BASIC =

- базовые понятия
- методы безопасности
- классификация сетей
- топологии сетей
- стандарты сетей
- основы организации сетей
- модель OSI
- модель TCP/IP
- физический уровень
- канальный уровень
  - ethernet
  - mac-адреса
  - vlan
  - stp
  - wi-fi
- сетевой уровень
  - ip adress
  - ip протокол - маршрутизация, фрагментация
  - управляющие протоколы - dhcp, arp, icmp
- взаимодействие канального и сетевого уровня
- транспортный уровень
  - upd
  - tcp
    - механизм сохранения порядка следования
    - скользящее окно
    - соединение и разрыв
    - формат заголовка
    - технология nat
    - управление потоком
    - управление перегрузкой
  - socket interface
  - межсетевой экран

WIRESHARK windows, linux

Отслеживание устройств через пассивное прослушивание Wi-Fi - link
ARP спуфинг на Python - link
Про VPN - link

BASICS

more about: авторский ресурс - link

Таненбаум, Уэзеролл = Компьютерные сети
Олифер, Олифер = Компьютерные сети. Принципы, технологии, протоколы
Куроуз, Росс = Компьютерные сети. Нисходящий подход

***** БАЗОВЫЕ ПОНЯТИЯ **********

Абонент =
Клиент =
Сервер =
Хост = в широком смыле - любой компьютер в сети
Хаб = любой узел сети
Коллизия = наложение сигналов нескольких источников данных

ПОМЕТКИ ДЛЯ ВЫЯСНЕНИЯ

как посмотреть заголовок MAC уровня
есть ли на промежуточных хабах буферы
как быть, если пропускная способность хаба недостаточна
как устройство понимает, какой протокол используется

***** МЕТОДЫ БЕЗОПАСНОСТИ **********

OSI модель, УРОВЕНЬ = МЕТОД

1. канальный = фильтрация на портах коммутатора по MAC-адресу
2. транспортный = межсетевой экран (brandmauer = firewall)
3. прикладной = a. proxy server , b. content filter

Intrusion detection system = система обнаружения вторжений
Intrusion prevention system = система предотвращения вторжений

***** КЛАССИФИКАЦИЯ СЕТЕЙ **********

ТИП КОММУТАЦИИ
Коммутация = соединение абонентов сети через транзитные узлы.

1. коммутация каналов
   фиксированный канал связи
   при разрыве - заново устанавливать связь
   пример = телефон
2. коммутация пакетов
   данные бьются на пакеты
   возможны разные пути = отказоустойчивость
   на каждом узле решается задача маршрутизации

ТЕХНОЛОГИИ ПЕРЕДАЧ

1. широковещательные
   передает один - доступно всем
   пример: WiFi, Ethernet
2. точка-точка
   передает один другому (вохможно, по цепочке)
   пример (коммутируемый Ethernet)

***** ТОПОЛОГИИ СЕТЕЙ **********

Топология = схема связи между компьютерами в сети
Геометрически, тополгоия = граф, где
   вершины = узлы сети
   ребра = связи между узлами

БАЗОВЫЕ ТОПОЛОГИИ (как правило, используются смешанные)

- полносвязная = каждый напрямую с каждым
- ячеистая = полносвязная минус некоторые связи
- звезда = все подключены к одному, передача через него
- кольцо = соединение с двумя соседними
- дерево = соединение в виде дерева
- общая шина = передает один, доступно всем

ВАЖНО! Отличие топологий:
   физическая = соединение устройств в сети
   логическая = правила распространения сигнала в сети
Например, классический Ethernet (физ. звезда + лог. общая шина)

***** СТАНДАРТЫ СЕТЕЙ **********

Формальные = принятые по формальным законам стандартизации
Фактические = установившиеся сами собой

1. ISO = международная организация по стандартизации
   -> эталонная модель взаимодействия открытых систем

2. IEEE = институт инженеров по электронике и электротехнике
   -> сетевое оборудование / технологии передачи данных
                              (для каждой свой комитет)

3. IAB = совет по архитектуре интернета
   -> протоколы интернет (называются RFC). Подразделения
          IRTF - долгосрочные перспективы
          IETF - сетевые протоколы

4. W3C консорциум
   -> стандарты World Wide Web (называются Рекомендации) :
          язык разметки HTML
          таблицы стилей CSS
          архитектура Web Services Architecture
          язык разметки XML

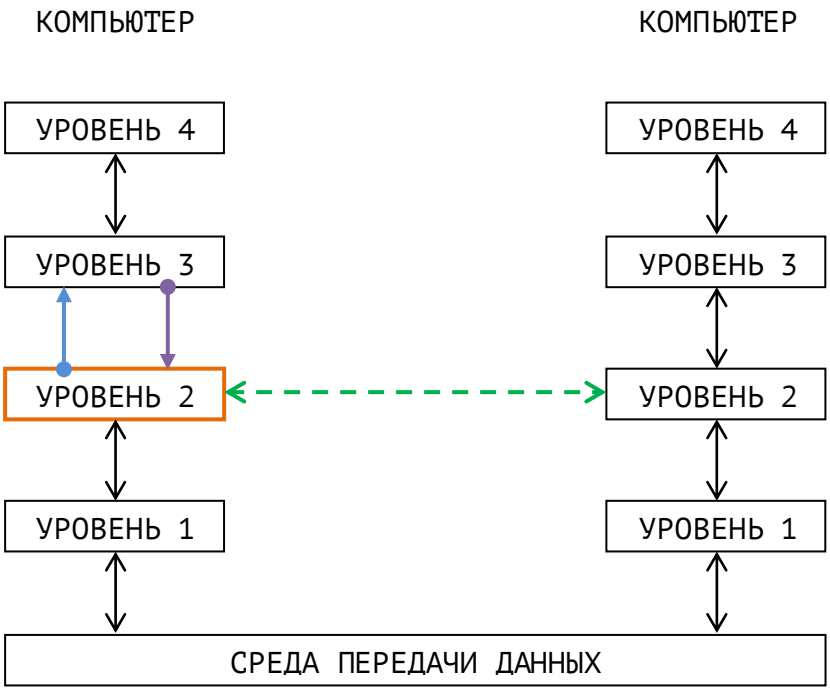
ЭТАЛОННЫЕ МОДЕЛИ ОРГАНИЗАЦИИ СЕТЕЙ

1. ISO OSI = формальная, на практике не используется
   - 7 уровней, протоколы не входят в модель
   - хорошая теоретическая проработка

2. TCP/IP = фактическая
   - 4 уровня
   - одноименный стек протоколов = основа сети Интернет

***** ОСНОВЫ ОРГАНИЗАЦИИ СЕТЕЙ **********

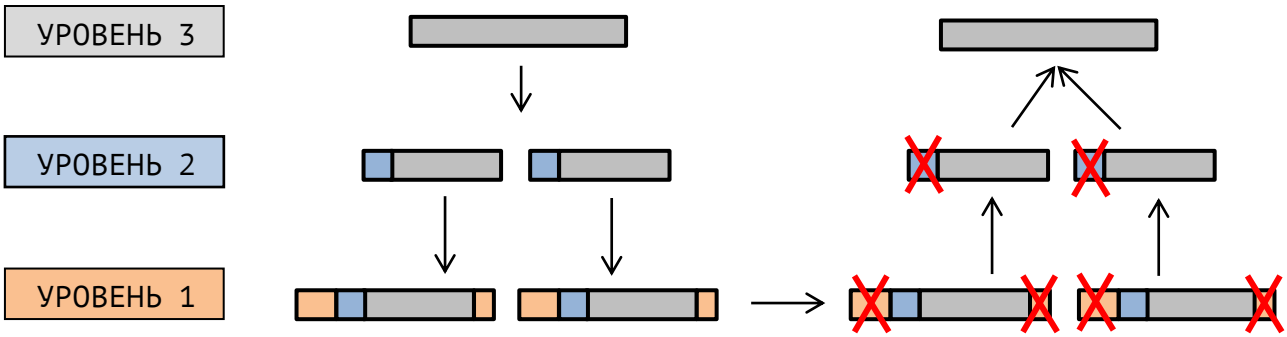
Для решения сложной задачи построения сетей используется принцип декомпозиции = разбиение на "уровни". Каждый уровень решает одну задачу (или набор тесно связанных).



- СЕРВИС = функции, реализуемые уровнем
- ИТЕРФЕЙС = набор операций, предоставляемых верзнему уровню
- ПРОТОКОЛ = правила, согласующие работу одинаковых уровней (взаимодействие через заголовки протоколов)

АРХИТЕКТУРА = набор уровней + протоколов (интерфейсы не входят)
СТЕК ПРОТОКОЛОВ = необходимая иерархия протоколов

- ИНКАПСУЛЯЦИЯ = включение сообщения верхнего уровня в сообщение нижнего уровня. **Сообщение = заголовок + данные + концевик (опц.)**

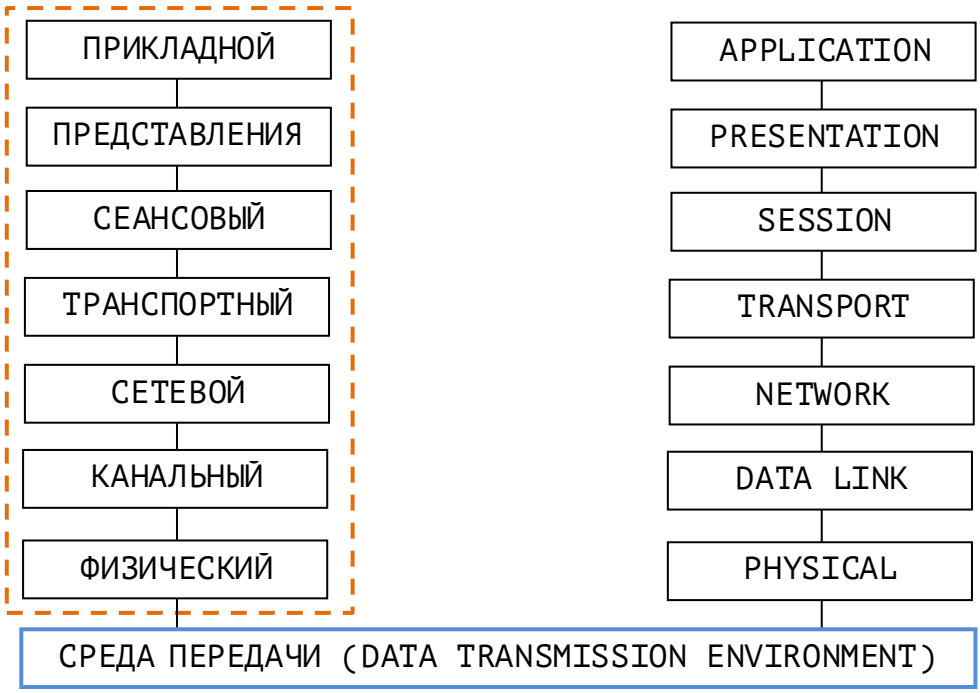


***** МОДЕЛЬ OSI **********

OSI = эталонная модель, принятая комитетом ISO в 1983 г.
OSI = open system interconnation

Открытая система в этом контексте =
система, построенная в соответствии с открытыми спецификациями

OSI :
- не включает описания протоколов = НЕ сетевая архитектура
- используется в качестве "общего языка" для описания сетей



ФИЗИЧЕСКИЙ
- передача битов по физическому каналу связи
- не вникает в смысл передаваемой информации
- работает концентратор

КАНАЛЬНЫЙ
- передает кадры
- обнаруживает и исправляет ошибки
- физическая адресация (конкретное устройство)
- управление доступом к разделяемой среде передачи
- работает коммутатор и точка доступа

СЕТЕВОЙ
- передает пакеты
- согласование различий объединяемых сетей
- общая адресация (сетевой / глобальные адреса)
- маршрутизация (поиск маршрутов через узлы сети)
- работает маршрутизатор

ТРАНСПОРТНЫЙ
- передает сегменты
- передача данных между процессами на хостах
- обеспечение надежности обмена данными на хостах
- сквозной / сетенезависмый = изолирован от сети

СЕАНСОВЫЙ
- передает сообщения
- очередность передачи сообщений
- синхронизация выполнения критических операций
- обнаржуние сбоя в сети и восстановления соединения

ПРЕДСТАВЛЕНИЯ
- передает сообщения
- представление данных в понятном формате
- включает синтаксис и семантику
- шифрование и дешифрование

ПРИКЛАДНОЙ
- передает сообщения
- набор приложений, доступный пользователю

***** **МОДЕЛЬ TCP/IP** **********

TCP/IP = фактическая модель, закрепленная … .
- основа Интернет
- назван по одноименному стеку протоколов TCP/IP

***** ФИЗИЧЕСКИЙ УРОВЕНЬ **********

Задача - просто передать данные
Характеристики:
- скорость = пропускная способность + время задержки
- количество ошибок

Представление информации:
- кодирование (прямоугольные импульсы) = медные провода
- модуляция (синусоидальные волны) = беспроводное, оптоволокно

***** КАНАЛЬНЫЙ УРОВЕНЬ **********

ТЕХНОЛОГИИ КАНАЛЬНОГО УРОВНЯ = Ethernet, Wi-Fi, 5/4/3G, MPLS

Задача - формирование локальной сети
Кадры формируются именно на канальном уровне
- добавление заголовка
- добавлнеие концевика
Существуют разные способы обозначения кадра (см. отдельно)

ОШИБКИ Обнаружение - контрольная сумма
Исправление - коды исправляющие ошибки
Повторная отправка - остановка и ожидание
- скользящее окно (исп. в TCP)
Исправлять ошибки эффективнее всего:
- частое возникновение -> канальный уровень
- редкое возникновение -> канальный только обнаружение

КАНАЛЬНЫЙ УРОВЕНЬ разделяется на два подуровня:

1. LLC (logical link control) = управление логическим каналом
- передача данных, обработка ошибок
- предоставляет мультиплексирование
- предотвращает "затопление" медленного получателя
- общий для разных технологий

2. MAC (media access control) = управление доступом к среде
- совместное использование разделяемой среды
- адресация
- специфичен для разных технологий

***** **ETHERNET** **********

ТИПЫ ETHERNET

Ethernet	10 Мб/с	II (DIX) / IEEE 802.3
Fast …	100	II (DIX) / IEEE 802.3u
Gigabit …	1Гб/с	II (DIX) / IEEE 802.3z , 802.3ab
5G …	2,5 и 5	II (DIX) / IEEE 802.3bz
10G …	10	II (DIX) / IEEE 802.3ae , 802.3an
100G …	40 и 100	II (DIX) / IEEE 802.3ba

КЛАССИЧЕСКИЙ ETHERNET
разделяемая среда
до Gigabit / 5G

КОММУТИРУЕМЫЙ ETHERNET
точка-точка
от Fast до 100G

ФИЗИЧЕСКИЙ УРОВЕНЬ может быть:
- коаксиальный кабель
- витая пара
- оптоволокно

КАНАЛЬНЫЙ УРОВЕНЬ: в классическом Ethernet смешаны LLC и MAC

ФОРМАТ КАДРА стандарта II (DIX)
общий для классического и коммутируемого

6 байт	6 байт	2 байта	46 – 1500 байт	4 байта
АДРЕС ПОЛУЧАТЕЛЯ	АДРЕС ОТПРАВИТЕЛЯ	ТИП	ДАННЫЕ*	КОНТРОЛЬНАЯ СУММА
заголовок			0x0800 – IPv4 0x0806 – ARP 0x86DD – IPv6	концевик

* Существует раширение JumboFrame для ДАННЫХ до 9000 байт

КЛАССИЧЕСКИЙ ETHERNET

КОНЦЕНТРАТОР (hub) = устройство для создания сетей Ethernet
Основа: витая пара
Топологии:
- физическая - звезда
- логическая - общая шина

Все устройства подключаются к концентратору. Концентратор работает на физическом уровне модели OSI. Единственная задача - распространить сигнал, принимаемый от одного из хабов на все остальные.

Доступ к среде осуществляется по специальному правилу CSMA/CD:
(carrier sense multiple access with collision detection):
- каждый хаб прослушивает сеть
- хаб получает все пакеты и сам обрабатывает mac-адреса
- состав сигнала: преамбула + кадр + межкадровый интервал
- при передаче данных хаб сам следит за коллизиями
  - прослушивает несущую частоту
  - если свободна - передает данные
  - одновременно - считывает свои же данные
  - если данные вдруг отличаются = коллизия
  - начинает передавать jam-последовательность
  - следующая попытка передать данные:
    - после паузы
    - пауза = L * 512 битовых интервалов (см. стандарт)
    - L = случайное число [0, 2^n - 1]
    - n = номер попытки
      - после 10 попытки n не увеличивается
      - после 16 попытки передача прекращается

Недостатки:
- небезопасность (пакеты доступны всем)
- множественные коллизии
- пропускная способность сети = самому медленному хабу

КОММУТИРУЕМЫЙ ETHERNET

КОММУТАТОР (switch) = устройство для содания сетей Ethernet
Основа: витая пара, оптоволокно
Топологии:
- физическая - звезда
- логическая - полносвязная сеть

Внешний вид похож на концентратор

Внутри коммутатора реализована
- аппаратная часть
- прошивка

такие, которые позволяют при передаче сигнала из одного порта, передавать его на любой другой / несколько портов. Концентратор самостоятельно анализирует заголовки кадров, извлекает из них mac-адрес получателя и передает кадр на соответствующий порт (таким образом, коммутатор функционирует на физическом + канальном уровне OSI).

ТАБЛИЦЫ КОММУТАЦИИ

При включении коммутатора, происходит заполнение таблицы, которая устанавливает соответствие mac-адреса и порта коммутатора (в таблице есть и другие поля - см. подробнее)

Заполнение называется "алгоритм обратного обучения" и состоит в прослушивании портов и считывании mac-адреса отправителя пакета.

В случае, если коммутатору попадается кадр, получателя которого не удается найти по таблице, этот кадр одновременно посылается на все порты в надежде, что такой получатель все же найдется (например, если он еще не отправил ни одного пакета и просто не успел попасть в таблицу коммутации)

АЛГОРИТМ ПРОЗРАЧНОГО МОСТА

Заключается в том, что коммутатор никаким образом не влияет на пакет. Поэтому он доходит до получателя так, как будто получатель и отправитель соединены по сети точка-точка.

***** MAC-адреса **********

MAC-адреса = на одноименном подуровне MAC канального уровня.
- длина адреса 6 байт (48 бит)
- формат записи = XX-XX-XX-XX-XX-XX или XX:XX:XX:XX:XX:XX

Задача = однозначная идентификация конкретного устройства (или, иначе, идентификация сетевого интерфейса узла сети)
- Ethernet : IEEE 802.3
- Wi-Fi : IEEE 802.11

Поведение сети при наличии несокльких устройств с совпадающими индивидуальными MAC-адресами не регламентировано.

ТИПЫ MAC-адресов
- индивидуальный (unicast) : 30-9C-23-15-E8-8C
- групповой (multicat) : 01- … (всегда 01)
- широковещательный (broadcast) : FF-FF-FF-FF-FF-FF

НАЗНАЧЕНИЕ MAC-адресов:
- централизовано (производителями*), IEEE 802 бит = 0
- локально (администраторы сетей**) бит = 1

* 4-6 байты = OUI (organization unique identifier)
** контроль уникальности и выставление бита = на их совести

MAC-адреса за пределами локальной сети может быть полезен только для поиска поставщика сетевой карты в целях поддержки. Для связи с хабами в других сетях нужны:
- протоколы более высокого уровня
- маршрутизаторы, которые понимают эти протоколы

IMEI-адрес = адрес устройства канального уровня в сети сотовой связи (также уникальный = параллель с MAC-адресом)

***** VLAN **********

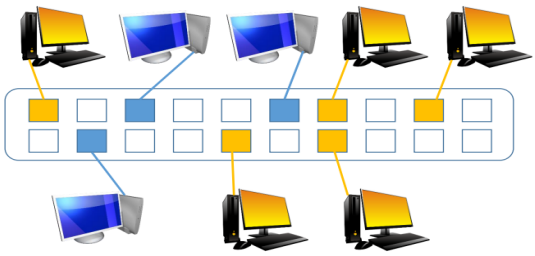
VLAN (virtual loval area network) = виртуальная локальная сеть
Действует: на канальном уровне модели OSI
реализуется коммутаторами

Задача = резделить общую локальную сеть на несколько автономных непересекающихся подсетей. "Виртуальная" - так как реализуется на уровне ПО коммутаторов, а не физически.

- безопасность
- масштабируемость
- снижение общей нагрузки (кадр не передается на все порты в случае если получатель не обнаружен по таблице коммутаций).

Для реализации:
- в таблицу коммутации добавляется дополнительное поле VLAN
- вводится стандарт IEE 802.1q со специальным форматом кадра

Порт коммутатора	MAC-адрес	VLAN
1	1C-75-08-D2-49-45	2
2	00-02-B3-A7-49-D1	3
3	00-04-AC-85-E7-03	3
4	54-BE-F7-88-15-47	3
5	00-40-D0-C0-08-BA	2
...		



IEEE 802.1q

Вводится для передачи информации о VLAN между несколькими коммутаторами. Для этого в стандартном кадре:

- поле ТИП стандартного кадра заменяется на 0x8100
- перед данными добавляется:
  - 2 байта = запись номера VLAN
  - 2 байта = ТИП стандартного кадра

6 байт	6 байт	2 байта	2 байта	2 байта	46-1500 байт	4 байта
Адрес получателя	Адрес отправителя	Тип	Тег	Тип	Данные	Контрольная сумма

VLAN в пределах одного коммутатора = нетегированный
VLAN для сети коммутаторов = тегированный

TRUNK PORT (магистральный порт) = порт сетевых устройств, через который проходит тегированный трафик.

***** STP **********

Проблема = если несколько коммутаторов объединены в виде кольцевой топологии, то это приведет к так называемому широковещательному шторму.

Широковещательный шторм = возникает, когда на вход коммутатора подается пакет с mac-адресом устройства, не входящего в текущую сеть. В таком случае, по правилу работы коммутатора, он посылает этот пакет на все порты. Если же вдруг сеть коммутаторов закольцована, этот пакет будет бесконечно бродить по сети, постоянно удваиваясь (можно нарисовать схему).

STP = spanning tree protocol = протокол для координации работы связанных между собой комутаторов. Действует:
- на канальном уровне модели OSI
- реализуется коммутаторами

СТАНДАРТЫ: IEEE 802.1d = классический STP
IEEE 802.1w = улучшенный STP (скорость работы)

ПРИНЦИП = формирование связующего дерева = подграф без циклов, содержащий все узлы исходного дерева
- автоматическое отключение дублирующих соединений
- в случае разрыва, отключенные соединеия могут включаться

ЭТАПЫ РАБОТЫ

1. выбор корневого коммутатора (по мин. mac-адресу / вручную)
2. определение кратчайших путей до корневого коммутатора
   - количество участков пути
   - скорость передачи на каждом из участков
3. отключение всех остальных соединений

BPDU = bridge protocol data units = собщение протокола STP, которое отправляется каждым из коммутаторов в сети на trunk-port с периодичностью в 2 секунды. С помощью этого сообщения происходит выбор корневого коммутатора и обнаружение замкнутых колец. Рассылается на групповой mac-адрес 01:80:C2:00:00:00

ВАЖНО! BPDU рассылается для каждой! VLAN, известной коммутатору. То есть в случае, если есть 100 VLAN, то каждые 2 секунды будет рассылаться 100 BPDU. Из-за этого классический STP несовместим с технологией VLAN. Для этого существует MSTP (IEEE 802.1s)

СОСТОЯНИЕ ПОРТОВ по технологии STP:

listening	= обработка BPDU, но без передачи данных
learning	= не передает кадры, составляет mac-таблицу
forwarding	= принимает и передает данные и BPDU
blocking	= заблокирован по технологии STP
disabled	= выключен администраторов

СТОИМОСТЬ СОЕДИНЕНИЙ при расчете кратчайшего пути

4 Mbit/s	250
10	100
16	62
100	19
1 Gbit/s	4
2	3
10	2

***** **WI-FI** **********

WI-FI является торговой маркой WI-FI ALLIANCE, IEEE 802.11
Действует:
на физическом + канальном уровне модели OSI (LLC + MAC)
реализуется ...

ОБОРУДОВАНИЕ = точка доступа

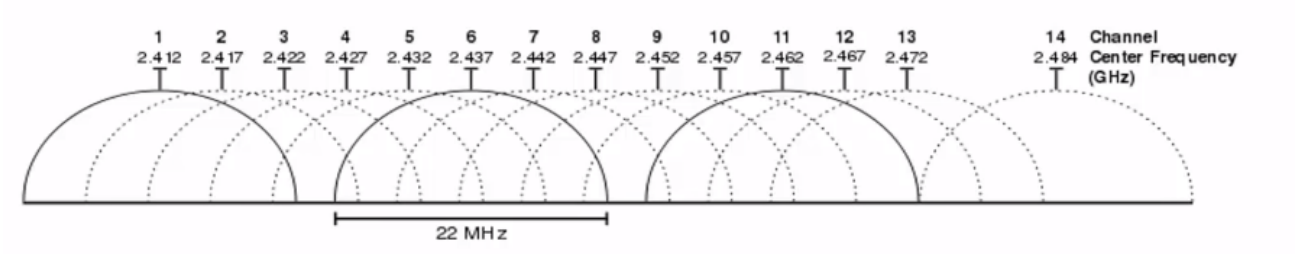
Тип кадра LLC = IEEE 802.2 = общий с Ethernet
Тип кадра MAC = ... = особенный для Wi-Fi

ПЕРЕДАЧА ДАННЫХ = электромагнитное излучение: 2.4 или 5 ГГц
ПРЕДСТАВЛЕНИЕ ДАННЫХ = метод OFMG

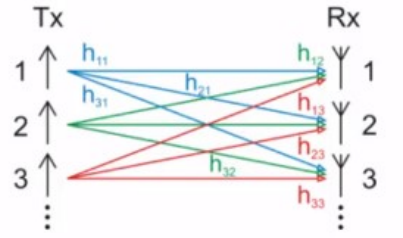
OFMG = orthogonal frequency division mutliplexing = передача данных параллельно на разных частотах. Частоты частично накладываются, однако технологичя позволяет их считывать.

В диапазоне 2.4 ГГц используется 14 каналов, таким образом, в пределах взаимной досягаемости возможно наличие только 14 сетей. Конфликт при превышении этого числа назвыается Wi-Fi джунгли.

Возможная ширина канала: 20 МГц / 40 МГц / 80 МГц / 160 МГц



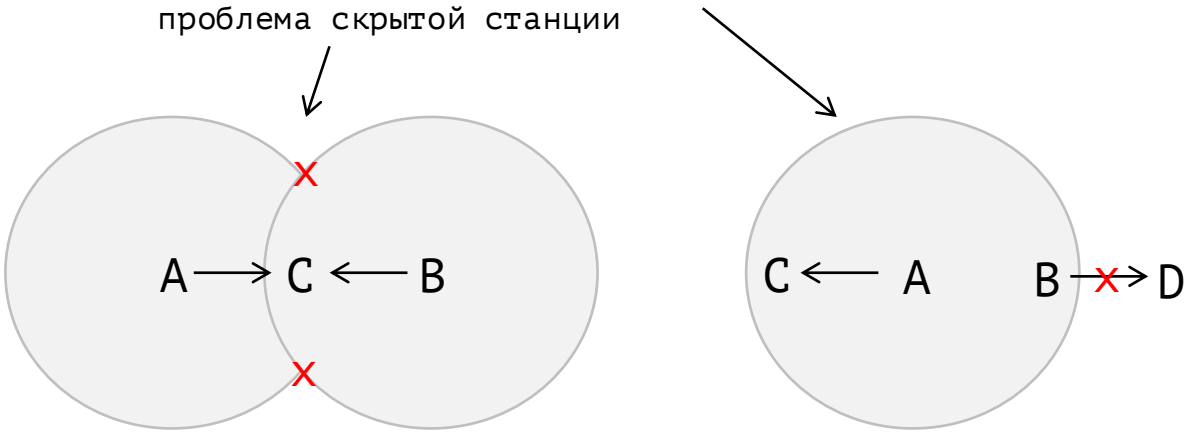
MIMO = multiple input multiple output = метод кодирования сигнала для использования нескольких антенн. Имеет смысл, если несколько антенн имеется как на предатчике, так и на приемнике.
стандарты : IEEE 802.11n и 802.11ac
пространственный поток = сигнла от одной антенны до другой. Схема передачи →



WI-FI позволяет менять скорость передачи при разном качестве сигнала. Адаптация происходит за счет изменения:
- ширины используемого канала
- методов модуляции
- guard interval (интервал между сигналами)

ОСОБЕННОСТИ БЕСПРОВОДНОЙ СРЕДЫ

- больше ошибок, чем в проводной среде
- мощность передаваемого сигнала > мощность принимаемого
- ограниченный диапазон распространения сигнала =
    проблема засвеченной станции
    проблема скрытой станции



ОБНАРУЖЕНИЕ КОЛЛИЗИЙ в WI-FI

Для удостоверения, что приемник получил пакет отправителя, используется подтверждение. Если подтверждения нет по истечении таймера_подтверждения, пакет считается непринятым.

Для обнаружения коллизий:
- невозможно использовать подход Ethernet
- допустимо обнаружение коллизий по отсутствию подтверждения но это очень дорогостоящая по времени операция. Поэтому подход заключается на в обнаружении коллизий, а в их предотвращении.

Доступ к среде осуществялется по правилу CSMA/CA
(carrier sense multiple access with collision avoidance):
- каждый хаб прослушивает сеть
- если сеть пустая, хаб отправляет пакет
- приемник
    получает кадр
    выжидает "короткий_межкадровый_интервал"
    отправляет подтверждение
- отправитель
    получает подтверждение
    выжидает "межкадровый_интервал"
    попадает в фазу "период_молчания"
- в периоде молчания находятся другие хабы
- для каждого хаба этот период выбирается случайным образом
- первый хаб, у которого истек период молчания
  (у которого меньше "слотов_ожидания"), отправляет пакет

Описанный выше метод, как правило, достаточен для корректного функционирования сети. Но теоретически он не решает проблем засвеченной / скрытой станции. Поэтому есть метод MACA, который, однако, не является обязательным и редко ипользуется. Суть:

1. перед отправкой основного пакета, отправитель посылает сообщение RTS = request to send = в котором содержится размер предполгаемого пакета.

2. если приемник готов принять пакет, то отправляет обратное сообщение CTS = clear to send = подтверждениие готовности, в котором также есть размер пакета.

3. CTS получают все компьютеры в зоне доступа
   - действтиельный отправитель отправляет пакет
   - остальные ждут достаточное время, чтобы не мешать

ФОРМАТ КАДРА в WI-FI

- подуровень MAC = особый формат
- подуровень LLC = стандартный как и в Ethernet

Трансофрмация от формата MAC к формату LLC
автоматическая на хабе подуровня LLC

АДРЕСА WI-FI

ИНФРАСТРУКТУРНЫЙ РЕЖИМ

DA = destination adress
SA = source adress

RA = reciever adress (кто получает из беспроводной среды)
TA = transmitter adress (кто передает в беспроводную среду)

РАСПРЕДЕЛЕНИЕ ПО ПОЛЯМ КАДРА

Z to Inet : A1 = Y (RA) , A2 = Z (SA = TA) , A3 = X (DA)
Inet to Z : A1 = Z (RA = DA) , A2 = Y (TA) , A3 = X (SA)

УПРАВЛ КАДРОМ	ДЛИТЕЛЬН	АДРЕС 1	АДРЕС 2	АДРЕС 3	УПРАВЛ ОЧЕРЕДН	АДРЕС 4	ДАННЫЕ	КОНТРОЛЬН СУММА
2 байта	2 байта	6 байт	6 байт	6 байт	2 байта	6 байт	0 - 2304 байт	4 байта

Используется совместно с управляющими кадрами MAC подуровня

Используется совместно с флагом MF
Включает в себя: sequnce number + fragment number

0-1 = версия протокола (текущее 00, прочее зарезервировано)
2-3 = тип (data frame / control frame / management frame)
4-7 = подтип (RTS, CTS, ACK … сервисы wi-fi)
8 = to DS (1 = от беспроводной среды к проводной)
9 = from DS (1 = от проводной среды к беспроводной)
10 = MF (1 = еще остались кадры = more fragments)
11 = RT (1 = повторная отправка)
12 = power (управление питанием, 1 = спящий режим)
13 = MD (управление питанием, 1 = данные в буфере точки доступа)
14 = WEP = protection frame (1 = используется шифрование)
15 = order (1 = используется порядок = по умолчанию)

СЕРВИСЫ (службы) WI-FI

BSS = basic service set
ESS = extended service set

В радиусе своего действия точка доступа рассылает идентификатор своего набора сервисов = BSSID, который эквивалентен mac-адресу точки доступа.

Пользователи, получающие идентификатор, воспринимают его как имя точки доступа. Например: BSSID = FA:F0:82:D9:0D:10
SSID = "my_wi_fi"

Аутентификация = подтверждение права на подключение
клиент отправляет кадр управления management frame
если запрос удовлетворен, ТД отправляет кадр подтверждения
- open = без защиты, подключается любой
- personal = один пароль для всех пользователей
- enterprise = уникальный пароль для каждого (необходим сервер аутентификации + протоколы RADIUS / LDAP etc.)

Внешняя аутентификация = подключение к ТД происходит открыто, но после этого пользователя перенаправляет на сервис авторизации, где аутентификация и происходит по дополнительным правилам (напрмер, смс пароль и проч.).

Ассоциация = установление связи с точкой доступа
клиент получил разрешение от точки доступа
клиент потправляет параметры wi-fi, с которыми может работать
если параметры подходят ТД, она отправляет подтверждение

Передача данных = после успешных аутентификации и ассоциации

Отключение от сети = выполняется одним из способов
- запрос от клиента на деаутентификацию / деассоциацию
- ТД автоматически отключает клиента через некоторое время после того, как тот покидает зону действия ТД

Расширенный набор сервисов = необходим, когда необходимо покрыть Wi-Fi сетью большую площадь, для чего используются связанные ТД. Согласование работы ТДа осуществляется контроллером. Каждая точка доступа передает идентичные SSID, но при этом их BSSID остаются уникальными.

В расширенный набор сервисов входит, например, роуминг (возможность беспрепятственно перемещаться по всей покрытой территории без необходимоти аутентификации с каждой ТД в локальной сети).

ПОИСК ТОЧКИ ДОСТУПА Wi-Fi КЛИЕНТОМ

Пассивное сканирование

- ТД периодически рассылают сигнальные пакеты = beacon frames
- клеинт принимает эти кадры и формирует список доступных ТД

Активное сканирование

- клиент сам рассылает пакеты = probe request
- TD в ответ на этот запрос посылают информацию о себе

ШИФРОВАНИЕ В Wi-Fi

Шифрование:

- подвергаются только данные, но не заголовки
- должен быть установлен флаг WEP = protection frame

Типы шифрования: WEP (устаревший), WPA, WPA2 (современный)

***** СЕТЕВОЙ УРОВЕНЬ **********

Задача - обеъединение локальных сетей, построенных на основе разных технологий канального уровня. Является основой Интернет.

- объединение сетей = internetworking
- маршрутизация
- обеспечение качественного обслуживания

АДРЕСАЦИЯ НА СЕТЕВОМ УРОВНЕ:

вводятся глобальные адреса, не связанные с mac-адресами
преобразование адресов = метод ARP для TCP/IP

СОГЛАСОВАНИЕ РАЗМЕРОВ ПАКЕТА = метод фрагментации

- промежуточное устройство оценивает технологию следующего канала
- если пропускная способность
  меньше размера пакета - пакет делится на части (фрагментация)
  достаточноая - пакет передается в неизменном виде
- очередное устройство объединяет фрагментированные пакеты
- отправитель и получатель не "заботятся" об этой процедуре

ETHERNET и WI-FI очень похожи, так как Wi-Fi является адптацией Ethernet для беспроводной среды. Поэтому для взаимодействия этих двух технологий достаточно канального уровня и сложное сетевое согласование не нужно. Однако эти технологии НЕ МАСШТАБИРУЕМЫ

МАСШТАБИРОВАНИЕ НА СЕТЕВОМ УРОВНЕ

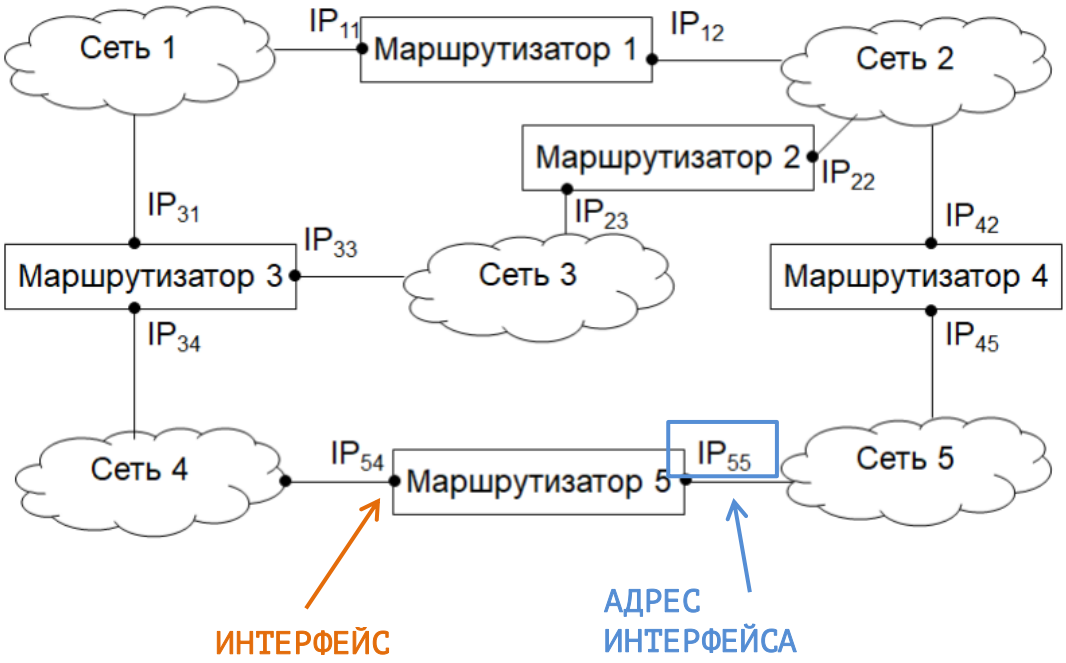
- агрегация адресов = работа с блоками адресов (сетями)
- зепрет на пересылку пакету, адресация которых не найдена
- возможность наличия нескольких путей в сети = маршрутизация

ПРОТОКОЛЫ СЕТЕВОГО УРОВНЯ:

IP = передача данных
ICMP = управление сетью
ARP = связь глобального и локального адресов
DHCP = автоматическое назначение глобальных адресов

управляющие протоколы сетевого уровня

ОБОРУДОВАНИЕ = маршрутизатор (router)



МАРШРУТИЗАЦИЯ (routing) = поиск маршрута доставки пакета между сетями через транзитные узлы. Задача выполняется маршрутизатором. При этом необходим:

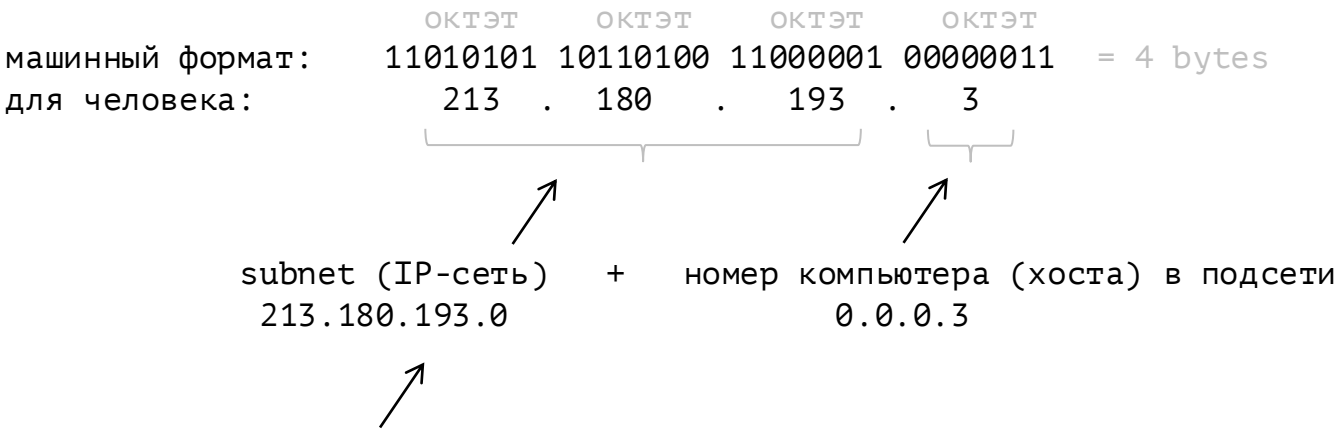
- учет изменений топологии сети
- учет загрузки каналов связи и маршрутизаторов

Для каждой порции данных задача решается отдельно

***** IP ADRESS **********

Глобальные адреса
- не привязаны к технологии локальных сетей
- уникальны в пределах для составной сети
- две версии протокола для модели TCP/IP:
  IPv4 : 4 байта
  IPv6 : 16 байт

IPv4



Маршрутизаторы работают с подсетями. Не всегда деление на подсеть и хост происходит в соотношении 3 к 1 (возможны и другие), однако общее правило: старшие биты = подсеть, младшие биты = хост

МАСКА ПОДСЕТИ позволяет выделить подсеть и хост из IP адреса
содержить те же самые 32 бита:
1 = на позициях, где подсеть
0 = на позициях, где хост

```
возможный вариант маски:      11111111 11111111 11111111 00000000
десятичная запись:            255.255.255.0
запись в виде префикса:       213.180.193.3 /24    (24 бита =  1)

возможный вариант маски:      11111111 11111111 11110000 00000000
десятичная запись:            255.255.240.0
запись в виде префикса:       213.180.193.3 /20    (20 бит  =  1)
                    подсеть =  213.180.192.0
                       хост =  0.0.1.3
```

КЛАССЫ IP адресов

класс D = групповые адреса = 224.0.0.0 - 239.255.255.255
класс E = зарезервировано = 224.0.0.0 - 255.255.255.255

ТИПЫ АДРЕСОВ в IPv4

индивидуальный (unicast) - только один компьютер
групповой (multicast) - несколько компьютеров подсети
шировоквещательный (broadcast) - все компьютеры подсети

ШИРОКОВЕЩАТЕЛЬНЫЙ АДРЕС

доступен только внутри одной подсети
невозможно указать "все компьютеры сети" ("Godzillagram")
имеет формат:
- биты адреса подсети = как обычно
- биты адреса хоста = все выставлены в 1

направленное шировковещание =
- от компьютера подсети A к компьютерам подсети B
- пакет будет пропущен только в сеть B
- формат адреса: описан выше

ограниченное широковещание =
- от компьютера подсети A к компьютерам подсети A
- пакет не будет пропущен в другие подсети
- формат адреса: 255.255.255.255

СЛЕДСТВИЯ из ФОРМАТА ШИРОКОВЕЩАТЕЛЬНОГО АДРЕСА

в IP адресе нельзя оставлять только 0 или только 1 в адресе хоста, так как это будет означать:
- только 0 = остается только адрес подсети
- только 1 = широковещательный адрес

ОСОБЫЕ IPv4 АДРЕСА

0.0.0.0	текущий хост
255.255.255.255	все хосты текущей подсети
127.0.0.0 /8	обратная петля, данные приходят обратно как правило, используется адрес хоста 1 но можно использовать любой адрес хоста
169.254.0.0 /16	ОС автоматически назначает адрес если недоступна другая конфигураци IP могут использовать только в одной подсети не проходят через маршрутизатор

РАСПРЕДЕЛЕНИЕ IP адресов в МИРЕ

IANA (Internet assigned numbers authority) =
- распределение IP мира
- для получения IP адресов необходимо обращение в IANA
- адреса распределяются через RIR

RIR (regional Internet register) =
- непосредственное назначение адресов
- всего несколько (пять) организация …
- … закрепленые за континентами

ЧАСТНЫЕ IP адреса = не нужно обращаться в IANA. Эти адреса зарезервированы для случаев, когда НЕ нужно подключение к Интернет. Регламентируются документом "RFC 1918"
- 10.0.0.0 /8 до 10.255.255.255
- 172.16.0.0 /12 до 172.31.255.255
- 192.168.0.0. /16 до 192.168.255.255 соответственно

При необходиомости подключения к Интернет, используется технология NAT = Network adress translation

NAT = внутри организации можно строить какие угодно локальные сети, и пользоваться внутренними IP адресам. Для выхода в Интернет используется всего один единственный IP адрес.

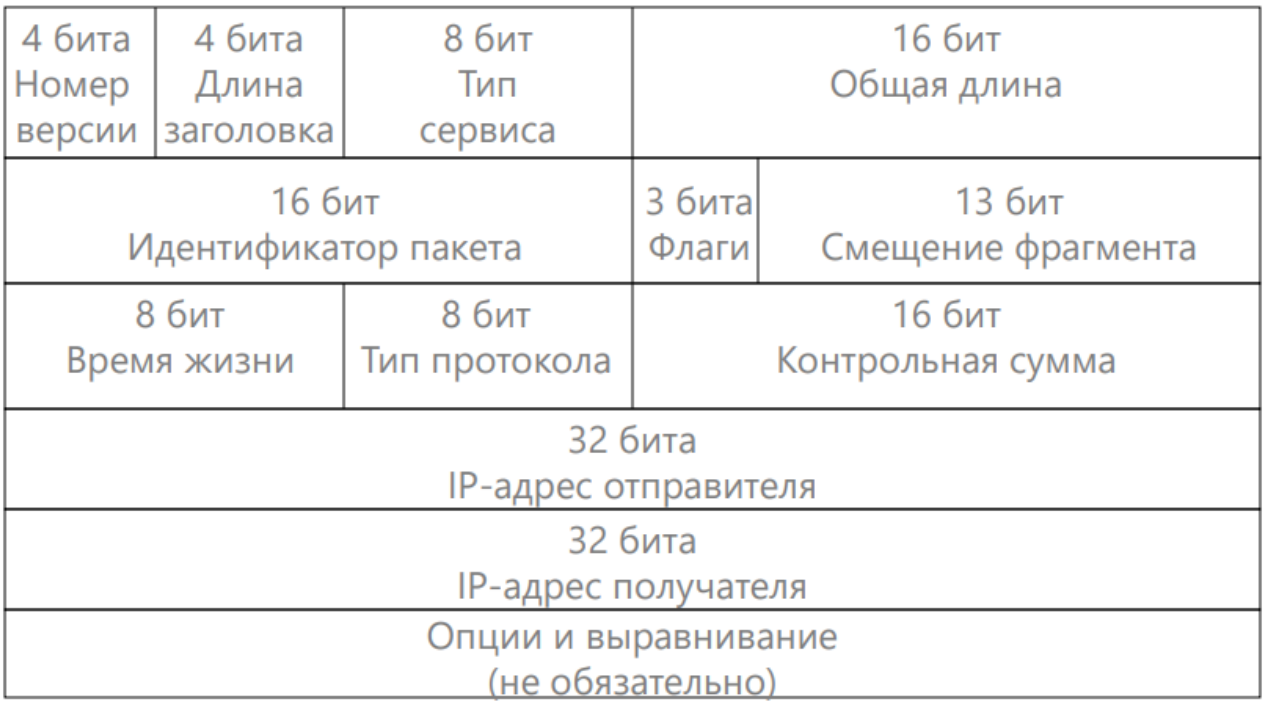
По мере исчерпания адресов IPv4 используются IPv6 (длина 16 байт)

***** IP ПРОТОКОЛ **********

Передача данных без гарантии доставки
без сохранения порядка следования сообщений

Передача данный без установки соединения. Обратного подтверждения нет, исправлением ошибок должен заниматься протокол уровня выше.

ФОРМАТ ЗАГОЛОВКА IP



номер версии IPv4 или IPv6
длина заголовка размер всего заголовка в байтах
тип сервиса редко используется на практике
общая длина размер всего пакета в байтах

идентификатор, флаги, смещение см. фрагментация

время жизни измеряется в hop'ax = отрезки между роутерами
тип протокола код протокола верхнего уровня
контрольная сумма расчитывается только по заголовку пакета

опции = необязательно поле, которое может включать:
- записать маршрут по маршрутизаторам
- временные метки прохождения по маршрутизаторам
- кастомизация маршрута

обязательно выравнивание размера заголовка до границы в 32 бита

***** IP ПРОТОКОЛ - маршрутизация **********

ЭТАПНОСТЬ

1. изучение структуры сети = производится "в фоновом режиме"
2. продвижение (forwarding) = куда отправит конкретный пакет

ИЗУЧЕНИЕ СТРУКТУРЫ

... (в продвинутом курсе) ...

FORWARDING

Осуществляется на основе таблицы маршрутизации, которая заполняется на этапе изучения структуры. Заполнение возможно:

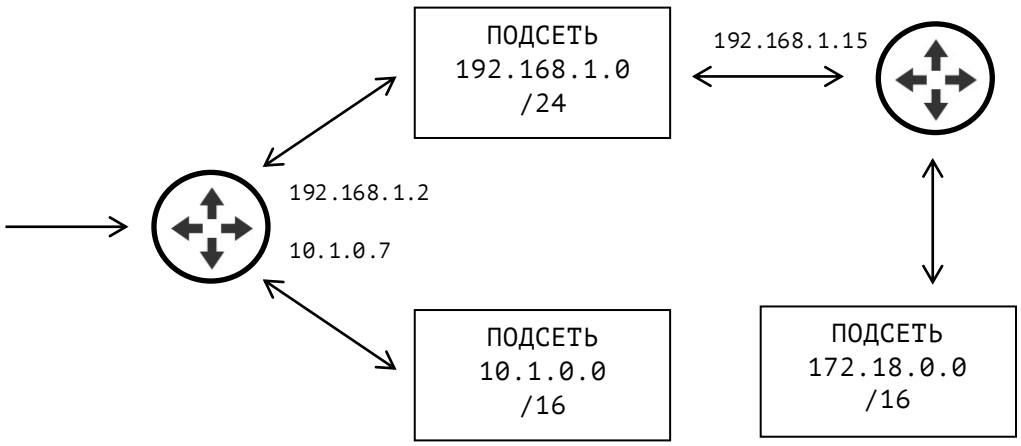
- статическое (вручную)
- динамическая (авто., протоколы RIP, OSPF, BGP etc.)

Обязательный поля таблицы (есть и другие специальные поля):

адрес, маска = задают подсети, о которых знает роутер
интерфейс = ip адреса (или имена) интерфейсов роутера
шлюз (gateaway) = что делать с пакетом, отправленным через интерфейс

- шлюз = соседний роутер
- "подсоединен" - подсеть подключенна напрямую
- ip адрес соседнего роутера, если передача дальняя

учитываются адреса только ближайшей подсети / роутера
метрика = показатель приоритетности пути от роутера до конечного адресата, (количество роутеров, пропускная спобоность, загруженность). Чем меньше метрика, тем путь приортетнее.



АДРЕС	МАСКА	ШЛЮЗ	ИНТЕРФЕЙС	МЕТРИКА
192.168.1.0	255.255.255.0	подсоединен	192.168.1.2	276
10.1.0.0	255.255.0.0	подсоединен	10.1.0.7	276
172.18.0.0	255.255.0.0	192.168.1.15	192.168.1.2	306
0.0.0.0	0.0.0.0	*ip шлюза*	*ip интерфейса*	0

Пакет, адреса подсети назначения которого в таблице маршрутизации не существует, либо:

- отбрасывается
- перенаправляется на роутер по умолчанию (default gateaway)

GATEAWAY = это маршрутизатор для отправки пакетов по умолчанию (для неизвестных роутеру сетей). Как правило, подключен к Интернет. Условное обозначение: 0.0.0.0 маска 0.0.0.0 или "default".

ДВЕ ПОДХОДЯЩИЕ ЗАПИСИ

Роутер принял пакет с адресом 192.168.100.12
Для него подходит две записи 192.168.100.0 /24 vs 192.168.0.0 /16
Общее правило маршрутизации:

- поиск маршрута к хосту = маска /32
- поиск маршрута к сети = маска от /31 до /1
- маршрут по умолчанию = маска /0 , подходят все пакеты

***** IP ПРОТОКОЛ - фрагментация **********

В заголовке IP есть следующие поля для фрагментации:
иднетификатор, флаги, смещение

идентификатор = присваивается пакету, идентичен у фрагментов

флаги = зарезервированный бит
DF : 1 = запрет на фрагментацию
MF : 1 = есть еще фрагменты, 0 = последний фрагмент
если установлен запрет на фрагментацию, но пакет по размеру больше максимального размера кадра нижнего уровня сети получателя, то
пакет отбрасывается
получателю отправляется сообщение о случившемся

смещение = способ упорядочить передаваемые / получаемые пакеты, так как технология IP не гарантирует последовательный прием пакетов. Смещение записывается в 8 байтовых блоках. Пример:
пакет = 4000 байт минус 20 байт заголовка = 3980 байт

	фрагменты		смещение
1	0 - 1479	0 / 8 =	0
2	1480 - 2959	1480 / 8 =	185
3	2960 - 3980	2960 / 8 =	370

ФРАГМЕНТАЦИЯ в IPv6 = отсутствует. Обязанность подбора допустимого размера пакета возлагается на хосты-отправитеи. Технология подбора размера пакета = Path MTU Discovery

***** УПРАВЛЯЮЩИЕ ПРОТОКОЛЫ СЕТЕВОГО УРОВНЯ **********

DHCP = dynamic host configuration protocol
позволяет автоматически назначать компьютерам в сети IP адреса

ARP = adress resolution protocol
позволяет по IP адресу получить MAC адрес компьютера

ICMP = internet control message protocol
- сообщение об ошибках
- тестирование работы сети:
    ping = проверка доступности получателя (linux - link)
    traceroute = определение маршрута к получателю (linux - link)

***** DHCP **********

DHCP = dynamic host configuration protocol
позволяет автоматически назначать компьютерам в сети IP адреса

Особенности:
- необходима инфраструктура = DHCP сервер
- IP адреса могут меняться

Протокол работает по ТЕХНОЛОГИИ КЛИЕНТ-СЕРВЕР
Клиент = компьютер, который получает IP адрес
Сервер = компьютер, выдающий адреса и контролирующий уникальность

Порядок сообщений DHCP для получения IP адреса
- К: D dicover = поиск сервера (FF:FF:FF:FF)
- С: O offer = предложение IP адреса
- К: R request = согласие на IP адрес
- С: A ack = подтверждение IP адреса

Дополнительные сообщения DHCP
- nack = (С) запрет использования запрошенного IP адреса
- release = (К) освобождение IP адреса
- inform = (К) запрос информации об уже имеющемся IP адресе

Условия возможности работы протокола DHCP:
- клиент и сервер в одной подсети либо
- роутер использует настроку DHCP RELAY
  (разрешает шировковещательные адреса для пакетов DHCP)

СПОСОБЫ НАЗНАЧЕНИЯ IP АДРЕСОВ

Фиксированный = один физический MAC vs один выделенный IP
Динамический =
- адрес выделяется из доступного серверу диапазона адресов
- адрес выделается на ограниченное время = lease time
- после окончания аренды IP адрес освобождается
- продление аренды = request + ack

ОПЦИИ DHCP = дополнительный конфигурационные параметры, небходимые хосту для работы в сети:
- маска подсети
- шлюз по умолчанию
- адреса DNS-серверов
- адреса серверов времени
- маршруты
- etc. …

***** ARP **********

ARP = adress resolution protocol
позволяет по IP адресу получить MAC адрес компьютера

Протокол работает по ТЕХНОЛОГИИ ЗАПРОС-ОТВЕТ
Порядок работы протокола:
- есть компьютер, который хочет узнать MAC по IP
- он формирует ARP-запрос ("чей IP … ?") и
- отправляет его на широковещательный адрес (FF:FF: …)
- его получают все компьютеры подсети
- компьютер, чей IP совпадает, отвечает на запрос
- MAC адрес извлекается из этого ответа

ФОРМАТ ARP-запроса / ответа

тип сети	1 = ethernet, …
тип протокола	2048 = IP
длина локального адреса	6 = MAC
длина глобального адреса	4 = IP
операция	1 = запрос, 2 = ответ
локальный адрес отправителя	соотв. адрес
глобальный адрес отправителя	соотв. адрес
локальный адрес получателя	пустой для запроса
глобальный адрес получателя	соотв. адрес

ARP формально принадлежит сетевому уровню, но ARP запросы и ответы вкладываются в кадры протоколов канального уровня. Из-за того, что роутеры не пропускают пакеты на широковещательные адресе, процедура возможна только в пределах одной подсети.

После получения MAC-адреса, он кэшируется на компьютере в таблице. ARP таблица имеет поля = IP адрес : MAC адрес : Тип (стат/динам)

GRATUITOUS ARP = добровольный ARP запрос
Запрос собственного IP адреса
- оповещение всех компьютеров подсети о моем IP адресе
- проверка наличия одинаковых IP адресов в подсети

***** ICMP **********

ICMP = internet control message protocol
- сообщение об ошибках (не обязательно должны обрабатываться)
- тестирование работы сети:
    ping = проверка доступности получателя
    traceroute = определение маршрута к получателю

ФОРМАТ ПАКЕТА ICMP

    1 байт   = тип сообщения (что произошло)
    1 байт   = код сообщения (еще конкретнее)
    2 байта   = контрольная сумма
    4 байта   = служебная информация (зависит от когда и типа)
    поле данных = как правило, фрагмент пакета, в котором ошибка

ТИПЫ ICMP сообщений

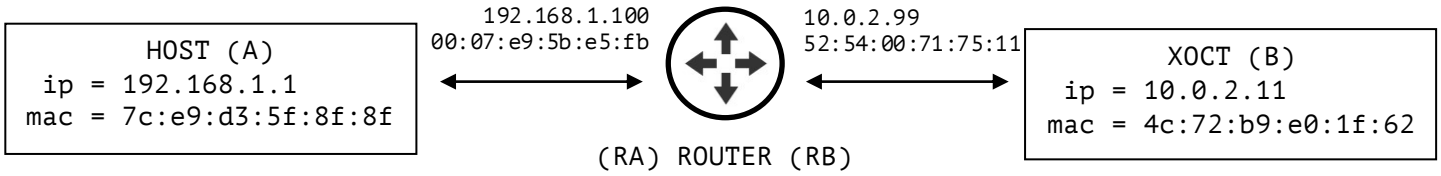
     0 - эхо-ответ
     3 - узел назначения недостижим
     5 - перенаправление маршрута
     8 - эхо-запрос
     9 - сообщение о маршрутизаторе
    10 - запрос сообщения о маршрутизаторе
    11 - истечение времени жизни пакета
    12 - проблемы с параметрами
    13 - запрос отметки времени
    14 - ответ отметки времени

ПАКЕТ ICMP вкладывается внутрь пакета IP

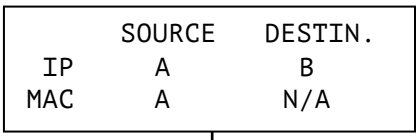
***** ВЗАИМОДЕЙСТВИЕ КАНАЛЬНОГО и СЕТЕВОГО УРОВНЕЙ **********

Канальный уровень =
    передача данных внутри подсети
    mac-адрес

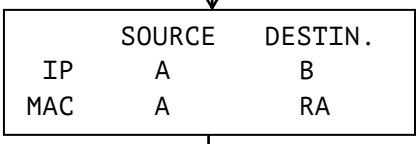
Сетевой уровень =
    передача данных между подсетями
    ip-адрес



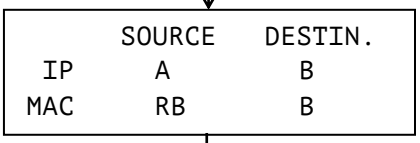
Начальное состояние:
    (a) хочет передать пакет в (b)
    (a) известен только ip адрес (b)



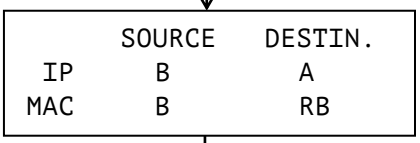
1 шаг:
- проверка по IP = входит ли (B) в подсеть (A)
- так как не входит, нужен маршрутизатор
- протокол ARP = узнать MAC маршрутизатора
- передать пакет на маршрутизатор



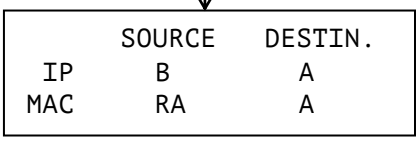
2 шаг:
- маршрутизатор определяет подсеть назначения
- протокол ARP = узнать MAC получателя (B)
- замена полей MAC адресов в пакете
    source = (RB)
    destination = (B)



3 шаг:
- (B) хочет отправить ответ обратно (A)
- (B) меняет местами IP адреса
- (B) меняет местами MAC адреса
- т.о. передает пакет на маршрутизатор



4 шаг:
- маршрутизатор определяет подсеть назначения
- скорее всего, сохранил данные MAC получателя
- замена полей MAC адресов в пакете
    source = (RA)
    destination = (A)



***** ТРАНСПОРТНЫЙ УРОВЕНЬ **********

Задача - взаимодействие процессов на хостах
- здесь задается адрес получателя формата ip:port
- обеспечение целостности данных
гарантия доставки = подтверждение получения
гарантия порядка следования сообщений = нумерация

Этот уровень является сетенезависмым (реализуется с использованием интерфейса сокетов). Весь стек протоколов нижних уровней можно заменить, но процесса на хосте это никак не коснется.

Протоколы
UDP = ненадежная доставка коротких 'сообщений'
TCP = надежная доставка длинных 'сообщений'

ПОРТ = адрес процесса на транспортном уровне (по сути просто номер, по которому сетевой процесс может быть идентифицирован в системе)
диапазон адресов: 1 - 65535
адреса не повторяются у разных процессов
форма записи: IP:port = 192.68.0.1:80

1 - 1024 = 'хорошо_известные_порты' (80 https , 53 dns …)
1025 - 49151 = регистрация в Interner Assigned Numbers Authority
49152 - 65535 = динамические порты (автоматически назнает ОС)

***** UPD **

UDP = user datagram prtocol
передаваемые данные = дэйтаграмма (по аналогии телеграммой)

Особенности:
- не устанавливает соединения
- нет гарантии доставки данных
- нет гарантии сохраненния порядка следования сообщений

таким образом, надежность НЕ ПОВЫШАЕТСЯ относительно протокола IP

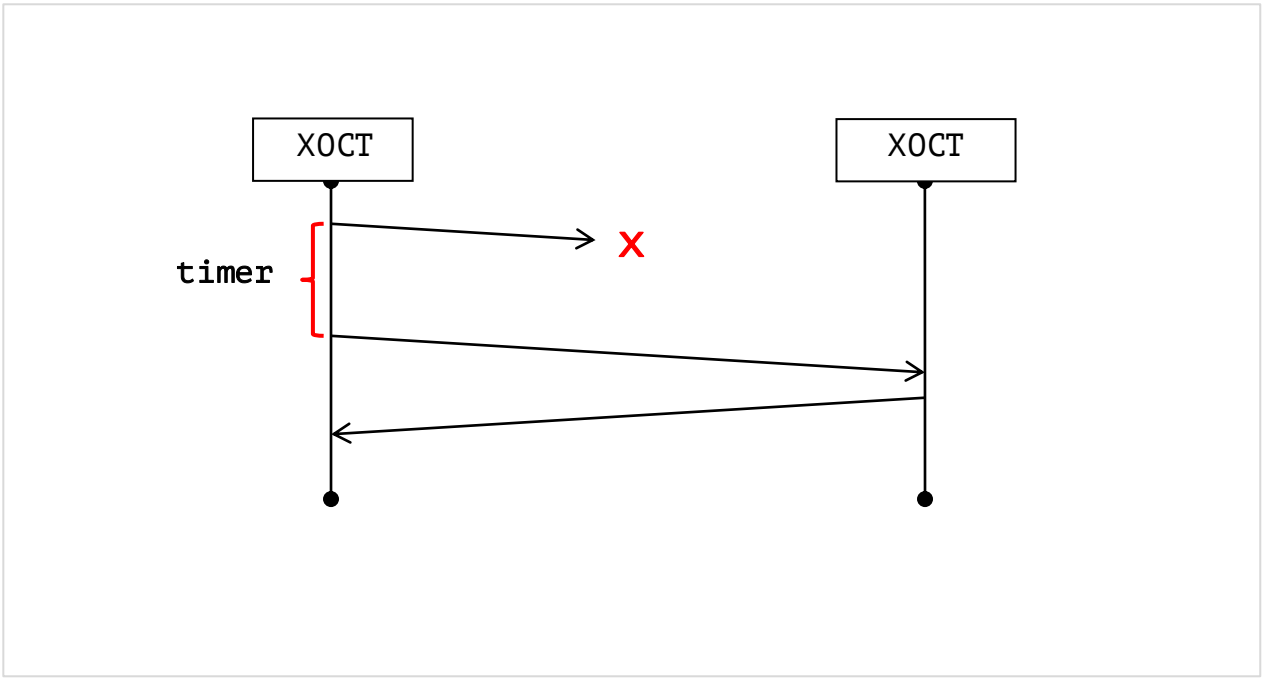
Применение = короткие сообщения / клиент сервер
(например в протоколе DNS)

ФОРМАТ ЗАГОЛОВКА

2 байта	2 байта	2 байта	2 байта
ПОРТ ОТПРАВИТЕЛЯ	ПОРТ ПОЛУЧАТЕЛЯ	ДЛИНА UDP	КОНТРОЛЬНАЯ СУММА

Длина дэйтаграммы:
min = 8 байт (только заголовок)
max = 65515 (ограничение длиной IP-пакета)

ПОРЯДОК ОБМЕНА ДЕЙТАГРАММАМИ



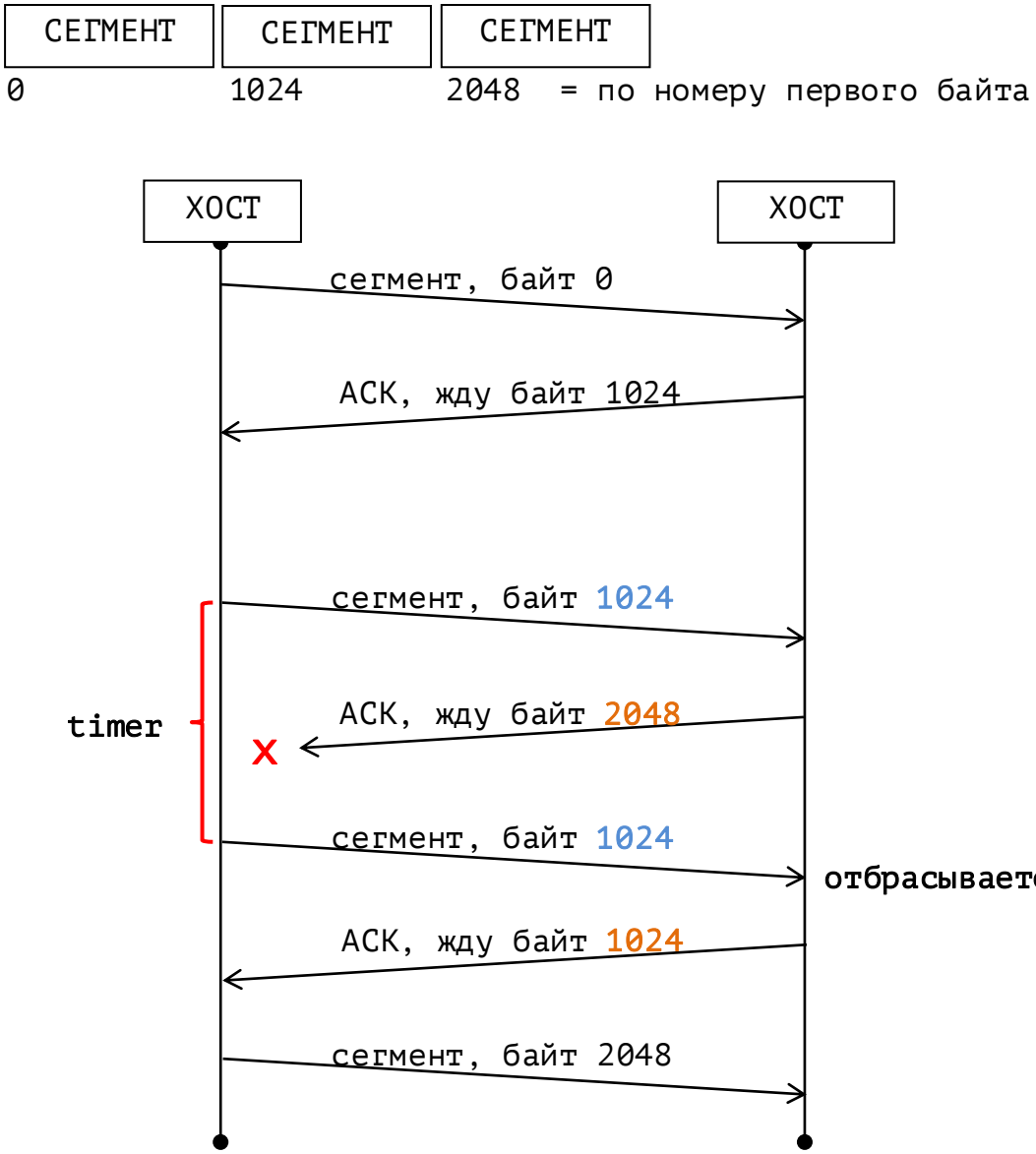
***** TCP **

TCP = transport control protocol
передаваемые данные = сегменты (reliable byte stream)

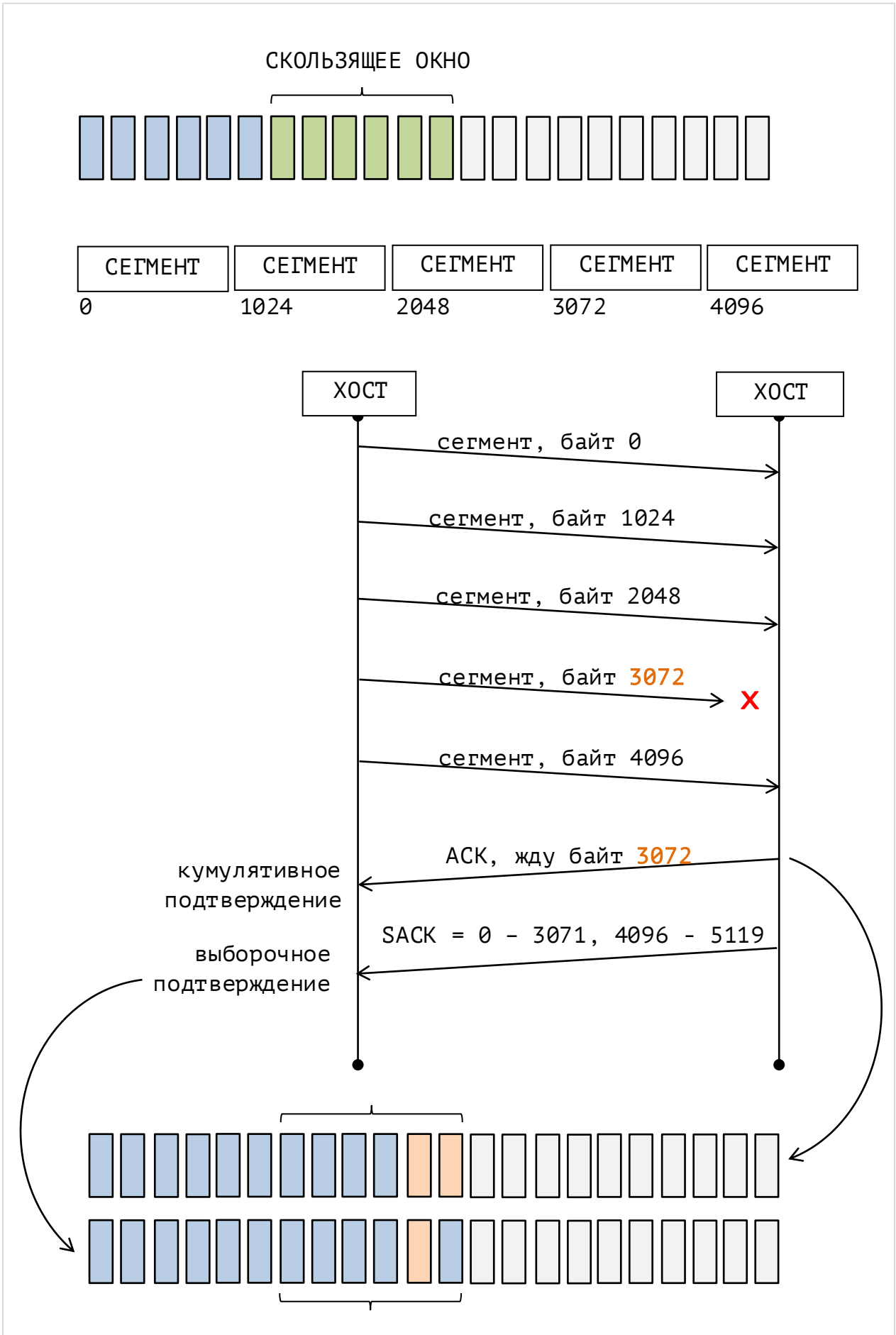
Особенности:
- установка соединения перед отправкой
- гарантия доставки данных
- сохранение порядка следования сообщений

1. приложение >> поток байт >> транспортная подсистема
2. транспортная подсистема >> сегменты >> транспортная подсистема
3. транспортная подсистема >> поток байт >> приложение

МЕХАНИЗМ СОХРАНЕНИЯ ПОРЯДКА СЛЕДОВАНИЯ

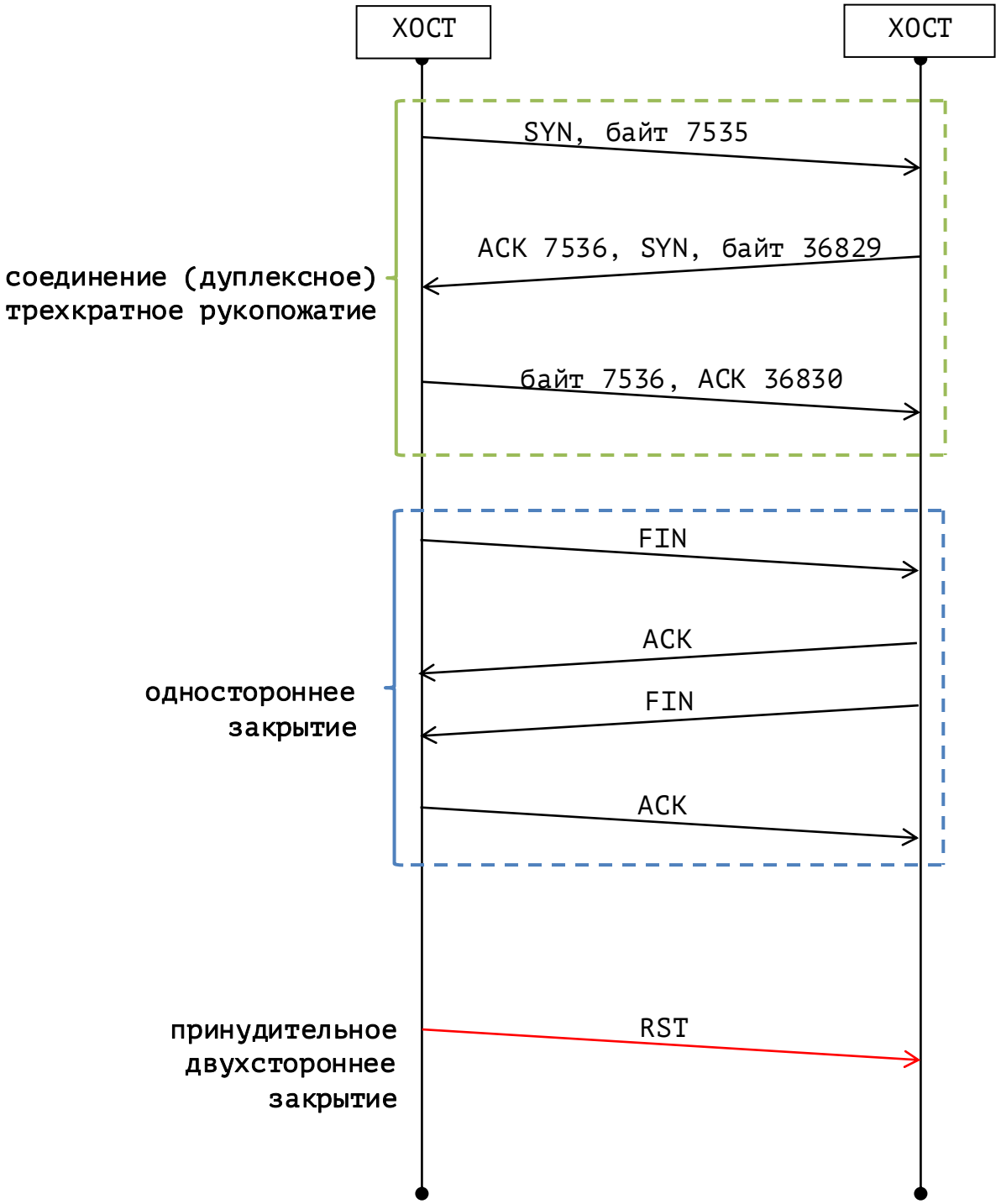


СКОЛЬЗЯЩЕЕ ОКНО

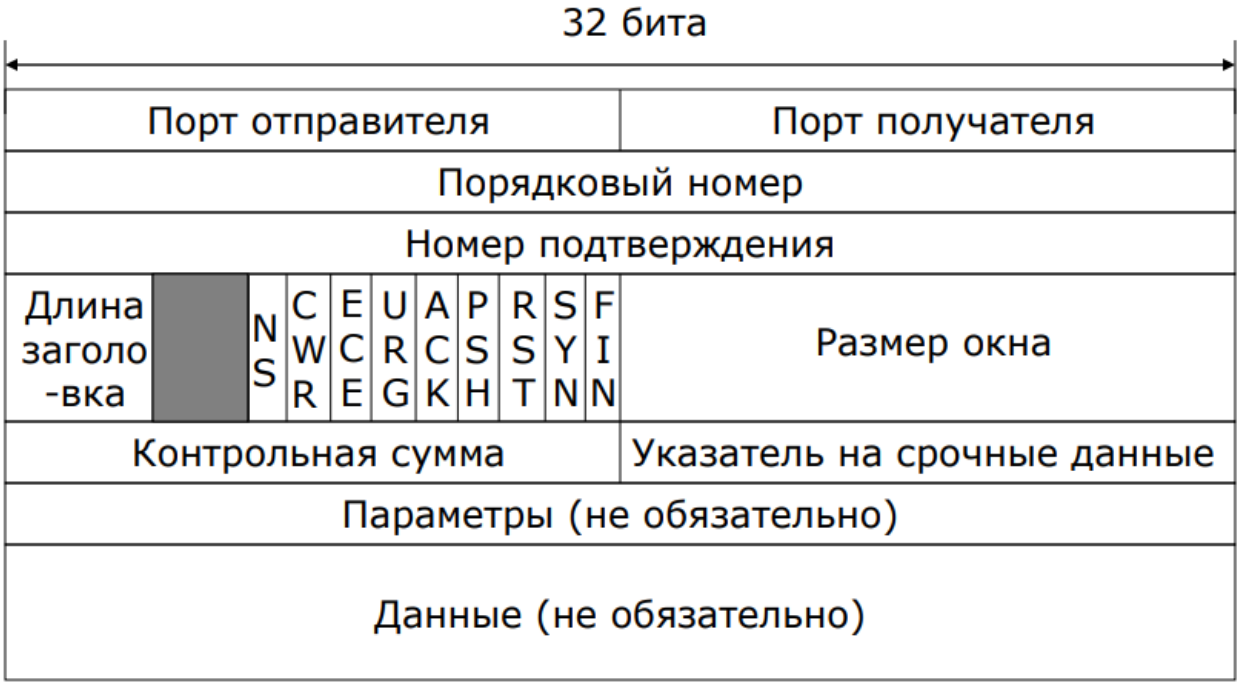


СОЕДИНЕНИЕ и РАЗРЫВ

SYN = synchronization (байт выбирается по сложной технологии)
ACK = acknowledge (запрос следующего байта = как при передаче)
FIN = finish (один FIN = закрытие только в одну сторону)
RST = reset (полное закрытие соединения)



ФОРМАТ ЗАГОЛОВКА



Порядковый номер = № байта для обозначения отправляемого сегмента
Номер подтверждения = № байта для кумулятивного повдтверждения

3 бита = зарезервированы
NS = защита от случайного/злонамеренного изменения CWR и ECE
CWR = congestion window reduced
ECE = explicit congestion echo
URG = есть срочные данные (исп. совместно с указателем)
ACK = подтверждение (потчи на всех пакетах, кроме первого = SYN)
PSH = push = напрямую в приложение, минуя буфер (не исп.)
RST = разрыв соединения (односторонный)
SYN = установка соединения
FIN = разрыв соединения (двухсторонний принудительный)

Размер окна = сколько данных готов принять (управление потоком)

Указатель на срочный денные = совместно с флагом URG.
Сейчас флаг и поле не используются

Параметры (часто используются в TCP)
   MSS = maximum size segment (задается при установке связи)
   SACK = selective acknowledgment (выборочное подтверждение)
   масштаб окна = изменить размер окна в большую сторону
   метки времени
   etc.

ТЕХНОЛОГИЯ NAT

Трансляция внутренних IP адресов во внешние осуществялется с помощью устройства NAT, содержащего NAT-таблицу

NAT-устройство имеет зарегистрированный в IANA уникальный IP-адрес

NAT-таблица, поля:
- внутренний IP = адрес хоста в сети
- внутренний порт = порт хоста в сети
- внешний IP = адрес NAT устройства
- внешний порт = случайный порт, назначаемый NAT'ом

При необходимости подключения внутреннего устройства к глобальной сети, NAT-устройство:
1. присваивает этому устройству случайный порт
2. заголовок IP >> адрес устройства заменяет на свой адрес
3. заголовок TCP/UDP >> случайный порт (см. п1)
4. заполняет NAT-таблицу

При получении данных из сети, NAT-устройство в обратном порядке по таблице восстанавливает адрес и порт внутреннего устройства.

Другими словами, на устройства NAT устанавливается однозначное соответствие между: внутренний IP:порт = внешний порт

Типы NAT отображения адресов:

- уникальное = 1 внешний IP : 1 внутренний IP
  эффективно при объединении локальных сетей, внутри каждой их которых есть повторяющиеся локальные IP адреса, но есть возможность получить набор уникальных адресов. Таким образом, внутри обеих сетей хосты продолжают взаимодействовать по прежним адрас, а при взаимодействии с устройством другой сети = уникальный адрес

- динамическое = 1 внешний IP : поочередно неск. внутренних IP
- masquerading = 1 внешний IP : одновременно все внутренние IP

Преимущество NAT = сглаживание нехватки IP + конфиденциальность
Недостатки NAT = отсутствие прозрачности +
сбои некоторых прикладных протоколов

УПРАВЛЕНИЕ ПОТОКОМ

ЗАТОПЛЕНИЕ = ситуация, при которой по какой-либо причине (особенность разработанного программного обеспечения, низкая производительность, сбой в работе получателя без оповещения отправителя …)

УПРАВЛЕНИЕ ПОТОКОМ = предотвращение затопления. Осуществляется с помощью поля "размер окна" заголовка TCP: получатель отправляет подтверждение, в котором содержится:

- ожидаемый байт (поле "номер подтверждения")
- суммарный ожидаемый объем данных (поле "размер окна")

Если получатель установил размер окна равным 0 (не готов принимать данные), то отправитель останавливается и ждет.

Для восставновления передачи, получатель повторно отправляет последнее подтверждение, но устанавливая уже не нулевой размер окна

Если отправитель ждет слишком долго, то отправляет сегмент под названием Zero Window Probe (уточнение, можно ли уже начать передавать данные или все еще следует ожидать). Этот запрос эффективен так же и для того, чтобы проверить, не оборвалось ли соединение.

В данном случае под ОКНОМ подразумеватеся ОКНО УПРАВЛЕНИЯ ПОТОКОМ

УПРАВЛЕНИЕ ПЕРЕГРУЗКОЙ

ПЕРЕГРУЗКА = состояние СЕТИ, при котором промежуточные хабы не успевают передавать сегменты в том количестве, в котором их отправляют отправители. В таком случае, сегменты просто теряются.

УПРАВЛЕНИЕ ПЕРЕГРУЗКОЙ осуществяется с помощь регулирования окна, используемого отправителем, при определении количества сегментов, передаваемых в сеть.

В данном случае под ОКНОМ подразумевается ОКНО ПЕРЕГРУЗКИ. Размер этого окна НЕ СОДЕРЖИТСЯ в заголовке TCP - расчет производится самим отправителем.

СИГНАЛЫ О ПЕРЕГРУЗКЕ = потеря сегмента
задержка сегмента
сигнал от маршрутизатора

Потеря сегмента : считается, что сети достаточно надежные, поэтому потеря сегмента, скорее всего, свидетельтсвует о перегрузке, а не о каком-то техническом/аппаратном сбое.

возможно Random Early Detection = маршрутизатор отбрасывает пакеты до того, как случается перегрузка

проблема 1 = реакция на перегрузку, а не предоствращение
проблема 2 = т.к. в TCP нет рандомизации задержки отправления (как, например, в Wi-Fi), поэтому после сбоя все снова начинают одновременно отпралять данные = новая перегрузка.

Задержка сегмента : измерение RTT (Round Trip Time) = время, за которое приходит получение подтверждения от получателя с момента отправки ему сегмента. Если RTT существенно увеличивается - уменьшается окно перегрузки

проблема 1 = надежность ниже, т.к. разные факторы задержки
проблема 2 = отправители с методом задержки подавляются в сети отправителями с методом потери сегмента - несправедливость.

Комбинированное = потеря + задержка (прим. Compound TCP Microsoft)

Сигнал от маршрутизатора = Explicit Congestion Notification.

маршрутизатор на принятом сегменте устанавливает флаг в заголовке IP = 2 младших бита поля "тип сервиса" >> 11
передает сегмент получателю

получатель, отправляя подтверждение, устанавливает флаг в заголовке TCP = ECE >> 1

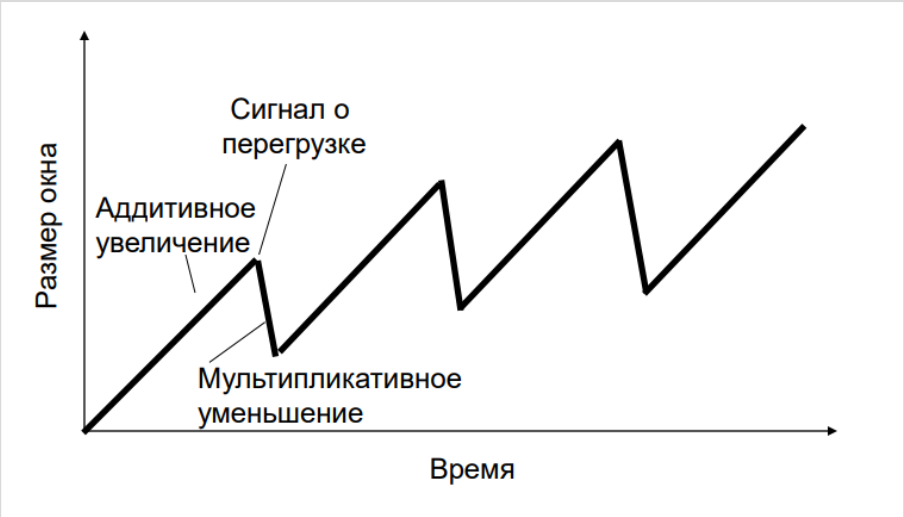
отправитель, обнаружив это, в новом сегменте устанавливает флаг в заголовке TCP = CWR >> 1
и передает в сеть

СПОСОБЫ РАССЧЕТА РАЗМЕРА ОКНА ПЕРЕГРУЗКИ

AIMD = additive increase / multiplicative decrease

$$w(t+1) = \begin{cases} w(t) + a, & \text{если нет перегрузки} \\ w(t) * b, & \text{если есть перегрузка , как правило :} \end{cases}$$

a = MSS (maximum segment size)
b = 1/2

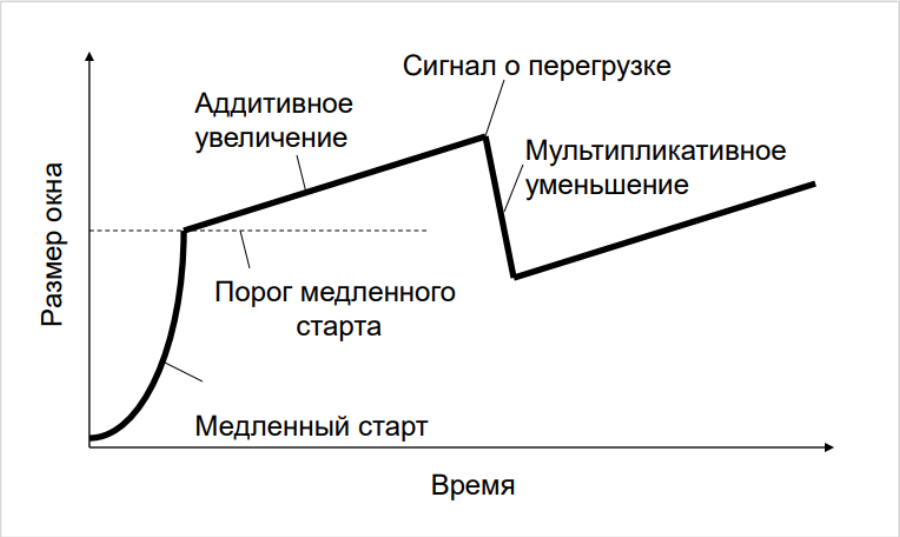


МЕДЛЕННЫЙ СТАРТ =

начинаем с маленько размера окна
на каждый 1 подтвержденный сегмент отправляем 2 новых
при получении сигнала о перегрузке, начинаем сначала

КОМБИНИРОВАННЫЙ СПОСОБ =

начинается с медленного старта
достижение "порога медлнного старта" >> AIMD

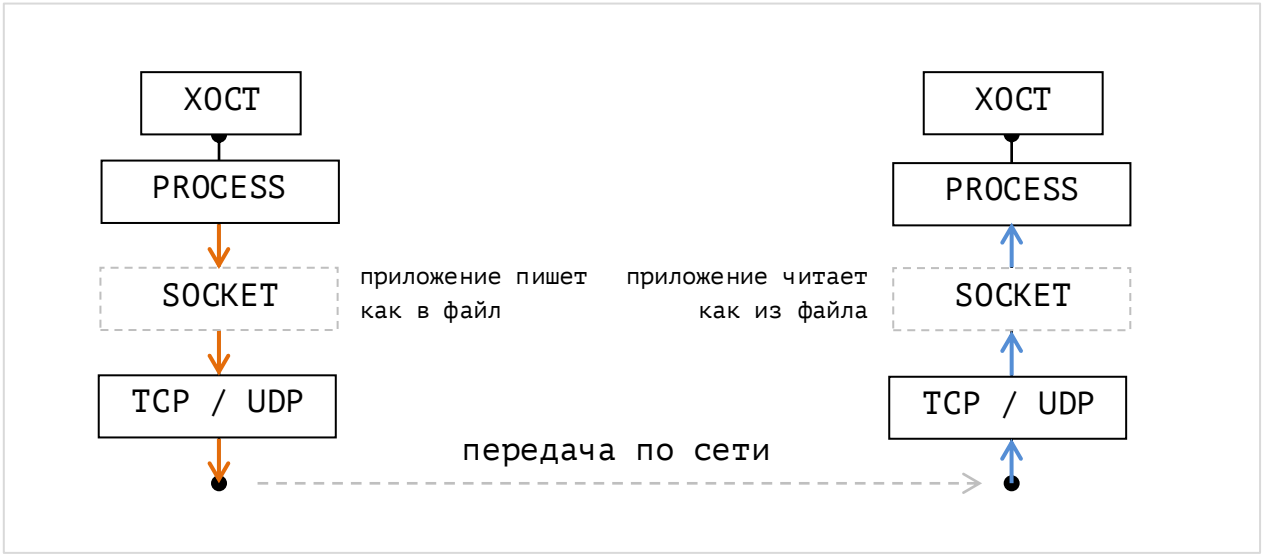


***** SOCKET INTERFACE **********

СОКЕТ = специальный тип файла, посредством записи в который и чтения из которого осуществляется взаимодействие компьютеров сети.

ИНТЕРФЕЙС СОКЕТА = интерфейс, который предоставляет транспортный уровень приложению, желающему работать с сетью.

Схема использования сокетов:

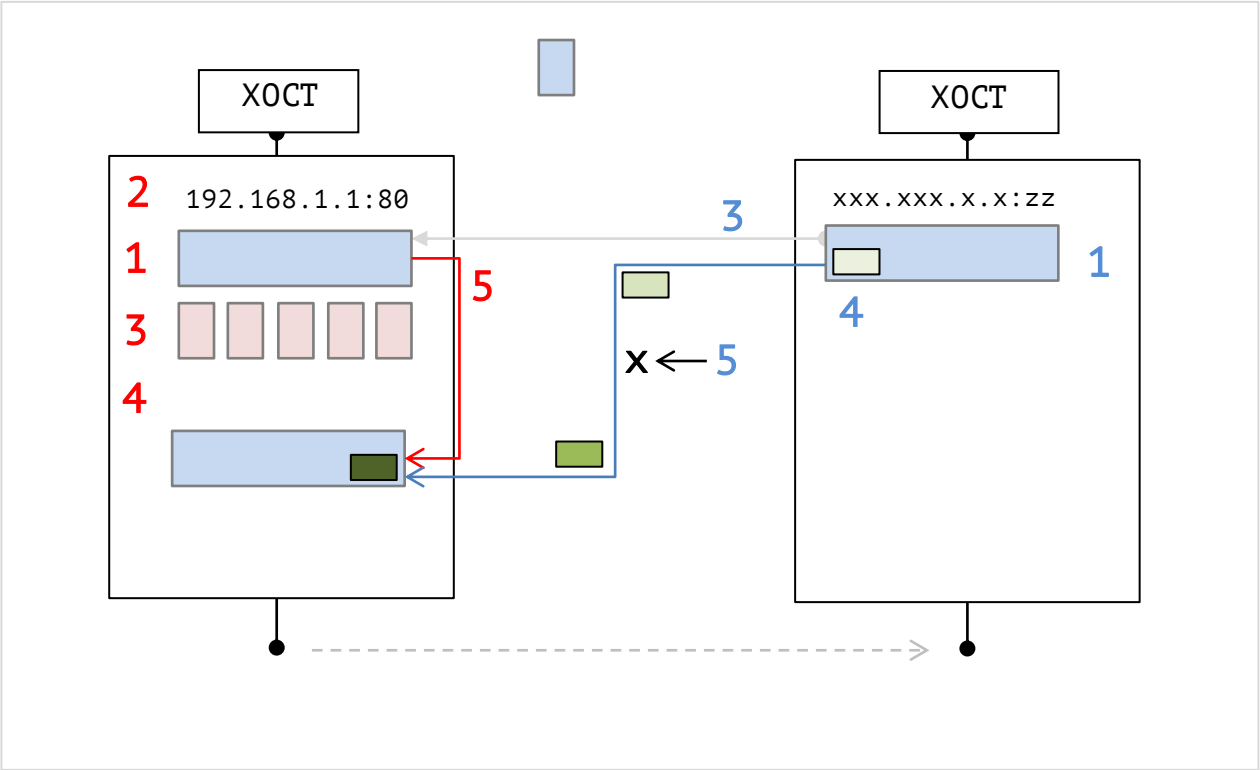


ОПЕРАЦИИ СОКЕТОВ БЕРКЛИ

(первые сокеты универсальные) клиент-сервер

socket = создание нового
bind = связать с IP:порт
listen = ждем соединения
accept = принять запрос на установку соединения
connect = установить соединение
send = отправить данные по сети
receive = получить данные из сети
close = закрыть соединение

СЕРВЕР = слуашает известный другим IP:port
КЛИЕНТ = активно устанавливает соединения с сервером



1 : socket = создание сокета
2 : bind   = назначение сокету IP:port
3 : listen = создание очереди для соединений (макс. количество)
4 : accept = готовность принимать запросы на соединения

1 : socket = создание сокета
2 : bind   = как правило, выполняется автоматически ОС
3 : connect = запрос на соединение
5 : создание копии сокета >> освобождается возможность подключения

4 : send   = передача данных
6 : receive = прием данных
повторяется до тех пор, пока необходимо передавать данные
5 : close   = закрыть соединение

***** МЕЖСЕТЕВОЙ ЭКРАН **********

Межсетевой экран = BRANDMAUER / FIREWALL (одно и тоже название) = является барьером между
    локальной сетью (аппаратный firewall) или
    отдельным хостом (програмный firewall)
и глобальной сетью ил другим хостом, которым мы не доверяем, с целью обеспечения безопасности сетевого взаимодействия

Работают на Сетевом + Транспортном уровне OSI

Для обеспечения безопасности, все данные проходят через firewall, и это устройство/по пропускает только данные, подходящие под правила в соответствующей таблице

Таблицы firewall:
- таблица соединений
- таблица правил

ТАБЛИЦА СОЕДИНЕНИЙ = регистрация всех установленных соединений.
Поля:
- отправитель = IP:порт
- получатель = IP:порт

ТАБЛИЦА ПРАВИЛ = проверка данных
Поля:
- отправитель = IP:порт
- получатель = IP:порт
- используемый в данных протокол
- установленные в заголовке протокола флаги
- проверить таблицу соединений (да/нет)
> действие (разрешить/запретить)