решается проблема идентификации получателя. Когда несколько устройств соединены шиной или концентратором, сигнал получают все. Если топология — «шина» или «звезда» с концентратором, сетевая карта анализирует заголовок канального уровня и определяет, ей ли предназначено это сообщение или нет. Если MAC-адрес получателя совпадает с адресом сетевой карты (либо является широковещательным), кадр принимается и отдается в реализацию сетевого стека операционной системы. Если нет — отбрасывается.

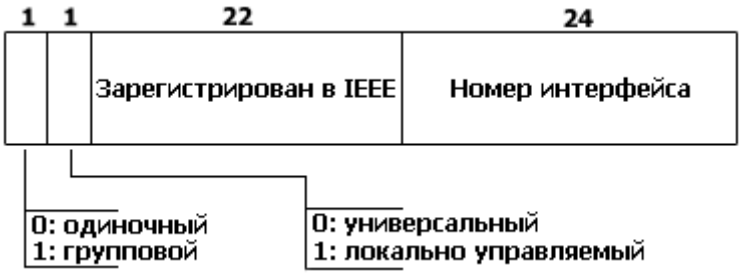
MAC-адрес

MAC расшифровывается как Media Access Control (управление доступом к среде). MAC-адрес, или MAC-48, имеет длину 48 бит (6 байт) и иногда называется Hardware Address, аппаратный адрес. MAC-48 — один из вариантов аппаратных адресов, используемых для сходных задач. Другие варианты — EUI-48, EUI-64. Последний используется в FireWire, а также в IPv6. MAC-48 получил широкое распространение, применялся в Token Ring, FDDI, используется в Ethernet, Wi-Fi, WiMAX и т. д.

Структура MAC-адреса

Задача MAC-адреса — обеспечить уникальную адресацию сетевых интерфейсов устройств, работающих в одной локальной сети (использующих одну и ту же среду передачи данных).

MAC-адрес состоит из 6 октетов, т. е. из 48 бит, и имеет следующую структуру:



Структура MAC-адреса

Первый бит показывает, является ли получатель одиночным или данный кадр — широковещательный.

Часто используемый широковещательный адрес FF:FF:FF:FF:FF:FF, который означает «все получатели внутри одной локальной сети». То есть фрейм с таким адресом получателя будет обработан всеми устройствами, которые физически могут получить его (т. е. находятся в одном локальном сегменте сети).

Каждый раз, когда срабатывает ARP-запрос, протокол DHCP или PPPoE пытается обнаружить сервер, а также в случаях бродкастного запроса применяется этот адрес.

Существуют и другие варианты широковещательного адреса, например, в случае мультикаст-рассылки. Все подписчики будут иметь MAC-адрес получателя, который принадлежит группе подписавшихся на мультикаст-поток хостов. Данный MAC-адрес является виртуальным и вычисляется исходя из IP-адреса группы рассылки. Мультикастный MAC-адрес не может быть адресом отправителя, только адресом получателя.

Второй бит определяет, будет ли данный адрес глобально уникальным («универсальный») или он назначен локально (т. е. возможны конфликты совпадающих MAC-адресов). Стоит отметить, что и глобально уникальный адрес также можно присвоить сетевой карте во многих случаях: «уникальным» он будет постольку, поскольку карте не присвоили новый адрес или адрес сетевой карты не присвоили другому устройству. Отметим, что, несмотря на глобальную уникальность, она нужна только в рамках локальной сети, чтобы можно было различать соседей по используемому каналу. Хотя идеи с использованием «уникальности» MAC-адреса для сетевой идентификации тоже существовали.

Как обеспечивается уникальность MAC-адреса

В случае универсального адреса 22 бита, следующие после первых 2 битов — префикс производителя, зарегистрированный в IEEE. Каждый производитель ведет учет выпущенных устройств, добавляя к своему префиксу порядковый номер выпущенного устройства (нечто вроде серийного номера) или сетевого интерфейса. Таким образом, мы можем считать, что никакие два устройства или сетевые интерфейсы с универсальными MAC-адресами не будут иметь один и тот же адрес.

00:0c:ad:1d:ab:11 — пример локально администрируемого адреса.

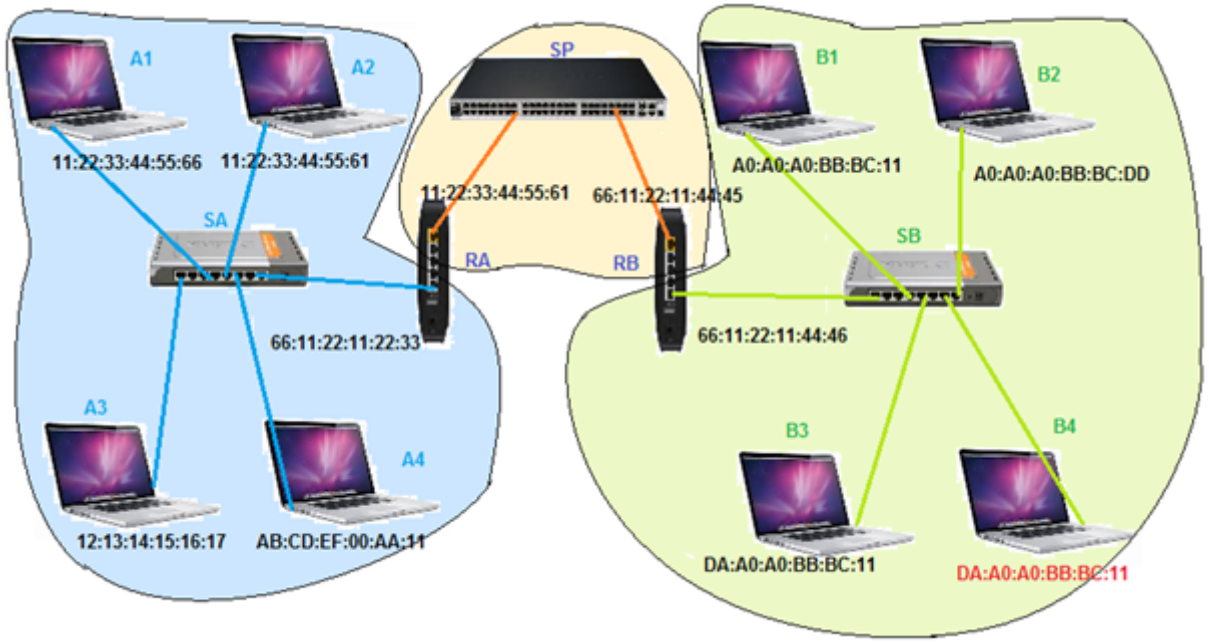
Локально администрируемые адреса выделяются для виртуальных сетевых интерфейсов (виртуальные машины, тоннели), хотя у сетевого интерфейса можно поменять прошитый заводской MAC-адрес на необходимый.

Домашние роутеры позволяют на внешнем интерфейсе (Uplink) поднимать (клонировать, привязывать) MAC-адрес сетевого интерфейса компьютера в локальной сети. Это может понадобиться, если провайдер идентифицирует ваш аккаунт (для биллинга, контроля трафика и журналирования) по MAC-адресу компьютера. Клонирование MAC-адреса позволяет к Интернету уже не один компьютер, а целую локальную сеть, используя домашний роутер.

Область использования MAC-адреса

Каждый MAC-адрес должен быть уникален среди всех адресов, доступных через одну и ту же среду передачи данных (то есть внутри одного широковещательного домена).

Обратите внимание на рисунок. Благодаря тому, что интерфейс вашего компьютера и внешний интерфейс домашнего роутера находятся в разных сетях (разных широковещательных доменах), конфликта не произойдет. Если же два сетевых интерфейса в одном сегменте будут иметь один и тот же MAC-адрес, произойдет конфликт адресов.



Сети объединяют роутеры (RA и RB), каждый из которых имеет по два сетевых интерфейса, а значит, и два MAC-адреса. Это не всегда так: для такого устройства можно использовать и только один MAC-адрес. Но мы рассмотрим простой пример, где в качестве маршрутизатора применяется обычный домашний роутер либо компьютер с двумя сетевыми картами.

В сети SA (левой) все MAC-адреса уникальны.

В провайдерской сети SP (средней) также все MAC-адреса уникальны. Несмотря на то, что роутер RA клонировал MAC-адрес компьютера A2 (видимо, он был первоначально подключен к провайдеру, поэтому его и понадобилось привязать по MAC-адресу), в «среднем» сегменте также все MAC-адреса уникальны.

В сети SB (правой) имеются два компьютера (B3 и B4) с неуникальными MAC-адресами. Конфликт MAC-адресов приведет к ошибкам в работе.

Можно ли забанить по MAC-адресу

Ответьте на вопрос. Есть форум. Злоумышленник повадился писать назойливый спам (легкий заработок, умерла тетя в Зимбабве и т. д.). Можно ли его забанить по MAC-адресу? В каких случаях?

Вариант «MAC-адрес можно сменить» в качестве корректного ответа не подойдет. Если злоумышленник работает в другой сети, мы будем видеть в приходящих кадрах в качестве

MAC-адреса отправителя MAC своего шлюза доступа. Если кто-то попытается забанить злоумышленника таким образом, пожелаем ему удачи.

Если же злоумышленник работает в той же сети, например, живет в соседнем подъезде, и форум работает на сервере в домашней сети, то есть доступен не через шлюз, либо сервер находится в серверной, и спамер — сосед по серверному шкафу, в заголовках кадра мы действительно будем видеть MAC-адрес его сетевой карты (или назначенный вручную). В этом случае бан по MAC-адресу технически действительно возможен.

Формат Ethernet-кадра

На канальном уровне Ethernet-кадр выглядит следующим образом:

Первые 6 байт — MAC-адрес получателя.

Вторые 6 байт — MAC-адрес отправителя.

Следующие 2 байта — тип протокола вышестоящего уровня (предназначен, чтобы оборудование или операционная система могли понять, какой протокол инкапсулируется — ARP, IPX, IPv4, IPv6, MPLS или другой).

Полезные данные могут иметь длину от 46 до 1500 байт. Такой размер был выбран исходя из особенностей работы через общую среду. Если размер данных меньше 46, такой кадр пройдет через сеть быстрее, чем успеет распространиться коллизия (что создает ряд сложностей), а кроме того, эффективно посчитать контрольную сумму не получится. Размер более 1500 байт, напротив, займет канал, не давая возможности другим узлам отправить сообщение. В высокоскоростных сетях (как правило, при использовании оптических каналов связи) объем полезных данных может быть больше, чем 1500 байт. В IPv6 такие фреймы называются jumbo-frame. На практике все еще часто используется 100 Мбит Ethernet с длиной полезных данных в 1500 байт. Стоит отметить, что фреймы-карлики (dwarf) с длиной полезной нагрузки 46 байт иногда встречаются в сети, например при выключении сетевого оборудования. Иногда может встречаться меньшее значение, например 42. Это связано с вариациями протокола Ethernet, содержащих те или иные дополнительные заголовки (LLC — Logical Link Control, 802.1Q VLAN и т. п.).

Контрольная сумма служит для определения, были ли данные повреждены в ходе передачи (из-за помех, наводок и т. д.). Если контрольная сумма не совпадает со вновь посчитанной получателем, кадр отбрасывается. Это единственная услуга, предоставляемая канальным уровнем, которая касается целостности данных. Контроль порядка принятых данных, оповещение о доставке, недоставке, повторной доставке, сборке последовательностей из принятых сообщений — эти задачи выполняются не на канальном уровне, а на транспортном и частично на сетевом. Контрольная сумма рассчитывается каждый раз в своей сети, то есть ничего не говорит о том, были ли повреждены данные на других участках маршрута (в других сетях). Контроль целостности самих данных, чтобы они пришли к конечному получателю от изначального отправителя, тоже не входит в задачи канального уровня.

MTU (Maximum Transmission Unit)

В выводе ifconfig в UNIX-подобных системах можно увидеть параметр MTU. На самом деле MTU — свойство любого сетевого интерфейса: это размер полезной нагрузки кадра. Сообщение, большее, чем MTU, не может быть передано одним кадром. У разных канальных технологий MTU разный. Традиционное значение для Ethernet – 1500. Для туннелирующих технологий MTU должен быть меньше, чем MTU сетевого устройства, через которое данные будут передаваться физически с учетом туннелируемых заголовков.

```
user@lvm-virtual-macshine:~$ ifconfig
ens33     Link encap:Ethernet  HWaddr 00:0c:29:1f:6b:1a
          inet addr:192.168.116.140  Bcast:192.168.116.255  Mask:255.255.255.0
          inet6 addr: fe80::bb62:f277:31cd:d92a/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:737409 errors:0 dropped:0 overruns:0 frame:0
          TX packets:3774151 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:109882858 (109.8 MB)  TX bytes:9476988136 (9.4 GB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:540029 errors:0 dropped:0 overruns:0 frame:0
          TX packets:540029 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:4808782049 (4.8 GB)  TX bytes:4808782049 (4.8 GB)

tun0      Link encap:UNSPEC  HWaddr 00-00-00-00-00-00-00-00-00-00-00-00-00-00-00-00
          inet addr:172.16.0.6  P-t-P:172.16.0.5  Mask:255.255.255.255
          inet6 addr: 2002:b9c3:1ba4:cafe::1000/64 Scope:Global
          inet6 addr: fe80::6286:fa36:73dd:52d8/64 Scope:Link
          UP POINTOPOINT RUNNING NOARP MULTICAST  MTU:1420  Metric:1
          RX packets:4756 errors:0 dropped:0 overruns:0 frame:0
          TX packets:6234 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:100
          RX bytes:3124862 (3.1 MB)  TX bytes:773195 (773.1 KB)
```

CSMA/CD

Технология Ethernet базируется на методе управления доступом, называемом CSMA/CD (Carrier Sense Multiply Access with Collision Detection) — это означает множественный доступ с контролем несущей и обнаружением коллизий. Коллизией называют столкновение информационных передач. Устройства в сети постоянно прослушивают эфир и останавливают любую передачу, если обнаружена коллизия.

Как работает этот механизм?

Устройство прослушивает несущую частоту. Если обнаруживается несущая частота, значит, ни один из передатчиков не модулирует сигнал, и канал свободен. Когда узел начинает передавать информацию, он одновременно проверяет сигнал на проводнике, по которому осуществляется передача. Если сигнал не совпадает, значит, кто-то другой тоже пытается передать данные, обнаружена коллизия. Тот же способ при отсутствии коллизий позволяет в Gigabit Ethernet и выше использовать один проводник в полнодуплексном режиме. Если коллизий нет, то благодаря коммутаторам мы можем отправлять сигнал в канал и вычислять разницу между поступившим в канал и отправленным сигналом. Это будет принятый сигнал. Если больше двух абонентов в одной сети используют общую разделяемую среду, такой метод не работает.

Почему же коллизии возникают, если мы прослушиваем несущую частоту? Из-за низких скоростей в первых версиях Ethernet и задержки сигнала оба узла могли зафиксировать несущую частоту (сигнал еще не дошел) и начать передачу. Коллизия будет обнаружена, когда часть данных уже отправлена.

Если узел обнаруживает коллизию, он перестает отправлять и формирует jam-сигнал — сигнал преднамеренной помехи, призванный информировать другие станции, что возникла коллизия и узлы не должны ничего отправлять. После этого каждый узел, пытавшийся отправить сообщение, выдерживает паузу псевдослучайной длины, после чего передача возобновляется. Если снова возникнет коллизия, длительность паузы будет увеличена.

Стоит отметить, что если в Ethernet используется технология обнаружения коллизии, то в Wi-Fi — технология избегания коллизий. Там jam-последовательность отправляется перед передачей и служит сигналом другим устройствам, что передавать сигнал не надо.

Концентраторы и коммутаторы

Концентраторы (хабы) реализовали топологию типа «звезда», но работали на физическом уровне (т. е. с точки зрения работы на канальном уровне хаб не отличается от шины). Задача обнаружения коллизий и определения того, предназначен данный кадр получателю или нет, возлагался ли непосредственно на сетевые интерфейсы.

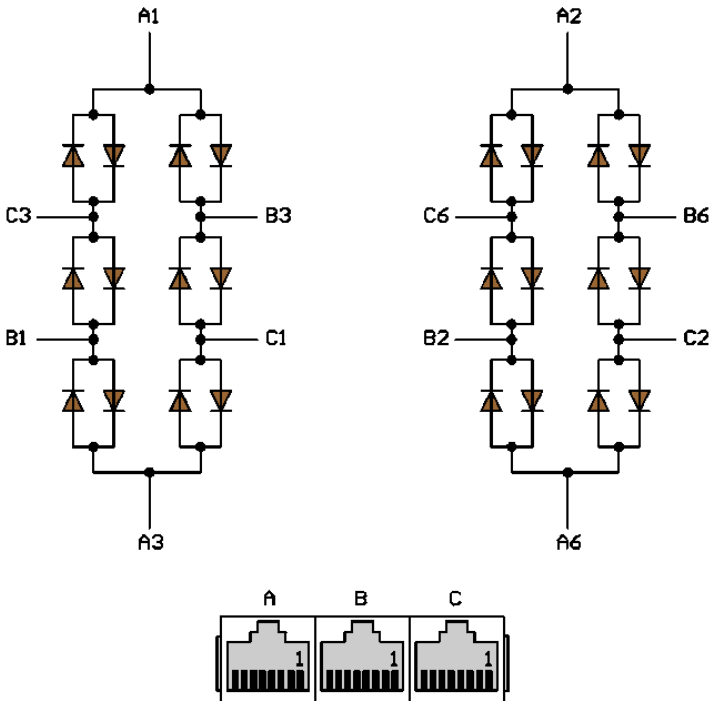
Простейший хаб можно спаять самостоятельно. Хаб работает на физическом уровне, у него нет программируемой логики.

В обычном состоянии на Ethernet-интерфейсе используется фильтрация пакетов канального уровня, и если MAC-адрес в заголовке назначения принятого кадра не совпадает с MAC-адресом текущего сетевого интерфейса и не является широковещательным, пакет отбрасывается.

Сетевая карта может работать в Promiscuous-режиме («неразборчивом»). В этом случае будут приниматься все кадры. Этим могут воспользоваться злоумышленники либо сетевые инженеры для поиска проблем в сети. Wireshark или tcpdump на компьютере, подключенном к хабу или к шине, с promiscous-режимом на сетевой карте, будет получать все фреймы, проходящие через данную сеть.

Как вы уже знаете, хабы — устаревший тип оборудования, они больше не производятся. Их вытеснили коммутаторы (которые иногда ошибочно называют хабами).

Напомним, чем отличаются схемы и внешний вид концентратора и коммутатора:



Простой самодельный концентратор на диодах. Электрическая схема и фото готового устройства. Устройство на три порта, пассивное, не требует даже питания. Работает в полудуплексном режиме, 10 Мбит/с.

Источник: http://www.zen22142.zen.co.uk/Circuits/Interface/pethhub.htm

48-портовый коммутатор Cisco Catalyst 9300

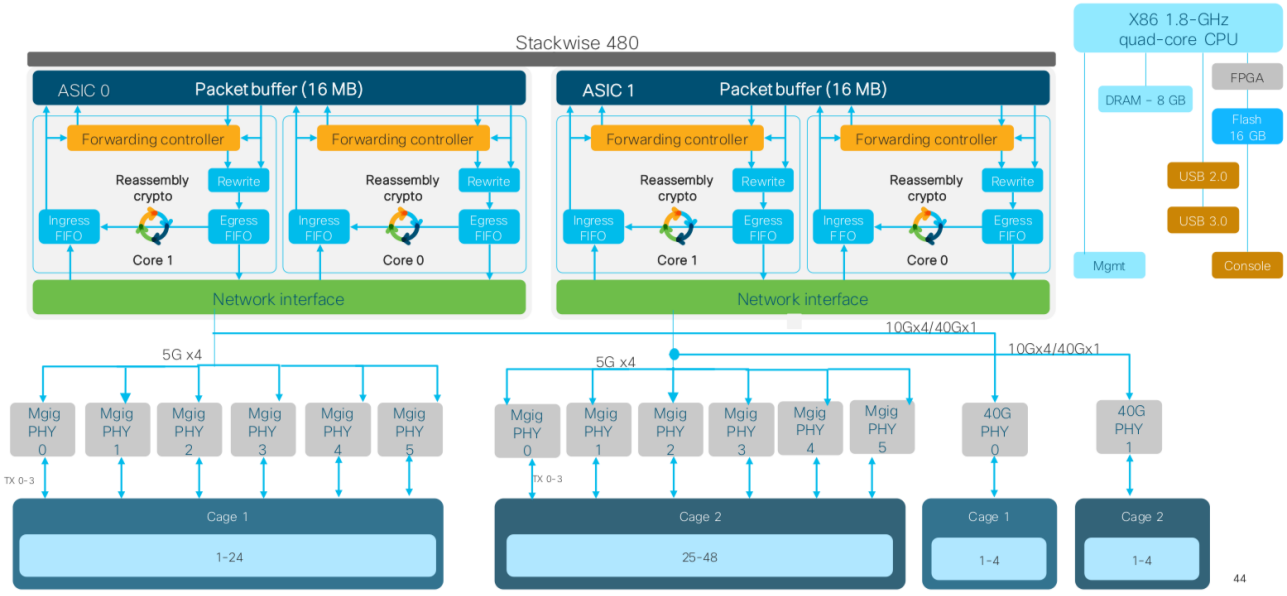


Схема коммутатора Cisco Catalyst 9300

Устройство и логика работы коммутатора (свитча) совершенно иные, нежели у концентратора.

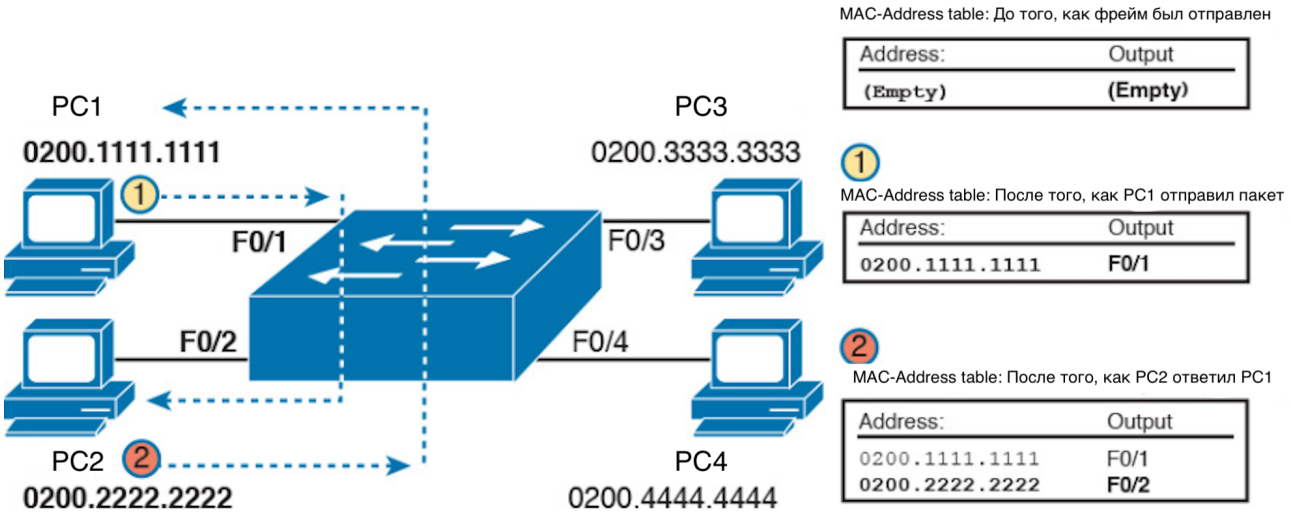
Коммутатор работает на канальном уровне и ведет таблицу соответствий номеров портов и MAC-адресов. Его не сделать самостоятельно с помощью паяльника: даже простейший коммутатор реализуется с помощью ASIC/ПЛИС (Application Specific Integrated Circuit — программируемые логические интегральные схемы) или процессора, которые позволяют ему выполнять алгоритм работы. На ASIC должна быть память для хранения MAC-адресов (таблицы соответствий MAC-адресов и номеров портов коммутатора), место для буфера пакетов и так далее.

Как правило, порты L2-коммутатора, в отличие от сетевых интерфейсов, не имеют собственных MAC-адресов (и тем более IP-адресов). Работа коммутатора в целом прозрачна для отправляющего и

принимающего устройств. Кроме L2-коммутаторов, которые принимают решение? основываясь на MAC-адресах (L2 модели OSI), также существуют L3-коммутаторы, которые фактически являются маршрутизаторами, то есть принимают решение, куда отправлять пришедший пакет, основываясь на IP-адресе (L3 модели OSI). В таком случае каждый интерфейс коммутатора может иметь свой собственный IP- и MAC-адрес. Главное отличие L3-коммутатора от маршрутизатора — в количестве функций обработки пакетов. Так, например, L3-свитч может иметь только Ethernet-порты, роутер же имеет возможность использовать Serial-интерфейсы, такие как PPP, имеет больше функций, обеспечивающих качество обслуживания, — QoS, большие размеры буфера и таблиц маршрутизации.

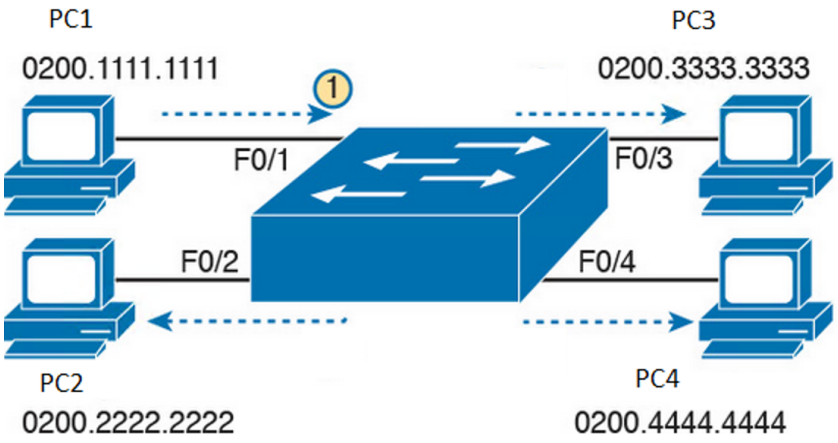
Так как коммутатор для подключенного в него хоста является прозрачным устройством, то нет никакого механизма предварительного заполнения таблицы коммутации. Поэтому коммутатор вынужден изучать MAC-адреса в момент их получения.

Представьте себе ситуацию, когда все устройства (4 компьютера и коммутатор) только что включили в розетку, они загрузились, но еще никто не отправлял никуда данные. Так как коммутатор еще не получал никаких пакетов, то его таблица MAC-адресов (таблица коммутации) пуста.



Далее наступает момент, когда PC1 хочет отправить фрейм PC2. Для простоты примем, что PC1 уже каким-то образом узнал, какой MAC адрес у PC2, и в состоянии заполнить все поля Ethernet-заголовка.

Как только фрейм был отправлен PC1, он поступает на порт коммутатора FastEthernet0/1 (F0/1). В этот момент таблица коммутации свитча пуста, и он не знает, на какой именно интерфейс надо отправить этот фрейм (всего активных интерфейсов 4). В связи с этим коммутатор сначала записывает MAC-адрес отправителя и порт, на который пришел фрейм, в таблицу коммутации (момент 1), после чего он вынужден отправить пришедший фрейм на все порты. Логика здесь такова: так как отправитель (PC1) знает, что такой MAC-адрес получателя существует, значит, он находится где-то внутри локальной сети, а следовательно, если отправить этот фрейм на все порты (исключая тот порт, из которого фрейм пришел), он должен найти получателя.



Address Table:

Address:	Output
(Empty)	(Empty)

Такой процесс называется флудингом — в данном случае это так называемый Unknown Destination Unicast flood. Если посмотреть на конечный результат (копия фрейма рассылается на все активные порты коммутатора), то процесс выглядит как широковещательная рассылка. Отличие от бродкаст-рассылки заключается в том, что MAC-адрес получателя в случае с unknown destination unicast-рассылкой, является валидным юникаст-адресом, а в случае с бродкастом это широковещательный MAC-адрес FF:FF:FF:FF:FF:FF. В итоге мы получаем ситуацию, когда все участники локальной сети получили один и тот же фрейм, но лишь PC2 имеет указанный во фрейме адрес получателя. Поэтому PC3 и PC4 просто отклоняют такой пришедший фрейм, а PC2 его принимает и обрабатывает в соответствии с запросом.

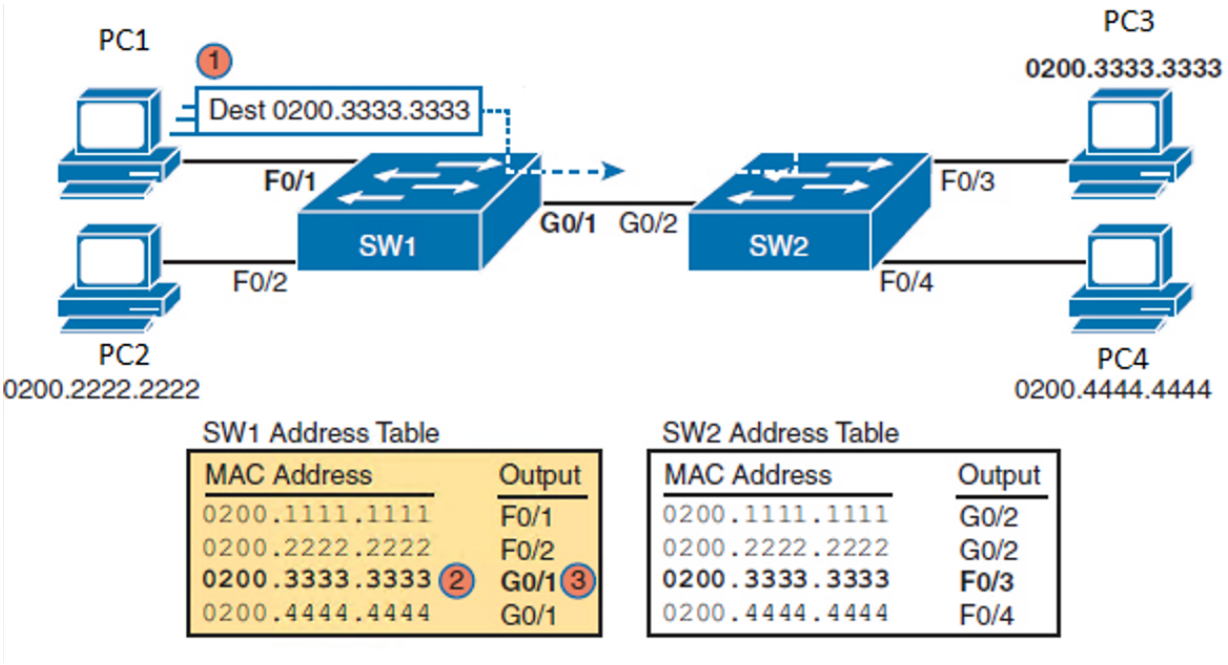
После того, как PC2 получил фрейм, он отвечает PC1. В этой ситуации, коммутатор вновь получает фрейм от PC2 в интерфейс F0/2. Коммутатор видит, что MAC-адрес отправителя ему неизвестен, и сразу же вносит этот MAC-адрес, а также интерфейс, через который был получен фрейм, в таблицу коммутации (момент 2). Далее коммутатор смотрит, кому предназначается фрейм, и видит, что адрес назначения ему известен (он был записан в момент 1). Теперь нет никакой смысла рассылать фрейм по всем портам, и коммутатор отправляет его сразу в интерфейс F0/1.

Важно понимать, что таблица коммутации является локальной для каждого L2 устройства, и каждое устройство принимает решение о коммутации в конкретный порт или флуде на все порты независимо от всех остальных. Также, не существует никакого протокола синхронизации таблиц между коммутаторами.

Если к коммутатору подключен другой коммутатор, то мы увидим, что напротив интерфейса, ведущего в сторону коммутатора, будут выучены несколько MAC-адресов. И если получатель фрейма будет подключен к другому коммутатору, будет сделана ровно такая же проверка таблицы, и фрейм будет отправлен на соответствующий порт.

Например, предположим, что PC1 отправляет фрейм в PC3. Свитч SW1 получает фрейм от PC1 из интерфейса F0/1 (момент 1), проверяет, есть ли у него MAC-адрес отправителя (0200.1111.1111) в таблице MAC-адресов, и если нет, то заносит этот адрес в таблицу. Далее SW1 смотрит, кому

предназначается этот фрейм, — MAC-адресу PC3 (0200.3333.3333). Он смотрит в свою таблицу коммутации и видит там, что адрес получателя находится за интерфейсом G0/1. SW1 отправляет этот фрейм в интерфейс G0/1, который ведет не в PC3, а в коммутатор SW2. Далее SW2 делает то же, что и SW1: он заносит MAC-адрес отправителя (0200.1111.1111) и интерфейс, через который был получен фрейм (G0/2), в свою таблицу MAC-адресов, после чего проверяет адрес назначения фрейма, видит, что адрес получателя уже изучен и находится за интерфейсом F0/3, следовательно фрейм может быть отправлен туда напрямую.



Таким образом решается задача идентификации. Но что делать с проблемой предотвращения коллизий?

В случае поступления нескольких фреймов одновременно устройство может обработать только один из них или будет накапливать их в буфере. В современных коммутаторах буферизация используется, когда порт назначения занят. Буферизация может осуществляться сразу на всё устройство либо для каждого порта отдельно, образуя очереди. При буферизации также возможны потери (например, когда буфер достигнет максимально допустимого размера).

Коммутаторы могут использовать два метода для пересылки пакетов между портами:

- Store-and-Forward (хранение и пересылка) коммутация;
- Cut-through (прямая) коммутация.

Store-and-Forward

Коммутатор при получении фрейма сохраняет его в буфере. При этом он анализирует адрес назначения фрейма. Также он вычисляет CRC фрейма и сравнивает с трейлером. При совпадении CRC коммутатор отправляет фрейм на порт назначения. При несовпадении — отбрасывает кадр. При

использовании этого метода увеличивается задержка, но увеличивается эффективность использования пропускной способности. Коммутаторы Catalyst используют именно этот метод.

Cut-through

При прямой коммутации коммутатор не сохраняет весь фрейм в буфер. Как только коммутатор получает часть фрейма, в которой содержится адрес назначения, он пересылает фрейм на порт назначения без проверки на ошибки. Данные передаются быстрее, но иногда пересылаются испорченные фреймы, которые все равно будут отброшены на конечном устройстве. То есть уменьшается эффективность использования пропускной способности.

С точки зрения физического уровня сегмент является доменом коллизий (верно для случая с концентратором или шиной). Устройствам приходится конкурировать за среду передачи данных.

С точки зрения канального уровня сегмент является широковещательным доменом.

Широковещательные ARP-, PPPoE-, DHCP-запросы ограничены широковещательным доменом.

Также возможен широковещательный **ping**, например, ping 192.168.0.255.

Не все узлы могут отвечать на широковещательный ICMP-запрос, поэтому надежный способ поиска узлов — не ping 192.168.0.255, а последовательный перебор всех адресов в сети.

Современные коммутаторы не только объединяют домены коллизий, но и обеспечивают промежуточный контроль ошибок. По этому принципу коммутаторы разделяют на Cut-Thru и Store-And-Forward. Коммутаторы Cut-Thru буферизируют MAC-адрес отправителя и получателя для выполнения коммутации и пересылают сообщение дальше. Кадры с неправильной контрольной суммой, карликовые кадры тоже пересылаются. Такие коммутаторы работают быстро, опережая в скорости сетевые мосты, но не могут обработать несколько сообщений, направляемых на один и тот же порт одновременно, либо одновременно принять и отправить сообщение на один порт. Коммутаторы Store-and-Forward полностью буферизируют сообщение и помимо коммутации выполняют проверку, не пересылая поврежденные кадры между сегментами.

Домен коллизий/широковещательный

Доменом коллизий называют часть сети Ethernet, где все узлы работают с общей разделяемой средой передачи данных и каждое устройство может создать коллизию в сети с любым другим абонентом этого домена.

Можно сказать, что домен коллизий — один сегмент сети Ethernet, работающий с общим канальным уровнем (Data Link layer) модели OSI, в котором одновременно осуществить передачу кадра может только один узел. Задержка передачи кадров между узлами или одновременная передача кадров приводит к возникновению коллизий, которые препятствуют передаче кадров, снижают пропускную способность и требуют работы специального алгоритма, уменьшающего вероятность возникновения

коллизий в сети. С увеличением числа узлов в сегменте возрастает и вероятность возникновения коллизии. Домены коллизий отделяются друг от друга сетевыми мостами или коммутаторами.

На практике домен коллизий — несколько устройств, объединенных в сеть с топологией «шина», либо «звезда» с концентратором, либо «точка-точка» (что имеет значение при полудуплексной передаче). В домене коллизий передача одного фрейма заставляет остальные устройства ждать завершения передачи, а при попытке одновременной передачи двумя или более узлами приведет к коллизии. Кроме того, в домене коллизии все слышат абсолютно весь трафик.

Широковещательным доменом (англ. broadcast domain) называют группу доменов коллизий, соединенных с помощью устройств второго уровня. Иными словами, это логический участок компьютерной сети, в котором все узлы могут передавать данные друг другу с помощью широковещания на канальном уровне сетевой модели OSI.

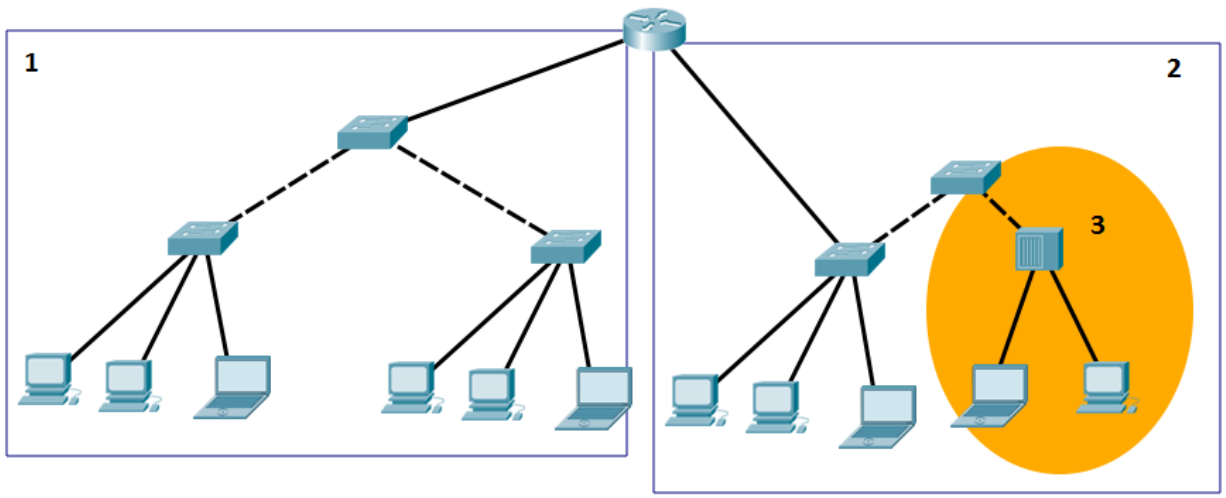
В широковещательном домене несколько доменов коллизий могут вести передачу внутри себя независимо друг от друга. В отличие от домена коллизий, в широковещательном домене каждое устройство слышит только трафик своего домена коллизий. Тем не менее широковещательный трафик распространяется по всему домену коллизий.

Чем является участок сети, легко обнаружить с помощью Cisco Packet Tracer. Если ICMP-сообщение распространилось по всем узлам, это домен коллизий. Если ARP-сообщение распространилось по всем узлам, это широковещательный домен.

С точки зрения модели OSI/ISO домен коллизий является физическим сегментом (L1), а широковещательный домен — логическим сегментом (L2).

Зона 1 и 2 — широковещательные домены, границами которых служат интерфейсы маршрутизатора.

Зона 3 — домен коллизий, границей которого является порт коммутатора.



Чем чреват конфликт MAC-адресов

Если два устройства (B3 и B4) будут иметь один и тот же MAC-адрес в пределах одного широковещательного сегмента, коммутатор будет постоянно перезаписывать таблицу коммутации, так как фреймы с одним и тем же MAC-адресом отправителя будут постоянно приходить из разных интерфейсов. При этом кадр может уйти не тому получателю, кадры будут теряться, скорость — падать. Также следует отметить, что скорость перезаписи таблицы конечна, и на современных коммутаторах составляет примерно 25 000 перезаписей в секунду. Когда этот порог будет превышен, коммутатор будет запоминать MAC-адреса с задержкой, что может привести к рассылке приходящих новых кадров во все интерфейсы.

Забегая вперед, отметим, что перед первой отправкой отправителю MAC-адрес неизвестен. Для его получения ARP-запрос широковещательно рассылается на MAC-адрес FF:FF:FF:FF:FF:FF.

Существует несколько атак канального уровня, среди которых можно назвать MAC-spoofing, DHCP-spoofing, ARP-spoofing. Рассмотрим пример ARP-spoofing. Ответив на ARP-запрос своим MAC-адресом, злоумышленник может представиться, например, маршрутизатором и реализовать атаку «человек посередине» (man in the middle, MITM). Для защиты от ARP-спуфинга используют VLAN, access-листы, туннелирование (в частности PPPoE), IPSEC и статичные ARP.

Локальные компьютеры тоже ведут таблицы соответствия **arp** (изучите команду **arp**, она доступна и в Windows и в GNU/Linux). Можно фиксировано прописать MAC-адрес маршрутизатора, например на каждой клиентской машине в сети, если он известен заранее.

Петля коммутации

Если есть несколько коммутаторов и к каждому подключено несколько узлов (обратите внимание: все они находятся **в одном сегменте**), вы можете попробовать увеличить надежность, замкнув коммутаторы в кольцо, или выстроить более сложную топологию. Без поддержки протокола STP в лучшем случае таблицы коммутатора могут постоянно перезаписываться, а в худшем — один кадр может передаваться по кругу бесконечно, нарушая работу коммутаторов. Такое состояние называется **петля коммутации**.

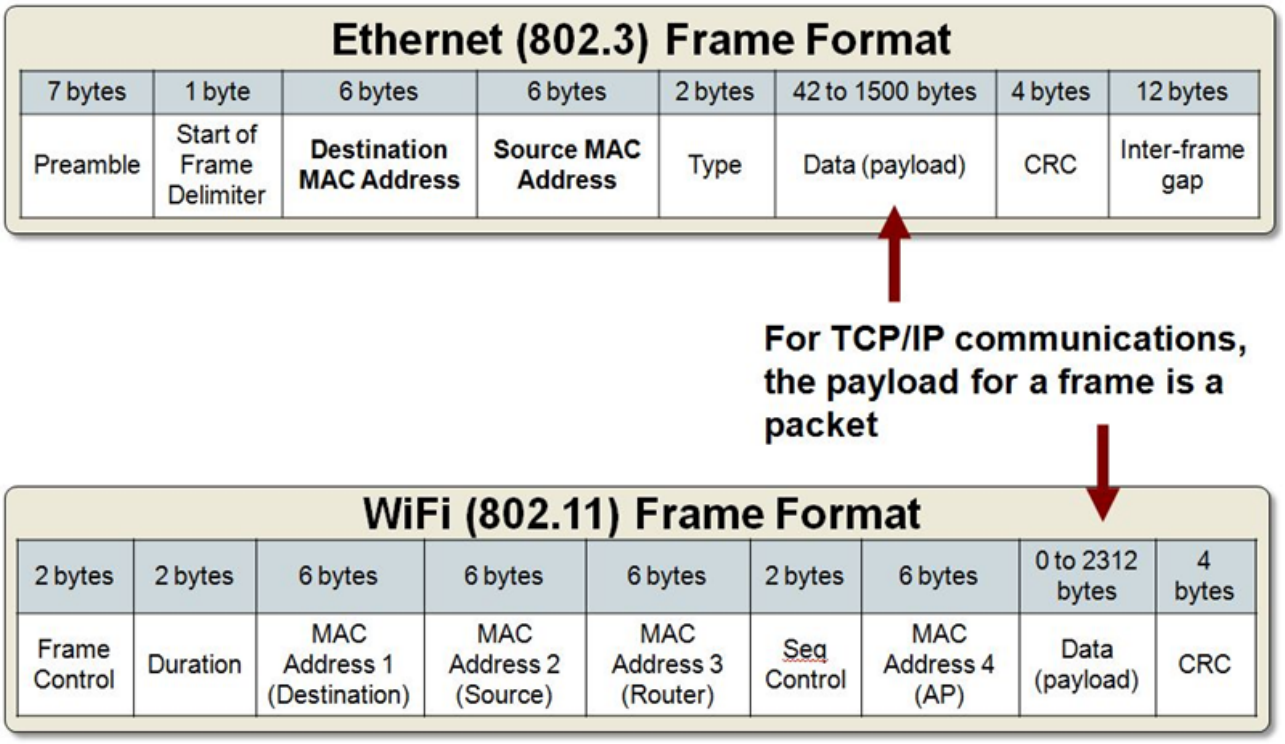
Алгоритм STP (Spanning Tree Protocol) определяет топологию в том числе благодаря обмену между коммутаторами и по другим признакам, в результате лишние пути отсекаются. Используются алгоритмы графов для нахождения кратчайшего пути и получения дерева без петель. Если физическая топология — кольцо, то логическая — дерево, позволяющее избежать петель коммутации.

В случае нарушения одного из путей он будет отсечен, а граф — перестроен (заработает ранее отключенный путь).

Более подробно о формате Ethernet-кадра

Как правило, когда рассматривают Ethernet-кадр, речь идет только о канальном уровне. На самом деле Ethernet-кадр содержит и структуры, относящиеся к физическому уровню: преамбулу (Preambule), ограничитель начала кадра (Start of Frame Delimiter) и межкадровый интервал (Inter-Frame Gap).

Рассмотрим структуру фрейма стандарта 802.3.



Формат фрейма IEEE 802.3 Ethernet. Для сравнения: формат фрейма IEEE 802.11

Вначале узел, собираясь начать передачу, отсылает преамбулу — семь байт вида 10101010. Для себя мы делим эту последовательность на байты, но технически это последовательность единиц и нулей, призванная сформировать особой формы сигнал, необходимый для синхронизации передатчика и приемника.

Далее следует ограничитель начала кадра, длиной в один байт (или, как принято говорить, октет). Он содержит комбинацию из 8 бит: 10101011 — она ровно на единицу отличается от последовательности преамбулы. Его значение соответствует названию, за ним следует осмысленная информация. Иногда его рассматривают как часть преамбулы: исторически в протоколах-предшественниках стандарта восьмой октет действительно был частью преамбулы из 8 октетов. Теперь последний байт логически выделен, имеет смысл и отличается на единицу от исходного значения. Этот бит, установленный в 1 вместо 0, фактически означает конец преамбулы и начало приема MAC-адреса получателя.

Следующие два поля кодируют так называемые физические адреса, или MAC-адреса. С ними мы уже познакомились. Могут присутствовать дополнительные поля в зависимости от используемого протокола канального уровня. В стандартном Ethernet II дополнительных вставок нет, скорее всего, вы не обнаружите их, используя Wireshark в обычном трафике.

Следующее поле: Длина/Тип протокола.

> Кадры формата 802.3 содержат поле Length вместо привычного нам Type (EtherType). Исторически сложилось так, что существует несколько стандартов для кадров Ethernet помимо перечисленных.
>
> Потом DEC, Intel и Xerox доработали их до универсального красивого решения Ethernet II (Ethernet DIX —по первым буквам названий компаний), которое стало крайне популярным — IP работает именно поверх него.
>
> Поле Length прежде говорило о общем размере полезной нагрузки. Это было малоинформативно, и такой кадр мог нести только один тип вышестоящего протокола. Значения Length могут быть до 1500 (0x05dc).
>
> В кадре Ethernet II отказались от поля Length и освободившиеся 2 байта использовали под поле Type (EtherType), которое определяет тип вышестоящего протокола. Чтобы чётко отличать их от 802.3, берутся значения выше 1536 (0x0600).
>
> Так, например, если кадр несёт IPv4, тип будет 0x0800, ARP — 0x0806, VLAN (802.1q) — 0x8100, IPv6 — 0x86DD, QinQ — 0x9100 и т. д.

Источник: http://pascal.tsu.ru/other/frames.html#as-h4-2325214 via https://habrahabr.ru/post/189268/.

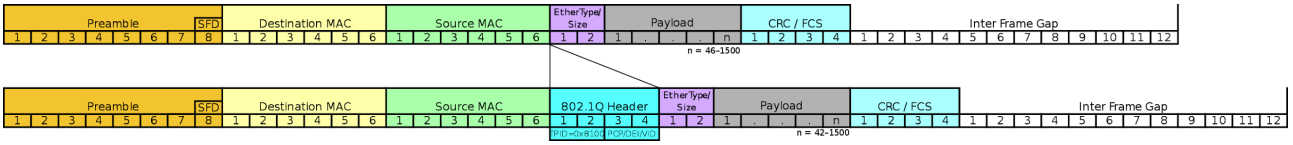
На https://habrahabr.ru/post/189268/ можно найти ответы на другие каверзные вопросы, касающиеся стека сетевых технологий для собеседований.

Следующее поле — данные. Полезная нагрузка: как правило, это IPv4-пакет, но не обязательно. Может присутствовать пакет ARP, IPv6, IPX, MPLS.

Замыкает поле контрольной суммы, которое служит для контроля ошибок.

Далее следует межкадровый интервал (Inter-Frame Gap); его часто не упоминают в формате кадра канального уровня, так как он относится к физическому уровню. Его длина — 12 байт, он служит для механизма работы с коллизиями. Это пауза, необходимая между передачей фреймов, чтобы не возникло коллизий.

Стандарты 802.3x и 802.X могут содержать дополнительные поля, но общая структура одна и та же.



Звезда. Пример для 802.1Q, определяющий тегирование трафика для VLAN

Микросегментация

В настоящее время концентраторы в технологиях Ethernet практически не встречаются. На протяжении развития Ethernet попытки снизить влияние коллизий привели к созданию мостов, потом коммутаторов, а потом коммутаторов, работающих в full-duplex. Если при использовании half-duplex домен коллизий включает порт устройства, порт коммутатора и соединение между ними, при full-duplex в домене коллизий остается только порт устройства, и фактически коллизии сходят на нет. Это явление называется микросегментацией, а порты устройств — микросегментами.

При этом Ethernet обратно совместим, он будет работать и со старыми стандартами. Все механизмы — CSMA/CD, проверка MAC-адреса на соответствие MAC-адресу получателя, — работают до сих пор. Несмотря на то, что топологии «шина» и «звезда» с концентратором ушли в прошлое, влияние тех времен до сих пор значительно в архитектуре Ethernet и в особенности Fast Ethernet, который все еще довольно широко используется провайдерами при подключении абонентов.

Стандарты Ethernet

Ethernet (читается «эзернет», от лат. aether — эфир) — сетевая технология канального уровня, использующаяся в большинстве локальных сетей. Технология осуществляет пакетную коммутацию.

Стандарты Ethernet определяют проводные и оптические соединения и электрические/волновые сигналы на физическом уровне, формат кадров и протоколы управления доступом к среде — на канальном уровне модели OSI. Технология, появившаяся в 70-х годах прошлого века, практически вытеснившая аналогичные сетевые технологии с рынка, на данный момент может быть использована для построения и локальных, и глобальных сетей. Современные телекоммуникационные операторы связи строят сети на базе этой технологии. Высокая пропускная способность и надежность линий связи, а также дополнительные функции (VLAN/QoS/STP) позволяют технологии минимизировать недостатки пакетной коммутации и полностью раскрыть возможности для передачи голосового трафика.

Ethernet-технологию описывают стандарты 802.3 IEEE. Первоначальная версия была разработана в 1979 году Робертом Меткалфом. В качестве передающей среды используется коаксиальный кабель (устарел), витая пара (UTP-5cat) или оптоволокно.

Технология характеризуется постоянно установленным соединением.

Скорость потока — 10 Мб/сек, 100 Мб/сек, 1 Гб/сек, 10 Гб/сек, 40 Гб/сек, 100 Гб/сек в обе стороны (режим полного дуплекса). Скорость определяется используемым видом стандарта.

Виды Ethernet

Классификация Ethernet по скорости (в хронологическом порядке):

1. Ethernet (коаксиал/витая пара/оптика) — 10 Мбит/с.
2. Fast Ethernet (витая пара 4 жилы/оптика) — 100 Мбит/с.
3. Gigabit Ethernet (витая пара 8 жил/оптика) — 1000 Мбит/с.
4. 10 Gigabit Ethernet (витая пара 8 жил/оптика) — 10 000 Мбит/с.
5. 40 Gigabit Ethernet (оптика/витая пара) — 40 000 Мбит/с.
6. 25 Gigabit Ethernet (оптика/витая пара) — 25 000 Мбит/с.
7. 50 Gigabit Ethernet (оптика) — 50 000 Мбит/с
8. Multigigabit Ethernet (витая пара) — 100/1000/2500/5000/10 000 Мбит/с
9. 100 Gigabit Ethernet (оптика) — 100 000 Мбит/с.

Как на витой паре получить вместо 100 Мбит/с 1 Гбит/с и больше

Мы не только переходим на оптоволокно, но и используем витую пару. Конечно, скорость упирается в качество витой пары. На кабеле категории 3 скорость 1 Гбит/с получить не выйдет, но на 5 — уже возможно. При этом скорость определяется не только улучшением характеристик проводника, но и алгоритмически.

Какие идеи используются в Gigabit, 40 Gigabit и т. д.:

1. 4 пары вместо двух. Логичный и простой шаг. При этом кросс-кабель уже применяться не должен, только прямой. Прирост — вместо 1 бита мы передаем 2 (2TX вместо 1) и получаем 2 (2RX вместо 1).
2. 4 пары одновременно для приема и передачи. Это позволит отправить сразу 4 бита либо принять 4 бита, но в полудуплексе.
3. Отправляем сигнал и прослушиваем. Если сигнал не совпадает, вычисляем разницу. Раньше этот способ использовался для контроля коллизий, теперь (в условиях микросегментации) позволяет поднять скорость. Теперь мы можем одновременно отправить и принять 4 бита. В таком случае при измерении скорости 1 бод будет равен не 1 бод в секунду, а целых 4.
4. Также мы можем увеличить количество состояний сигнала. Если в компьютерах применяется 0 и 1, на практике в Ethernet применяется большее число состояний.
5. Можно увеличить частоту передачи. Таким образом в единицу времени мы будем передавать еще больше информации.

В результате получаем прирост скоростей в 10, 40, 100 раз.

Практическое задание

Шаг 1

Скачать файл **Lesson3Homework.pkt.** Открыть в Cisco Packet Tracer.

Исправить проблемы с сетевыми подключениями (IP-адреса не трогать, они выделяются динамически). Если компьютер не получил IP-адрес с помощью DHCP, зайти в настройки TCP/IP, снять и заново установить радиокнопку DHCP.

Изучите устройства в наличии. Будьте внимательны с подключением портов.

Если всё сделано правильно, то любой компьютер из левой сети должен пинговать любой компьютер из правой сети. Компьютеры правой сети не смогут пинговать компьютеры левой сети, так и должно быть (домашний роутер не пропустит во внутреннюю сеть. Позже мы узнаем, что эта технология называется NAT).

Шаг 2

Подписать работу (указать фамилию и инициалы после нее).

Убедитесь, что в меню в Options > Preferences установлен флаг **Always Show Port Labels in Logical Workspace.**

С помощью инструмента Draw Polygon (а также Draw Ellipse/Draw Rectangle) красным цветом отметьте все домены коллизий и синим цветом все бродкастные домены.

Сохраните файл. Сделайте скриншот с помощью сервиса Lightshot (https://prnt.sc/) или аналогичного, загрузите его в облако и сохраните ссылку (также допускается прилагать скриншот к практическому заданию).

Загрузите .pkt-файл, в комментарии опишите все свои действия, укажите ссылку на скриншот или приложите его в формате png.

Для успешной сдачи практического задания скриншот обязателен.

Дополнительные материалы

1. http://itandlife.ru/technology/computer-networks/setevye-ustrojstva-tipy-setevyx-ustrojstv-i-ix-funkcii/2147483618/.

2. https://habr.com/ru/company/cbs/blog/276863/.
3. https://www.atraining.ru/arp-inarp-rarp-proxy-gratuitous-dai-sticky-and-more/.

Используемая литература

Для подготовки данного методического пособия были использованы следующие ресурсы:

1. http://itandlife.ru/technology/computer-networks/setevye-ustrojstva-tipy-setevyx-ustrojstv-i-ix-funkcii/2147483618/.
2. https://habr.com/ru/company/cbs/blog/276863/.