

Евдокимова 21205 🕶️

Лаба 1

ARP и таблица коммутации

Собрать сеть из четырёх оконечных устройств, используя два коммутатора (по два устройства на каждом коммутаторе).

Настроить интерфейсы (статически) на устройствах в сети таким образом, чтобы они находились в одном адресном пространстве (IPv4).

Проверить, что все оконечное оборудование может обмениваться ICMP-пакетами.

- * Проследить за прохождением ICMP-пакетов по сети.
- * Проследить за прохождением ARP-пакетов.
- * Проследить за заполнением таблиц коммутации на коммутаторах.
- * Проследить за заполнением ARP-таблиц на оконечном оборудовании.

Для того, чтобы проследить, следует отключить симуляцию пакетов [CDP](#) и [STP](#) (но нужно почитать, что это за протоколы)

=====

Решение лабы

*Находимся во вкладке realtime

Добавляем в сеть 4 конечных устройства и 2 [коммутатора](#) (они же свичи).

Надо связать комп со свичем.

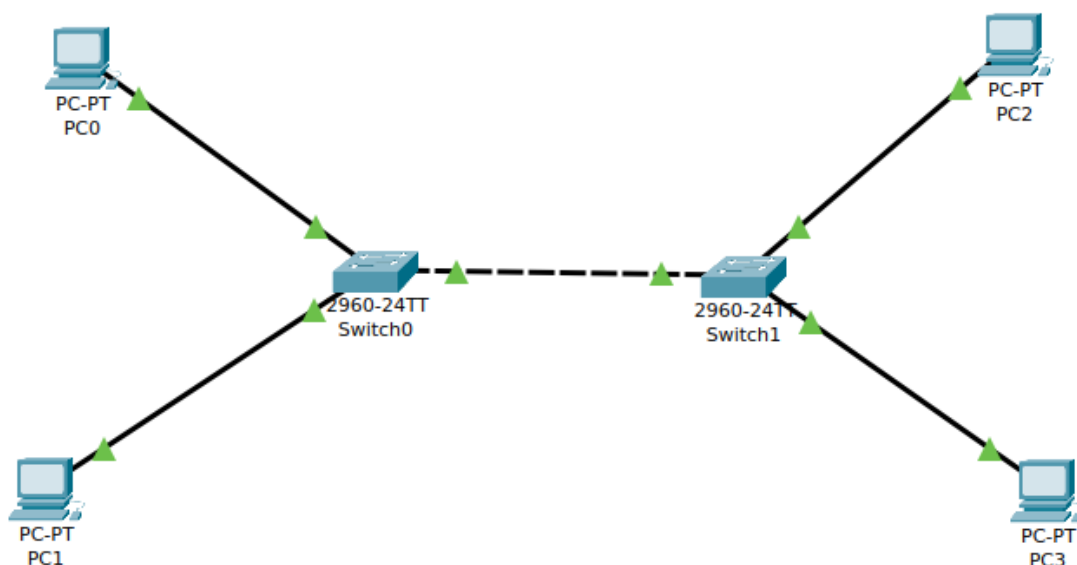
А с помощью какого кабеля соединяем комп со свичем? Витой парой (она же прямой кабель, aka straight through).

А свич со свичем? Перекрестным кабелем (aka crossover cable).

Теор часть про [разные типы кабелей тут](#).

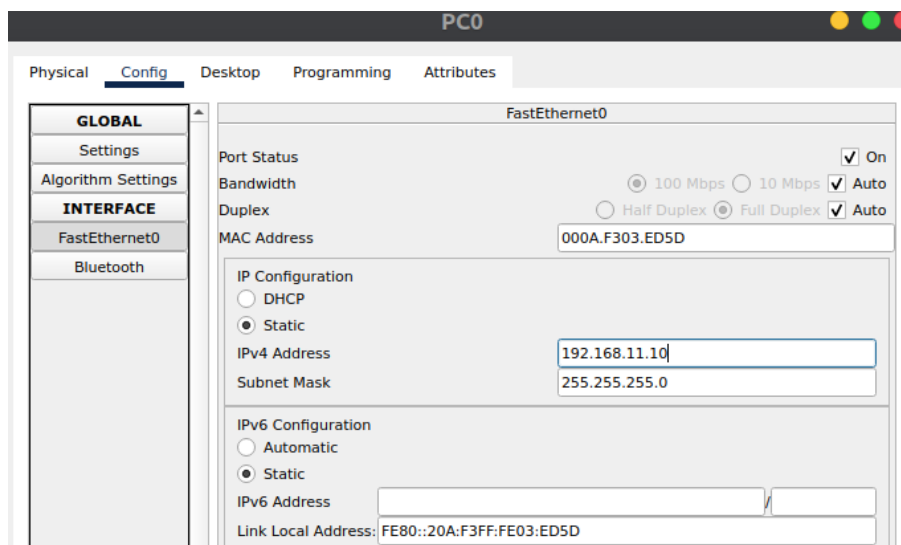
Когда кабели выбираем, жмякаем на Ethernet.

Ура, вот что должно получиться. Сеть собрана 😊



Теперь надо настроить [интерфейсы](#) так, чтобы устройства находились в одном адресном пространстве.

Проставим каждому компу ip-адрес: тыкаем на комп PC0 -> Config -> FastEthernet0 -> Ip COnfigurations -> IPv4 Address и записываем сюда, н-р, 192.168.11.10, жмякаем enter, получаем автоматически маску сети:



Аналогично для каждого компа делаем.

Пусть

на PC0 - адрес 192.168.11.10

PC1 - 192.168.11.11

PC2 - 192.168.11.12

PC3 - 192.168.11.13

Чтобы показать, что устройства находятся в одном адресном пространстве, попробуем пропинговать PC0 до PC2:

*В режиме реал тайм:

PC0 -> Desktop -> Command Prompt

Пишем команду 'ping 192.168.11.12'

```
C:\>ping 192.168.11.12

Pinging 192.168.11.12 with 32 bytes of data:

Reply from 192.168.11.12: bytes=32 time<1ms TTL=128
Reply from 192.168.11.12: bytes=32 time=1ms TTL=128
Reply from 192.168.11.12: bytes=32 time=1ms TTL=128
Reply from 192.168.11.12: bytes=32 time<1ms TTL=128

Ping statistics for 192.168.11.12:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
    Approximate round trip times in milli-seconds:
        Minimum = 0ms, Maximum = 1ms, Average = 0ms
```

Видим, что все ок, пакетики отправляются. Если бы компы (хосты) находились в разном адресном пространстве, то ничего не будет передано.

ping - отправляет запросы на хосты (конечные устройства то есть), смотри man ping.

Посмотрим на таблицу MAC-адресов:

Тыкаем на switch0 -> CLI, enter.

‘Switch>’ - означает, что мы находимся в режиме пользователя, которому доступны только чтения (изменять ничего не можем).

Чтобы перейти в привилегированный режим, пишем enable, тогда получим ‘Switch#’.

Чтобы посмотреть MAC-таблицу адресов пишем ‘show mac-address’

```
Switch>enable
Switch#show mac-address
      Mac Address Table
-----
Vlan    Mac Address      Type    Ports
----    -
1       0001.9671.7d13   DYNAMIC Fa0/2
1       0001.9720.61bb   DYNAMIC Fa0/3
1       0010.11cc.d656   DYNAMIC Fa0/3
1       0030.f225.4303   DYNAMIC Fa0/3
```

*Переходим в режим симуляции.

Simulation Panel -> Edit Filters -> Show All/None(снизу справа на панели. Так мы уберем все галочки быстро) -> выбираем только ICMP and ARP.

Будем отправлять пакеты с компа PC0 на PC3.

Очищаем arp-таблицу на PC0 И PC3 с помощью команды ‘arp -d’.

Тыкаем на значок письма (Add Simple PDU), жмякаем на PC0 и PC3.

Тыкаем на свичи и очищаем мак-таблицу: ‘clear mac-address’

Для проверки очистки просим вывести таблицу (и 1ю, и 2ю)

```
Switch#clear mac-address
Switch#show mac-address
      Mac Address Table
-----
Vlan    Mac Address      Type    Ports
----    -
```

Они должны быть пустыми (это важно).


Также arp-таблицы PC0 and PC3 должны быть очищены(‘arp -d’) и для проверки очистки в command prompt пишем ‘arp -a’ - получаем таблицу соответствия IP и MAC адресов для данного компьютера (должно быть пустым).

Получаем вот что:

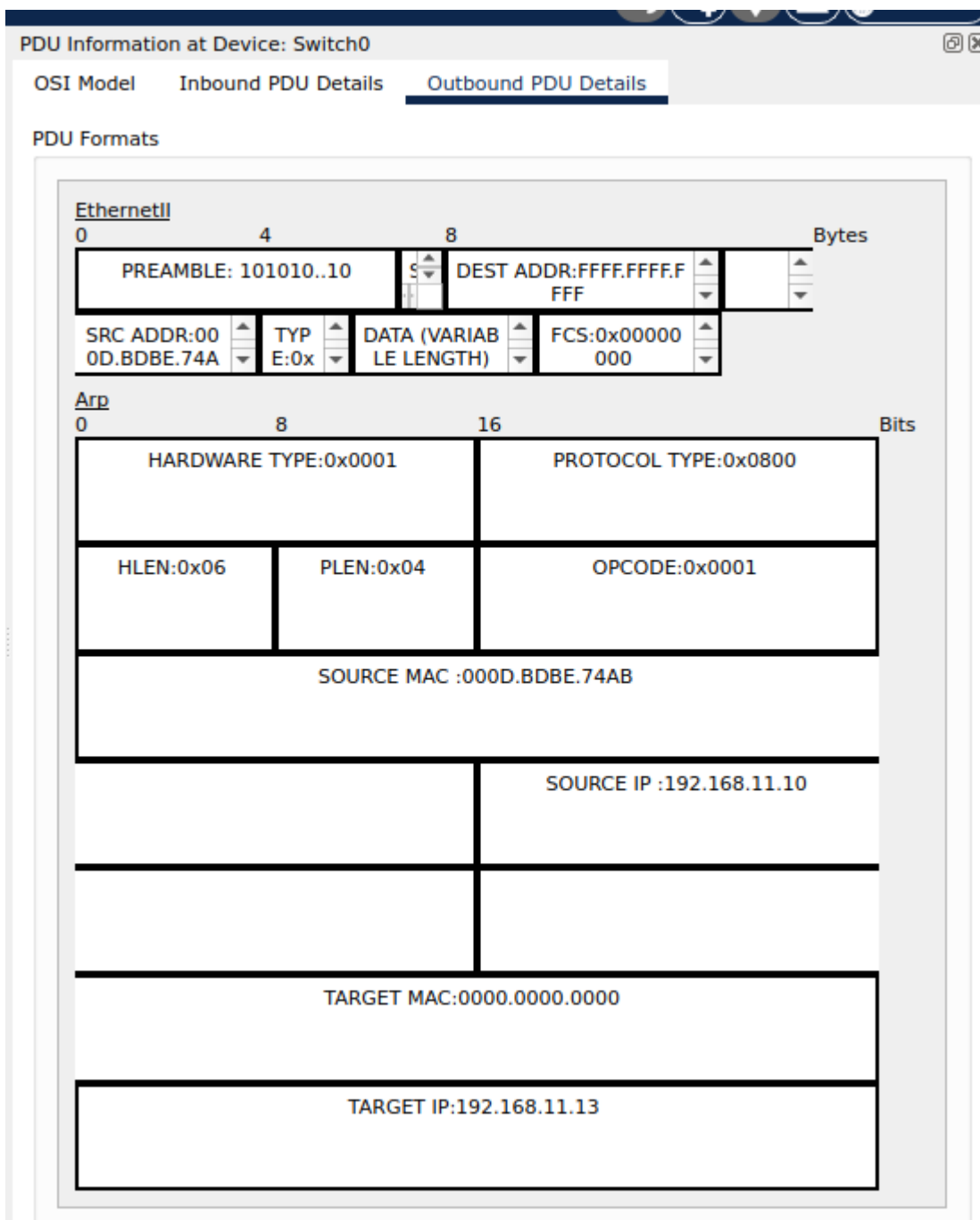
The screenshot shows the Cisco Packet Tracer interface. At the top, there's a network diagram with two switches (2960-24TT Switch0 and Switch1) connected by a dashed line. Each switch has two PCs connected to it. Below the diagram are four tables: ARP Table for PC0, ARP Table for PC3, MAC Table for Switch0, and MAC Table for Switch1. The Simulation Panel on the right shows an event list with the following data:

Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	PC0	ICMP
	0.000	--	PC0	ARP

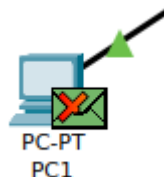
Окей. Тыкаем 1 (!!!) раз на  . Полетел пакет ARP, таблица изменилась следующим образом:

Simulation Panel				
Event List				
Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	PC0	ICMP
	0.000	--	PC0	ARP
	0.001	PC0	Switch0	ARP

То есть мы знаем мак адрес отправителя, а получателя не знаем. У получателя известен ip-адрес, а mac-адрес - нет. Тыкнем на зеленый арп-пакет и получим вот что:



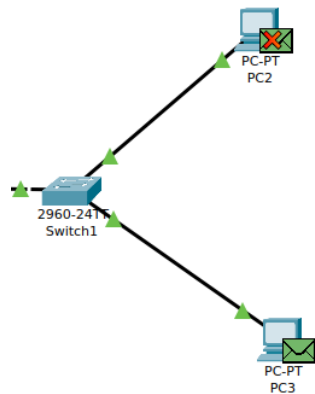
На следующем шаге пакет улетает в PC1 и switch1. Но PC1 этот



пакет не принимает

. На следующем шаге switch1

отправляет пакет PC2 (который не принимает этот пакет) и PC3



PC3 принимает этот пакет.

PDU Information at Device: Switch1

OSI Model Inbound PDU Details Outbound PDU Details

PDU Formats

EthernetII

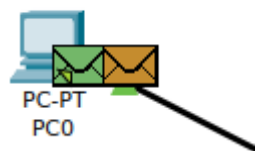
0		4		8		Bytes	
PREAMBLE: 101010...10				DEST ADDR:000D.BD BE.74AB			
SRC ADDR:0 090.0C37.86		TYP E:0		DATA (VARIA BLE LENGTH		FCS:0x0000 0000	

Arp

0		8		16		Bi	
HARDWARE TYPE:0x0001				PROTOCOL TYPE:0x0800			
HLEN:0x06		PLEN:0x04		OPCODE:0x0002			
SOURCE MAC :0090.0C37.8675							
				SOURCE IP :192.168.11.13			
TARGET MAC:000D.BDBE.74AB							
TARGET IP:192.168.11.10							

Получаем мак-адрес PC3.

и на следующем шаге отправляется пакет switch1 -> switch0 -> PC0.



Ура, arp-пакет доставлен

Таблица после отправления arp-пакета от отправителя до получателя и обратно, от получателя до отправителя, выглядит так:

Event List				
Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	PC0	ICMP
	0.000	--	PC0	ARP
	0.001	PC0	Switch0	ARP
	0.002	Switch0	Switch1	ARP
	0.002	Switch0	PC1	ARP
	0.003	Switch1	PC2	ARP
	0.003	Switch1	PC3	ARP
	0.004	PC3	Switch1	ARP
	0.005	Switch1	Switch0	ARP
	0.006	Switch0	PC0	ARP
	0.006	--	PC0	ICMP

Теперь полетел пакет ICMP. Летит в switch0 -> сразу в switch1 (а почему в PC1 не полетел? Ответ ниже) -> PC3. Потом обратно: PC3 -> switch1 -> switch0 -> PC0.

После всех действий таблица выглядит так:

Event List				
Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	PC0	ICMP
	0.000	--	PC0	ARP
	0.001	PC0	Switch0	ARP
	0.002	Switch0	Switch1	ARP
	0.002	Switch0	PC1	ARP
	0.003	Switch1	PC2	ARP
	0.003	Switch1	PC3	ARP
	0.004	PC3	Switch1	ARP
	0.005	Switch1	Switch0	ARP
	0.006	Switch0	PC0	ARP
	0.006	--	PC0	ICMP
	0.007	PC0	Switch0	ICMP
	0.008	Switch0	Switch1	ICMP
	0.009	Switch1	PC3	ICMP
	0.010	PC3	Switch1	ICMP
	0.011	Switch1	Switch0	ICMP
	0.012	Switch0	PC0	ICMP

arp и mac - таблицы после прохождения всех хостов выглядят вот так:

ARP Table for PC0			ARP Table for PC3			MAC Table for Switch0			MAC Table for Switch1		
IP Address	Hardware Address	Interface	IP Address	Hardware Address	Interface	VLAN	Mac Address	Port	VLAN	Mac Address	Port
192.1...	0090.0C...	FastEth...	192.1...	000D.B...74AB	FastEth...	1	000D.BDBE.74AB	FastEthernet0/2	1	000D.BDBE.74AB	FastEthernet0/1
						1	0090.0C37.8675	FastEthernet0/1	1	0090.0C37.8675	FastEthernet0/3

Вот “конечная станция”:

The screenshot displays a network simulation environment. At the top, a topology diagram shows two switches (Switch0 and Switch1) connected by a dashed line. Each switch is connected to two PCs (PC0, PC1 on Switch0; PC2, PC3 on Switch1). Below the diagram are four tables: ARP Table for PC0, ARP Table for PC3, MAC Table for Switch0, and MAC Table for Switch1. To the right is an Event List panel showing a series of ICMP events. At the bottom, there are play controls, a status bar, and a scenario dropdown menu.

IP Address	Hardware Address	Interface
192.1...	0090.0C...	FastEth...

IP Address	Hardware Address	Interface
192.1...	000D.B...74AB	FastEth...

VLAN	Mac Address	Port
1	000D.BDBE.74AB	FastEthernet0/2
1	0090.0C37.8675	FastEthernet0/1

VLAN	Mac Address	Port
1	000D.BDBE.74AB	FastEthernet0/1
1	0090.0C37.8675	FastEthernet0/3

Vis.	Time(sec)	Last Device	At Device	Type
	0.000	--	PC0	ICMP
	0.000	--	PC0	ARP
	0.001	PC0	Switch0	ARP
	0.002	Switch0	Switch1	ARP
	0.002	Switch0	PC1	ARP
	0.003	Switch1	PC2	ARP
	0.003	Switch1	PC3	ARP
	0.004	PC3	Switch1	ARP
	0.005	Switch1	Switch0	ARP
	0.006	Switch0	PC0	ARP
	0.006	--	PC0	ICMP
	0.007	PC0	Switch0	ICMP
	0.008	Switch0	Switch1	ICMP
	0.009	Switch1	PC3	ICMP
	0.010	PC3	Switch1	ICMP
	0.011	Switch1	Switch0	ICMP
	0.012	Switch0	PC0	ICMP

Теоретическая часть

Опр. Пакет - полученная порция информации.

Опр. Хаб (концентратор)

Сетевой хаб или концентратор (Hub), обеспечивающий объединение нескольких компьютеров в единую локальную сеть и обмен данными между ее узлами на первом уровне сетевой модели OSI. Главной его задачей является получение сигнала с информацией на один из портов и передача дальше на другие порты. Однако в работе хаба возможны коллизии из-за возможного столкновения пакета данных на одном из портов, что значительно замедляет процесс передачи данных.

Концентратор (хаб) — это устройство физического уровня, которое оперирует отдельными битами, а не кадрами.

Опр. Маршрутизаторы (роутеры).

Маршрутизаторы используются для поиска оптимального маршрута передачи данных на основании специальных алгоритмов маршрутизации, например выбор маршрута (пути) с наименьшим числом транзитных узлов. Работают на сетевом уровне модели OSI.

Опр. Коммутаторы (свитчи).

Коммутаторы - это устройства, работающие на канальном уровне модели OSI и предназначенные для объединения нескольких узлов в пределах одного или нескольких сегментов сети. Передаёт пакеты коммутатор на основании внутренней таблицы - таблицы коммутации, следовательно трафик идёт только на тот MAC-адрес, которому он предназначается, а не повторяется на всех портах (как на концентраторе).

Концентратор повторяет пакет, принятый на одном порту на всех остальных портах.

Хорошая [статья](#) с таблицей сравнения маршрутизатора, коммутатора и хаба.

Еще [здесь норм](#).

Различия между витой парой и кроссовером

Смотри сайты [номер раз](#), [номер два](#).

Вывод: перекрестный кабель соединяет два устройства одного типа для связи друг с другом, например, компьютер и компьютер, или коммутатор и коммутатор. **Витая пара соединяет два разных устройства друг с другом**, например, компьютер и коммутатор.

Для общего развития: Патч корд — это отрезок кабеля типа витая пара, оба конца которого обжаты коннекторами.

Пара определений еще.

Опр. Сетевой интерфейс

Сетевой интерфейс – это программный интерфейс для сетевого оборудования. Например, если на вашем компьютере есть две сетевые карты, вы можете управлять и настраивать связанные с ними сетевые интерфейсы по отдельности. Сетевой интерфейс может быть связан с физическим устройством, а также может быть виртуальным. Примером последнего является устройство закольцовывания (loopback) – виртуальный интерфейс локальной машины.

Про протоколы

Опр. Сетевой протокол

Протокол определяет формат и порядок сообщений, которыми обмениваются два или более взаимодействующих объектов, а также действия, предпринимаемые при передаче и/или приеме сообщения либо при возникновении другого события.

Про протокол [тут](#), [тут](#).

Опр. Протокол IP

Internet Protocol (IP, досл. «межсетевой протокол») — маршрутизируемый протокол сетевого уровня стека TCP/IP. Именно IP стал тем протоколом, который объединил отдельные [компьютерные сети](#) во всемирную сеть [Интернет](#).

Про ip читать [тут](#).

Опр. Протокол CDP

CDP (*Cisco Discovery Protocol*) — протокол второго уровня, разработанный компанией Cisco Systems, позволяющий

обнаруживать подключённое (напрямую или через устройства первого уровня) сетевое оборудование Cisco, его название, версию IOS и IP-адреса.

Протокол достаточно полезный, так как он может показать, что за устройство (версия ПО, номера портов, платформа и ещё много другой информации) подключено в сеть. Это может быть удобно для составления карты сети, ведения документации и мониторинга сети. Однако также это облегчает атаку на сеть. В связи с этим протокол CDP в большинстве случаев отключают.

Читать [тут](#) и [тут](#).

Опр. Протокол STP

STP - Spanning Tree Protocol (протокол [остовного дерева](#)) — канальный протокол. Основной задачей STP является устранение [петель](#) в топологии произвольной сети [Ethernet](#), в которой есть один или более [сетевых мостов](#), связанных избыточными соединениями. STP решает эту задачу, автоматически блокируя соединения, которые в данный момент для полной связности коммутаторов являются избыточными.

Про стандарты

Опр. Стандарт

Стандарт - это набор правил и соглашений, используемых при создании локальной сети и организации передачи данных с применением определенной топологии, оборудования, протоколов и т. д.

Опр. Стандарт Ethernet

Ethernet это стандарт, который относится только к построению локальных сетей LAN (Local Area Network).

Ethernet - это набор описаний способов физической передачи сигналов (электричество) на первом уровне модели OSI и формирования кадров (фреймов) на втором уровне модели OSI внутри локальных сетей LAN.

Важно! Ethernet относится **только** к проводным сетям.

!!!Про Ethernet Frame смотреть [ниже](#).

Подробности смотреть [здесь](#) и [здесь](#).

Про адреса

Опр. MAC-адреса

Ethernet адресация определяет либо конечное устройство, либо группу адресов в сети. Эти адреса называют MAC-адресами (Media Access Control), состоящими из 48 бит и записываемыми в шестнадцатеричной системе счисления. Например, 6AFE:834B:32C4.

НО! Есть 2 исключения:

000..00 - адрес сети

255.255.255.255 - адрес зарезервирован для broadcast

MAC-адрес состоит из двух частей – уникальный идентификатор организации/производителя (24 бита) и оставшая часть, которую назначил сам производитель (еще 24 бита). Зная MAC адрес (или первые 24 бита) и имея под рукой интернет, мы можем определить производителя, например здесь.

Опр. Unicast адрес – адрес, принадлежащий одному устройству в сети (например, 6AFE:834B:32C4).

Опр. Broadcast адрес – широковещательный канал — метод передачи данных в компьютерных сетях, при котором поток данных (каждый переданный пакет в случае пакетной передачи) предназначен для приёма всеми участниками сети, имеет значение FFFF:FFFF:FFFF.

Когда пишем приложения, оперируем ip-адресами. А откуда берутся mac адреса? С завода))

Подробности [туточки](#).

Держу в курсе: Самое маленькое адресное пространство состоит из 2 бит.

Опр. IP-адрес

IP-адрес — уникальный числовой идентификатор устройства в компьютерной сети, работающей по протоколу [IP](#).

В 4-й версии IP-адрес представляет собой 32-битное число. Как правило, адрес записывается в виде четырёх десятичных чисел значением от 0 до 255 (эквиваленты четырём восьмибитным числам), разделенных точками, например, 192.168.0.3.

Особые ip-адреса

В [протоколе IP](#) существует несколько соглашений об особой интерпретации IP-адресов: если все двоичные разряды IP-адреса равны 1, то [пакет](#) с таким адресом назначения должен рассылаться всем узлам, находящимся в той же сети, что и источник этого пакета. Такая рассылка называется ограниченным [широковещательным](#) сообщением (*limited broadcast*). Если в поле номера узла назначения стоят только единицы, то пакет, имеющий такой адрес, рассылается всем узлам сети с заданным номером сети. Например, в сети 192.168.5.0 с [маской подсети](#) 255.255.255.0 пакет с адресом 192.168.5.255 доставляется всем узлам этой сети. Такая рассылка называется широковещательным сообщением (*direct broadcast*).

Опр. Маска подсети

Маска подсети — [битовая маска](#) для определения по [IP-адресу](#) адреса [подсети](#) и адреса [узла](#) (хоста, компьютера, устройства) этой подсети. В отличие от IP-адреса маска подсети не является частью [IP-пакета](#).

Смотреть примерчики [тут](#).

Держу в курсе: Адресная часть должна быть непрерывной: сначала должны быть биты относящиеся к сетевой части, а потом уже к хостовой. Т.е. такого адреса 255.0.255.0 быть не может.

Опр. CIDR (вкусный нямням) 🍷

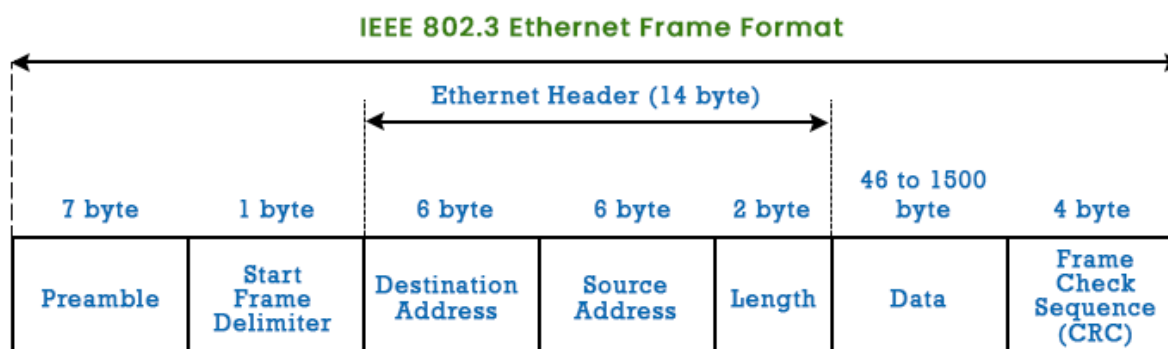
Бесклассовая адресация (*Classless Inter-Domain Routing*, [англ. CIDR](#)) — метод [IP-адресации](#), позволяющий гибко управлять пространством [IP-адресов](#), не используя жёсткие рамки [классовой адресации](#). Использование этого метода позволяет экономно использовать ограниченный ресурс IP-адресов, поскольку возможно применение различных [масок подсетей](#) к различным подсетям.

Ethernet Frame

Стартовый маркер и финишный маркер, а между ними сначала хэдер (мак адреса отправителя и получателя). Есть контрольная сумма crc - побитовая сумма всех битов (в конце получается один

бит, хог всех битов. Позволяет определить поврежденные биты, но это не норм. Тк определяет четное или нечетное количество бит было повреждено). Контр сумма- хэш-функция, чтоб определять количество поврежденных данных.

Коммутаторы не меняют заголовок.



Сначала коммутатор принимает кадр, потом парсится хедер.

Смотрим на мак получателя и надо понять, куда отправлять? Как это работает?

Порты коммутатора пронумерованы. Есть табличка мак-адресов:

MAC	Port (interface)
M2	2

Как заполняется эта талица? Поиск по табличке происходит аппаратно - специальным устройством, которое мгновенно находит адрес по таблице. Как попала эта запись (M2, 2) в таблицу и что делать если нет этой записи?

Ответ: Коммутатор никого не опрашивает, *он использует только ту инфу которая есть в фрейме.*

На канальном уровне если 2 устройства находятся в сети, то нет гарантии безопасности. Нужно использовать протоколы например на сетевом уровне.

Когда коммутатору приходит кадр, коммутатор проверяет мак адрес получателя, отправителя и контрольную сумму.

В итоге пакет доходит до получателя. В самом протоколе канального уровня нет запросов и тд. Кадр нужен для

транспортировки в более высокие уровни. Возможно там есть двусторонние обмены данными.

Когда переполняется таблица? Когда оч много устройств, но есть механизм очистки: когда отправитель долго ничего не отправляет, коммутатор помечает строку с отправителем как нерабочую или удаляет эту строку.

Операция определения куда отправлять (используя таблицу коммутации) - это оч быстрая операция. Интернет на одной только коммутации не получится. Между сетями должна быть маршрутизация, которая работает более логично: определяет один только путь, который приведет к получателю и если мы не знаем пути до получателя, то пакет выбрасывается.

Про протоколы ARP & ICMP

Протокол ARP

стр 521

Протокол разрешения адресов (ARP), который обеспечивает трансляцию (преобразование) IP-адресов в адреса канального уровня.

При передаче дейтаграмм используются и адреса сетевого уровня (например, IP-адреса Интернета), и адреса канального уровня (то есть MAC-адреса), поэтому возникает потребность в преобразовании одних адресов в другие. В Интернете эту работу выполняет протокол разрешения адресов (Address Resolution Protocol, ARP).

Как работает?

ARP-модуль преобразует IP-адрес в MAC-адрес узла.

Как он это делает?

У ARP-модуля каждого узла есть оперативная память, в которой хранится ARP-таблица. В этой таблице прописаны IP-адреса хостов локальной сети и соответствующие им MAC-адреса.

передающий узел определяет

нужный ему адрес при помощи протокола ARP. Сначала передающий узел формирует специальный ARP-пакет. В ARP-пакете содержится несколько полей, среди которых есть IP-адреса и MAC-адреса передающего и принимающего узлов. Для обоих ARP-пакетов (запроса и ответа) используется один и тот же

формат. Цель ARP-пакета с запросом состоит в том, чтобы опросить все остальные узлы локальной сети и определить LAN-адрес, соответствующий интересующему нас IP-адресу.

Кадр с ARP-запросом принимается всеми остальными адаптерами подсети, и (поскольку в запросе использовался широковещательный адрес) каждый адаптер передает содержащийся в кадре ARP-пакет своему узлу.

Каждый узел проверяет, совпадает ли его IP-адрес с указанным IP-адресом получателя в ARP-пакете. Узел, обнаруживший совпадение, посылает запрашивающему узлу ответный ARP-пакет с указанным в нем соответствующим MAC-адресом. После этого запрашивающий узел может обновить свою ARP-таблицу и отправить IP-дейтаграмму, заключенную в кадр канального уровня, где MAC-адрес назначения соответствует адресу хоста или маршрутизатора, ответившего на предыдущий ARP-запрос.

<https://www.ibm.com/docs/ru/aix/7.2?topic=protocols-address-resolution-protocol>

Протокол ICMP.

Читать [здесь](#).

Вторым протоколом сетевого уровня является **Протокол управляющих сообщений Internet (ICMP)**. **ICMP** - обязательная часть любой реализации **IP**. **ICMP** отправляет сообщения об ошибках и управляющие сообщения протоколу **IP**.

<https://www.ibm.com/docs/ru/aix/7.2?topic=protocols-internet-control-message-protocol>

Таблица коммутации сопоставляет мак адрес и порт

=====

Коммутаторы Ethernet ничего не знают об IP-адресах и для передачи данных используют MAC-адреса. позволяет по IP-адресу компьютера определить его МАК-адрес.

Кайф <https://zvondozvon.ru/tehnologii/protokoli/arp>

<https://zvondozvon.ru/tehnologii/protokoli/icmp>

Супер [крутая статья про arp](#)

Для определения соответствия между логическим адресом сетевого уровня (IP) и физическим адресом устройства (MAC) используется протокол ARP (Address Resolution Protocol, протокол разрешения адресов).

Как происходит передача arp пакетов? Вот [тут написано](#).

Очень крутой индус, который по сути [слил](#) лабу.